Introduction aux Data Sciences/nAvec r

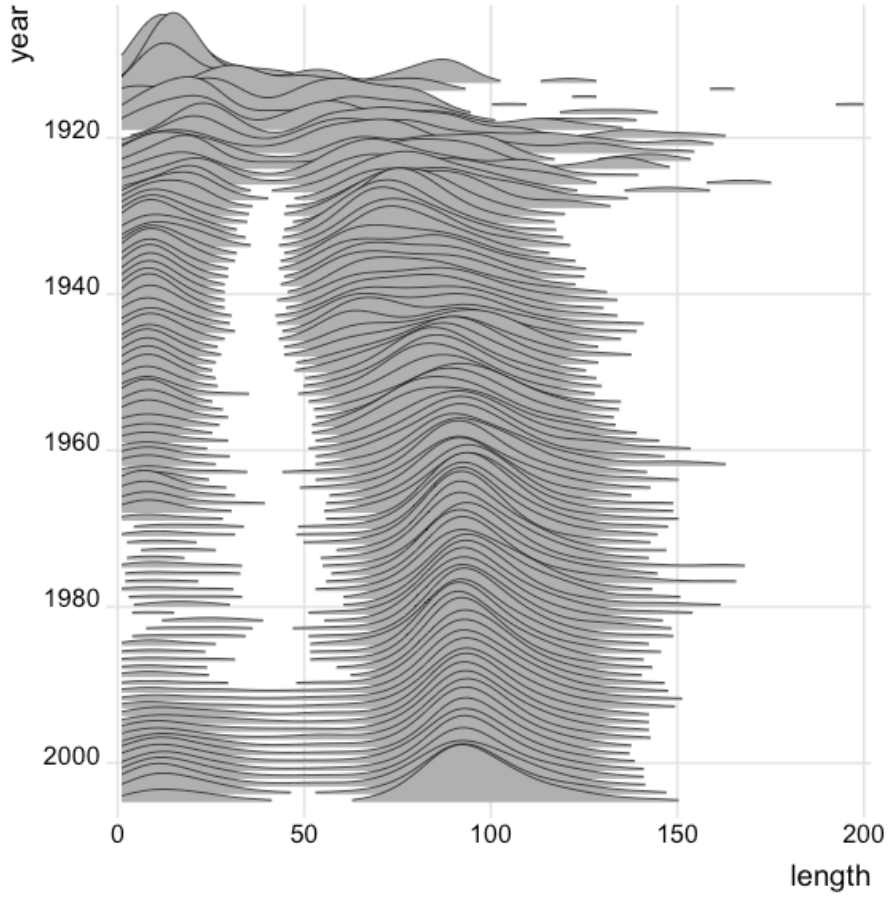
Christophe Benavent - Université Paris Dauphine

2022-10-19

Contents

Chapter 1

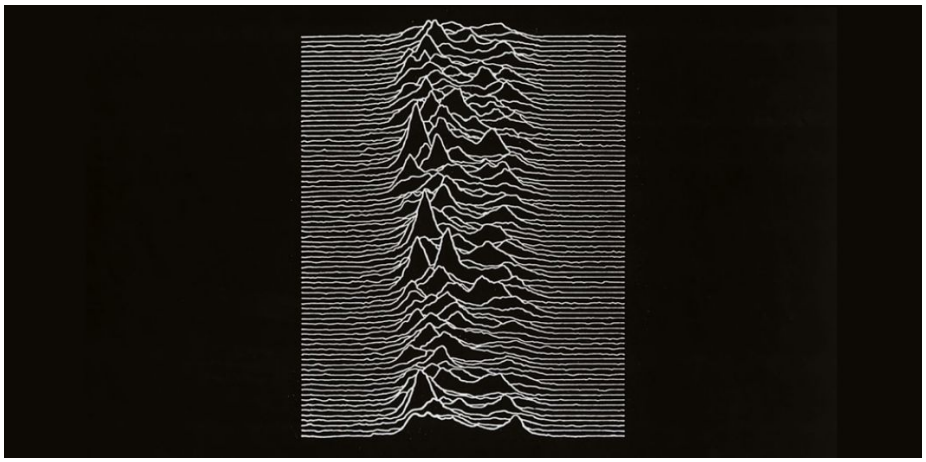
Avant propos



Ce bookdown présent les éléments d'un cours de data science avec r. Il est reproductible, on peut en cloner les éléments à partir du repository. Le texte est encore hasardeux mais les codes sont vérifiés. Il sera dynamique, modifié à mesure de nos cours, séminaires et ateliers.

L'illustration de couverture représente l'évolution de la longueur des films de la base Imbd et raconte en chiffres un aspect de l'histoire du cinéma. Jusqu'aux années 30, la longueur est hétérogène puis elle se se stabilise : les courts-métrages ont une durée de l'ordre de 15 mn qui se raccourcit avec les décennies, ce genre menace de disparaître dans les années 80 et reprend du poil de la bête dans les années 2000. Les films longs voient leur longueur s'accroître et se stabiliser autour d'un peu moins de 100 mn, soit une heure et quarante minutes. On observera enfin qu'au cours des années 1990 les films de taille intermédiaires réapparaissent. On devinera dans cette évolution l'émergence de standards, ou de conventions. Dans ce graphique il y a tous les éléments des data sciences contemporaines : un jeu de données riche et systématique, un modèle statistique fondamental avec la notion de densité de probabilité, une mesure, un critère de comparaison.

Les diagrammes ridges, c'est ainsi qu'on les appele, sont inspirés de la pochette de l'album Unknown Pleasures de Joy division sorti en pleine période New Wave, en 1979. Un article de Vice en rappele l'origine et le destin du graphisme qu'on connait mieux imprimé sur des t-shirt que dans les cours de statistiques.



1.1 Plan du manuel

C'est un projet en cours, Le plan général projeté est le suivant. Certains chapitres sont publiés (mêmes incomplets) d'autres sont dans les limbes. On les ajoutera progressivement.

- 1 - L'environnement r x
- 2 - Installation et prise en main x
- 3 - Usage de ggplot - uni et bivarié x
- 4 - Usage de ggplot - multivarié x
- 5 - Tables avec flex
- 6 - Modèles factoriels (Psych) x
- 7 - AFC x
- 8 - MDS
- 9 - Clustering x
- 10 - Analyse de réseaux
- 11 - Analyse de variance et régression linéraire x
- 12 - Modèle linéaire généralisé x
- 13 - Modèles à décomposition d'erreur x
- 14 - Modèle d'équations structurelles (Lavaan)
- 15 - Times series
- 16 - Analyse spatiale et géographique
- 17 - Machine learning x

1.2 Les jeux de données

Au cours du développement, plusieurs cas pratiques - souvent réduit en volume pour rester exemplaire, seront employés. Les données sont partagées.

Voici la présentation des sets de données utilisées dans le syllabus. Ils sont disponibles dans le répertoire "./data/"

- ESS : c'est une très belle base de données de sociologie
- happydemics : observatoire de la présidentielle2022
- NSPools
- Arpur : commerce de paris
- Botanic
- …

1.3 Le cadre technique et les packages utilisés

Ce *syllabus* est écrit en **Markdown** (**?**) et avec le package **Bookdown** (**?**). Le code s'appuie sur `tidyverse` et emploie largement les ressources de `ggplot`. Les

packages seront introduits au fur et à mesure mais un voici la liste complète.

```
options(tinytex.verbose = TRUE)
knitr::opts_chunk$set(echo = TRUE, include=TRUE, cache=TRUE, message=FALSE, warning=FA

#boite à outils et dataviz
library(tidyverse) # inclut ggplot pour la viz, readr et
library(cowplot) #pour créer des graphiques composés
library(ggridges) # le joy division touch
library(ggmosaic)
library(ggcorrplot)
library(corrplot) #à supprimer
library(ggthemes)
library(colorspace) #pour les couleurs
library(wesanderson)
library(RColorBrewer)

#networks
library(igraph)
library(ggraph)

# Accéder aux données
library(rtweet)  # une interface efficace pour interroger l'api de Twitter

# NLP
library(tokenizers)
library(quanteda)
library(quanteda.textstats)
library(udpipe) #annotation syntaxique
library(tidytext)
library(cleanNLP) #annotation syntaxique

#sentiment
library(syuzhet)              #analyse du sentimeent

#mise en page des tableaux
library(flextable)

#statistiques et modèles
library(lme4) #pour des modèles plus complexe que les mco
library(jtools) #une série d'utiltaire pour bien représenter les résultats
library(interactions) #traitement des interactions
library(nlme)#pour les hlm
library(psych) #pour la psychometrie

#ACP et AFCM
```

```
library("FactoMineR")
library("factoextra")

#ML
library(caret)

#regression
library(lme4)
library(jtools)
library(interactions)
library(betareg)
library(lavaan)

# Utilitaires
library(citr) #pour insérer des références dans le markdown

library(MASS)


#config plot
theme_set(theme_minimal())
```

L'ensemble du code est disponible sur github. A ce stade c'est encore embryonnaire. Les proches et nos étudiants pourrons cependant y voir l'évolution du projet et de la progression. Une version pdf est disponible ici.

Quelques conventions d'écriture du code r

- On dénomme les data frames de manière générale `df`, les tableaux intermédiaires sont appelé systématiquement `foo`
- Gestion des palettes de couleurs ** une couleur :" royalblue" ** deux couleurs ** 3 à 7 couleurs
- On emploie autant que possible le dialecte tidy.
- Les chunks sont notés en 4 chiffres : 2 pour le chapitre et deux pour le chunck. 0502 est le second chunk du chapitre 5.
- On commente au maximum les lignes de code pour épargner le corps du texte et le rendre lisible

Chapter 2

Introduction aux data sciences

L'objet du manuel est de donner un aperçu général des méthodes d'analyses de données et de data sciences Mais avant de s'engager dans les procédures une réflexion épistémologique et historique peut être utile. Si les méthodes sont puissantes, inventives, ils faut aussi s'interroger sur leur conditions d'émergences. La discipline fût la statistique, elle alimenta milles champs spécifiques : économétrie, psychométrie, biostatistiques. Derrières les problèmes l'avance des mathématiques pour caractériser les modèles proposés. Elle s'est laissée aller à d'autres terminologies : analyse des données, data mining, Machien learning, deep learning.

2.1 Science, art, technique et pratiques

Plutôt que le terme consacré de data sciences, il vaudrait mieux parler de data ingénierie dans la mesure où le data scientiste participe à un processus de production qui va de l'acquisition des données à leur propagation dans l'organisation ou la société. La technique domine sur la science et l'unité se trouve dans l'intégration de ce processus. La révolution des données vient de l'interopérabilité croissante de ces techniques et d'une intégration qui fluidifie le passage d'une étape à une autre. Standards et langages en sont les éléments clés.

Du côté des sciences, ce dont bénéficie l'univers des data sciences, c'est l'héritage de cultures statistiques foisonnantes qui après s'être développées dans leur cocon disciplinaire, se retrouvent désormais rassemblées dans un même langage. Bien sur il y a de manière sous-jascente les mathématiques et les statistiques qui construisent les fondements des modèles et des techniques.

Mais leur développement s'est fait souvent quand le scientifique se retrouve face à un problème où une observation. Prenons le cas des psychologues qui ont inventé l'analyse factorielle dans le but de pouvoir tester certains de leurs concepts : un degré d'intelligence, une personnalité, des attitudes.

Ou celui des écologues qui souhaitent estimer une population de poisson dans une rivière, problème qui a donné naissance aux modèles de capture/recapture. On pourrait ajouter les géographes avec les modèles d'analyse spatiale, les financiers face à la variabilité des cours des places boursières, etc. Celui des économètres est peut-être le plus évident. Les biostatisticiens sont des contributeurs importants.

Ce que la technique apporte c'est l'intégration par un langage et donc un ensemble de conventions, incarnées par r et python, algorithmes, et de programmes qui ne sont plus spécifiques à un domaine, mais peuvent circuler de l'un à l'autre. C'est ainsi que le catalogues de toutes les techniques psychométriques devient accessible aux autres disciplines par le biais d'un package en particulier, psych. De la même manière l'outillage des linguiste devient accessible aux autres disciplines, pensons aux économiste qui intégrent dans le indicateurs des sources textuelle telle que l'analyse du sentiment. (ref)

L'interopérabilité apportée par ces langages ne se définit pas que par l'algorithme qui aurait été porté d'un autre langage vers celui-ci (des cas de réécriture ?) mais aussi par des programme passerelle qui à partir de r permettent d'activité des algorithme écrit en C, en javascript ou tout autre langage "plus informatique" et souvent plus efficace.

2.2 Une courte histoires des logiciels statistiques

Ce qu'on observe dans l'évolution des logiciels

- 1980 : stat-itcf
- 1980 : SAS comme accès à r
- 1990 : SPSS
- stata
- 1997 : s dès 976 puis r, 1996 fre sofsware as r. le CRAN nait en 1997. 2003 création dez r foundation.
- 1991 - 2001: Python - Guido van Rossum - Python Software Foundation, créée en 2001
- keras
- tensor flow

http://www.deenov.com/blog-deenov/histoire-du-logiciel-spad.aspx

Un des grand mouvement du domaine est l'hésitation entre le programmatique et le no-code. La pression commerciale conduit certains acteurs à encourager

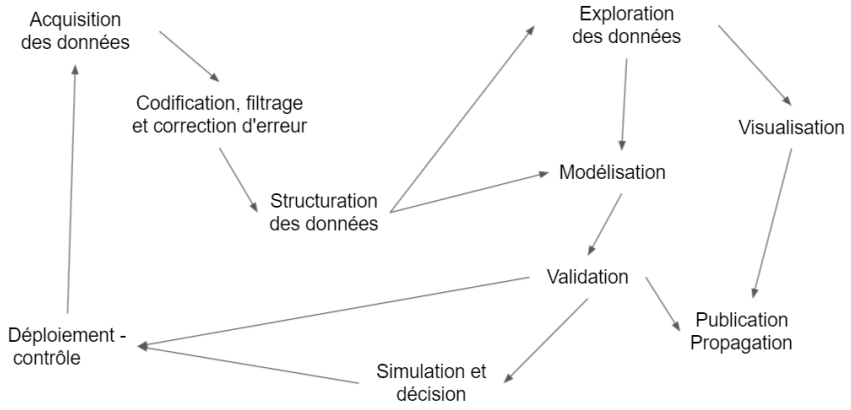
l'usage de sur-couche logiciel qui débarrasse l'utilisateur de l'exigence techniques, il peut se laisser guider par l'intuition, mais l'aliène en dissimulant la mécanique profonde des processus de traitement des données.

Le succès de python et de r réside dans

- La modularisation : langage de base /fonction/ package et notion de dépendance
- L'interopérabilité pas toujours parfaite (versions, classes de données)
- La cumulativité : les fonction s'ajoutent aux fonctions, se sédimentent
- L'accès

2.3 Le processus de traitement des données

Les data sciences ne sont finalement que l'intégration d'un flot de traitement des données qui va de l'acquisition à la divulgation.



- Acquisition
- Codification , filtrage et correction d'erreur
- Structuration des données : api, open data
- Exploration
- Modélisation :
- validation : tests versus AB testing
- Simulation et décision
- Vizualisation et sensemaking
- Déploiement :
- Contrôle :
- Publication : dash board, pdf , slide etc, webb site

2.4 Les facteurs sociaux du développement des datasciences

Ces développements sont favorisés par un environnement fertile dont quatre facteurs se renforcent mutuellement. La constitution d'un système de communication commun organisé autour de peu de langage, et d'un ensemble de normes de données mais aussi la vitalité d'une communauté. La multiplicité des sources de données et l'évolution des technologies de la mesure et du nombre constitue un second groupe de facteurs.

2.4.1 Une lingua franca

La lingua franca est la langue des ports et du commerce de la méditerranée au XIVème siècle, un mélange de langage qui sert l'échange, un commun pourrait-on dire aujourd'hui. C'est ce que sont devenus python et r parmi d'autres, la seconde langue après l'anglais qui s'est imposé comme la langue d'écriture. Le langage des scientifiques est sans doute désormais un pidgin, un créole d'anglais, et de r ou de python, sans compter les sparc, les C++ ou javascript. Les langues de la donnée se mêlent volontiers, elle sont de plus en plus agnostique.

L'environnement r par exemple devient de plus en plus ouverts à python, à la fois de manière directe en permettant de coder dans un même document des calculs en r pluis en python, mais aussi de manière indirecte parce que prolifèrent des packages passerelles permettant d'aller chercher des ressources écrites dans un autre langage.

2.4.2 Une communauté

Le second facteur, intimement lié au premier, est la constitution d'une large communauté de développeurs et utilisateurs qui se retrouvent aujourd'hui dans des plateformes de dépôts (Github, Gitlab), de plateformes de type quora (StalkOverFlow), de tutoriaux, de blogs (BloggeR), de journaux (Journal of Statistical Software) et de bookdown. Des ressources abondantes sont disponibles et facilitent la formation des chercheurs et des data scientists. Toutes les conditions sont réunies pour engendrer une effervescence créative.

Cette communauté se reproduit à petite échelles dans les procédures de laboratoire et les conventions de travail en commun des chercheurs. Elle peut se développer autant verticalement qu'horizontalement : des hubs qui concentrent l'ensemble des acteurs et des ressources, qu'un grand nombre de micro communauté focalisés sur des problèmes très locaux.

2.4.3 La multiplication des sources de données.

Le troisième est la multiplication des sources de données et leur facilité d'accès. Les données privées, et en particulier celles des réseaux sociaux, même si un péage doit être payé pour accéder aux APIs, popularisent le traitement de données massives.

Le mouvement des données ouvertes (open data) proposent et facilitent accès à des milliers de corps de données : retards de la SNCF, grand débat, le formidable travail de l'Insee, european survey etc.

2.4.4 de la statistique à l'IA

Le retour au boites noires dans les années 2000. Ce qui distingue les statistiques traditionnelles de l'approche machine learning réside d'abord par une approche de la modélisation différente.

Les modèles statistiques et économétriques considèrent une structure de relation, la spécification du modèle (ex : le modèle linéaire), mais aussi des modèles de distribution des erreurs qui définissent le cadre d'estimation. L'évaluation passe par le test des hypothèses sur les paramètres et par la qualité d'ajustement.

Le machine learning, se concentre sur la valeur prédictive, et considère n'importe quelle spécification même si elle est peu intelligible et comprend de grandes quantités de paramètres sur lesquels aucun test n'est produit.

Les deux approches ont plutôt tendance à ce compléter, les première testant des théories, les secondes procurant aux première de nouvelles hypothèse par de nouvelle mesure. Pour en donner un exemple simple, l'analyse de sentiment emploie des modèles complexes pour le prédire avec le seul texte, l'IA permet d'enrichir des données empiriques par exemple en testant en finance la relation de cet indicateurs aux prix de marché. Un autre exemple en marketing.

Les méthodes disponibles se sont accumulées depuis ces dernières 20 années. faisons-en une courte liste.

- 1956 :perceptron
- 1963 : arbre de décision
- 2005 : CNN
- 2008 : lda topic
- 2013 : word2vec
- 2018 : transformers
- ….

KNN, SVM, rf et le retour des réseaux de neurones.

2.5 Conclusion

Il ne reste plus qu'à soulever le capot et de mettre les mains dans le cambouis.

Et à se rappeler que si la nécessité de se faire remarquer à conduit les acteurs du domaine à envisager des data sciences, que c'est d'abord un art d'écriture, et une pratique qui permet à leurs artisans de s'échanger des secrets de fabrique.

On remerciera tous ceux qui développent des Packages, nous aurons le point de vue de ceux qui les utilisent. Ce cours est aussi un livre de recette, celui d'un chercheur en sciences sociales qui picore dans l'immensité de la production pour trouver des procédures reproductibles par ses étudiants.

Chapter 3

Prise en main de r

Pour démarrer :

- 1 - Télécharger et installer r sur le site du Comprehensive r Archive Network
- 2 - Télécharger et installer Rstudio.(version free)
- 3 - Dans le cadre de cet atelier, on adopte la méthode du rmarkdown. On recommande fortement de lire l'ouvrage de référence, même si la prise en main est très rapide.
- 4 - Il est désormais indispensable d'utiliser le package `tidyverse` et en particulier les fonctions de manipulation et de pipe (%>%) fournies par `dplyr`. Ce sera donc le premier package à installer (attention, il appele de nombreuses dépendences, l'installation peut prendre plusieurs minutes)

3.1 La convention du Rmarkdown

Différentes manières d'interagir avec r sont possibles : la première est le mode console, pour de petite opérations et un utilisateur chevronné, celà peut être commode car rapide mais très rapidement on sera amené à enregistrer les opérations dans des scripts. Une idée novatrice a été d'intégrer l'ensemble des éléments dans un seul document : le script découpé en petits éléments : des chunks, le commentaire et l'analyse verbabe dans un format texte, et le résultat. Dans l'univers python il s'agit des carnets Jupiter, pour r c'est le rmarkdown.

C'est un dialecte du markdown générique adapté au langage r. On recommande au lecteur d'en lire le manuel et de le garder dans ses onglets.

Quelques éléments de base :

un document markdown est composé de plusieurs éléments

1. Yalm : dans cet entête les éléments essentiels sont définis et paramétrés
2. Texte : il suit les conventions de mise en forme du html :
    - des # pour les niveau de titres
    - une syntaxe (x)[.xxx] pour des liens vers les URLS ou des images.
3. Les chunks sont isolés par 3 tiks au début et à la fin.
4. Résultats apparaissent sous les chunks après avoir été exécutés

Ce document peut être publié sous différents formats : html, pdf ou même word.

Il comprend les éléments suivants :

- Plan
- Texte
- Code
- Résultats
- Bibliographie
- Références
- Liens
- Images

3.2 Lire les données

La première étape c'est la lecture des données. On commence par lecture de fichiers locaux, dont les formats sont multiples : csv, tsv, xlsx, Spss, etc… Pour chacun d'eux existe une fonction dédiée. Le package `readr` contribue à cette tâche pour les fichiers *.csv.

```
df <- read_csv("./Data/BXL_listings.csv")
head(df,5)
```

```
## # A tibble: 5 x 16
##      id name       host_id host_~1 neigh~2 neigh~3 latit~4 longi~5 room_~6 price
##   <dbl> <chr>        <dbl> <chr>   <lgl>   <chr>     <dbl>   <dbl> <chr>   <dbl>
## 1  2352 Triplex-2~    2582 Oda     NA      Molenb~    50.9    4.31 Entire~    91
## 2  2354 COURT/Lon~    2582 Oda     NA      Molenb~    50.9    4.31 Entire~    74
## 3 45145 B&B Welco~  199370 <NA>    NA      Bruxel~    50.9    4.37 Hotel ~   120
## 4 48180 Top Apart~  219560 Ahmet   NA      Woluwe~    50.8    4.41 Entire~   200
## 5 52796 Bright ap~  244722 Pierre  NA      Ixelles    50.8    4.36 Entire~    74
## # ... with 6 more variables: minimum_nights <dbl>, number_of_reviews <dbl>,
## #   last_review <date>, reviews_per_month <dbl>,
## #   calculated_host_listings_count <dbl>, availability_365 <dbl>, and
## #   abbreviated variable names 1: host_name, 2: neighbourhood_group,
## #   3: neighbourhood, 4: latitude, 5: longitude, 6: room_type
```

Il est possible aussi d'accéder en direct aux données du web, c'est bien utile pour s'assurer que les données sont bien fraîches. Par exemple une connexion à Nsppolls qui propose une compilation de tous les sondages d'intention de vote de la présidentielle 2022.

```
df_pol <- read_delim("https://raw.githubusercontent.com/nsppolls/nsppolls/master/presidentielle.c
                     delim = ",", escape_double = FALSE, trim_ws = TRUE)
```

Bien d'autre possibilités sont offertes, on pourra utiliser des API, des programmes de scrapping., ire en bouche des fichiers dans un répertoire, interroger des bases SQL des SGBD) ou d'autres systèmes.

3.2.1 La diversité des formats

Peu de formats échappent à r, ils peuvent faire appel à des packages spécifiques

- excell
- Json
- shape et autre données géographiques.
- les formats bibliographiques sont plus exotique : bib et ris
- les xml pourront donner des maux de têtes.

3.3 `Dplyr` pour manipuler les données

Dès lors que les données sont chargées en mémoire il va souvent être nécessaire d'en travailler, l'aspect et la structure. L'aspect concerne les formats et les significations, les recodages. La structure est relative à la forme des tableaux. Il faudra souvent traiter les données brutes pour proposer à nos modèles des structures appropriées.

`Dplyr` est un des packages essentiels de la suite tidyverse. Il permet de manipuler aisément les données et mérite une étude approfondie. Un point de départ ou en français : dplyr.

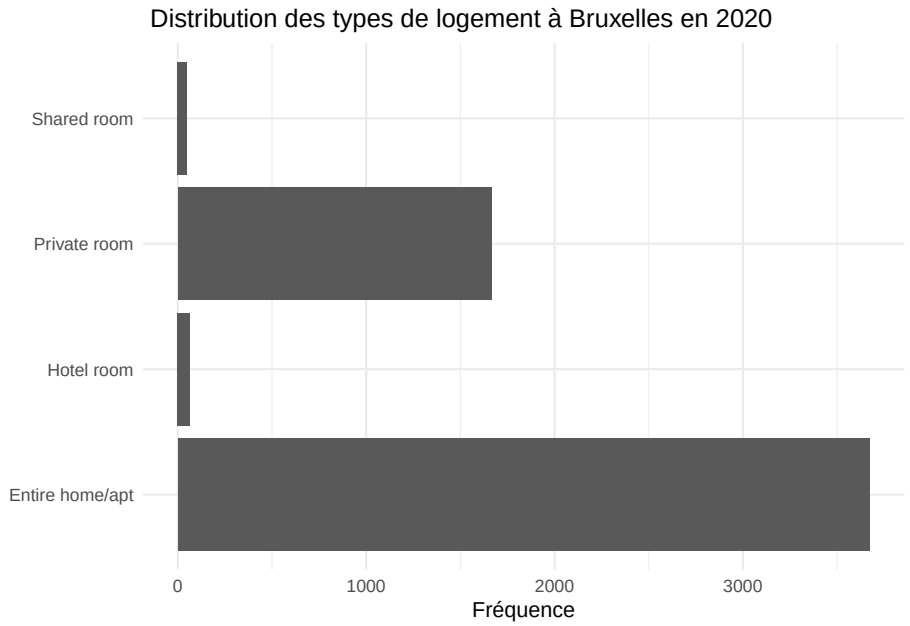
Deux idées sont au coeur de `Dplyr` d'abord celle du pipe, ensuite celle du verbe. `Dplyr` encourage une approche processus et performative.

3.3.1 Des pipes %>%

Une grand part de l'intérêt de dplyr est de reprendre un opérateur de magritr très utile : le pipe noté `data %>% f() %>% g()...` Celui ci permet de passer le résultat de l'opération à gauche f() sur les données `data`, dans la fonction g() à droite.

Un exemple simple : dans la ligne de code suivante, une première fonction lit le fichier CSV, et envoie le résultat de cette lecture dans une fonction graphique élémentaire: compter les occurrences des modalités de la variable room_type.

```
g <- read_csv("./Data/BXL_listings.csv") %>%
  ggplot(aes(x=room_type))+
  geom_bar()+
  coord_flip()+
  labs(x=NULL, y= "Fréquence", title=" Distribution des types de logement à Bruxelles e
g
```



3.3.2 Des verbes

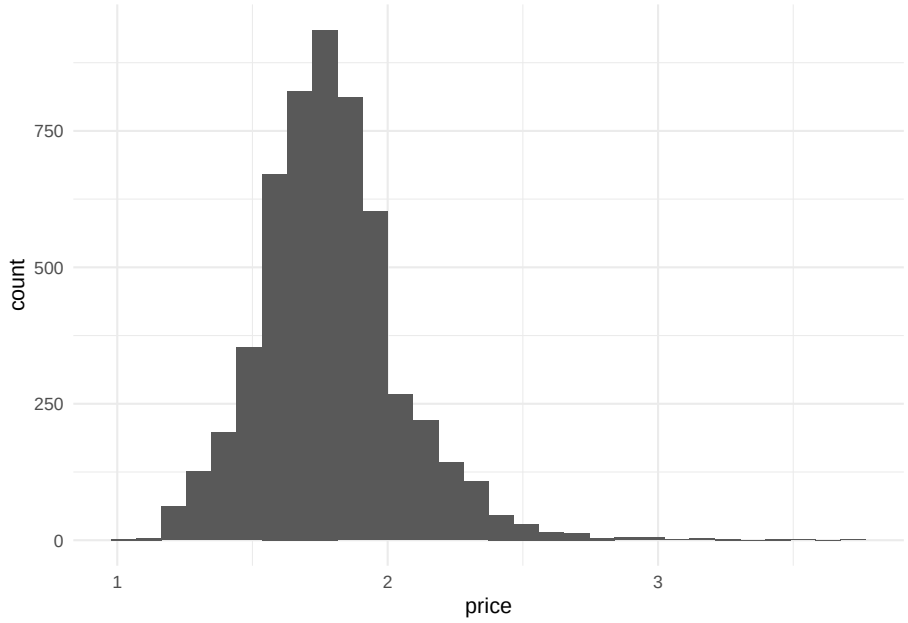
L'originalité de `dplyr`est de définir les fonctions comme des verbes. Chaque verbe désigne une action particulière. On va les examiner progressivement. * transformer une variable, * filtrer les observations selon un critère, * isoler des variables, * les grouper pour en calculer des résultats statistiques (somme, moyenne, variance, max min etc), * les déployer selon un format long ou les distribuer en différents critères, * les fusionner enfin.

3.3.2.1 Mutate

En Français c'est "transformer". On modifie la valeur d'une variable par une fonction plus ou moins complexe, éventuellement en ajoutant des conditions.

Dans notre exemple, faisant au plus simple, puisque la distribution est asymétrique, une transformation du prix par les log10 peut donner des résultats intéressants. Et c'est le cas, on retrouve une distribution qui semble être gaussienne.

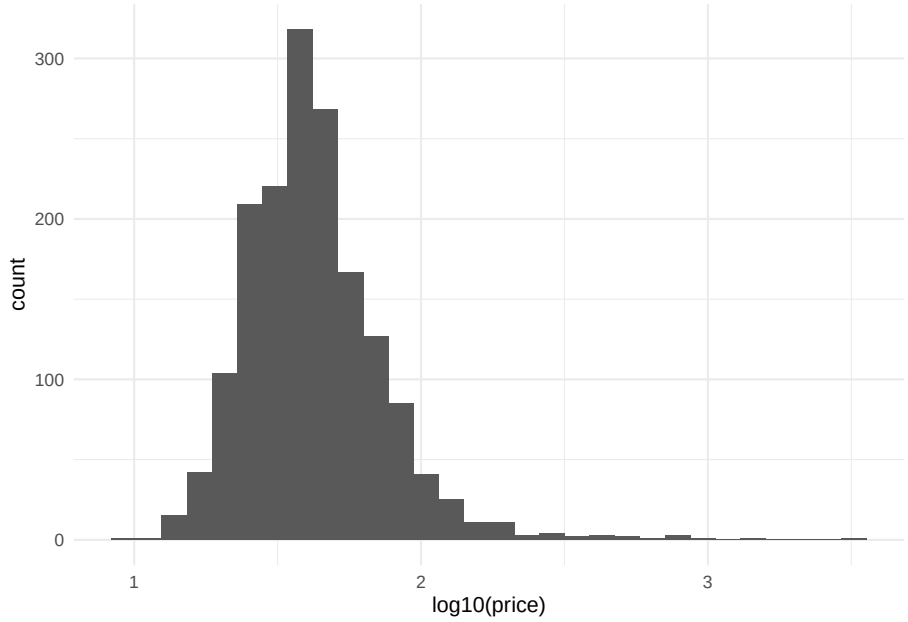
```
g <- read_csv("./Data/BXL_listings.csv") %>%
  mutate(price=log10(price))%>%
  ggplot(aes(x=price))+
  geom_histogram()
g
```



3.3.2.2 Filter

On peut vouloir se concentrer sur une sous population. Par exemple les chambres privées.

```
g <- read_csv("./Data/BXL_listings.csv") %>%
  filter(room_type=="Private room" ) %>%
    mutate(price=price)%>%
  ggplot(aes(x=log10(price)))+
  geom_histogram()
g
```



3.3.2.3 select

On peut sélectionner des colonnes pour créer un tableau spécifique. On en profite pour introduire 'flextable' , une solution élégante pour éditer des tableaux en html.

```
foo <- read_csv("./Data/BXL_listings.csv") %>%
  dplyr::select(room_type,price)

ft <- flextable(foo[ sample.int(10),])%>%
   set_header_labels(room_type="Type de logement",
  price = "Prix en euros")%>%
  theme_vanilla()%>% fontsize(size = 9)%>%
  autofit()
ft
```

Type de logement	Prix en euros
Entire home/apt	65
Entire home/apt	85
Entire home/apt	74
Entire home/apt	80

Type de logement	Prix en euros
Hotel room	120
Entire home/apt	80
Entire home/apt	95
Entire home/apt	91
Entire home/apt	74
Entire home/apt	200

3.3.2.4 Group_by et summarize

c'est une opération clé, en groupant les observations selon les modalités d'une variables, on peut construire des tableaux agrégés avec `summarise` qui permet de calculer de nombreuses statistiques : somme, moyenne, variance, max, min .. à travers les groupes.

```
foo <- read_csv("./Data/BXL_listings.csv")%>%
  dplyr::select(neighbourhood, price)%>%
    group_by(neighbourhood ) %>%
  summarise(averageprice=round(mean(price),1),
            nombreoffre=n())

#mise en forme flextable
ft <- flextable(foo)%>%
  set_header_labels(neighbourhood="Quartier",
  averageprice = "Prix en euros",
  nombreoffre="Nombre d'offre", size=9)%>%
  fontsize(size = 9)%>%
  theme_vanilla()
ft
```

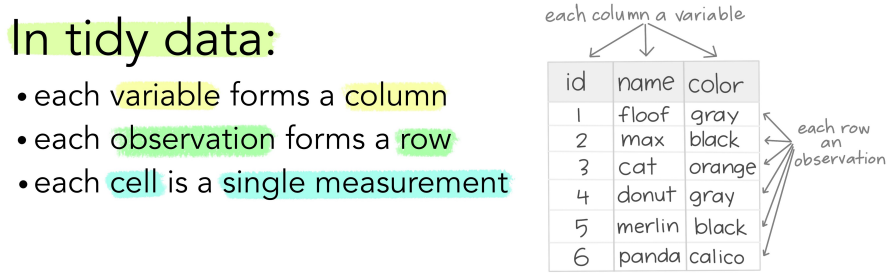
Quartier	Prix en euros	Nombre d'offre
Anderlecht	71.9	232
Auderghem	66.3	77
Berchem-Sainte-Agathe	65.9	31
Bruxelles	91.0	1,759

Quartier	Prix en euros	Nombre d'offre
Etterbeek	75.8	296
Evere	70.0	41
Forest	64.9	226
Ganshoren	50.5	21
Ixelles	81.5	849
Jette	70.3	75
Koekelberg	70.7	37
Molenbeek-Saint-Jean	67.4	179
Saint-Gilles	76.1	589
Saint-Josse-ten-Noode	55.2	136
Schaerbeek	61.9	364
Uccle	75.5	274
Watermael-Boitsfort	74.8	65
Woluwe-Saint-Lambert	62.5	121
Woluwe-Saint-Pierre	111.0	81

3.3.2.5 Pivot_wider et pivot_longer

Si pour l'habitué des feuilles de calculs, les données croisent des observations avec des variables, ce format n'est pas le seul moyen de représenter des données, et pas forcément le meilleur.

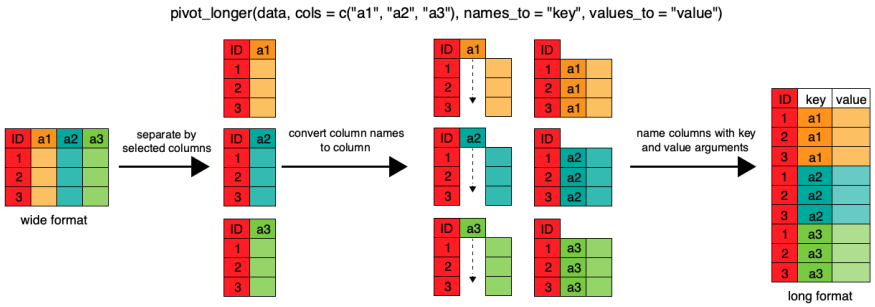
Une théorie des tidy data a été proposé par wickham : Un ensemble de données est une collection de valeurs, généralement des nombres (si elles sont quantitatives) ou des chaînes de caractères (si elles sont qualitatives). Les valeurs sont organisées de deux manières. Chaque valeur appartient à une variable et à une observation. Une variable contient toutes les valeurs qui mesurent le même attribut sous-jacent (comme la hauteur, la température, la durée) dans différentes unités. Une observation contient toutes les valeurs mesurées sur la même unité (comme une personne, ou un jour, ou une course) à travers les attributs.



Wickham, H. (2014). Tidy Data. Journal of Statistical Software *59* (10). DOI: 10.18637/jss.v059.i10

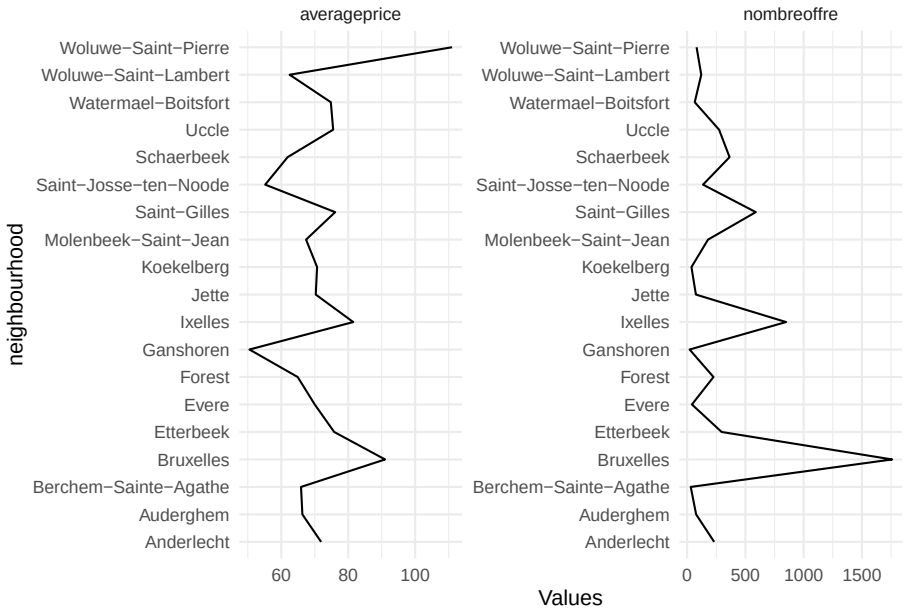
Figure 3.1: merge

Pour passer d'un tableau individu x variable à une structure ordonnée, la fonction pivot_longer est particulièrement appropriée. En voici l'anatomie.



Et un exemple numérique :

```
foo <- foo %>%
  pivot_longer(-neighbourhood,names_to= "Variables",values_to = "Values")

ggplot(foo, aes(x=neighbourhood, y=Values, group=Variables))+
  geom_line()+facet_wrap(vars(Variables),scales="free")+
  coord_flip()
```



L'opération inverste est de partir d'un tableau long vers un tableau large.

country	year	cases
Angola	1999	800
Angola	2000	750
Angola	2001	925
Angola	2002	1020
India	1999	20100
India	2000	25650
India	2001	26800
India	2002	27255
Mongolia	1999	450
Mongolia	2000	512
Mongolia	2001	510
Mongolia	2002	586

country	1999	2000	2001	2002
Angola	800	750	925	1020
India	20100	25650	26800	27255
Mongolia	450	512	510	586

Pivot data wider

```
data %>%
  pivot_wider(
    names_from = "year",
    values_from = "cases"
  )
```

On remarquera que l'usage de cette fonction est nécessaire dans l'emploi de ggplot qui suit la logique des tidy data, ou données ordonnées

3.3.3 Fusionner les données

On sera souvent amené à construire des tableaux de données en les enrichissant par d'autres tableaux et à fusionner les données.

Le cas le plus simple est d'ajouter d'autres observations à un fichier de données. On distingue deux cas :

- les deux tableaux concernent les mêmes individus classé dans le même ordre, seules les colonnes diffèrent. On utilisera la fonction `cbind()`
- si les variables sont identiques mais que les individus sont différents on peut concatène des données avec `rbind()` (L'équivalent de DPLYR est row_bind et column_bind)

```
x1<-as.data.frame(c(1,2,3,4,5)) %>%rename(x=1)

y<-as.data.frame(c("a","b","c","d","e"))  %>%rename(y=1)

z<-cbind(x1,y)

ft<-flextable(z)
ft
```

x	y
1	a
2	b
3	c
4	d
5	e

```
x2<-as.data.frame(c(9,8,7,6)) %>%rename(x=1)

w<-rbind(x1,x2)

ft<-flextable(w)
ft
```

x
1
2
3
4
5
9

x
8
7
6

mais très souvent on sera dans des cas différents et la fusion des données devra suivre des index

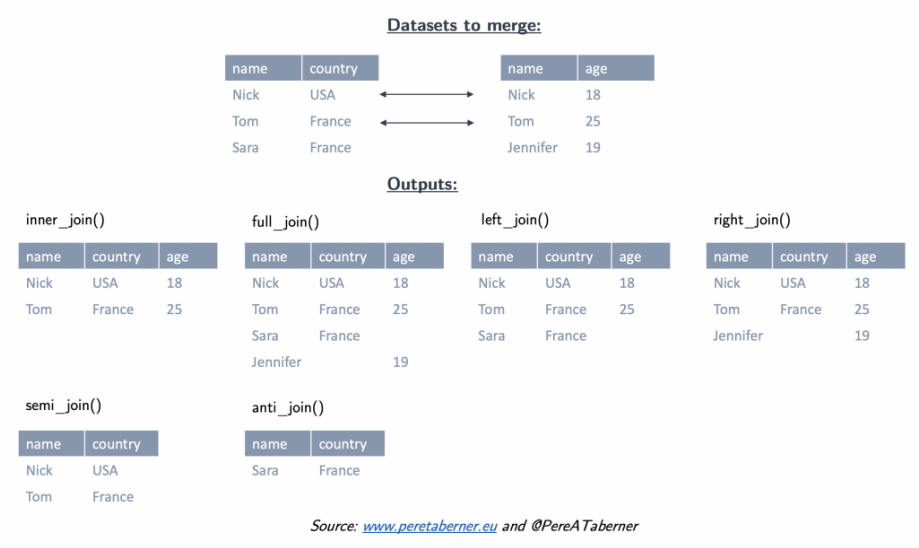


Figure 3.2: merge

Plusieurs types de fusion sont proposées.

générale

fusion à gauche

fusion à droite

(Voir ce cours)[https://coletl.github.io/tidy_intro/lessons/dplyr_join/dplyr_join.html]

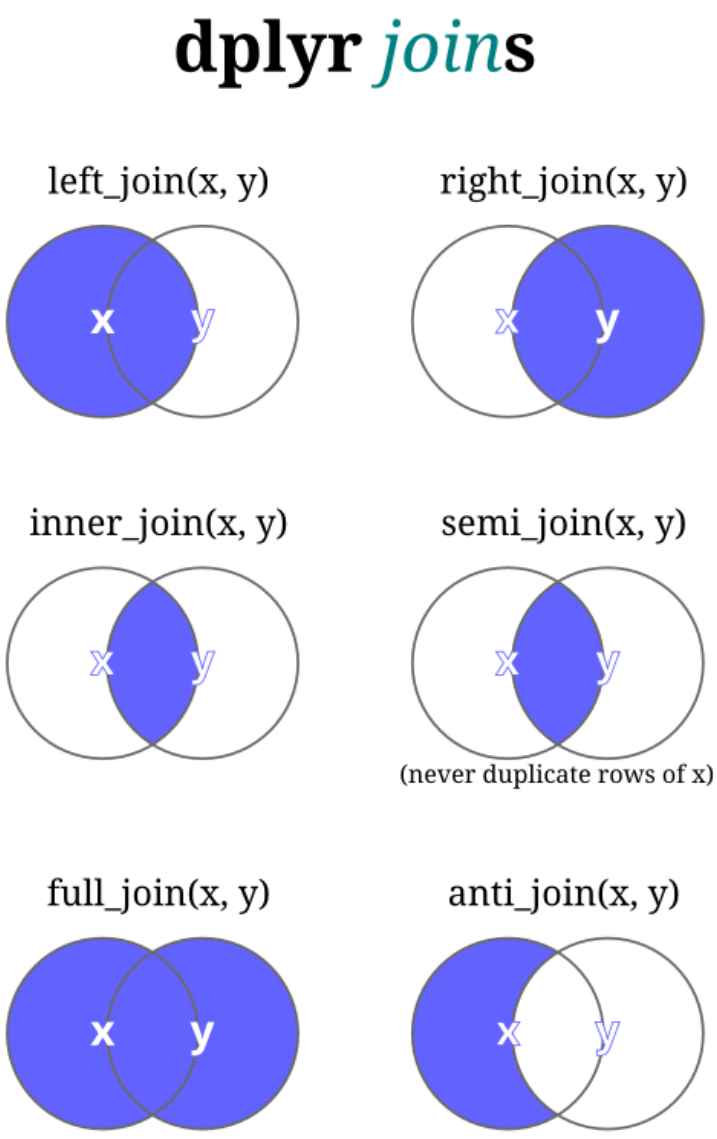


Figure 3.3: Mode de fusions

Chapter 4

Introduction à la grammaire des graphiques et à ggplot

Nous avons appris à lire des données, à les manipuler, Nous allons nous intéresser à la manière de les représenter en introduisant le concept de grammaire des graphiques et en appliquant ggplot au traitement des données univariées.

4.1 La grammaire des graphiques

C'est sans doute une des percées conceptuelles la plus intéressante des data sciences. La représentation graphiques des données fait l'objet à la fois d'une explosion créative mais aussi d'une synthèse théorique. C'est l'apport de la grammaire des graphiques.

Ces outils s'appuient sur l'idée de grammaire des graphiques. En voici un clair résumé.En français il y a toujours le larmarange

4.1.1 Un modèle en couche

Celle-ci met un ordre dans les éléments qui composent un graphique et les superpose.

- l'aesthetic definit les éléments que l'on veut représenter : ce qu'on met en abscisse, ce qu'on met en ordonnné, les groupes que l'on veut distinguer.
- la geométrie (geom_x)qui définit la forme de représentation

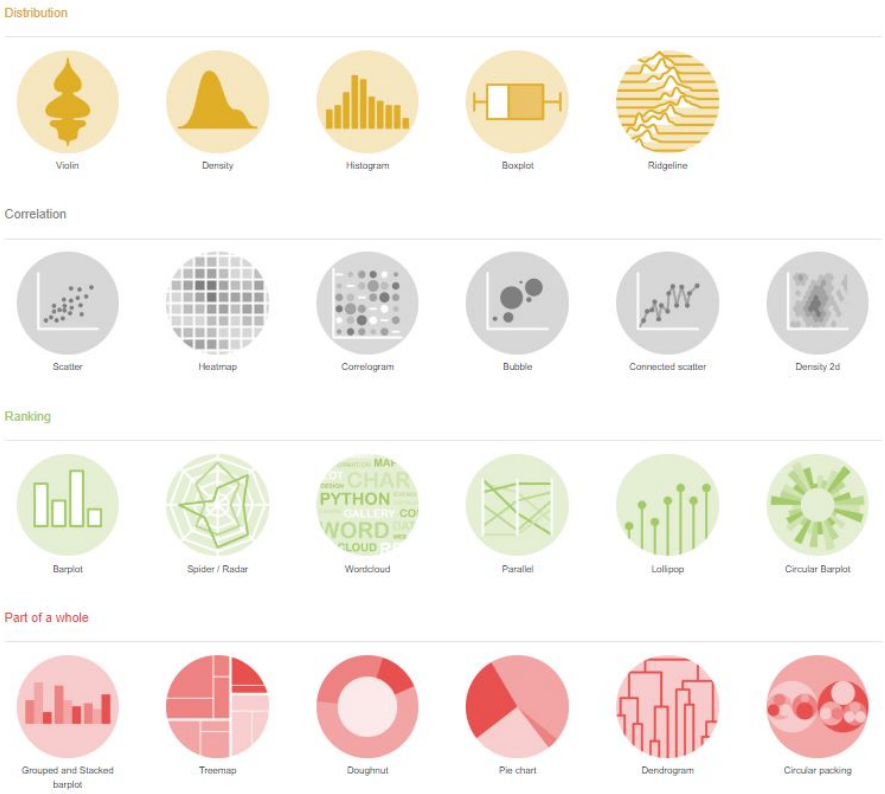
Figure 4.1: layers

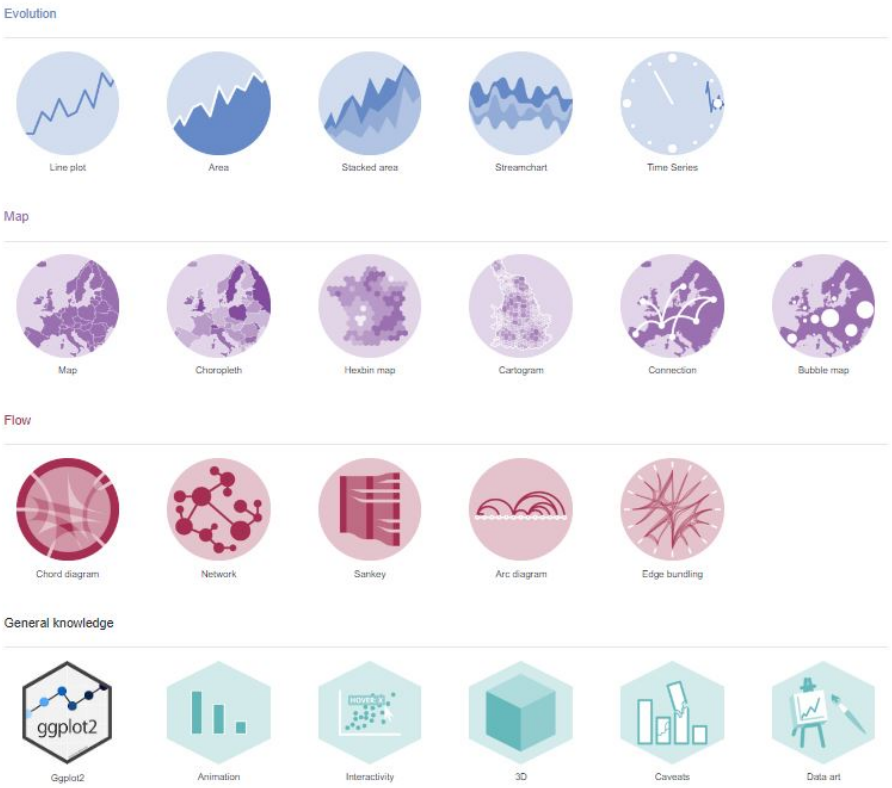
- les échelles (scale_x)
- Labelisation (labs)
- les templates
- le facetting

ggplot est construit selon cette structure. Voici le book de référence, qui est au centre de ce cours. On aura besoin de manière assez systématique de manipuler les données avant de les représenter, dplyr nous permet de le faire aisément.

4.1.2 Une typologie des représentations

Un point de départ fondamental est la gallery de ggplot„ elle présente de manière synthétique la plupart les types de figures qui peuvent être représentées, avec du code facilement reproductible.


Distribution
Violin
Density
Histogram
Boxplot
Ridgeline
Correlation
Scatter
Heatmap
Correlogram
Bubble
Connected scatter
Density 2d
Ranking
Barplot
Spider / Radar
Wordcloud
Parallel
Lollipop
Circular Barplot
Part of a whole
Grouped and Stacked barplot
Treemap
Doughnut
Pie chart
Dendrogram
Circular packing



Une classification simple

- Analyse univariée : une seule variable quantitative ou qualitative.
- Analyse bivariée : deux variables quali ou quanti/
- Analyse multivariée

  - les variables sont quantitatives : on analyse des matrices de corrélations
  - les variables sont qualitatives : on analyse des tableaux croisés

- Analyse temporelles :
- Analyse géospatiale : les chloroplèthes.
- Analyse de réseaux : représenter des noeuds et les arcs qui les relient. Depuis Moreno,
- analyse d'arbres : ils sont des sortes de réseaux mais avec une structure hiérarchiques. le dendogramme est le plus connus.
- Diagramme de flux

4.1.3 L'esthétique

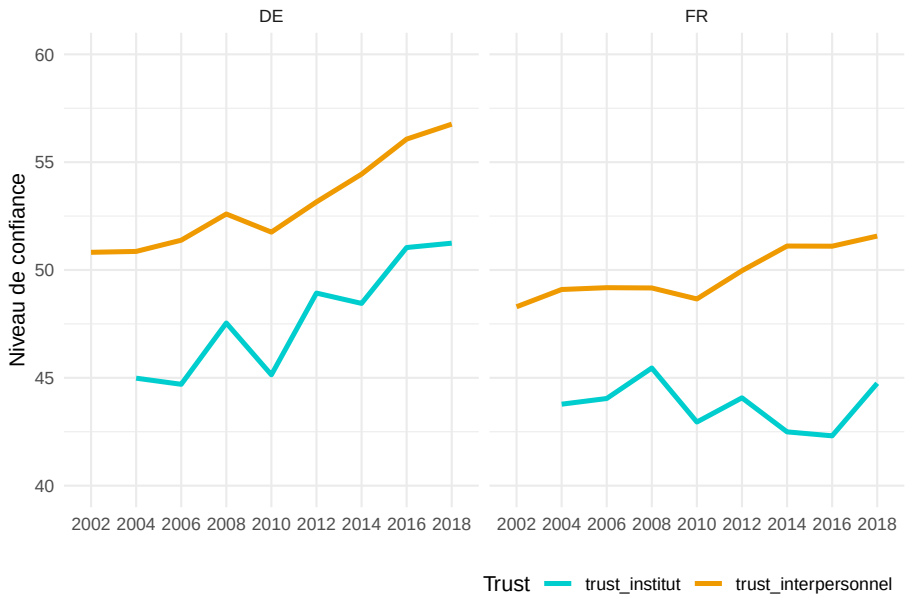
L'art des couleurs tient dans les palettes on aimera celles de Wes Anderson, on peut adorer `fishualize` si on a un faible pour les poissons tropicaux.

Pour la théorie voir ...

Si l'on est pas créatif on se reportera à des modèles, certains étant fameux. On laisse le lecteur les rechercher ici ou là . A titre d'exemple, une belle leçon, pour reproduire le célèbre BigMac index de The Economist.

Application à l'analyse univariée

Les données sont extraites de l'ESS, une sélection est disponible ici. Elle couvre les 9 vagues et concernent la France et L'Allemagne. Les variables dépendantes (celles que l'on veut étudier et expliquer) sont les 9 items de la confiance, les variable considérées comme indépendantes (ou explicatives) sont une sélection de variables socio-démographiques : âge, genre, perception du pouvoir d'achat, orientation politique, type d'habitat.



L'analyse univarié, comme son nom l'indique, ne s'intéresse qu'à une seule variable. Celle-ci peut être **quantitative** ou **qualitative** et ne comporter qu'un nombre limité de modalités entre lesquels aucune comparaison de grandeur ne peut être faite. Les premières ont le plus souvent dans r un format numérique, les autres correspondent au format *factor*.

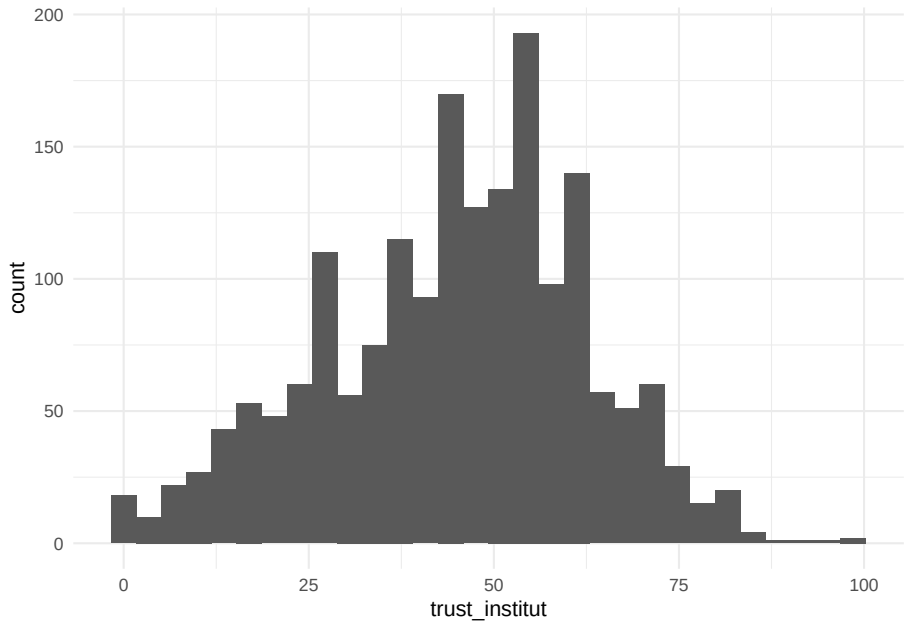
4.1.4 Le cas des variables quantitatives

Les variables quantitatives décrivent une variable dont les valeurs décrivent les quantités d'une grandeur. Elle peuvent être discrètes (dénombrement du d'un nombre d'unités) - le nombre d'habitant), ou continue (le nombre de km parcourus). l'**histogramme** est l'outil de base pour représenter la distribution d'une telle variable. Il représente pour des intervalles de valeurs donnés, la fréquence des observations.

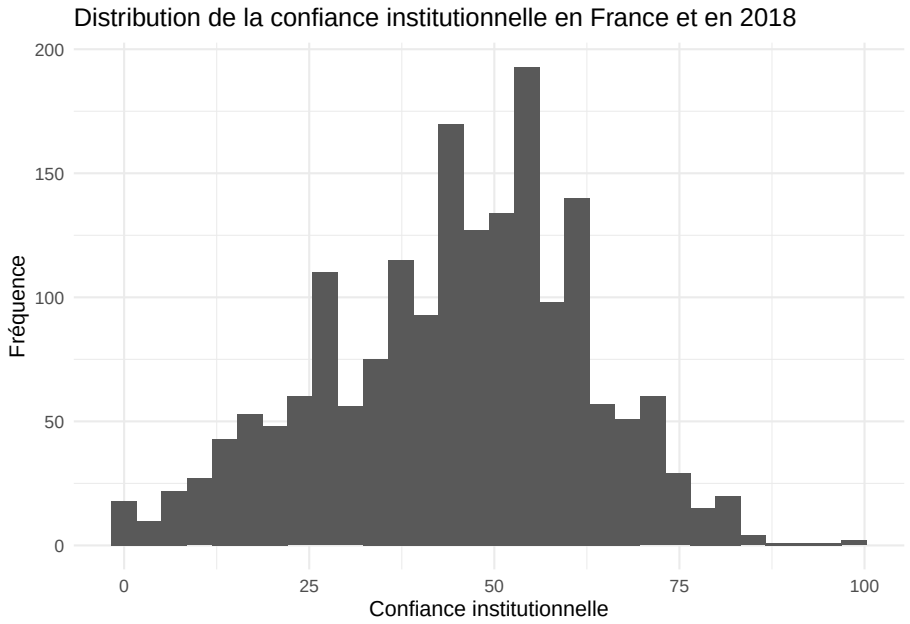
Sa syntaxe simple comporte d'abord la définition de la variable et de la source de données, puis une des "géométrie" de ggplot : la fonction `geom_histogram`. Dans notre exemple, on va représenter le score de confiance institutionnelle pour la France en se concentrant sur la dernière vague d'enquête.

```
df<-readRDS("./data/dfTrust.rds)")

#filtrage sur 2018 et la France.

foo<-df%>%
```

```
  filter(Year=="2018" & cntry=="FR" & !is.na(trust_institut))

# on stocke le diagramme dans l'objet g00,
# pour le réutiliser ultérieurement et pouvoir le compléter.

g00<-ggplot(foo,aes(x=trust_institut))+
  geom_histogram()

g00
```




```
g00+labs(title="Distribution de la confiance institutionnelle en France et en 2018",
         x="Confiance institutionnelle", y="Fréquence")
```



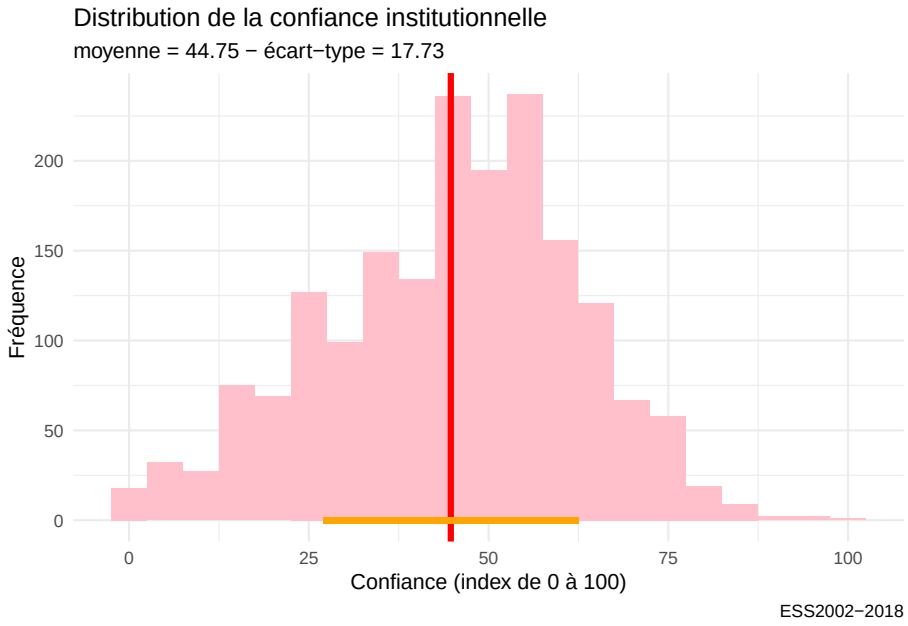
On va améliorer l'aspect en

1. modifiant la couleur et la largeur des barres,
2. ajoutant un thème,
3. en précisant les éléments textuels (titres, label)
4. en calculant et en représentant la valeur moyenne et l'écart-type . Pour ces statistiques, on emploie les fonction de base : mean, sd et round.

On notera que le titre est défini par la concaténation de plusieurs chaines de caractères avec la fonction `paste0`. On peut ainsi injecter dans le graphique des éléments externes au jeu de données.

```
#on calcule la moyenne
moy=mean(foo$trust_institut, na.rm=TRUE)
sd=sd(foo$trust_institut, na.rm=TRUE)

#avec tous les éléments
g01 <-ggplot(foo,aes(x=trust_institut))+
  geom_histogram(binwidth=5,fill="pink")+
  labs(title= "Distribution de la confiance institutionnelle",
       subtitle= paste0("moyenne = ",round(moy,2), " - écart-type = ", round(sd,2)),
       caption="ESS2002-2018",
       y= "Fréquence",
       x="Confiance (index de 0 à 100)")+
    geom_vline(xintercept=moy, color="red",size=1.5)+
```

```
        geom_segment(y = 0, yend=0,x=moy-sd,xend=moy+sd, color="orange",size=1.5)

g01
```

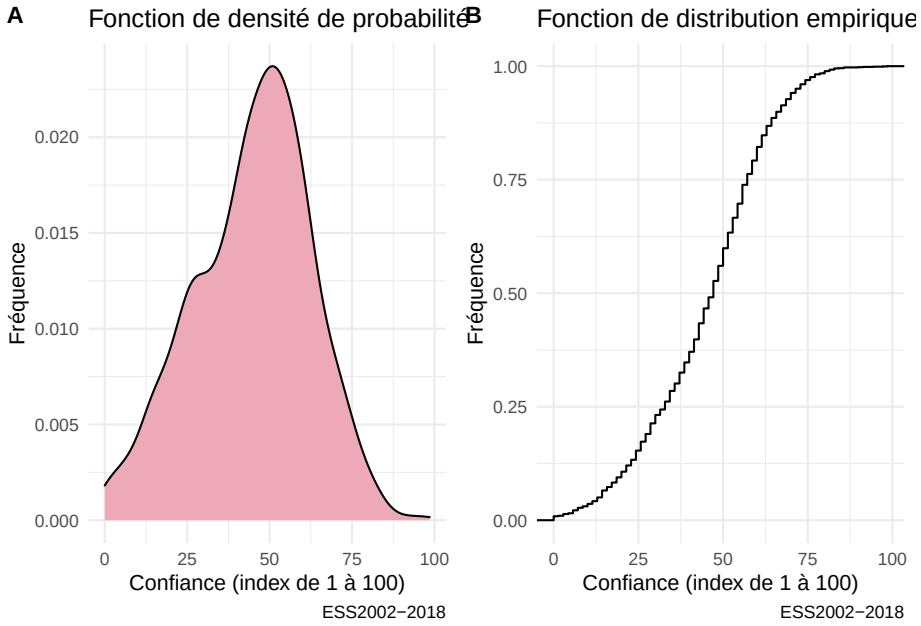


On peut souhaiter normaliser un tel graphe et prendre pour convention que la surface soit égale à 1. On représentera donc une fonction de densité de probabilité, à laquelle on peut associer une fonction cumulée de la distribution.

On en profite pour introduire l'usage de `cowplot` qui permet d'associer des graphiques en un seul document.

```
g04<-ggplot(foo,aes(x=trust_institut))+
  geom_density(fill="pink2") +
  labs(title= "Fonction de densité de probabilité",
       caption="ESS2002-2018",
       y= "Fréquence",
       x="Confiance (index de 1 à 100)")

g05<-ggplot(foo,aes(x=trust_institut))+
  stat_ecdf(geom = "step")+
  labs(title= "Fonction de distribution empirique cumulée",
       caption="ESS2002-2018",
       y= "Fréquence",
       x="Confiance (index de 1 à 100)")
```

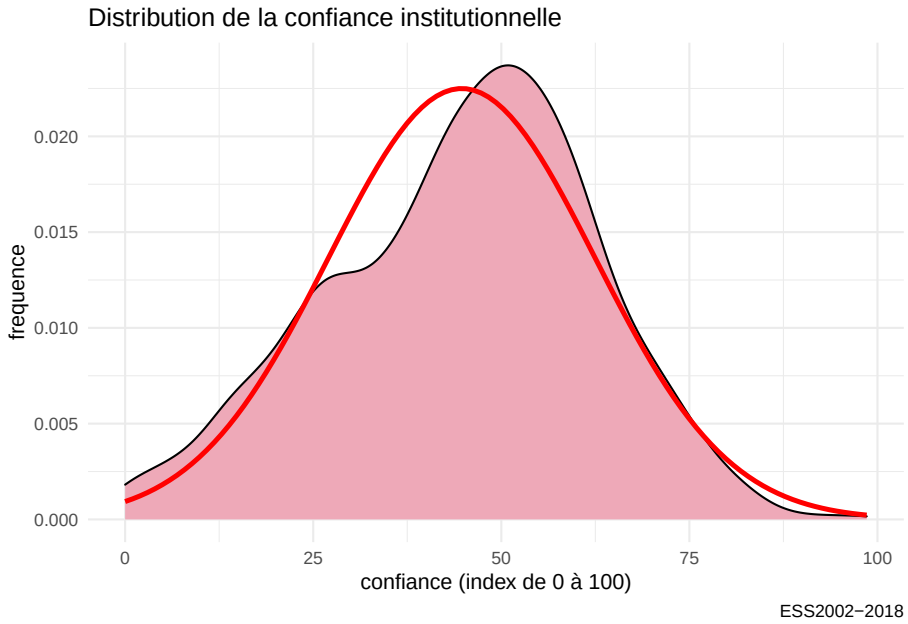
```
plot_grid(g04, g05, labels = c('A', 'B'), label_size = 12)
```



Enfin on peut examiner par rapport à une distribution théorique, en l'occurrence une distribution gaussienne, ou normale, de paramètres égaux à la moyenne et la variance empirique de la distribution. C'est ce que `stat_function` permet de réaliser.

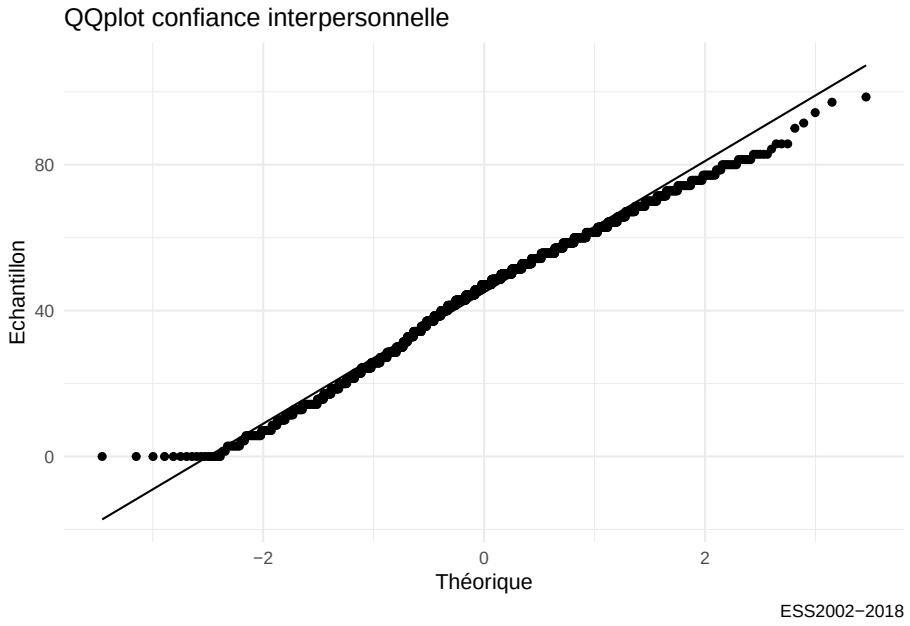
L'ajustement est convenable même si on observe une déviation sur la droite. C'est pourquoi on calcule aussi la Kurtosis et le skewness de la distribution.

```
#On a déjà calculé la moyenne : mean
#il nous manque l'écart-type et
sd<-sd(foo$trust_institut, na.rm=TRUE)
library(moments)
sk<-skewness(foo$trust_institut)
ks<-kurtosis(foo$trust_institut)


g05<-ggplot(foo,aes(x=trust_institut))+
  labs(title= "Distribution de la confiance institutionnelle", caption="ESS2002-2018",y
  geom_density(fill="pink2")+
  stat_function(fun = dnorm,color="red",size=1.2, args = list(mean =moy, sd=sd))

g05
```



Un grand classique du test de normalité d'une distribution est le diagramme QQ.

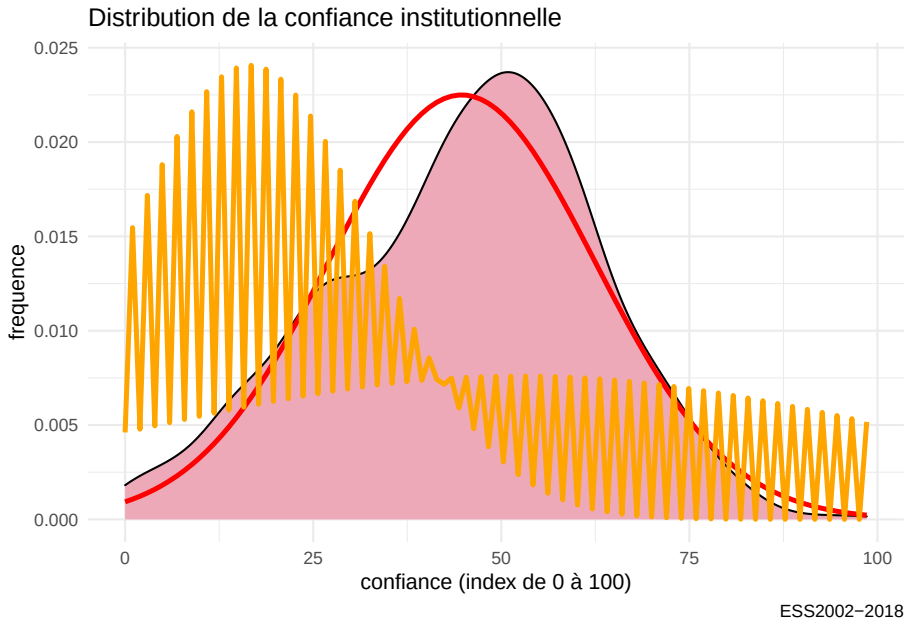
```
g06 <- ggplot(foo, aes(sample = trust_institut)) +
  stat_qq() + stat_qq_line()+
  labs(title= "QQplot confiance interpersonnelle",
       caption="ESS2002-2018",
       y= "Echantillon",x="Théorique")
g06
```



On finit cette étude détaillée par l'ajustement d'abord d'un modèle (loi normale) aux données. Ensuite d'un modèle de mélange (Mixture model) par lequel on définit la loi de distribution sous-jascente, comme un mélange entre deux populations normale de paramètres distincts.

https://tinyheero.github.io/2015/10/13/mixture-model.html

```
df0<-df %>%
  na.omit()
#library(MASS)
fit<-fitdistr(df0$trust_interpersonnel,"normal")
param<-as.data.frame(fit$estimate)
mean<-param[,1]
sd<-param[,1]

g07<- g05+stat_function(fun =  dnorm ,color="orange",size=1.2, args = list( mean=mean,
g07
```




```
library(mixtools)
trust = foo$trust_institut
mixmdl = normalmixEM(trust, k=2)
```

```
## number of iterations= 237
```

```
mixmdl$mu
```

```
## [1] 20.77247 50.86798
```

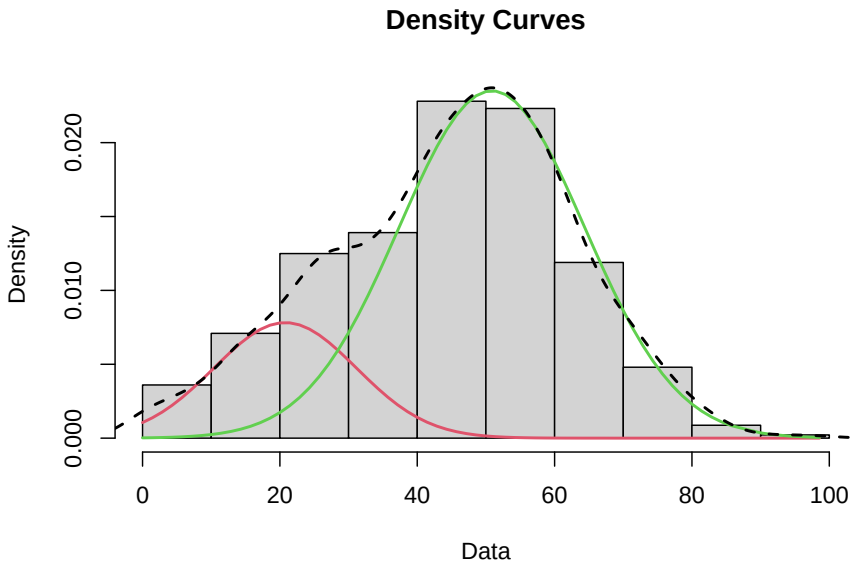
```
mixmdl$sigma
```

```
## [1] 10.37739 13.51802
```

```
mixmdl$lambda
```

```
## [1] 0.2034332 0.7965668
```

```
plot(mixmdl,which=2)
lines(density(trust), lty=2, lwd=2)
```

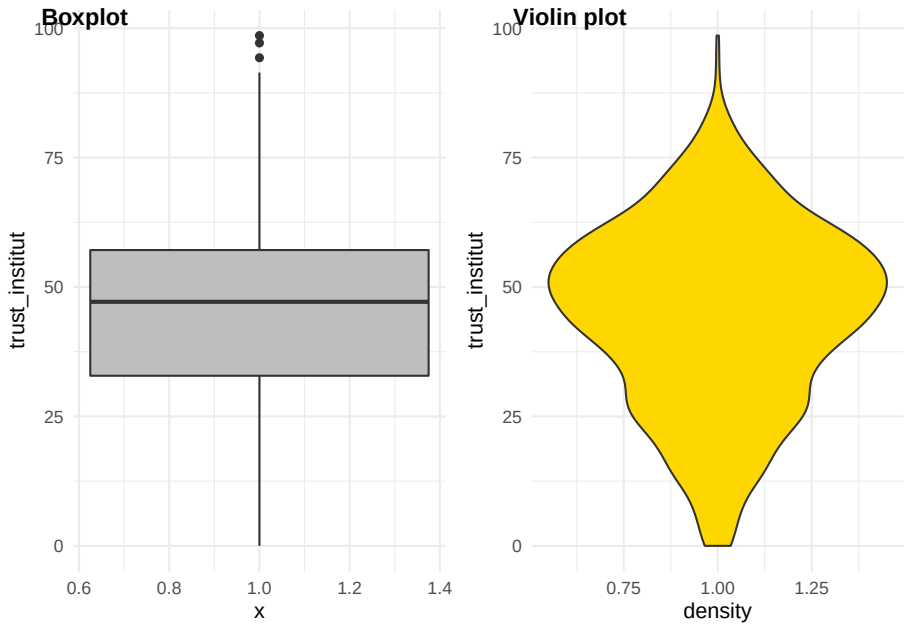


Finalement si notre distribution est univariée, car n'étudiant qu'une variable, on peut quand distinguer deux population distinctes.

4.1.5 D'autres méthodes

Il n'y a pas que l'histogramme ou le diagramme de densité, d'autres méthodes sont utiles, surtout quand on veut comparer des groupes (ce sera l'objet du prochain chapitre). Il s'agit du diagramme à moustache et du diagramme en violon.

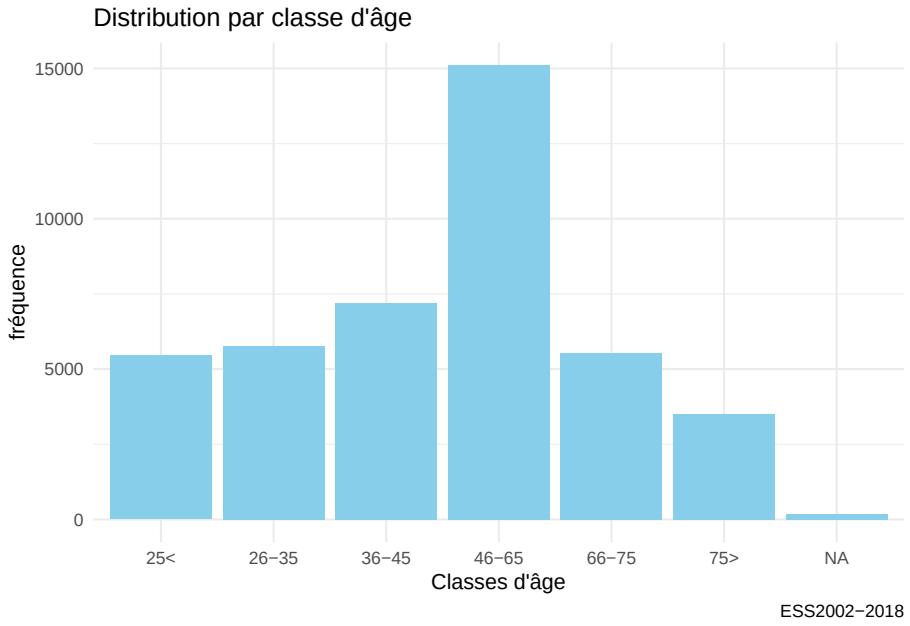
```
g0306 <- ggplot(foo, aes(y = trust_institut, x=1)) +
geom_boxplot(fill="Grey")

g0307 <- ggplot(foo, aes(x=1,y = trust_institut)) +
geom_violin(fill="Gold") + labs(x="density")

plot_grid(g0306, g0307, labels = c("Boxplot","Violin plot"),
  label_size = 12
)
```



4.1.6 Quand la variable est qualitative

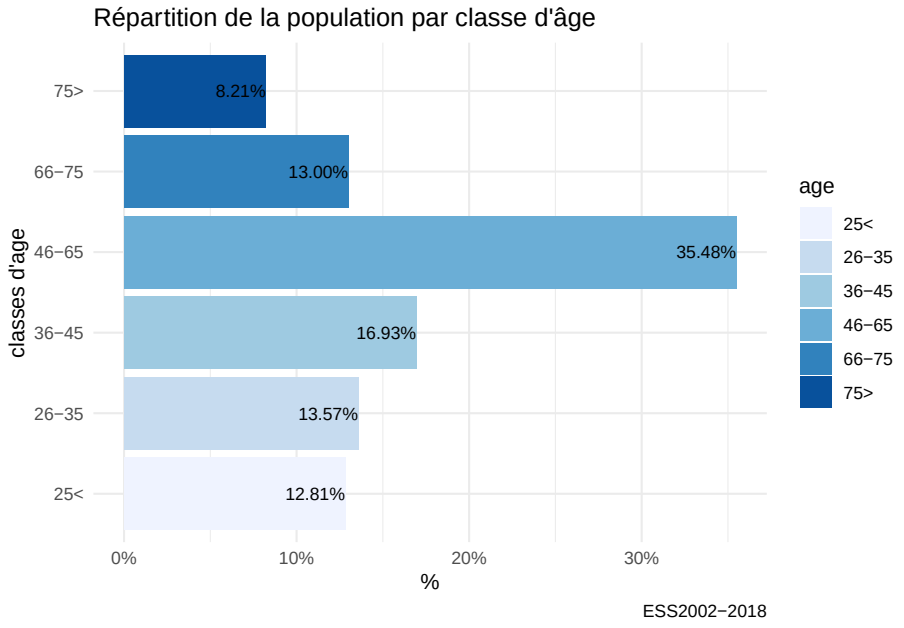
Quand la variable est qualitative, que ses modalités sont discrètes, la manière de représenter la plus commune est le fameux *camembert* que les experts abhorrent, n diagramme en barre représente mieux les proportions.

```
g08<-ggplot(df,aes(x=age))+
  geom_bar(fill="skyblue")+
  labs(title= "Distribution par classe d'âge",
       caption="ESS2002-2018",
       y= "fréquence",
       x="Classes d'âge")
g08
```

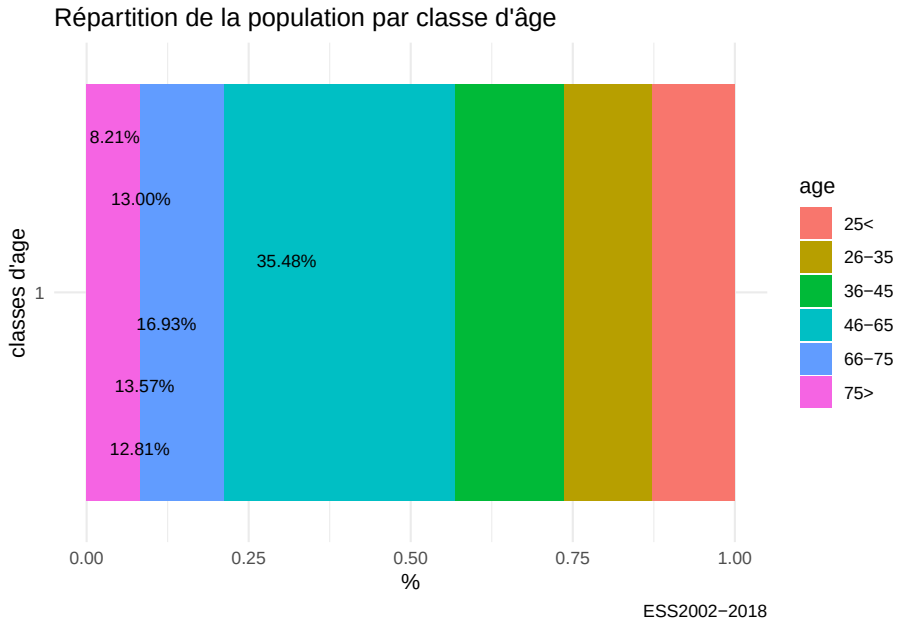


Avec quelques améliorations : contrôle de la couleurs des barres, ajout des % et pivot pour une meilleure lecture.

```
foo<-df %>%
  filter(!is.na(age))

g10<-ggplot(foo,aes(x=age, y = prop.table(stat(count)),label = scales::percent(prop.tab
  geom_bar(aes(fill = age)) +
  coord_flip()+
  labs(title= "Répartition de la population par classe d'âge", caption="ESS2002-2018",y
  scale_y_continuous(labels = scales::percent)+ #contrôle de l'échelle des % et du for
  scale_fill_brewer()+
  geom_text(stat = 'count',position = position_dodge(.9),  hjust = 1, size = 3)


g10
```

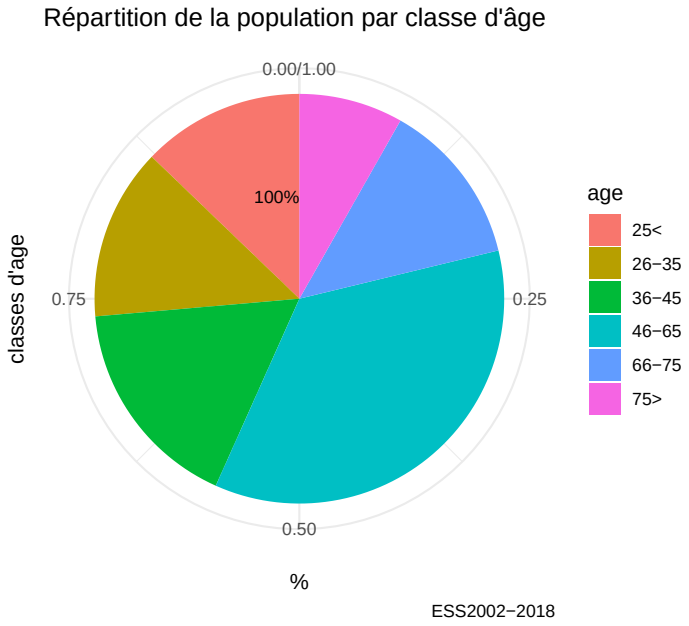



```
g11<-ggplot(foo,aes(x=factor(1),fill=age, y = prop.table(stat(count)),label =scales::percent(prop
  geom_bar(width=1) +
  coord_flip()+
  labs(title= "Répartition de la population par classe d'âge", caption="ESS2002-2018",y= "%",x="c
  geom_text(stat = 'count',position = position_dodge(.9),  hjust = 1, size = 3)

g11
```



si on tient au diagramme en secteur

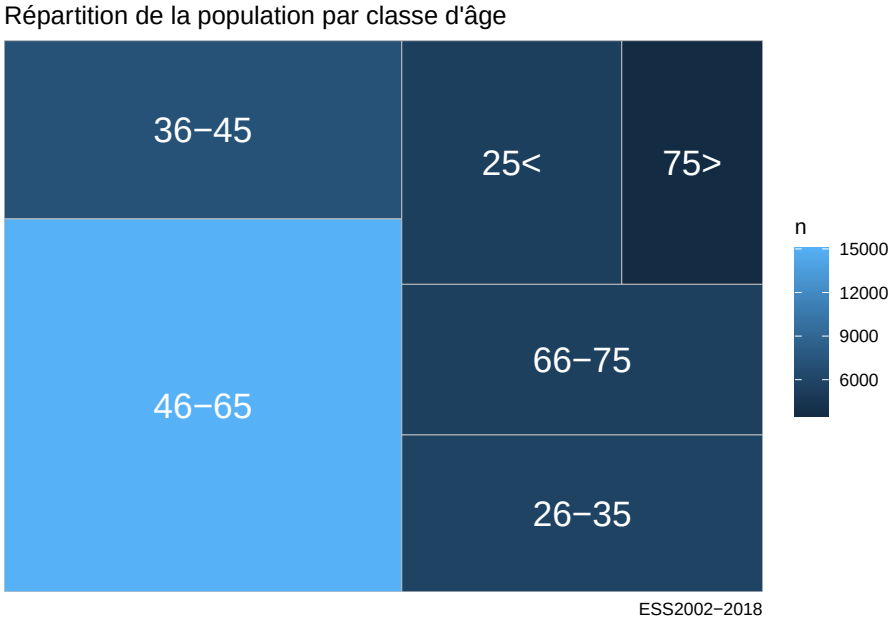
```
foo<-df %>%filter(!is.na(age))
g10<-ggplot(foo,aes(x="", y = prop.table(stat(count)),
                    label = scales::percent(prop.table(stat(count)))))+
  geom_bar(aes(fill = age)) +
  labs(title= "Répartition de la population par classe d'âge",
       caption="ESS2002-2018",y= "%",x="classes d'age") +
  geom_text(stat = 'count',position = position_dodge(.9),  hjust = 1, size = 3) +
  coord_polar("y", start=0)

g10
```



https://cran.r-project.org/web/packages/treemapify/vignettes/introduction-to-treemapify.html

si on tient au diagramme en cercle, autant opter pour un treemap avec la bibliothèque treemapifi

```
library(treemapify)
tree1<-df %>%
  mutate(n=1)%>%group_by(age) %>%
  summarize(n=sum(n)) %>%
  filter(!is.na(age))

g11 <- ggplot(tree1, aes(area = n, fill=n),label=age) +
  geom_treemap() +
  geom_treemap_text(aes(label=age),colour = "white", place = "centre",grow = FALSE)+
  labs(title= "Répartition de la population par classe d'âge", caption="ESS2002-2018",y= NULL,x=N

g11
```


Répartition de la population par classe d'âge
36−45
25<
75>
46−65
66−75
26−35
n
15000
12000
9000
6000
ESS2002−2018

Chapter 5

Analyse bi variée

Comme son nom l'indique, il s'agit d'examiner la relation entre deux variables et d'étudier leur distribution conjointe. On distinguera 3 situations et on examinera pour chaune les modes de représentations graphiques ainsi que les tests associés qui permette de s'assurer que la relation apparente est effective.

a) Deux variables quantitatives : scatterplot et corélations

b) deux variable qualitatives : tableau croisé et test du chi2

c) une variable quanti et une variable quali. Compariaons de moyennes et ANOVA

d) par comparer des distribution de plusieurs groupes (variables catégorielles)

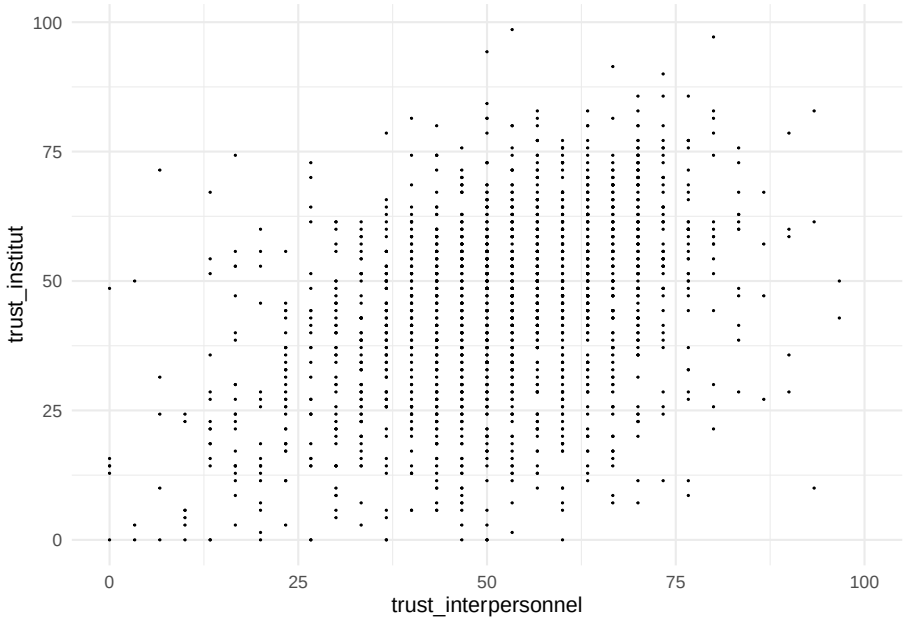
e) par comparer des moyennes d'une variable dépendante en fonction de plusieurs variables indépendantes catégorielle

f) mesurer l'association entre deux variables qualitatives

5.1 Diagrammes xy - la magie des corrélations

Venons en à analyser les relations entre deux variables quantitatives.

```
foo<-df %>%
  filter(cntry=="FR" & Year=="2018") #selection de l'echantillon

g31<- ggplot(foo, aes(x= trust_interpersonnel,y=trust_institut)) +
  geom_point( size=0.1)

g31
```



Ce graphe est peu clair, il y a trop de points qui prennent des valeurs discrètes. Une astuce est de donner une position aléatoire pour sur disperser, on fait mieux apparaitre la densité de points. On ajoute la représentation de deux courbe d'ajustement, l'une linéraire et l'autre non linéaires.

Mais en attendant en voici un calcul élémentaire.

le calcul de la variance

$$SS_{xx} = \sum (x - \bar{x})^2 = \sum x^2 - \frac{(\sum x)^2}{n}$$

le calcul de la covariance

$$SS_{xy} = \sum (x - \bar{x})(y - \bar{y}) = \sum xy - \frac{(\sum x)(\sum y)}{n}$$

et la corrélation qui est le rapport de la covariance sur la racine carrée du produit des variances de x et y.

$$r = \frac{SS_{xy}}{\sqrt{SS_{xx} SS_{yy}}}$$

La corrélation est de l'ordre d'un peu plus de 0,40 ce qui est assez élevé mais laisse une certaine indépendance des variables. Elle désignent des objets liés mais distinct. On peut tester l'hypothèse qu'en réalité cette corrélation est nulle.

Le test conduit au rejet de l'hypothèse nulle de manière très nette, compte-tenu de l'échantillon l'intervalle de confiance est compris entre 0.36 et 0.44.

```
#psych
r<-cor.test(foo$trust_interpersonnel, foo$trust_institut) #le test vient du package psych
r
```

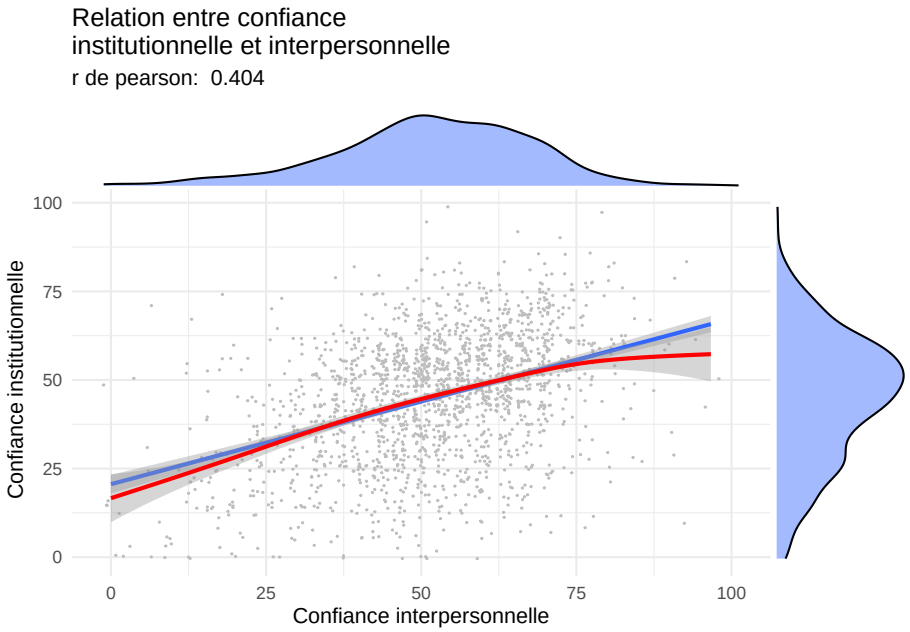
```
##
##  Pearson's product-moment correlation
##
## data:  foo$trust_interpersonnel and foo$trust_institut
## t = 18.861, df = 1821, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.3651235 0.4419644
## sample estimates:
##      cor
## 0.404257
```

```
rp<-round(r$estimate,3)
rp
```

```
##   cor
## 0.404
```

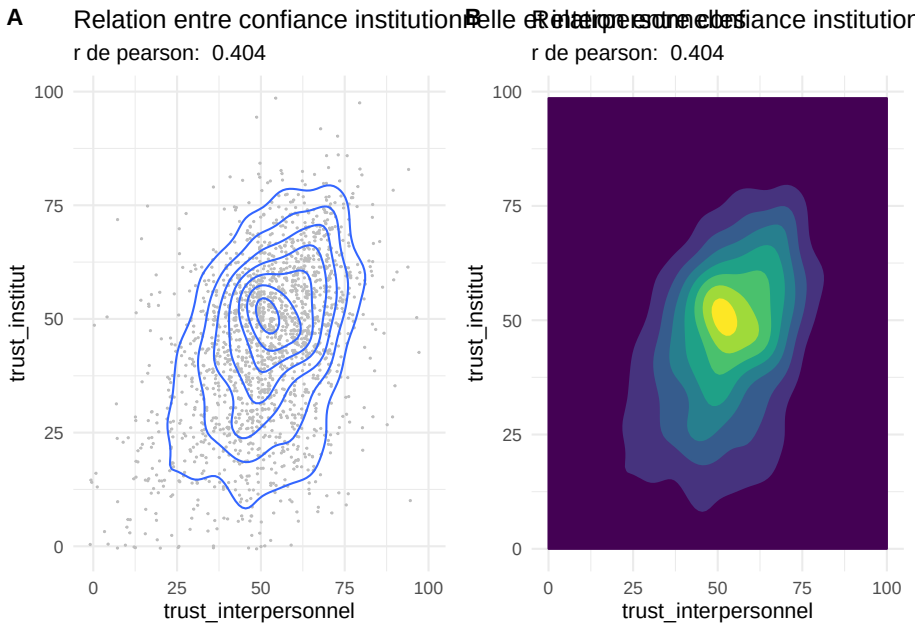
Améliorons le graphe On peut souhaiter ajouter une droite des moindre carrés (calculée pour chaque vague d'enquête pour évaluer la stabilité de la relation dans le temps). Les lignes sont parallèles, la corrélation ne change pas dans le temps, c'est une relation stable. Les deux formes de confiance vont dans le meme sens. On verra dans un autre chapitre comment calculer ces droites de corrélations.

```
library(ggExtra)
g32<-ggplot(foo, aes(x= trust_interpersonnel,y=trust_institut)) +
  geom_point(position = "jitter", size=0.1, color="grey")+
  geom_smooth(method="lm", se=TRUE) +
  geom_smooth(method="gam",color="red")     +
  labs(title = "Relation entre confiance \ninstitutionnelle et interpersonnelle",
       subtitle = paste("r de pearson: ",rp ),
       x= "Confiance interpersonnelle",
       y=" Confiance institutionnelle")

ggMarginal(g32  ,type = "density", fill = "Royalblue1", alpha=.5)
```



Une autre façon de représenter est celle de carte de densité de probabilité.

```
g32<-ggplot(foo, aes(x= trust_interpersonnel,y=trust_institut)) +
  geom_point(position = "jitter", size=0.1, color="grey")+geom_density2d()+
  labs(title = "Relation entre confiance institutionnelle et interpersonnelles", subti

g33<-ggplot(foo, aes(x= trust_interpersonnel,y=trust_institut)) +
  geom_density2d_filled(aes(fill = ..level.., color = ..level..),
    contour_var = "density")+
  labs(title = "Relation entre confiance institutionnelle et interpersonnelles", subti


plot_grid(g32, g33, labels = c('A', 'B'), label_size = 12)
```



5.2 Comparer les distributions et des moyennes

Dans notre base on a pris les données de l'Allemagne et de la France. On va comparer leur distribution. Et tant qu'à faire, puisque qu'on a deux variables, on va faire deux comparaisons : par pays et par type de confiance.

A cette fin, nous construisons un tableau de données spécifique.

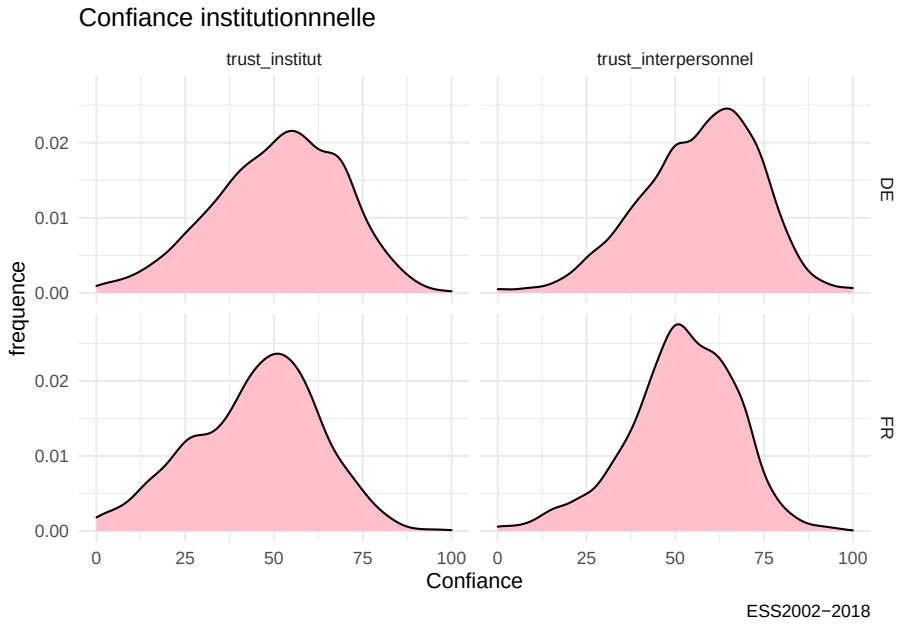
```
#on recode en facteur la variable

foo <- df %>%
  dplyr::select(cntry,trust_institut, Year,trust_interpersonnel) %>%
  filter( Year=="2018") %>%
 dplyr::select(-Year)%>%
 drop_na() %>%
  gather(variable, value, -cntry) #attention plutôt utiliser pivot_longer

head(foo)
```

```
## # A tibble: 6 x 3
##   cntry        variable       value
##   <chr+lbl>    <chr>          <dbl>
## 1 DE [Germany] trust_institut  58.6
```

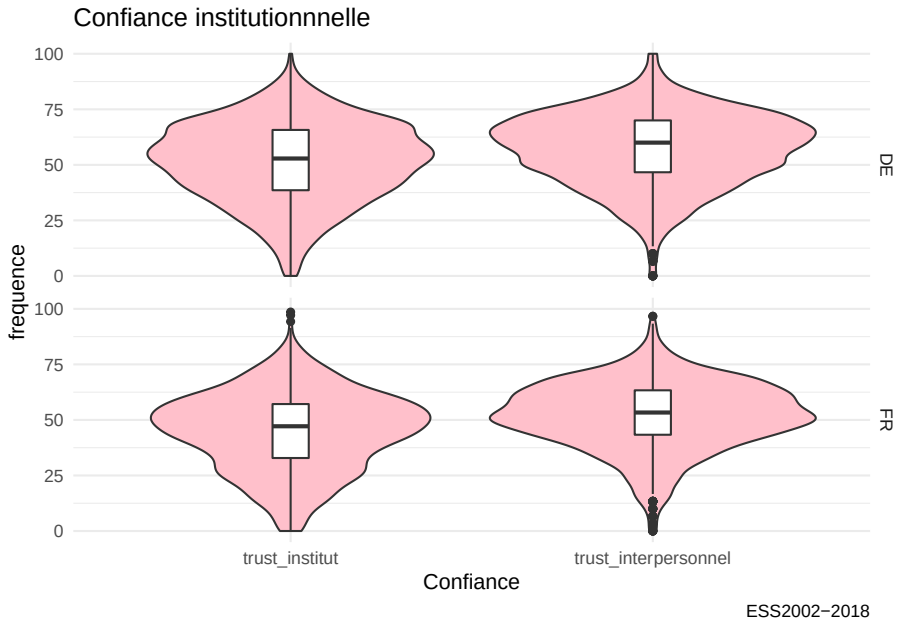
```
## 2 DE [Germany] trust_institut  65.7
## 3 DE [Germany] trust_institut  58.6
## 4 DE [Germany] trust_institut  65.7
## 5 DE [Germany] trust_institut  48.6
## 6 DE [Germany] trust_institut  37.1
```

Pour la représentation, en plus de la représentation en terme de densité, on va choisir une méthode de violon et de boxplot. On utilise une couche de "facetting" pour éclater ainsi la distribution des deux variables selon un critère de pays.

```
#on peut utiliser "facet"
g20<-ggplot(foo,aes(x=value))+ geom_density(binwidth=10, fill="pink")+ facet_grid(cntry
  labs(title= "Confiance institutionnnelle", caption="ESS2002-2018",y= "frequence",x="C
g20
```




```
g21<-ggplot(foo,aes(x=variable, y=value))+
  geom_violin( fill="pink") +
  geom_boxplot(width=0.1)+
  facet_grid(cntry~.)+
  labs(title= "Confiance institutionnnelle", caption="ESS2002-2018",y= "frequence",x="C
g21
```



5.2.1 Comparaison de moyennes

Comparer des distributions est une étape initiale nécesséaire, mais en général on sera plutôt intéresser de comparer des moyennes. Par exemple, on souhaiterais savoir si les degrés de confiances institutionnnelle et interpersonnelles varient en France selon les situations de revenu.

Calculons d'abord ces moyennes avec la fonction group_by et summarise.

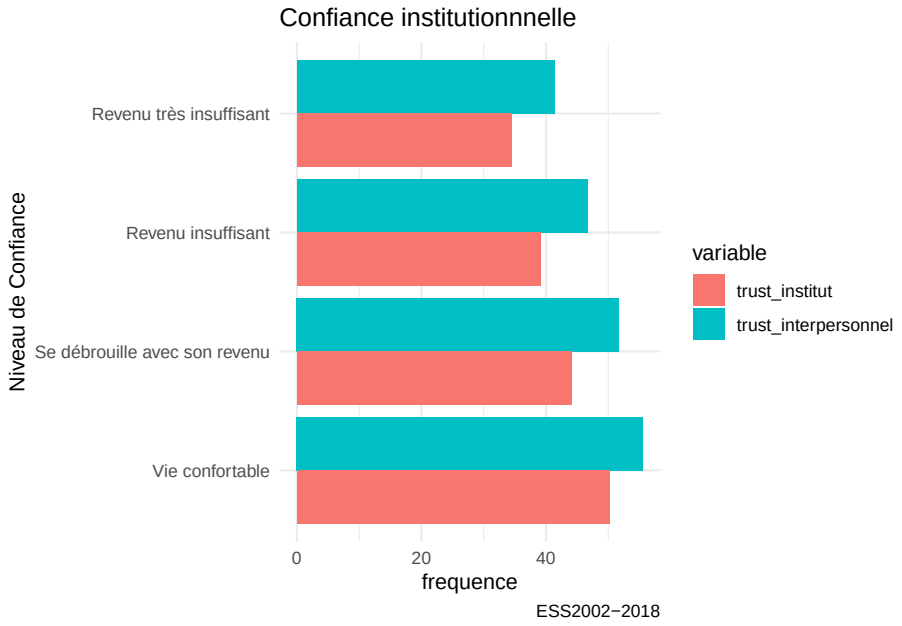
```
df_wave<-df %>% filter(cntry=="FR" & Year=="2018") %>%
  group_by(revenu) %>%
  summarise(trust_interpersonnel=mean(trust_interpersonnel, na.rm=TRUE),
            trust_institut =mean(trust_institut, na.rm=TRUE)) %>%
  filter(!is.na(revenu)) %>%                                    #filtrer les valeurs ma
  gather(variable, value, -revenu)                              #fichier long ( pivot l
head(df_wave)
```

```
## # A tibble: 6 x 3
##   revenu                          variable             value
##   <fct>                           <chr>                <dbl>
## 1 Vie confortable                 trust_interpersonnel  55.6
## 2 Se débrouille avec son revenu   trust_interpersonnel  51.7
## 3 Revenu insuffisant              trust_interpersonnel  46.7
## 4 Revenu très insuffisant         trust_interpersonnel  41.4
```

```
## 5 Vie confortable                trust_institut        50.2
## 6 Se débrouille avec son revenu trust_institut        44.1
```

Représentons ces moyennes graphiquement avec un geom_bar.

```
g06a<-ggplot(df_wave,aes(x=revenu,y=value, group=variable))+
  geom_bar(stat="identity",aes(fill=variable), position =position_dodge())+
  labs(title= "Confiance institutionnnelle", caption="ESS2002-2018",y= "frequence",x="
  coord_flip()

g06a
```



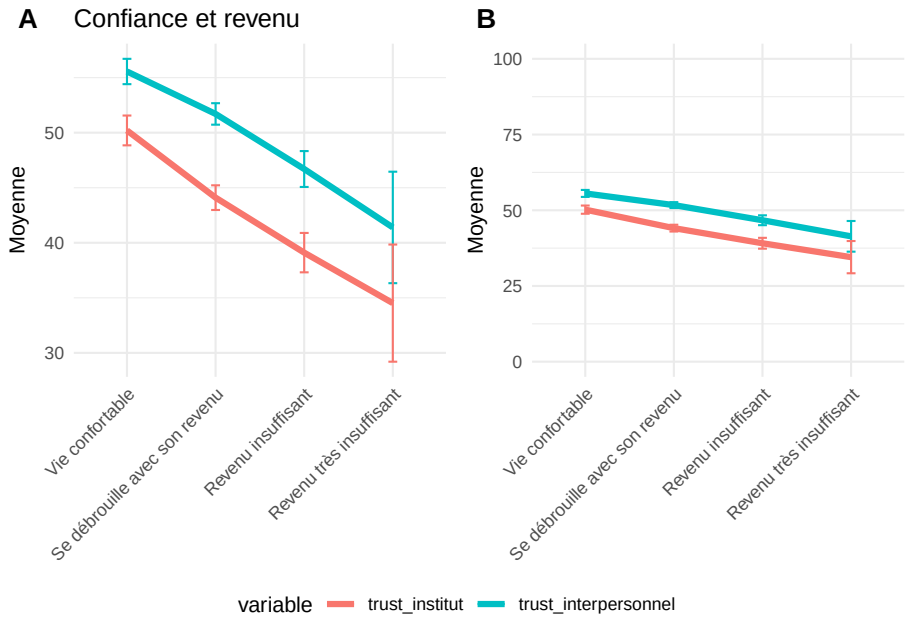
On a une solution mais pas la meilleure, on perd l'idée de variance et ce serait bien d'ajouter des barres d'intervalle de confiances , un diagramme en lignes serait plus élégant. On en profite pour corriger l'aspect des labels peu lisibles en les inclinants, et à choisir une échelle qui omettent les valeur supérieur à 70 et inférieure à 30 pour donner une vision plus respectueuses de la totalité de l'échelle qui va de 0 à 100.

Au passage on emploie à nouveau cowplot pour combiner les graphes, et ici plus précisément partager la légende des deux graphiques.

On observera que si le niveau de confiance diminue avec le revenu, la confiance interpersonnelle est plus forte, et de manière parallèle, à la confiance institutionnelle. On remarquera enfin que c'est pour les revenu les plus faibles que l'estimation est la plus imprécise ou la variance la plus grande.

```
df_wave2<-df %>%
  filter(cntry=="FR" & Year=="2018")%>%
  group_by(revenu) %>%
  mutate(n=1) %>%
  summarise(trust_interpersonnel_se=sd(trust_interpersonnel, na.rm=TRUE), #on calcule l'écartype
            trust_institut_se =sd(trust_institut, na.rm=TRUE),
            n=sum(n),
            trust_interpersonnel_se= 2*trust_interpersonnel_se/sqrt(n), # on calcule l'erreur typ
            trust_institut_se=2*trust_institut_se/sqrt(n)
            ) %>% dplyr::select(-n) %>%
  filter(!is.na(revenu)) %>%
  gather(variable, value, -revenu) %>% #on passe en format long
  dplyr::select(-revenu,-variable)%>%
  rename(se=value)

df_wave3<-cbind(df_wave,df_wave2) #on concatène les moyennes et les erreurs types

#on peut enfin produire le graphique

g06a<-ggplot(df_wave3,aes(x=revenu,y=value, group=variable))+
  geom_line(stat="identity",aes(color=variable), size=1.5)+
  geom_errorbar(aes(ymin=value-se, ymax=value+se, color=variable), width=.2,position=position_doc
  labs(title= "Confiance et revenu",y= "Moyenne",x=NULL)+
  theme(axis.text.x = element_text( angle=45, hjust =1)) #on controle l'angle et la position hori


g06b<-ggplot(df_wave3,aes(x=revenu,y=value, group=variable))+
  geom_line(stat="identity",aes(color=variable), size=1.5)+
  geom_errorbar(aes(ymin=value-se, ymax=value+se, color=variable), width=.2,position=position_doc
  ylim(0,100)+
  labs(title= "",y= "Moyenne",x=NULL)+
  theme(axis.text.x = element_text( angle=45, hjust =1)) #on controle l'angle et la position hori

prow <- plot_grid(
  g06a + theme(legend.position="none"),
  g06b + theme(legend.position="none"),
  align = 'vh',
  labels = c("A", "B", "C"),
  hjust = -1,
  nrow = 1
)
# extract a legend that is laid out horizontally
legend_b <- get_legend(
  g06a +
    guides(color = guide_legend(nrow = 1)) +
```

```
    theme(legend.position = "bottom")
)

# add the legend underneath the row we made earlier. Give it 10%
# of the height of one plot (via rel_heights).
plot_grid(prow, legend_b, ncol = 1, rel_heights = c(1, .1))
```



La visualisation est utile, encore faut-il qu'on soit bien certain que les variations ne soit pas le produit du hasard, des fluctuations d'échantillonnage. Si en moyenne la perception du pouvoir d'achat est associée à des moyennes de confiance décroissantes, les différences observées sont-elle significatives? Dans les représentations précédentes c'est le choix de l'échelle qui oriente l'analyse.

On a un besoin d'un test plus objectif. Celui est le très classique test d'analyse de variance (ANOVA).

Celui-çi est le test d'analyse de variance qui consiste à comparer la variance à l'intérieur des groupes (intra), et la variance entre les moyennes des groupes (inter ou between).

On note qu'ici on introduit la méthode flextable pour présenter des tableaux au formats scientifique. L'astuce ici est d'utiliser aussi xtable.

```
foo<-df %>%
  filter(cntry=="FR" & Year=="2018") %>%
  drop_na() #selection des données
```

```
fit<-lm(trust_institut~revenu, foo) #calcul du modèle linéaire

anova(fit) #test d'analyse de variance
```

```
## Analysis of Variance Table
##
## Response: trust_institut
##             Df Sum Sq Mean Sq F value    Pr(>F)
## revenu       3  27651  9217.1  32.052 < 2.2e-16 ***
## Residuals 1686 484846   287.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
library(xtable) #xtable transforme en table certains type d'objet dont les résultats de l'anova
ft <- xtable_to_flextable(xtable(anova(fit)), hline.after = c(0,2)) #la fonction permet d'exploi
ft
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
revenu	3	27,651.4	9,217.1	32.1	0.0
Residuals	1,686	484,845.7	287.6		

5.2.2 Deux variables qualitatives

L'étude de la relation éventuelle entre deux variables qualitative s'apprécie traditionnellement par une méthode de tableau croisé.

5.2.2.1 Tableau croisé

Pour calculer le tableau croisé on utilise la fonction très simple `table` et la fonction `prop.table`

```
t<-table(foo$revenu,foo$habitat)
t
```

```
##
##                                Big city Suburbs Town Village Countryside
##   Vie confortable                   118      82  161     142          31
##   Se débrouille avec son revenu     120     109  275     227          58
##   Revenu insuffisant                 48      38  129      88          22
##   Revenu très insuffisant             9       5   18      10           0
```

```
prop.table(t,2)
```

```
##
##                                    Big city    Suburbs       Town    Village
##   Vie confortable                0.40000000 0.35042735 0.27615780 0.30406852
##   Se débrouille avec son revenu  0.40677966 0.46581197 0.47169811 0.48608137
##   Revenu insuffisant             0.16271186 0.16239316 0.22126930 0.18843683
##   Revenu très insuffisant        0.03050847 0.02136752 0.03087479 0.02141328
##
##                                   Countryside
##   Vie confortable                  0.27927928
##   Se débrouille avec son revenu    0.52252252
##   Revenu insuffisant               0.19819820
##   Revenu très insuffisant          0.00000000
```

Mais ce n'est pas esthétique, avec la fonction `proc_freq` de flextable on obtient une meilleure présentation. Elle nous donne en peu de mots les effectif par cellule, les pourcentages en lignes, et en colonnes.

```
ft1<- proc_freq(foo, "revenu", "habitat", include.table_percent = FALSE,
                include.row_percent = FALSE, include.column_total = FALSE,
  include.column_percent = TRUE)
ft1
```

revenu	label	Big city	Suburbs
Vie confortable	Frequency	118	82
	Col Pct	40%	35.04%
Se débrouille avec son revenu	Frequency	120	109
	Col Pct	40.68%	46.58%
Revenu insuffisant	Frequency	48	38
	Col Pct	16.27%	16.24%
Revenu très insuffisant	Frequency	9	5
	Col Pct	3.05%	2.14%

```
ft2<- proc_freq(foo, "revenu", "habitat", include.table_percent = FALSE,
                include.row_percent = TRUE,
  include.column_percent = FALSE)
ft2
```

hab

revenu	label	Big city	Suburbs	Town
Vie confortable	Frequency	118	82	161
	Row Pct	22.1%	15.36%	30.15%
Se débrouille avec son revenu	Frequency	120	109	275
	Row Pct	15.21%	13.81%	34.85%
Revenu insuffisant	Frequency	48	38	129
	Row Pct	14.77%	11.69%	39.69%
Revenu très insuffisant	Frequency	9	5	18
	Row Pct	21.43%	11.9%	42.86%
Total	Frequency	295	234	583

5.2.2.2 le valeureux chi²

Le test du chi2 s'appuie sur une idée très simple qui de fait est un théorème : Si deux variables X et Y sont indépendantes, la fréquence de leur combinaison est le produit des fréquences marginales.

On peut donc sur cette base, calculer l'effectif attendu (expected frequency) puis le comparer à ce qu'on a observé pour chacune des cellules du tableau. On somme enfin ces écarts.

$$\chi^2 = \sum \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

Naturellement , une même valeur de cette quantité pour un petit tableau(2x2) n'a pas la même signification que si le tableau est grand(par ex 20x 10). On l'appréciera donc en fonction des degrés de liberté (n-1 x m-1).

Le test proprement dit consiste à examiner quelles sont les chances qu'on obtienne la valeur du chi2 calculé, pour un nombre de degré de liberté donné. Si cette probabilité est faible on rejetera l'hypothèse d'indépendance des deux variables.

Avec r la fonction chsq.test nous simplifie

```
chi2<-chisq.test(t)
chi2
```

```
##
```

```
##  Pearson's Chi-squared test
##
## data:  t
## X-squared = 23.853, df = 12, p-value = 0.0213
```

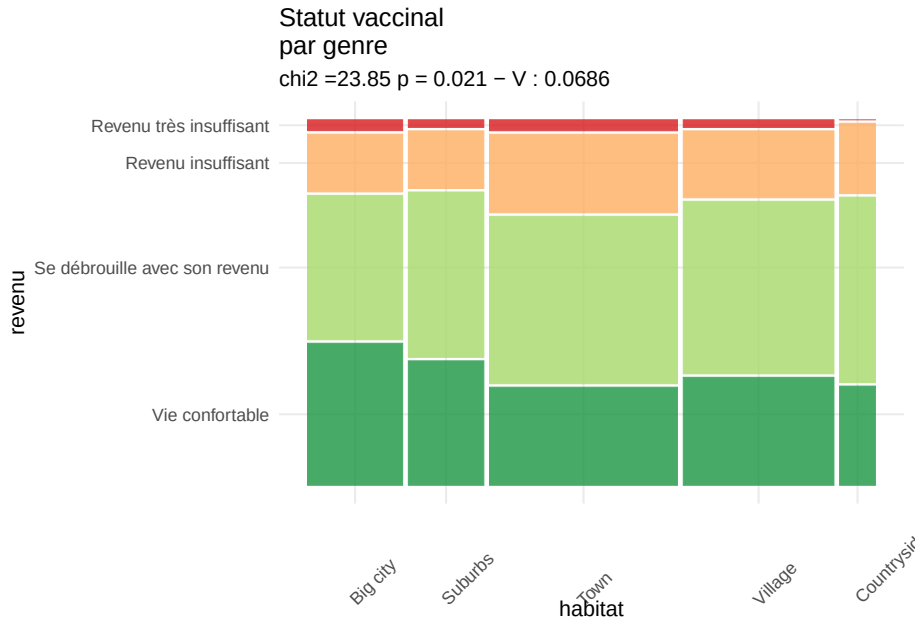
L'objet chi2 est une liste

```
# On isole les éléments qui nous intéresse

#library()
chi<-round(chi2$statistic,2)
p<-round(chi2$p.value,3)
#V<-cramerV(t, digit=3) ( le package n'est plus maintenu)
```

5.2.2.3 diagramme en mosaique

Le diagramme en mosaïque détermine la largeur des barres en fonction de l'effectif de la variable en abscisse et leur hauteur en fonction de la variable en ordonnée. Les couleurs permettent de mieux comparer. On s'aperçoit ici que les plus à l'aise avec leur revenu sont proportionnellement plus nombreux dans les grandes villes, et que ceux qui se débrouille sont plus fréquents dans les campagnes.

```
library(ggmosaic)
g1 <- ggplot(data = foo) +
  geom_mosaic(aes(x=product( revenu ,habitat), fill = revenu))+
  theme(axis.text.x = element_text(angle = 45, hjust = -0.1, vjust = -0.2))+
  theme(legend.position = "none")+
  labs(title="Statut vaccinal \npar genre",
       subtitle=paste0("chi2 =",chi, " p = ", p, " - V : ", V))+
  scale_fill_brewer(palette = "RdYlGn", direction = -1)

g1
```



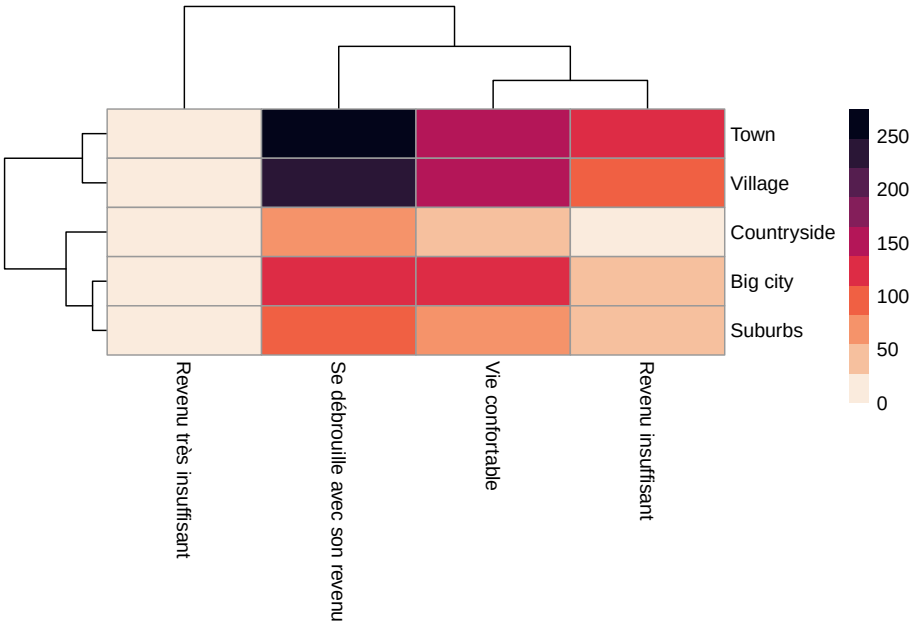
5.2.2.4 les chi2s partiels et des cartes de chaleur.

Une carte de chaleur représente une grandeur par un gradient de couleur pour chaque cellule définie par des variable x et y.

Faisons un premier essai pour représenter les effectifs, plutôt qu'avoir un tableau de nombres on va obtenir un tableau de couleurs.

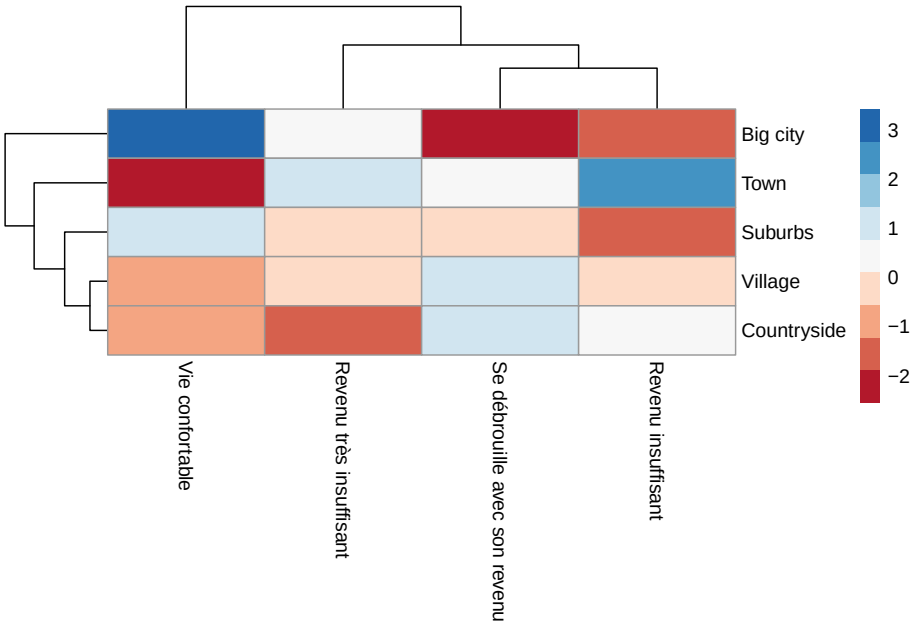
L'arbre qui apparait en ligne et en colonne correspond au résultat d'une classification hiérarchique que nous développons dans le chapitre X.

```
library(pheatmap)
library(viridis)

table2<-as.data.frame(t) %>%
  pivot_wider(names_from = Var1, values_from = Freq) %>%
  column_to_rownames( var = "Var2")
pheatmap(table2 , color = rocket(10,direction =-1))
```



On utilise la même technique mais en représenant une grandeur différentes : les tests du chi2 partiels, pour apprécier les sous ou les sur-représentation.

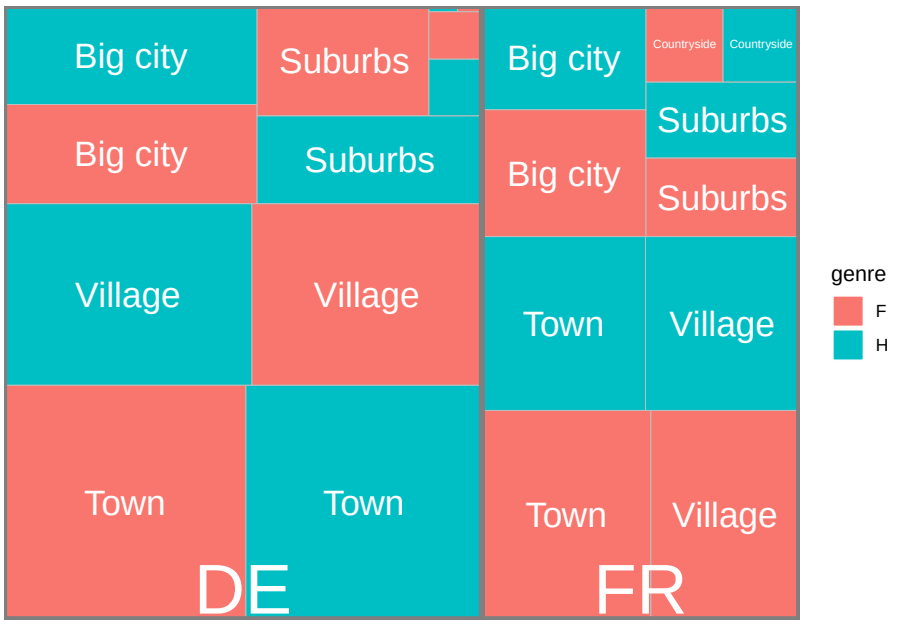
```
chi2df<- as.data.frame(chi2$stdres)

table2<-chi2df %>%
  pivot_wider(names_from = Var1, values_from = Freq) %>%
  column_to_rownames( var = "Var2")
pheatmap(table2 , color = brewer.pal(n = 9, name = "RdBu"))
```



5.2.2.5 Les treemaps, c'est merveilleux

D'autre graphiques et des emboîtements

```
library(treemapify)
tree1<-df %>% mutate(n=1)%>%group_by(cntry,genre,habitat) %>% summarize(n=sum(n),mean=mean(trust_

g10 <- ggplot(tree1, aes(area = n, fill=genre,subgroup=cntry)) +
  geom_treemap() +
  geom_treemap_text(aes(label=habitat),colour = "white", place = "centre",grow = FALSE)+
      geom_treemap_subgroup_text(color="white",grow = FALSE)+
  geom_treemap_subgroup_border()
g10
```


Big city
Suburbs
Big city
Suburbs
Village
Village
Town
Town
DE
Big city
Countryside
Countryside
Suburbs
Big city
Suburbs
Town
Village
Town
Village
FR
genre
F
H

Chapter 6

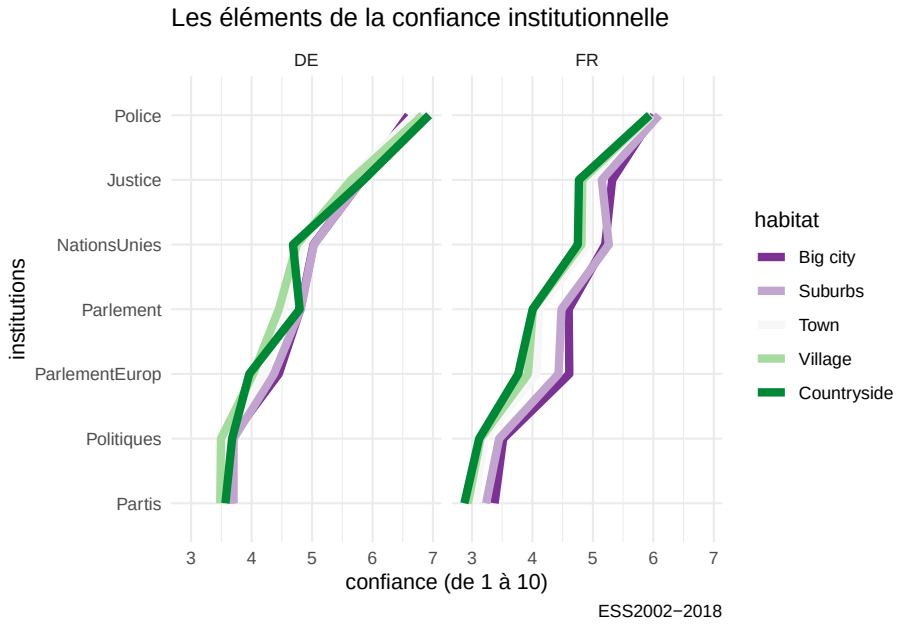
Analyse graphique multivariée

Dans ce chapitre, on généralise à des ensembles de variables.

6.1 La confiance institutionnelle, en détail

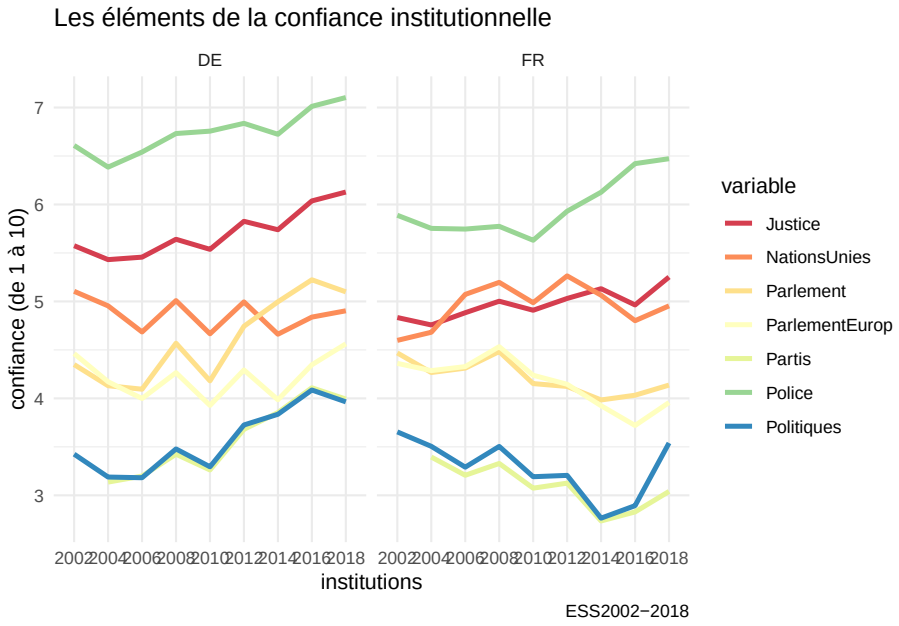
On veut reprénter 6 variables, correspondant à 5 types d'habitats et 2 pays.

```
df<-readRDS("./data/dfTrust.rds)")


rad<-df %>%
  group_by (habitat,cntry) %>%
  summarize(Partis=mean(Partis, na.rm=TRUE),
  Parlement=mean(Parlement, na.rm=TRUE),
  Politiques=mean(Politiques, na.rm=TRUE),
  Police=mean(Police, na.rm=TRUE),
  Justice=mean(Justice, na.rm=TRUE),
  NationsUnies=mean(NationsUnies, na.rm=TRUE),
  ParlementEurop=mean(ParlementEurop, na.rm=TRUE)) %>%
  filter(!is.na(habitat)) %>%
  gather(variable, value, -habitat, -cntry)

ggplot(rad, aes(x=reorder(variable, value),y=value, group=habitat))+
  geom_line(aes(color=habitat), size=2)+
  facet_grid(.~cntry) +coord_flip()+
  scale_color_brewer(type="div",palette=3)+labs(title= "Les éléments de la confiance institutionn
```



Une autre variante qui donne l'évolution de l'évolution de les éléments de la confiance institutionnelle

```
rad<-df %>%
  group_by (Year,cntry) %>%
  summarize(Partis=mean(Partis, na.rm=TRUE),
  Parlement=mean(Parlement, na.rm=TRUE),
  Politiques=mean(Politiques, na.rm=TRUE),
  Police=mean(Police, na.rm=TRUE),
  Justice=mean(Justice, na.rm=TRUE),
  NationsUnies=mean(NationsUnies, na.rm=TRUE),
  ParlementEurop=mean(ParlementEurop, na.rm=TRUE)) %>%
  gather(variable, value, -Year, -cntry)

ggplot(rad, aes(x=Year,y=value, group=variable))+
  geom_line(aes(color=variable), size=1.2)+
  facet_wrap(.~cntry, nrow=1) +
  scale_color_brewer(palette="Spectral")+labs(title= "Les éléments de la confiance inst
```

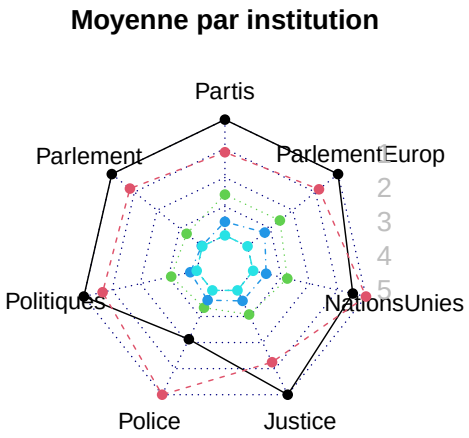


La différence entre les deux pays est claire, la rupture est accusée plus fortement en France qu'en Allemagne. L'explication n'est sans doute pas culturelle mais démographique, un coup d'oeil à la carte des densité permet de comprendre mieux : https://www.populationdata.net/cartes/allemagne-france-densite-de-population-2011/.

On pourra tenté un graphe en radar. Mais il n'est pas si convaincant.

```
library(fmsb)

rad<-df %>% filter(cntry=="FR") %>%
  group_by (habitat) %>%
  summarize(Partis=mean(Partis, na.rm=TRUE),
  Parlement=mean(Parlement, na.rm=TRUE),
  Politiques=mean(Politiques, na.rm=TRUE),
  Police=mean(Police, na.rm=TRUE),
  Justice=mean(Justice, na.rm=TRUE),
  NationsUnies=mean(NationsUnies, na.rm=TRUE),
  ParlementEurop=mean(ParlementEurop, na.rm=TRUE)) %>%
  filter(!is.na(habitat)) %>%
  dplyr::select(-habitat)

#on doit indiquer les valeurs minimale et maximale - la fonction rep permet de repeter (ici 7 fo
data <- rbind(rep(7,7) , rep(3,7) , rad)
#l'autre method c'est ce choisir maxmin=FALSE
```

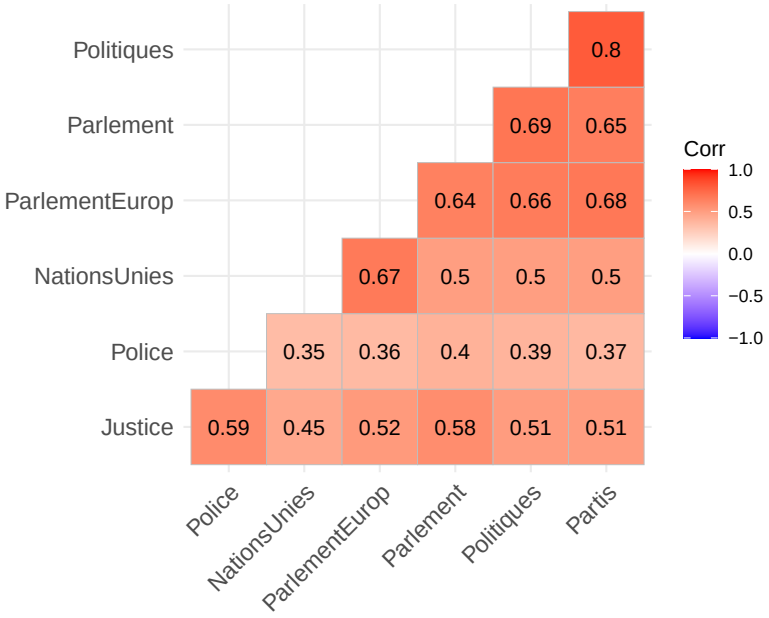
```
#rownames(rad) <- c("big city", "suburbs" ,"town","village", "countryside")
radarchart(rad, axistype=0, seg=4, title="Moyenne par institution", maxmin=FALSE)
legend(x=0.7, y=1, legend = rownames(rad), bty = "n",text.col = "grey", cex=1.2, pt.cex
```



6.2 Table de corrélation

Comparer les moyennes est une chose, on souhaiter en plus savoir quelle structure de corrélation les caractérisent. Rien de plus simple

```
library(ggcorrplot)
df<-readRDS("./data/dfTrust.rds)")%>%filter(Year==2018)

foo<-df %>% dplyr::select(NationsUnies,ParlementEurop, Parlement, Justice, Police, Poli
  drop_na()
r<-cor(foo)

ggcorrplot(r, hc.order = TRUE, type = "lower",
   lab = TRUE)
```




```
g<-paste0("./plot/g1",".jpg")
ggsave(g,plot=last_plot(), width = 27, height = 19, units = "cm")
```

6.3 Un cas plus complexe : présidentielle2020

Nsppolls cumulent les sondages publiés des grands instituts. On utilise ces données , ainsi qu'une boucle, pour explorer différents paramètre d'un modèle de lissage.

Le but : mieux percevoir les tendance par une sorte de méta-analyse des différents sondages :

6.4 une boucle pour produire de multiple graphe en variant un paramètre

```
library(lubridate)
alph<-.5

for (alph in seq(from=0, to= 1, by=.05)){
df_pol <- read_delim("https://raw.githubusercontent.com/nsppolls/nsppolls/master/presidentielle.c
```

```
                      delim = ",", escape_double = FALSE, trim_ws = TRUE)%>%
  filter(tour=="Premier tour") %>%filter(candidat=="Eric Zemmour"|
                                           candidat== "Marine Le Pen"|
                                           candidat== "Emmanuel Macron"|
                                           candidat== "Jean-Luc Mélenchon"|
                                           candidat== "Yannick Jadot"|
                                           candidat== "Valérie Pécresse"|
                                           candidat=="Fabien Roussel"|
                                           candidat=="Anne Hidalgo") %>%
  filter(fin_enquete>ymd("2022-01-09")) # on commence en septembre , octobre est-il me


table(df_pol$candidat)
SensiP1<-c("pink", "orange", "gray20", "red","firebrick", "Royalblue", " skyblue", "Cha

ggplot(df_pol, aes(y=intentions, x=fin_enquete))+
  geom_point(aes(color=candidat), size=.5, alpha=1-alph)+
  geom_smooth(span = alph, aes(col=candidat,fill=candidat), alpha=0.2)+
  scale_color_manual(values=SensiP1)+
  scale_fill_manual(values=SensiP1)+
  labs(title= "Evolution des intentions de vote #présidentielle2022 1er tour",
       subtitle =paste("Lissage méthode loess. alpha=",alph, " - ci=95%"),
       caption = "data @nsppolls viz @benavent",
       x=NULL)+theme_minimal()+scale_x_date(date_breaks = "1 month", date_minor_breaks
             date_labels = "%B")

sondage_nsppolls<-paste0("./nsppolls/sondage_nsppolls", alph*20, ".jpg")
ggsave(sondage_nsppolls,plot=last_plot(), width = 27, height = 19, units = "cm")

}


n<-df_pol%>%
  mutate(n=1)%>%
  group_by(id)%>%summarise(n=sum(n))
#nombre de sondage
n<-nrow(n)
```

Pour créer le gif on emplie magick. On a pris soin de sauvegarder les graphes dans un répertoire propre, ça facilite la lecture en boucle et la fabrication du gif.

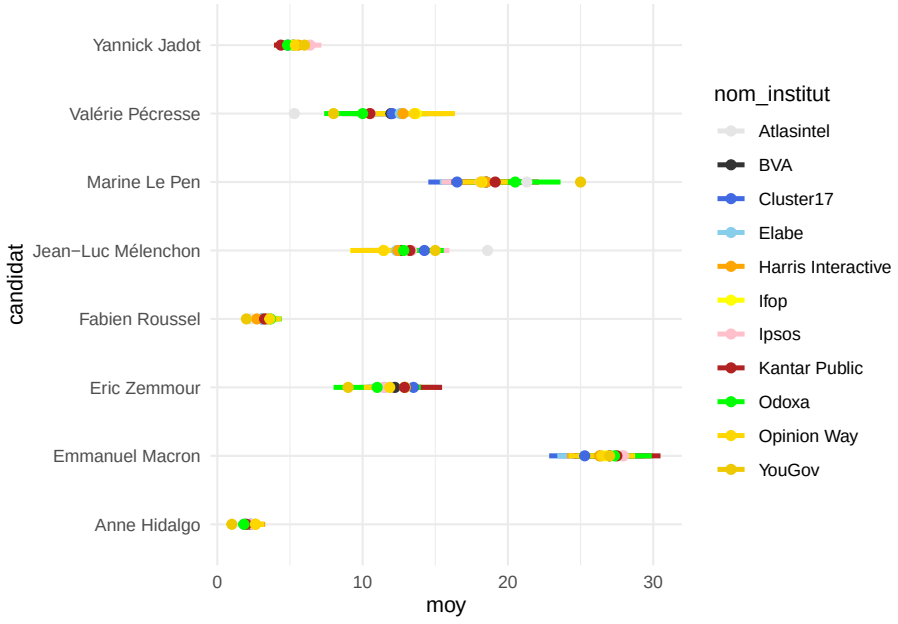
```
library(magick)

#gif
```

```
#on constitue une liste des noms des fichier *.jpg que l'on veut associer
frames <- paste0("./nsppolls/","sondage_nsppolls", 0:20,".jpg")

#on lit et on stoke dans m les images
m <- image_read(frames)

#on fabrique et on sauvergarde le gif
m <- image_animate(m, fps=1)
image_write(m, "./plot/sondages_lissage.gif")
```

6.4.1 effet sondeur

pour anticiper sur le chapitre suivant

```
foo<-df_pol%>%
  dplyr::select(candidat, intentions, fin_enquete, echantillon,nom_institut)%>%
  group_by(nom_institut, candidat)%>%
  summarise(moy=mean(intentions, na.rm=TRUE),
            std=sd(intentions, na.rm=TRUE))


SensiP2<-c("gray90","gray20", "Royalblue", "skyblue", "orange", "yellow", "pink", "firebrick", "g


g<-ggplot(foo,aes(x=candidat,y=moy))+
  geom_segment(aes(x = candidat,
                   y = -std+moy,
                   xend = candidat,
                   yend = std+moy,
                   color = nom_institut), size=1.2)+
    geom_point(aes(color=nom_institut), size=2)+
  scale_color_manual(values = SensiP2)+
  theme_minimal()+
  coord_flip()
g
```



6.5 Modéliser le biais du sondeur

http://www.stat.yale.edu/Courses/1997-98/101/anovareg.htm

```
df_pol$tps<-2
df_pol$tps[df_pol$fin_enquete < ymd("2022-01-31")]<-1

df_pol$tps[df_pol$fin_enquete > ymd("2022-03-01")]<-3

df_pol$tps<- as.factor(df_pol$tps)

fit1<- lm(intentions~candidat*tps,data=df_pol)
anova(fit1)
```

```
## Analysis of Variance Table
##
## Response: intentions
##                Df Sum Sq Mean Sq    F value    Pr(>F)
## candidat        7 122735 17533.6 10705.9435 < 2.2e-16 ***
## tps             2     25    12.7     7.7658 0.0004363 ***
## candidat:tps   14   4534   323.9   197.7554 < 2.2e-16 ***
## Residuals    2096   3433     1.6
## ---
```

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

```
fit2<- lm(intentions~candidat*tps+candidat*nom_institut,data=df_pol)
anova(fit2)
```

```
## Analysis of Variance Table
##
## Response: intentions
##                          Df Sum Sq Mean Sq    F value    Pr(>F)
## candidat                  7 122735 17533.6 12742.2074 < 2.2e-16 ***
## tps                       2     25    12.7     9.2428  0.000101 ***
## nom_institut             10     25     2.5     1.8283  0.051096 .
## candidat:tps             14   4534   323.9   235.3684 < 2.2e-16 ***
## candidat:nom_institut    70    633     9.0     6.5768 < 2.2e-16 ***
## Residuals              2016   2774     1.4
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(fit1,fit2)
```

```
## Analysis of Variance Table
##
## Model 1: intentions ~ candidat * tps
## Model 2: intentions ~ candidat * tps + candidat * nom_institut
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1   2096 3432.7
## 2   2016 2774.1 80    658.65 5.9832 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(fit1)
```

```
##
## Call:
## lm(formula = intentions ~ candidat * tps, data = df_pol)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.2585 -0.6350 -0.0278  0.6021  5.6493
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      3.1765     0.1792  17.726  < 2e-16 ***
## candidatEmmanuel Macron         21.5392     0.2534  84.992  < 2e-16 ***
```

```
## candidatEric Zemmour              9.7549     0.2534  38.492  < 2e-16 ***
## candidatFabien Roussel           -0.8039     0.2534  -3.172 0.001535 **
## candidatJean-Luc Mélenchon        6.4118     0.2534  25.300  < 2e-16 ***
## candidatMarine Le Pen            13.7353     0.2534  54.198  < 2e-16 ***
## candidatValérie Pécresse         13.2255     0.2534  52.187  < 2e-16 ***
## candidatYannick Jadot             2.7255     0.2534  10.755  < 2e-16 ***
## tps2                             -0.6765     0.2342  -2.888 0.003915 **
## tps3                             -0.9765     0.2089  -4.674 3.14e-06 ***
## candidatEmmanuel Macron:tps2      0.6844     0.3312   2.066 0.038934 *
## candidatEric Zemmour:tps2         2.0645     0.3312   6.233 5.52e-10 ***
## candidatFabien Roussel:tps2       2.1511     0.3312   6.494 1.04e-10 ***
## candidatJean-Luc Mélenchon:tps2   1.6438     0.3312   4.963 7.52e-07 ***
## candidatMarine Le Pen:tps2        0.5842     0.3312   1.764 0.077955 .
## candidatValérie Pécresse:tps2    -0.9616     0.3312  -2.903 0.003734 **
## candidatYannick Jadot:tps2       -0.1630     0.3312  -0.492 0.622725
## candidatEmmanuel Macron:tps3      4.6819     0.2955  15.847  < 2e-16 ***
## candidatEric Zemmour:tps3        -1.0570     0.2955  -3.578 0.000355 ***
## candidatFabien Roussel:tps3       2.0919     0.2955   7.080 1.95e-12 ***
## candidatJean-Luc Mélenchon:tps3   5.1319     0.2955  17.370  < 2e-16 ***
## candidatMarine Le Pen:tps3        3.4154     0.2955  11.560  < 2e-16 ***
## candidatValérie Pécresse:tps3    -4.8670     0.2955 -16.473  < 2e-16 ***
## candidatYannick Jadot:tps3        0.4724     0.2955   1.599 0.109995
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.28 on 2096 degrees of freedom
## Multiple R-squared:  0.9737, Adjusted R-squared:  0.9735
## F-statistic:  3379 on 23 and 2096 DF,  p-value: < 2.2e-16
```

```
summary(fit2)
```

```
##
## Call:
## lm(formula = intentions ~ candidat * tps + candidat * nom_institut,
##     data = df_pol)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.7981 -0.5456 -0.0272  0.5924  5.2019
##
## Coefficients:
##                                                        Estimate Std. Error
## (Intercept)                                             3.39409    1.18971
## candidatEmmanuel Macron                                20.60082    1.68251
## candidatEric Zemmour                                   10.59844    1.68251
```

```
## candidatFabien Roussel                                  -2.19508    1.68251
## candidatJean-Luc Mélenchon                              11.04593    1.68251
## candidatMarine Le Pen                                   15.30968    1.68251
## candidatValérie Pécresse                                 7.55835    1.68251
## candidatYannick Jadot                                    2.30826    1.68251
## tps2                                                    -0.68106    0.21565
## tps3                                                    -0.99409    0.19847
## nom_institutBVA                                         -0.30224    1.23753
## nom_institutCluster17                                   -0.84531    1.21657
## nom_institutElabe                                       -0.71509    1.21215
## nom_institutHarris Interactive                          -0.18347    1.21291
## nom_institutIfop                                        -0.18249    1.18474
## nom_institutIpsos                                       -0.08904    1.19038
## nom_institutKantar Public                               -0.47826    1.31223
## nom_institutOdoxa                                       -0.67101    1.35576
## nom_institutOpinion Way                                 -0.04925    1.18126
## nom_institutYouGov                                      -1.40000    1.65893
## candidatEmmanuel Macron:tps2                             0.73921    0.30498
## candidatEric Zemmour:tps2                                2.12294    0.30498
## candidatFabien Roussel:tps2                              2.12788    0.30498
## candidatJean-Luc Mélenchon:tps2                          1.71555    0.30498
## candidatMarine Le Pen:tps2                               0.60025    0.30498
## candidatValérie Pécresse:tps2                           -0.97779    0.30498
## candidatYannick Jadot:tps2                              -0.15932    0.30498
## candidatEmmanuel Macron:tps3                             4.79918    0.28068
## candidatEric Zemmour:tps3                               -0.99844    0.28068
## candidatFabien Roussel:tps3                              2.09508    0.28068
## candidatJean-Luc Mélenchon:tps3                          5.15407    0.28068
## candidatMarine Le Pen:tps3                               3.59032    0.28068
## candidatValérie Pécresse:tps3                           -4.65835    0.28068
## candidatYannick Jadot:tps3                               0.29174    0.28068
## candidatEmmanuel Macron:nom_institutBVA                  0.75768    1.75014
## candidatEric Zemmour:nom_institutBVA                    -0.46013    1.75014
## candidatFabien Roussel:nom_institutBVA                   1.43661    1.75014
## candidatJean-Luc Mélenchon:nom_institutBVA              -4.47432    1.75014
## candidatMarine Le Pen:nom_institutBVA                   -1.61439    1.75014
## candidatValérie Pécresse:nom_institutBVA                 5.43117    1.75014
## candidatYannick Jadot:nom_institutBVA                    0.47709    1.75014
## candidatEmmanuel Macron:nom_institutCluster17            0.21696    1.72049
## candidatEric Zemmour:nom_institutCluster17               0.93708    1.72049
## candidatFabien Roussel:nom_institutCluster17             1.75386    1.72049
## candidatJean-Luc Mélenchon:nom_institutCluster17        -1.72026    1.72049
## candidatMarine Le Pen:nom_institutCluster17             -2.63348    1.72049
## candidatValérie Pécresse:nom_institutCluster17           5.22876    1.72049
## candidatYannick Jadot:nom_institutCluster17              0.59139    1.72049
## candidatEmmanuel Macron:nom_institutElabe                1.38103    1.71424
```

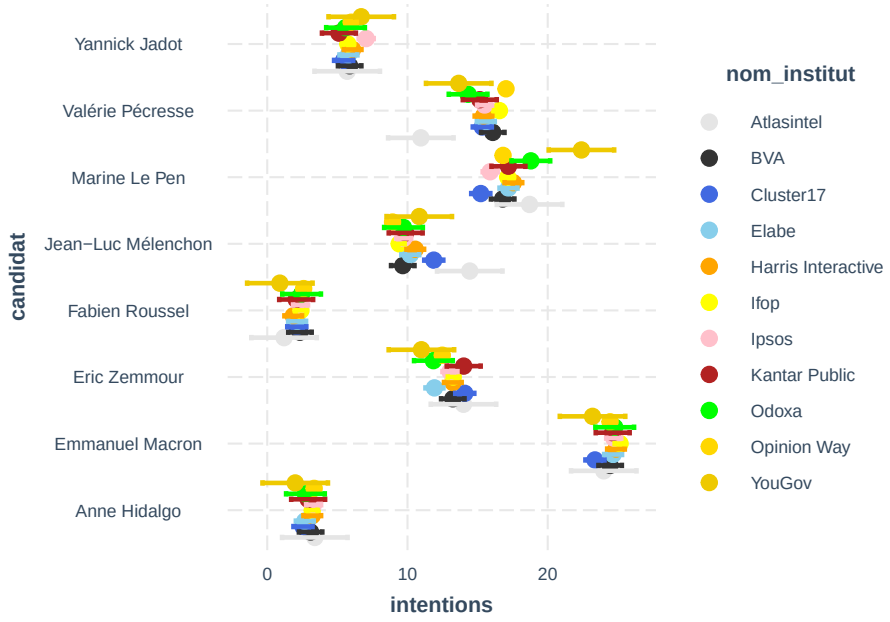
```
## candidatEric Zemmour:nom_institutElabe                    -1.35629    1.71424
## candidatFabien Roussel:nom_institutElabe                   1.64477    1.71424
## candidatJean-Luc Mélenchon:nom_institutElabe              -3.52444    1.71424
## candidatMarine Le Pen:nom_institutElabe                   -0.78677    1.71424
## candidatValérie Pécresse:nom_institutElabe                 5.34773    1.71424
## candidatYannick Jadot:nom_institutElabe                    0.80139    1.71424
## candidatEmmanuel Macron:nom_institutHarris Interactive     1.05598    1.71531
## candidatEric Zemmour:nom_institutHarris Interactive       -0.57494    1.71531
## candidatFabien Roussel:nom_institutHarris Interactive      0.83821    1.71531
## candidatJean-Luc Mélenchon:nom_institutHarris Interactive -3.70231    1.71531
## candidatMarine Le Pen:nom_institutHarris Interactive      -0.96863    1.71531
## candidatValérie Pécresse:nom_institutHarris Interactive    4.65034    1.71531
## candidatYannick Jadot:nom_institutHarris Interactive       0.56962    1.71531
## candidatEmmanuel Macron:nom_institutIfop                   1.34434    1.67547
## candidatEric Zemmour:nom_institutIfop                     -0.55201    1.67547
## candidatFabien Roussel:nom_institutIfop                    1.38693    1.67547
## candidatJean-Luc Mélenchon:nom_institutIfop               -4.84893    1.67547
## candidatMarine Le Pen:nom_institutIfop                    -1.37146    1.67547
## candidatValérie Pécresse:nom_institutIfop                  5.78598    1.67547
## candidatYannick Jadot:nom_institutIfop                     0.22809    1.67547
## candidatEmmanuel Macron:nom_institutIpsos                  0.79936    1.68345
## candidatEric Zemmour:nom_institutIpsos                    -0.87532    1.68345
## candidatFabien Roussel:nom_institutIpsos                   1.26446    1.68345
## candidatJean-Luc Mélenchon:nom_institutIpsos              -4.62283    1.68345
## candidatMarine Le Pen:nom_institutIpsos                   -2.71909    1.68345
## candidatValérie Pécresse:nom_institutIpsos                 4.63358    1.68345
## candidatYannick Jadot:nom_institutIpsos                    1.44225    1.68345
## candidatEmmanuel Macron:nom_institutKantar Public          1.11499    1.85577
## candidatEric Zemmour:nom_institutKantar Public             0.49465    1.85577
## candidatFabien Roussel:nom_institutKantar Public           1.34180    1.85577
## candidatJean-Luc Mélenchon:nom_institutKantar Public      -4.09037    1.85577
## candidatMarine Le Pen:nom_institutKantar Public           -1.02748    1.85577
## candidatValérie Pécresse:nom_institutKantar Public         4.67986    1.85577
## candidatYannick Jadot:nom_institutKantar Public           -0.11224    1.85577
## candidatEmmanuel Macron:nom_institutOdoxa                  1.45332    1.91734
## candidatEric Zemmour:nom_institutOdoxa                    -1.47379    1.91734
## candidatFabien Roussel:nom_institutOdoxa                   1.92240    1.91734
## candidatJean-Luc Mélenchon:nom_institutOdoxa              -4.05383    1.91734
## candidatMarine Le Pen:nom_institutOdoxa                    0.76336    1.91734
## candidatValérie Pécresse:nom_institutOdoxa                 4.03981    1.91734
## candidatYannick Jadot:nom_institutOdoxa                    0.55035    1.91734
## candidatEmmanuel Macron:nom_institutOpinion Way            0.50026    1.67056
## candidatEric Zemmour:nom_institutOpinion Way              -1.46076    1.67056
## candidatFabien Roussel:nom_institutOpinion Way             1.44706    1.67056
## candidatJean-Luc Mélenchon:nom_institutOpinion Way        -5.45426    1.67056
## candidatMarine Le Pen:nom_institutOpinion Way             -1.84075    1.67056
```

```
## candidatValérie Pécresse:nom_institutOpinion Way          6.12136    1.67056
## candidatYannick Jadot:nom_institutOpinion Way             0.33856    1.67056
## candidatEmmanuel Macron:nom_institutYouGov                0.60000    2.34608
## candidatEric Zemmour:nom_institutYouGov                  -1.60000    2.34608
## candidatFabien Roussel:nom_institutYouGov                 1.10000    2.34608
## candidatJean-Luc Mélenchon:nom_institutYouGov            -2.20000    2.34608
## candidatMarine Le Pen:nom_institutYouGov                  5.10000    2.34608
## candidatValérie Pécresse:nom_institutYouGov               4.10000    2.34608
## candidatYannick Jadot:nom_institutYouGov                  2.40000    2.34608
##                                                          t value Pr(>|t|)
## (Intercept)                                                2.853 0.004377 **
## candidatEmmanuel Macron                                   12.244  < 2e-16 ***
## candidatEric Zemmour                                       6.299 3.66e-10 ***
## candidatFabien Roussel                                    -1.305 0.192163
## candidatJean-Luc Mélenchon                                 6.565 6.59e-11 ***
## candidatMarine Le Pen                                      9.099  < 2e-16 ***
## candidatValérie Pécresse                                   4.492 7.44e-06 ***
## candidatYannick Jadot                                      1.372 0.170242
## tps2                                                      -3.158 0.001611 **
## tps3                                                      -5.009 5.96e-07 ***
## nom_institutBVA                                           -0.244 0.807081
## nom_institutCluster17                                     -0.695 0.487241
## nom_institutElabe                                         -0.590 0.555298
## nom_institutHarris Interactive                            -0.151 0.879779
## nom_institutIfop                                          -0.154 0.877598
## nom_institutIpsos                                         -0.075 0.940383
## nom_institutKantar Public                                 -0.364 0.715551
## nom_institutOdoxa                                         -0.495 0.620703
## nom_institutOpinion Way                                   -0.042 0.966747
## nom_institutYouGov                                        -0.844 0.398816
## candidatEmmanuel Macron:tps2                               2.424 0.015445 *
## candidatEric Zemmour:tps2                                  6.961 4.55e-12 ***
## candidatFabien Roussel:tps2                                6.977 4.07e-12 ***
## candidatJean-Luc Mélenchon:tps2                            5.625 2.11e-08 ***
## candidatMarine Le Pen:tps2                                 1.968 0.049184 *
## candidatValérie Pécresse:tps2                             -3.206 0.001367 **
## candidatYannick Jadot:tps2                                -0.522 0.601458
## candidatEmmanuel Macron:tps3                              17.098  < 2e-16 ***
## candidatEric Zemmour:tps3                                 -3.557 0.000383 ***
## candidatFabien Roussel:tps3                                7.464 1.24e-13 ***
## candidatJean-Luc Mélenchon:tps3                           18.363  < 2e-16 ***
## candidatMarine Le Pen:tps3                                12.791  < 2e-16 ***
## candidatValérie Pécresse:tps3                            -16.597  < 2e-16 ***
## candidatYannick Jadot:tps3                                 1.039 0.298749
## candidatEmmanuel Macron:nom_institutBVA                    0.433 0.665115
## candidatEric Zemmour:nom_institutBVA                      -0.263 0.792645
```

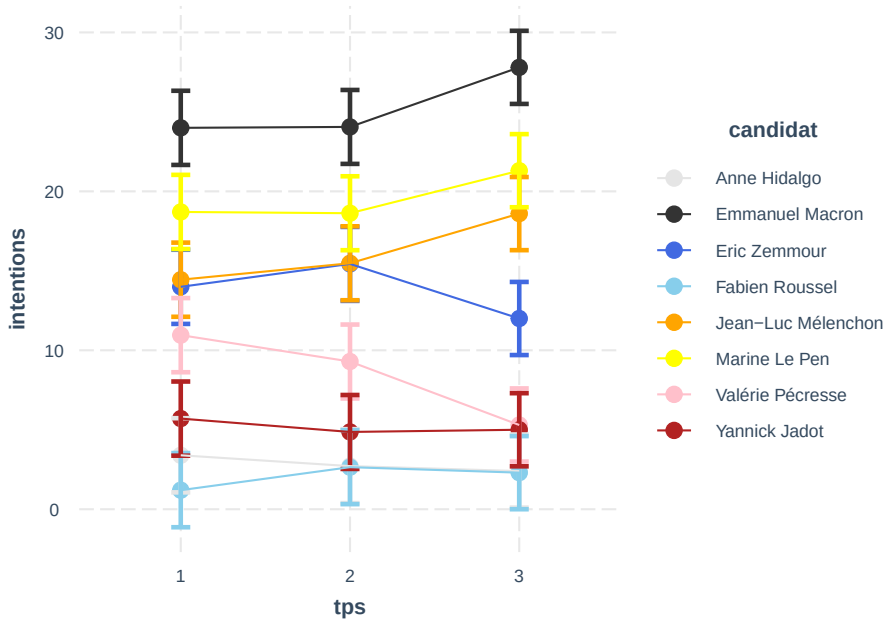
```
## candidatFabien Roussel:nom_institutBVA                      0.821 0.411826
## candidatJean-Luc Mélenchon:nom_institutBVA                 -2.557 0.010644 *
## candidatMarine Le Pen:nom_institutBVA                      -0.922 0.356411
## candidatValérie Pécresse:nom_institutBVA                    3.103 0.001940 **
## candidatYannick Jadot:nom_institutBVA                       0.273 0.785186
## candidatEmmanuel Macron:nom_institutCluster17               0.126 0.899664
## candidatEric Zemmour:nom_institutCluster17                  0.545 0.586049
## candidatFabien Roussel:nom_institutCluster17                1.019 0.308139
## candidatJean-Luc Mélenchon:nom_institutCluster17           -1.000 0.317496
## candidatMarine Le Pen:nom_institutCluster17                -1.531 0.126012
## candidatValérie Pécresse:nom_institutCluster17              3.039 0.002403 **
## candidatYannick Jadot:nom_institutCluster17                 0.344 0.731085
## candidatEmmanuel Macron:nom_institutElabe                   0.806 0.420553
## candidatEric Zemmour:nom_institutElabe                     -0.791 0.428924
## candidatFabien Roussel:nom_institutElabe                    0.959 0.337434
## candidatJean-Luc Mélenchon:nom_institutElabe               -2.056 0.039913 *
## candidatMarine Le Pen:nom_institutElabe                    -0.459 0.646309
## candidatValérie Pécresse:nom_institutElabe                  3.120 0.001837 **
## candidatYannick Jadot:nom_institutElabe                     0.467 0.640200
## candidatEmmanuel Macron:nom_institutHarris Interactive      0.616 0.538214
## candidatEric Zemmour:nom_institutHarris Interactive        -0.335 0.737521
## candidatFabien Roussel:nom_institutHarris Interactive       0.489 0.625133
## candidatJean-Luc Mélenchon:nom_institutHarris Interactive -2.158 0.031015 *
## candidatMarine Le Pen:nom_institutHarris Interactive       -0.565 0.572343
## candidatValérie Pécresse:nom_institutHarris Interactive     2.711 0.006763 **
## candidatYannick Jadot:nom_institutHarris Interactive        0.332 0.739863
## candidatEmmanuel Macron:nom_institutIfop                    0.802 0.422437
## candidatEric Zemmour:nom_institutIfop                      -0.329 0.741837
## candidatFabien Roussel:nom_institutIfop                     0.828 0.407892
## candidatJean-Luc Mélenchon:nom_institutIfop                -2.894 0.003844 **
## candidatMarine Le Pen:nom_institutIfop                     -0.819 0.413138
## candidatValérie Pécresse:nom_institutIfop                   3.453 0.000565 ***
## candidatYannick Jadot:nom_institutIfop                      0.136 0.891729
## candidatEmmanuel Macron:nom_institutIpsos                   0.475 0.634955
## candidatEric Zemmour:nom_institutIpsos                     -0.520 0.603149
## candidatFabien Roussel:nom_institutIpsos                    0.751 0.452670
## candidatJean-Luc Mélenchon:nom_institutIpsos               -2.746 0.006085 **
## candidatMarine Le Pen:nom_institutIpsos                    -1.615 0.106425
## candidatValérie Pécresse:nom_institutIpsos                  2.752 0.005968 **
## candidatYannick Jadot:nom_institutIpsos                     0.857 0.391697
## candidatEmmanuel Macron:nom_institutKantar Public           0.601 0.548025
## candidatEric Zemmour:nom_institutKantar Public              0.267 0.789843
## candidatFabien Roussel:nom_institutKantar Public            0.723 0.469738
## candidatJean-Luc Mélenchon:nom_institutKantar Public       -2.204 0.027628 *
## candidatMarine Le Pen:nom_institutKantar Public            -0.554 0.579867
## candidatValérie Pécresse:nom_institutKantar Public          2.522 0.011753 *
```

```
## candidatYannick Jadot:nom_institutKantar Public            -0.060 0.951779
## candidatEmmanuel Macron:nom_institutOdoxa                   0.758 0.448546
## candidatEric Zemmour:nom_institutOdoxa                     -0.769 0.442182
## candidatFabien Roussel:nom_institutOdoxa                    1.003 0.316155
## candidatJean-Luc Mélenchon:nom_institutOdoxa               -2.114 0.034612 *
## candidatMarine Le Pen:nom_institutOdoxa                     0.398 0.690574
## candidatValérie Pécresse:nom_institutOdoxa                  2.107 0.035242 *
## candidatYannick Jadot:nom_institutOdoxa                     0.287 0.774112
## candidatEmmanuel Macron:nom_institutOpinion Way             0.299 0.764623
## candidatEric Zemmour:nom_institutOpinion Way               -0.874 0.381997
## candidatFabien Roussel:nom_institutOpinion Way              0.866 0.386475
## candidatJean-Luc Mélenchon:nom_institutOpinion Way         -3.265 0.001113 **
## candidatMarine Le Pen:nom_institutOpinion Way              -1.102 0.270647
## candidatValérie Pécresse:nom_institutOpinion Way            3.664 0.000254 ***
## candidatYannick Jadot:nom_institutOpinion Way               0.203 0.839420
## candidatEmmanuel Macron:nom_institutYouGov                  0.256 0.798174
## candidatEric Zemmour:nom_institutYouGov                    -0.682 0.495325
## candidatFabien Roussel:nom_institutYouGov                   0.469 0.639216
## candidatJean-Luc Mélenchon:nom_institutYouGov              -0.938 0.348494
## candidatMarine Le Pen:nom_institutYouGov                    2.174 0.029834 *
## candidatValérie Pécresse:nom_institutYouGov                 1.748 0.080687 .
## candidatYannick Jadot:nom_institutYouGov                    1.023 0.306439
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.173 on 2016 degrees of freedom
## Multiple R-squared:  0.9788, Adjusted R-squared:  0.9777
## F-statistic: 902.8 on 103 and 2016 DF,  p-value: < 2.2e-16
```

```
library(jtools)

library(interactions)
cat_plot(fit2, pred=candidat,modx= nom_institut, color.class="Spectral")+
  scale_color_manual(values = SensiP2)+coord_flip()
```




```
cat_plot(fit2, pred= tps,modx=candidat, color.class="Spectral", dodge.width=0)+
  scale_color_manual(values = SensiP2)+geom_line(aes(color=candidat))
```



Chapter 7

Données géographique

voir étude de cas Airbnb avec le package sf

Chapter 8

Analyses factorielles exploratoires

8.1 Origine et histoire

Par analyse factorielle, on entend finalement un ensemble de méthodes dont l'objectif est d'extraire d'un ensemble multivariée de données, un petit nombre de dimensions, les facteurs, qui rendent compte de l'essentiel des variations. On peut en distinguer deux écoles, l'une alimentée par des questions de psychométrie, plutôt américaine, et l'autre française s'intéresse aux variables qualitatives, et à une perspective plus descriptive.

8.1.1 Une petite histoire de la psychométrie

L'analyse factorielle trouve son origine, en psychologie, dans l'intuition que dans des épreuves multiples un facteur principal contrôle les variation des items (les performance à différents tests). Mais c'est avec Thurstone (1931) que l'idée prend toute son ampleur en permettant que plusieurs facteurs traduisent la structure de la matrice de corrélations entre les tests. Spearman, Hotelling,.

Dans le monde de la gestion et en particulier de la GRH et du marketing, largement inspirés par la psychologie et la psychologie sociale, ces méthodes se sont propagées et ont formalisé un processus d'étude largement fondé sur ces techniques. Il est bien connu sous le terme de paradigme de Churchill qui synthésise une manière de construire et de développer des échelles multi-items de mesure par questionnaire.

Un grand virage c'est produit avec les méthodes d'équations structurelles qui ont permis le développement de méthodes à variables latente, dont la finalité est confirmatoire.

8.1.2 L'école française de l'analyse des données appliquée aux sciences sociales

dès le début des années 60

Un personnage : Emile Benzekri

Boudieu en premier applicateurs

Une école Française : pagès, escoffier, morisseau, Husson a repris le flambleau en développant FactoMiner.

Une série de logiciels : Alceste, Statitcf

8.2 Le modèle en facteurs communs et spécifiques

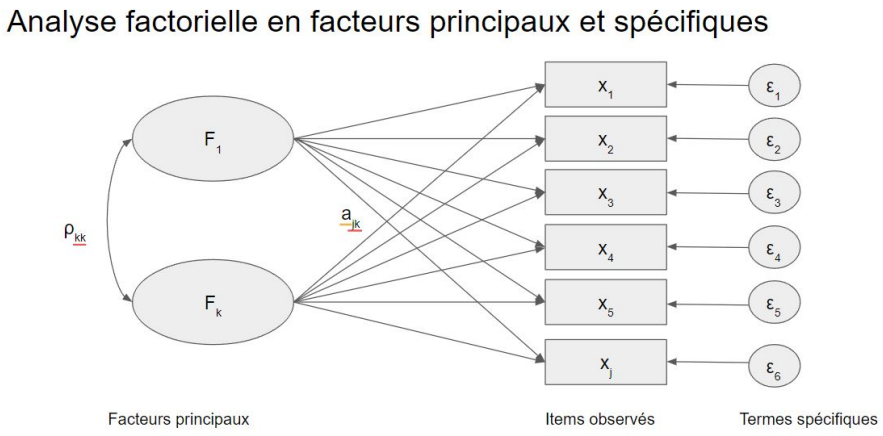
8.2.1 Un peu de théorie

La première historiquement est celle des psychologues et en particulier le modèle en terme de facteurs communs et spécifiques. Elle vise à partir de l'analyse d'une matrice de corrélation à identifier des éléments de structures sous-jascents.

La structure du modèle factoriel peut être présentée de manière simple. On supposera que chaque variables observées peut être décrites comme composées de facteurs généraux (F_{ik}) et de facteurs spécifiques $ _{i}$. Le modèle suppose ainsi que la valeur de l'individu i pour la variable j, dépend de k facteurs sous-jacents, les facteurs communs, et d'un terme spécifique à l'item.

$$x_{ij} = a_{1j}F_{i1} + a_{2j}F_{i2} + \cdots + a_{jk}F_{ik} + \varepsilon_{ij}$$

On peut représenter cela de manière graphique, en utilisant les conventions symboliques des modèles structurels qu'on examine dans le chapitre SEM. On y verra d'ailleurs comment ce modèle peut être spécifié de manière confirmatoire. On remarquera que dans cette structure les facteurs peuvent être corrélés.



Certains lecteurs seront surpris de cette présentation, ils sont sans doute plus habitués à factoriser en employant une méthode de l' ACP. Effectivement cette méthode sur laquelle on va revenir avec plus de détail dans la seconde section de ce chapitre, est une des techniques qui permettent d'approcher le modèle théorique que l'on vient de présenter. Elle n'est pas la seule.

L'estimation du modèle requiert deux décisions : l'une sur la méthode d'extraction des facteurs, et l'autre sur la méthode de rotation.

Les méthodes d'extraction : * ACP * ML

Les méthodes de rotation.

- Varimax
- Promax
- Oblimin
- ...

L'objectif des méthodes d'analyses factorielles est de réduire un ensemble de variables à un petit nombre de dimensions qui résument l'essentiel de l'information.

8.2.2 Ressources

On utilise principalement le package `psych` développé par Revelle et dédié à la psychométrie. Il couvre le plus complètement le champs de l'analyse factorielle et de la psychométrie.

S'y ajoutent deux fonctions très utiles pour représenter les résultats des analyses sous une forme lisible et au standard des publications scientifiques. Elles utilisent les ressources de `flextable`.

```
# Une fonction utile pour créer

flex <- function(data, title=NULL) {
  # this grabs the data and converts it to a flextbale
  flextable(data) %>%
  # this makes the table fill the page width
  set_table_properties(layout = "autofit", width = 1) %>%
  # font size
  fontsize(size=10, part="all") %>%
    #this adds a ttitlecreates an automatic table number
      set_caption(title,
                  autonum = officer::run_autonum(seq_id = "tab",
                                                 pre_label = "Table ",
                                                 post_label = "\n",
                                                 bkm = "anytable")) %>%
  # font type
  font(fontname="Times New Roman", part="all")
}

#Une seconde fonction pour le tableaux des loadings

fa_table <- function(x, cut) {
  #get sorted loadings
  loadings <- fa.sort(x)$loadings %>% round(3)
  #supress loadings
  loadings[loadings < cut] <- ""
  #get additional info
  add_info <- cbind(x$communality,
                    x$uniquenesses,
                    x$complexity) %>%
    # make it a data frame
    as.data.frame() %>%
    # column names
    rename("Communality" = V1,
           "Uniqueness" = V2,
           "Complexity" = V3) %>%
    #get the item names from the vector
    rownames_to_column("item")
  #build table
  loadings %>%
    unclass() %>%
    as.data.frame() %>%
    rownames_to_column("item") %>%
    left_join(add_info) %>%
    mutate(across(where(is.numeric), round, 3))
```

```
}
```

8.3 Cas d'application

Pour appliquer la méthode on va s'intéresser à l'échelle des valeurs de Schwartz (2006), qui sont mesurées dan différents pays au cours des différentes vagues de l'enquête European Social Survey.

Les variables mesurées sont un ensemble de 21 questions qui proposent des niveaux d'importances accordées à 21 questions, ou items, dont voici les formulation en anglais. Les répondants ont le choix sur une échelle de 0 à 10 qui va de "pas du tout important" à "très important". On se concentre sur les observations de la dernière vague.

Cette échelle a été développée par Schwartz

En voici les items dans leur formulation anglaise.

- IPCRTIV Important to think new ideas and being creative
- IMPRICH Important to be rich, have money and expensive things
- IPEQOPT Important that people are treated equally and have equal opportunities
- IPSHABT Important to show abilities and be admired
- IMPSAFE Important to live in secure and safe surroundings
- IMPDIFF Important to try new and different things in life
- IPFRULE Important to do what is told and follow rules
- IPUDRST Important to understand different people
- IPMODST Important to be humble and modest, not draw attention
- IPGDTIM Important to have a good time
- IMPFREE Important to make own decisions and be free
- IPHLPPL Important to help people and care for others well-being
- IPSUCES Important to be successful and that people recognise achievements
- IPSTRGV Important that government is strong and ensures safety
- IPADVNT Important to seek adventures and have an exciting life
- IPBHPRP Important to behave properly
- IPRSPOT Important to get respect from others
- IPLYLFR Important to be loyal to friends and devote to people close
- IMPENV Important to care for nature and environment
- IMPTRAD Important to follow traditions and customs
- IMPFUN Important to seek fun and things that give pleasure

```
# On renomme les variables pour une meilleure lecture et on selectionne
# le tableau de données utile à l'analyse.
```

```
df <- read_csv("./Data/ESS1-9e01_1.csv") %>%
  rename(
    V_creative=ipcrtiv,
    V_richness= imprich,
    V_justice =ipeqopt,
    V_admiration=ipshabt,
    V_security=impsafe,
    V_novelty=impdiff,
    V_conformism=ipfrule,
    V_openmindedness=ipudrst,
    V_modesty=ipmodst,
    V_fun=ipgdtim,
    V_autonomy=impfree,
    V_Care=iphlppl,
    V_Success=ipsuces,
    V_Autority =ipstrgv,
    V_Adventures=ipadvnt,
    V_wellbehavior=ipbhprp,
    V_respect=iprspot,
    V_loyalty=iplylfr,
    V_environnement=impenv,
    V_tradition=imptrad,
    V_pleasure=impfun)

foo1<-df %>% filter(essround==9)%>%
  dplyr::select(matches("V_.*"), cntry) %>% #notons la selection fondée sur des regex
  drop_na()
```

8.3.1 Examen de la matrice de corrélation

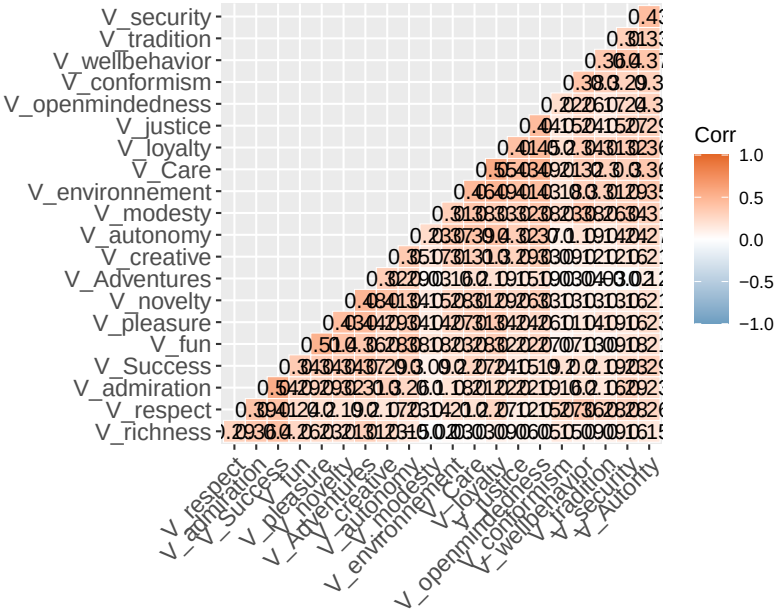
Calculons la matrice de corrélation, et présentons là en organisant l'ordre des variables selon leur corrélation. A ce stade indiquons qu'il s'agit de mettre un ordre dans les variables, tel que des variables fortement corrélées soient adjascentes (on revient sur la méthode utilisée dans le chapitre suivant).

On s'aperçoit qu'une structure émerge. Quatre groupes de variables peuvent être discernés: * la jouissance * le succès social * l'ouverture aux autres * la sécurité

Dans le filigrane de la matrice de corrélation, on devine une structure factorielle.

```
foo<- foo1 %>%
  dplyr::select(matches("V_.*"))

M <- cor(foo)
```

```
ggcorrplot(M, hc.order = TRUE, type = "lower",
   outline.col = "white",
   ggtheme = ggplot2::theme_gray,
   colors = c("#6D9EC1", "white", "#E46726"), lab = TRUE, lab_size = 4)
```



8.3.2 Modèle factoriel

Testons un modèle d'analyse factorielle à 4 dimensions. Nous l'augmentons d'un procédure de rotation oblimin pour un meilleur ajustement.

```
fa <- fa(foo,4, rotate="oblimin")  #principal axis

fa_table(fa, .30)%>%
  flex("A Pretty Factor Analysis Table")
```

Table 8.1: A Pretty Factor Analysis Table

item	MR1	MR4	MR3	MR2	Communality	Uniqueness	Complexity
V_openmindedness	0.692				0.472	0.528	1.002
V_justice	0.665				0.399	0.601	1.044

item	MR1	MR4	MR3	MR2	Communality	Uniqueness	Complexity
V_Care	0.621				0.516	0.484	1.145
V_environnement	0.561				0.427	0.573	1.171
V_loyalty	0.518				0.493	0.507	1.570
V_autonomy	0.438				0.364	0.636	1.603
V_creative	0.429				0.341	0.659	2.225
V_modesty	0.421		0.301		0.339	0.661	1.967
V_admiration		0.697			0.505	0.495	1.049
V_Success		0.644			0.539	0.461	1.094
V_richness		0.526			0.336	0.664	1.402
V_respect		0.419	0.379		0.383	0.617	2.157
V_wellbehavior			0.604		0.461	0.539	1.094
V_tradition			0.531		0.323	0.677	1.111
V_conformism			0.464		0.265	0.735	1.166
V_security			0.445		0.386	0.614	2.047
V_Autority			0.395		0.369	0.631	1.921
V_pleasure				0.758	0.561	0.439	1.035
V_fun				0.531	0.413	0.587	1.145
V_Adventures				0.506	0.465	0.535	1.829
V_novelty				0.392	0.431	0.569	2.553

```
fa[["Vaccounted"]] %>%
  as.data.frame() %>%
  rownames_to_column("Property") %>%
    mutate(across(where(is.numeric), round, 3)) %>%
    flex("Eigenvalues and Variance Explained for Oblimin Factor Solution")
```

Table 8.2: Eigenvalues and Variance Explained for Oblimin Factor Solution

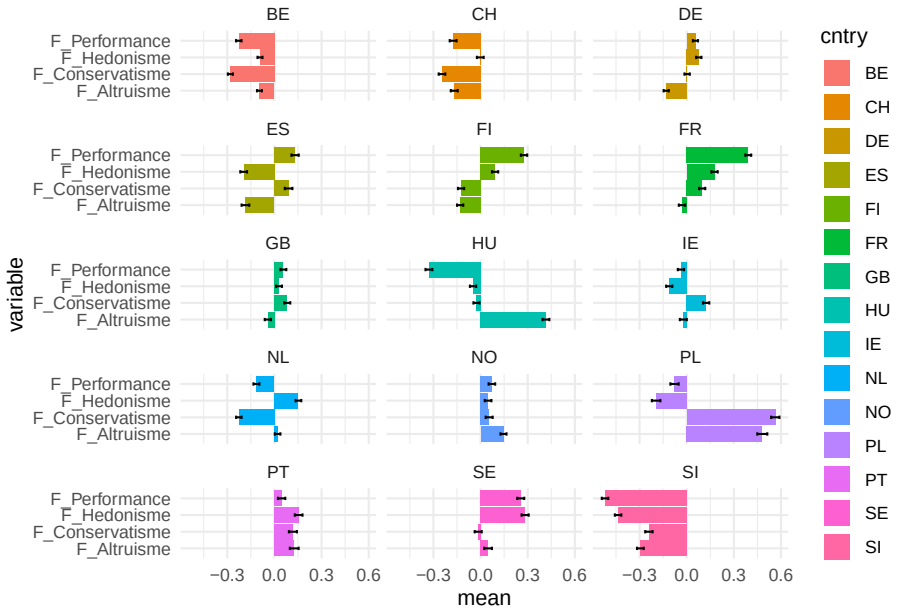
Property	MR1	MR4	MR3	MR2
SS loadings	3.122	1.958	1.897	1.813
Proportion Var	0.149	0.093	0.090	0.086

Property	MR1	MR4	MR3	MR2
Cumulative Var	0.149	0.242	0.332	0.419
Proportion Explained	0.355	0.223	0.216	0.206
Cumulative Proportion	0.355	0.578	0.794	1.000

Le set de données que nous avons traité est composé de 15 échantillons venant d'autant de pays. Puisque nous avons réduits les 22 mesures initiales à 4 grands facteurs, il est temps d'analyser les différences entre les pays.

On va d'abord récupérer les scores de chaque observation sur les quatre dimensions obtenues qu'on ajoute à notre fichier de travail pour récupérer la variable pays.

```
#récupérer les scores
scores<-fa$scores
scores<-as.data.frame(unclass(scores))

#matcher pour récupérer la variable pays et renommer pour plus de lisibilité
df_typo<-cbind(foo1, scores) %>%
  rename(F_Altruisme = MR1,
         F_Conservatisme=MR2,
         F_Performance=MR4,
         F_Hedonisme=MR3)

# On calcule les scores moyens par pays et les erreurs d'échantillonnage
df_g <- df_typo %>%
  dplyr::select(matches("F_.*"), cntry)%>%
  gather(variable, value,-cntry)%>%
  mutate(n=1)%>%
  group_by(variable,cntry)%>%
  summarize(mean=mean(value),
            n=sum(n),
            se=sd(value)/sqrt(n))

#on représente les résultats
ggplot(df_g,aes(x=variable, y=mean))+
  geom_bar(stat="identity",aes(fill=cntry), size=1.5)+
  coord_flip()+
  geom_errorbar(aes(ymin=mean-se, ymax=mean+se), width=.2, position=position_dodge(.9)) +
  facet_wrap(vars(cntry ),ncol=3)
```



Analyse en composante principale

L'ACP, dont l'optique est différente dans le sens où l'on cherche moins à rendre compte d'une structure sous-jascente à la matrice de corrélation , qu'à réduire l'information dans un espace limité.

8.3.3 le problème théorique

De manière intuitive l'ACP est la technique qui permet de représenter un poisson, une structure, sous son jour le plus intelligible, c'est à dire celui qui magnifie ses variations.

Examinons un poisson sous différentes projections. La première image rend mieux compte de la forme du poisson que la seconde, elle ne diffère que par la projection. De l'une à l'autre il n'y a qu'y rotation à 90°C vers la droite. C'est la même image, le même phénomène mais représenté selon deux perspectives, deux bases en terme de mathématiques. On comprend que pour représenter un objet au mieux dans un faible nombre de dimensions, il faut trouver la base vectorielle qui maximise les variations de taille.

résoudre ce problème est ce que fait l'ACP

8.3.4 Une représentation symbolique

L'idée va donc être de décomposer une matrice de variance-covariance (ou de corrélation) en respectant une contraintes : faire en sorte que le maximum de

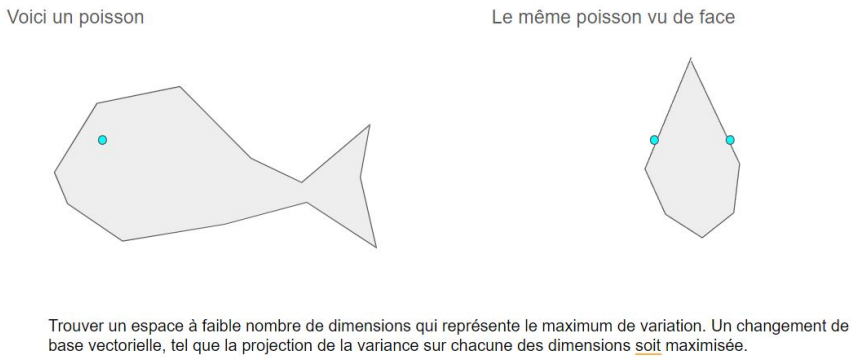


Figure 8.1: Modèle Factoriel exploratoire - EFA

variance soit capturée par la première dimension, puis par les suivantes successivement. La solution à ce problème se trouve dans la résolution d'un problème matriciel. Il faut procéder à un changement de base, autrement dit à un changement de référentiel.

La matrice de variance-covariance, ou de corrélation, si on a, au préalable, centré et standardisé les valeurs des variables, est obtenue simplement en multipliant la matrice de données (individus x variable) par sa transposée.

$$\Sigma = XX^t$$

Comme

$$\Sigma$$

est symétrique, elle est diagonalisable et peut-être représentée par une matrice de score W et une matrice diagonale D.

$$\Sigma_e = WDW^T$$

où D est la matrice diagonale des valeurs propres et W la matrice des composantes comprenant les j variables (en ligne) et les k dimensions (en colonne). L'équivalence suppose que le nombre de composantes est égal au nombre de variables initiales, Cependant l'usage conduit à ne retenir qu'un petit nombre de dimensions de telles sorte à ce que la différence entre Σ et Σ_e soit relativement petite. La matrice de score comprend autant de lignes que d'individus et de colonnes que de dimensions-sous-jascentes.

On remarquera que dans ce modèles on a autant de composantes que de variables, mais que ces dernières représentent une part décroissante de la variance. Certaines composantes n'ont pas de sens on se concentrera sur les premières rejoignant l'idée de l'analyse factorielle : peu de composantes, de facteurs, rendent compte des variations des données.

On restera cependant conscient que l'ACP n'est au fond qu'une manière de représenter les données, juste une projection. Ne retenir que les premières composantes va au-delà du modèle, c'est une démarche qui consiste à considérer que seules les premières composantes sont significatives, en apportant du sens, et les dernières peuvent être négligée. C'est une manière approximative de rejoindre le modèle factoriel, une solution simple pour en obtenir une solution.

8.3.5 Application

En guise d'application on va utiliser un tout petit jeu de données issu de l'analyse précédente : le tableau des profils pays, sur les 21 valeurs de @{schwartz_les_2006, Avec cette procédure d'agrégation on réduit fortement la variance individuelle, pour ne garder que des différences en moyenne d'un pays à l'autre.

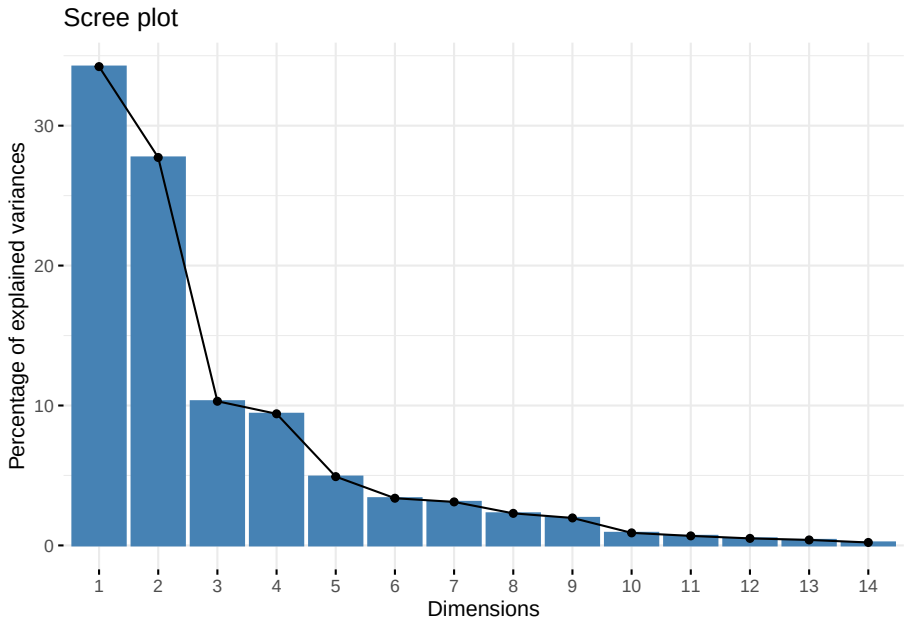
Le plus ici ne va plus être de comprendre la structure profonde des données, mais simplement de représenter ces différences dans un espace de dimension réduite.

```
foo<-foo1%>%
  group_by(cntry)%>%
  summarise(across(V_creative:V_pleasure, ~ mean(.x, na.rm = TRUE)))
#on note la fonction qui permet de résumer plusieurs variables à la fois

X<- foo%>%
  dplyr::select(-cntry)%>%
  as.data.frame()
rownames(X) <- foo$cntry
```

Plusieurs bibliothèque, en plus de la fonction de base princomp, propose une solution d' ACP. On choisit d'utiliser celle du package `Factominer` qu'on accompagne de la bibliothèque `factoextra` pour ses ressources graphiques.

Les résultats portent sur 3 éléments : les valeurs propres de chacune des dimensions retenues, les coordonnées des vecteurs variables, et celles des points individus.

```
#ACP
res.pca<-PCA(X, scale.unit = TRUE, ncp = 2, graph = FALSE)

fviz_screeplot(res.pca, ncp=21)
```



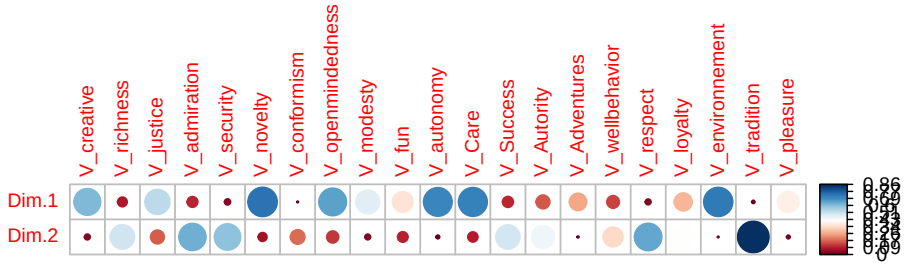

```
x<-res.pca$eig
```

Le premier élément d'analyse et le graphe des éboulis (ou scree plot) qui représente les variances projetées sur chacune des composantes. Ici deux composantes représentent les deux tiers.

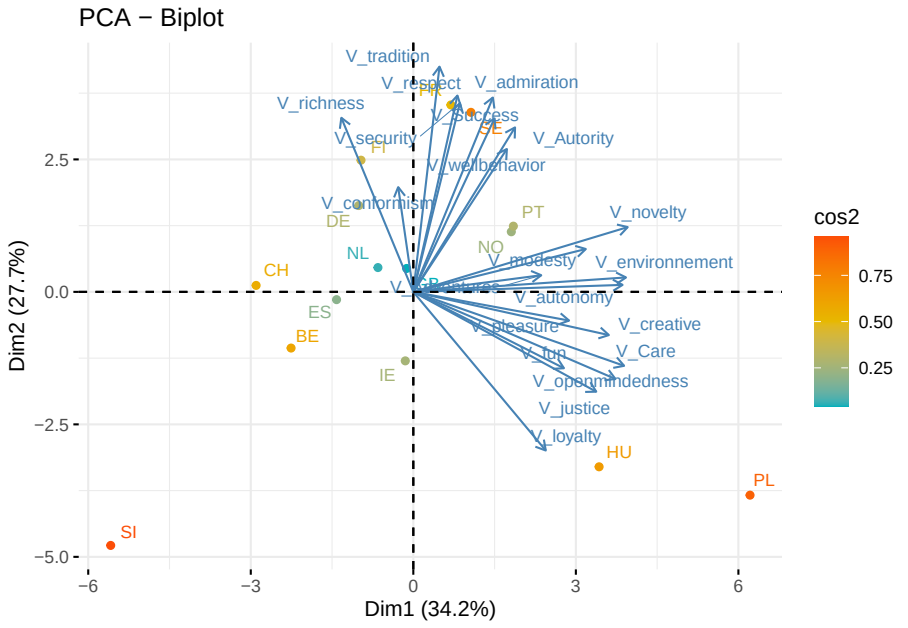
On représente les corrélations des valeurs aux deux composantes par un corrélogramme qui distingue nettement une composante d'ouverture aux autres, et une composante plus autoritaire et égocentrée.

Le biplot permet en plus de représenter les individus qui sont dans ce cas les différents pays. On laisse au lecteur de comprendre les différences entre les pays.

```
library("corrplot")
corrplot(t(res.pca$var$cos2), is.corr=FALSE, tl.cex = 0.8)
```




```
### ce merveilleux bi plot

fviz_pca_biplot(res.pca, col.ind = "cos2", labelsize = 3,
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE )# Biplot des individus et variables
```



8.4 Une généralisation de l'ACP : l'AFC

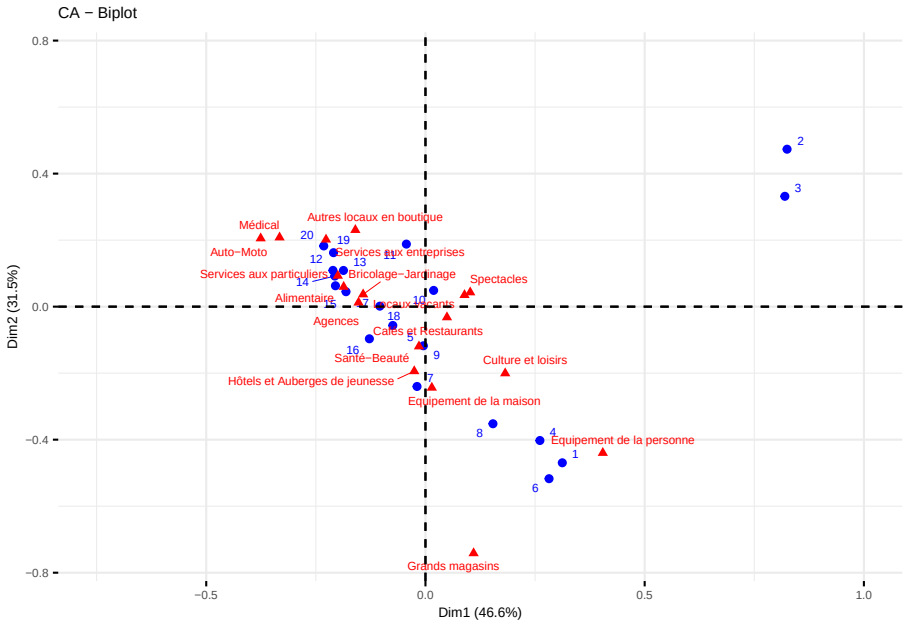
L'AFC trouve une application remarquable dans l'analyse de tableaux croisés. Elle est une méthode de réprésentation des profils lignes et colonnes: on s'aperçoit que deux analyses peuvent être menées : l'une sur les colonnes, et l'autre sur les lignes. Dans les deux cas cette analyse peut se faire en comparant les colonnes (lignes) selon la formule suivante

$$d_{i,j} = (f_{.i} - f_{.j})^2$$

L'idée maintenant est claire : on mène deux acp, en ligne et en colonne, et on projettent conjointement (dans un même espace)

```
library(readr)
BDCOM_2020 <- read_csv("Data//BDCOM/BDCOM_2020.csv") %>%rename(CODACT=CODE_ACTIVITE)
BDCOM_2017_CODACT_OD <- read_delim("Data/BDCOM/BDCOM_2017_CODACT_OD.csv",
                                   delim = ";", escape_double = FALSE, trim_ws = TRUE)

df<-BDCOM_2020%>%left_join(BDCOM_2017_CODACT_OD, by = "CODACT")%>%rename(ACT=27)
t<-table(df$ARRONDISSEMENT,df$ACT )

res.ca <- CA(t,
             graph = FALSE)
```

```
fviz_ca_biplot(res.ca, labelsize = 2, repel=TRUE)+
  theme(text = element_text(size =7)) +xlim(-0.75, 1)+ylim(-.75,0.75)
```



règles d'interprétation

- le point (0,0) représente le barycentre du nuage de point,et donc l'invidu moyens
- les lignes/colonne les plus excentrée sont les moins fréquentes, la distance d'une modalité d'une variable à une autres, indique la correspondance des deux modalités qui partagent les mêmes individus.
- l'inertie totale est chi²/n et donc une véritable méthode : analyse de la décomposition du khi2.

Dans notre exemple on note de suite les arrondissement 2 et 3 qui sont les plus proches de la catégorie commerce de gros.

On note aussi une disposition linéaire qui opposent les arrondissements excentrés, aux arrondissements du centre. Un univers commercial résidentiel vs un univers de transit (spectacles et grands magasins)

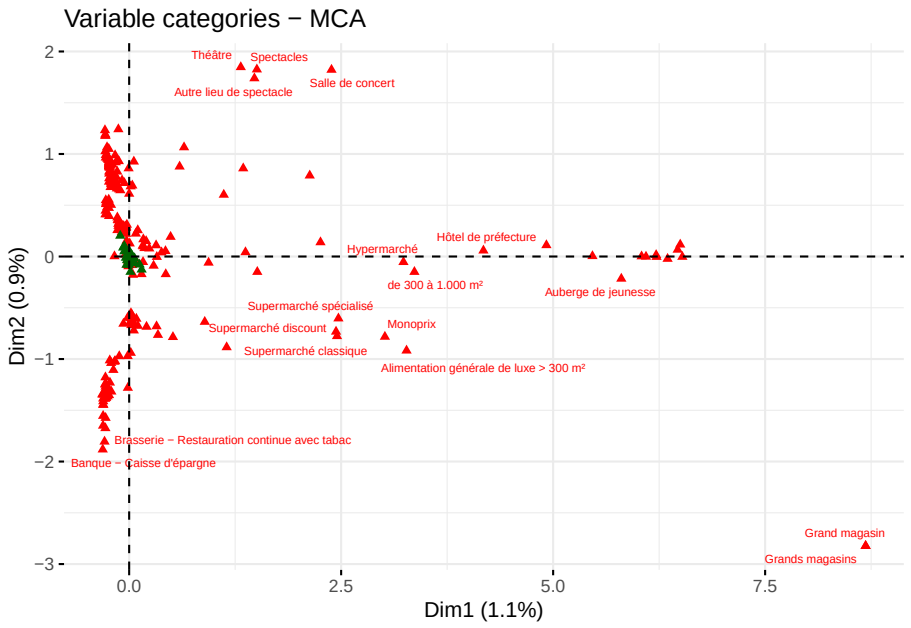
8.4.1 AFCM multiple

Très rapidement la méthode a été appliquée à une généralisation des tableaux croisés : le tableau de burt, ou son équivalent : le tableau disjonctif complet.

exemple

La mise en oeuvre par factominer permet d'employer une techniques de représentation de variables complémentaires : elles n'interviennent pas dans le calcul de la configuration factorielles, mais leurs positions dans l'espace sont calculées comme le barycentre des individus qui possède le trait considéré. Leur projection a un rôle illustratif.

```
library(FactoMineR)

foo<-df%>% dplyr::select(ACT, SURFACE, SITUATION, LIBACT, ARRONDISSEMENT)%>%
  as.matrix()

res<-MCA(foo,graph = FALSE,quali.sup=5)

fviz_mca_var(res, labelsize = 2, repel=TRUE)
```



remarques complémentaires : * pas de signification de l'inertie globale qui dépend de la structure du tableau (nombres de variables et de leurs modalités)

8.5 Développements

derrière les méthodes il y a un principe mathématique fondamental qui est au fondement de bien d'autres méthodes factorielles. C'est celle de la Singular Variance décomposition dont l'ACP est finalement un cas particulier.

8.5.1 le SVD

Le modèle mathématique fondamental

décomposer une matrice en plusieurs matrices l'acp une application à une matrice de nature particulières : la matrice de covariance ou de corrélation si standardisée

de nombreuses autres applications :

- à des matrices de comptage
- compression d'image
- information retrieval

d'autres méthodes s'appuient sur ce principe fondamental, et permettent de traiter des données textuelle .

On reporte le lecteur au chapitre X de du book NLP.

8.5.2 ACM , analyse canonique , analyse discriminante

Si ACP, AFC et AFCM ont pris le devant de la scène, bien d'autre méthodes analogues ont été développées

- ACM
- Analyse canonique
- Analyse factorielle discriminante qui a perdu du terrain au profit du modèle de régression logistique.

Un regain avec le machine learning et LSA, NFM etc..

8.6 En conclusion

1) une idée essentielle : réduire de nombreuses variables à un petit jeu de variables synthétiques

2) des méthodes au cœur de l'analyse des données

3) une autre idée essentielle : celle de vectoriser les données qu'on observe.

Chapter 9

Clustering

L'objectif des méthodes de classification automatique est de regrouper des observations qui se ressemblent sur un ensemble multidimensionnel de caractéristiques.

insérer image

Dans ce chapitre nous examinons deux familles de méthodes qui le distingue par la procédure de calcul : hierarchique d'une part, non hiérarchique de l'autre. On garde pour le chapître suivant l'étude des modèles de décisions qui ont une longue et riche histoire en marketing et ont préparé le développement de certains modèles de machine learning.

9.1 Les méthodes hiérarchiques ascendantes

Elles trouvent leur origine en biologie où dès les années 1930 Sokal et Sneath(Sneath and Sokal, 1973) ont proposé des méthodes pour analyser l'évolution des espèces. L'idée réside dans la comparison de specimens sur la base d'un certains nombre de caractéristiques, d'abord des caractères phénotypiques, puis dans ce domaine en s'appuyant sur les caractéristiques génétiques. Nous n'entrerons pas dans une discussion plus approfondis mais signalons que ces choix déterminent des méthodes et des hypothèses très différentes et largement débattues (cladistique etc)

Prenons le cas de différences phénotypiques et le tableau suivant.

tableau

Le but du jeu est de regrouper successivement les spécimens en fonction de leur ressemblance. L'algorithme consiste simplement à 1) calculer toutes les ressemblances deux à deux et 2) à fondre en une classe les deux éléments qui se

ressemble le plus. On réitère l'opération jusqu'à ce qu'on obtienne plus qu'une classe.

Le résultat est une arborescence dont chaque noeud représente un regrouppement de classe à un certain niveau de distance.

figure

Leurs variétés dépend de deux paramètres :

- le choix de la mesure de dissimilarités : Une distance euclidienne ? Son carré ? Une distance binaire comme l'indice de Jaccard?
- le choix de la méthode d'agrégation : que choisit-on pour calculer la distance entre deux classes A et B : la plus grande des distances entre les éléments de A et ceux de B ? La plus petite ? La distance moyennes, la médiane ?

9.1.1 Mise en oeuvre

On utilise l'enquête d'happydemics sur la période de fin mars.

```
library(lubridate)
df<-readRDS("./data/last.rds") %>%
  filter(date2>=make_datetime(year=2022, month=3, day = 19))


n_t<-nrow(df)

period<-" apres le 19 mars"
```

Il y a un trick de traitement des données. La question QCM a été encodée en une colonne, ajoutant les chaines de caractère des 16 thématiques avec un séparateurs $.

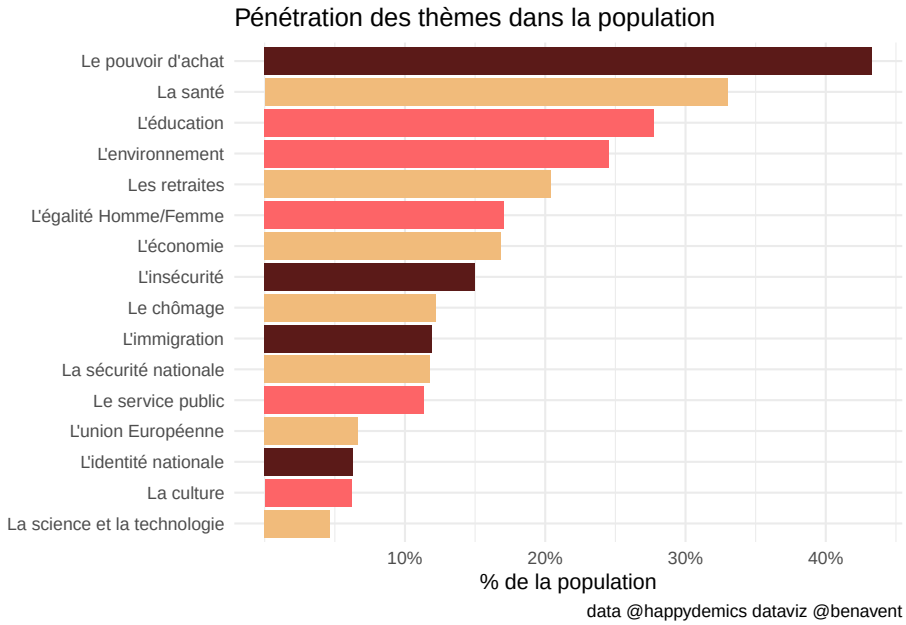
```
foo <-as.data.frame(str_split_fixed(df$themes, "\\$",n=3)) # On splite la colonne thèm

foo1<-cbind(df,foo)%>%
  rename(V1=23, V2=24, V3=25) %>%
  dplyr::select(id,V1,V2,V3)%>%
  pivot_longer(!id,names_to="rank",values_to="theme")%>%
  mutate(rank=ifelse(rank=="V1", 3,ifelse(rank=="V2", 2, ifelse(rank=="V3",1, 0)))) %>%
  filter(theme!="")%>%
  mutate(theme=str_trim(theme))%>%
  mutate(r=as.numeric(rank))%>%
  dplyr::select(-rank)
```

```
n1<-nrow(df) # le nombre d'individus
n2<-nrow(foo1) #le nombre de mentions
```

Dans une première étape faisons le bilan global

```
#on calcule la proportion et la pénétration des items

foo2 <-foo1%>%
  mutate(m=1)%>%
  group_by(theme)%>%
  summarise(frequence=sum(m),
            proportion=frequence/n2,
            penetration=frequence/n1)


col<-c("#F1BB7B",
       "#FD6467",
       "#FD6467",
       "#FD6467",
       "#5B1A18",
       "#5B1A18",
       "#5B1A18",
       "#F1BB7B",
       "#FD6467",
       "#F1BB7B",
       "#F1BB7B",
       "#F1BB7B",
       "#F1BB7B",
       "#5B1A18",
       "#FD6467",
       "#F1BB7B",
       "#F1BB7B"
       )

brks<-c(0.1, 0.2, 0.3,0.4,0.5,0.6)
ggplot(foo2,aes(x=reorder(theme, frequence), y=penetration))+
  geom_bar(stat="identity", aes(fill=theme))+
  coord_flip()+
  scale_fill_manual(values=col)+
  labs(title = "Pénétration des thèmes dans la population",
       x=NULL,
       y= "% de la population",
       caption = "data @happydemics dataviz @benavent")+
  theme_minimal()+
```

```
  theme(legend.position = "none")+
  scale_y_continuous(breaks = brks, labels = scales::percent(brks))
```




```
ggsave(paste0("./plot/theme_",period,".jpg"),plot=last_plot(), width = 27, height = 17
```

9.2 segmentation simplifiée

On commence va reconstruire un tableaux des individus x les thèmes. On garde les rangs comme indicateurs de l'importance .

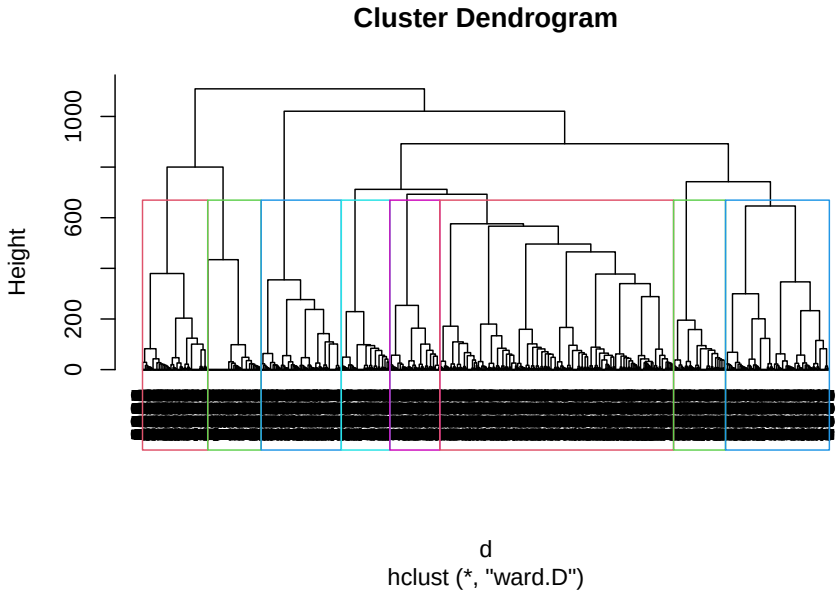
```
foo3<-foo1%>%
  pivot_wider(names_from="theme", values_from="r") %>%
  replace(is.na(.), 0)
head(foo3, 8)
```

```
## # A tibble: 8 x 17
##       id L'imm~1 Le po~2 L'édu~3 L'éga~4 Les r~5 L'env~6 Le se~7 L'ins~8 La sa~9
##    <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
## 1 2.36e8       3       0       0       0       0       0       0       0       0
## 2 2.36e8       0       3       2       1       0       0       0       0       0
## 3 2.36e8       0       3       0       0       2       0       0       0       0
```

```
## 4 2.36e8       0       0       0       0       3       2       1       0       0
## 5 2.36e8       0       1       0       0       3       0       2       0       0
## 6 2.36e8       0       2       3       0       0       1       0       0       0
## 7 2.36e8       0       0       3       2       0       1       0       0       0
## 8 2.36e8       1       2       0       0       0       0       0       3       0
## # ... with 7 more variables: `Le chômage` <dbl>, `L'économie` <dbl>,
## #   `La science et la technologie` <dbl>, `L'identité nationale` <dbl>,
## #   `La sécurité nationale` <dbl>, `L'union Européenne` <dbl>,
## #   `La culture` <dbl>, and abbreviated variable names 1: `L'immigration`,
## #   2: `Le pouvoir d'achat`, 3: `L'éducation`, 4: `L'égalité Homme/Femme`,
## #   5: `Les retraites`, 6: `L'environnement`, 7: `Le service public`,
## #   8: `L'insécurité`, 9: `La santé`
```

On calcule un tableau de distance et on performe la classification automatique.
dans cet essai on tente un modèle à 8 groupes.

```r
foo4<-foo3[,2:17]

#distance
d<-dist(foo4)

#clustering
h.D  <- hclust(d, method="ward.D")

#dendogramme
plot(h.D,  hang=-1)

#identification des clusters
rect.hclust(h.D , k = 8, border = 2:6)
```

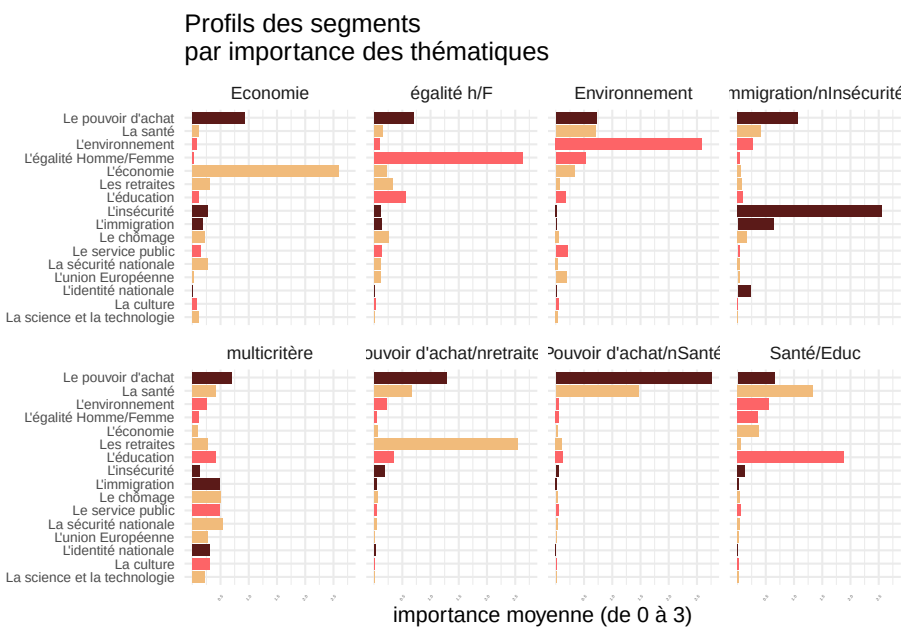



```
#attribution des clusters
memb <- cutree(h.D, k = 8)

#maj du fichier de données avec l'appartenace des individus aux groupes
foo5<-cbind(foo4, memb)
```

Il reste à décrire les différents types sur les 16 variables qui les décrivent. On choisit une méthode de barre ordonnée avec un facetting par groupe.

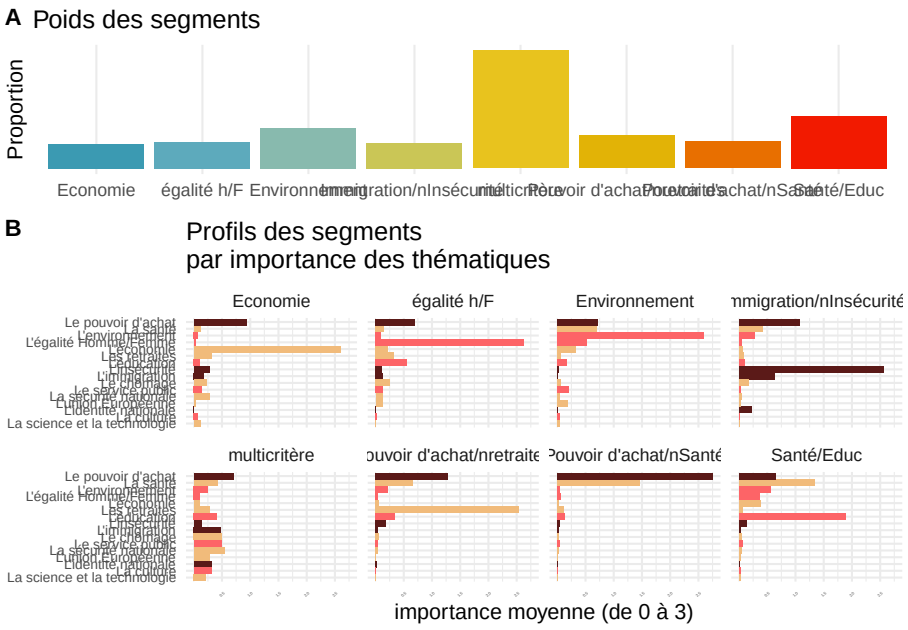
```
foo6<-foo5 %>%
  group_by(memb) %>%
  pivot_longer(-memb,names_to="Thèmes",values_to="Valeurs")%>%
  group_by(memb,Thèmes)%>%
  summarise(Valeurs=mean(Valeurs))

foo6$group[foo6$memb==1]<-"multicritère"
foo6$group[foo6$memb==2]<-"Santé/Educ"
foo6$group[foo6$memb==3]<-"Pouvoir d'achat/nretraites"
foo6$group[foo6$memb==5]<-"Immigration/nInsécurité "
foo6$group[foo6$memb==4]<-"égalité h/F"
foo6$group[foo6$memb==6]<-"Pouvoir d'achat/nSanté"
foo6$group[foo6$memb==7]<-"Economie"
foo6$group[foo6$memb==8]<-"Environnement"
```

```
library(scales)
brks<-c(0.5,1,1.5,2, 2.5,3)
p2<- ggplot(foo6, aes(x=reorder(Thèmes, Valeurs), y=Valeurs))+
  geom_bar(stat="identity",aes(fill=as.factor(Thèmes)))+
  facet_wrap(vars(group), ncol=4)+
  coord_flip()+
  scale_fill_manual(values=col)+
  theme_minimal()+
  scale_y_continuous(breaks=brks)+
  theme(legend.position = "none", axis.text=element_text(size=7),axis.text.x=element_text(angle =
  labs(title = "Profils des segments\npar importance des thématiques", x=NULL, y="importance moye

p2
```




```
ggsave("./plot/g_segment_p2.jpg",plot=last_plot(), width = 27, height = 17, units = "cm")

library(wesanderson)
seg_col<-wes_palette("Zissou1", 8, type = "continuous")
n<-nrow(foo5)
foo6<-foo5 %>% mutate(n=1) %>%
  group_by(memb)%>%
  summarise(freq=sum(n, na.rm=TRUE))%>% mutate( freq=freq/n)
```

```
foo6$group[foo6$memb==1]<-"multicritère"
foo6$group[foo6$memb==2]<-"Santé/Educ"
foo6$group[foo6$memb==3]<-"Pouvoir d'achat/nretraites"
foo6$group[foo6$memb==5]<-"Immigration/nInsécurité "
foo6$group[foo6$memb==4]<-"égalité h/F"
foo6$group[foo6$memb==6]<-"Pouvoir d'achat/nSanté"
foo6$group[foo6$memb==7]<-"Economie"
foo6$group[foo6$memb==8]<-"Environnement"


p1<- ggplot(foo6, aes(x=group, y=freq))+
  geom_bar(stat="identity", aes(fill=group))+
  scale_fill_manual(values=seg_col) +
  theme_minimal()+
  labs(title="Poids des segments", x=NULL, y="Proportion")+
  scale_y_continuous(breaks=brks,labels=percent)+
  theme(legend.position = "none")

ggsave("./plot/g_segment_p1.jpg",plot=last_plot(), width = 27, height = 17, units = "cm

plot_grid(p1, p2, labels = c('A', 'B'), label_size = 12, ncol=1,rel_heights =  c(1, 2))
```



```
ggsave("./plot/g_segment.jpg",plot=last_plot(), width = 27, height = 17, units = "cm")
```

9.3 tableaux croisés de la typologie et des critères sociaux démos

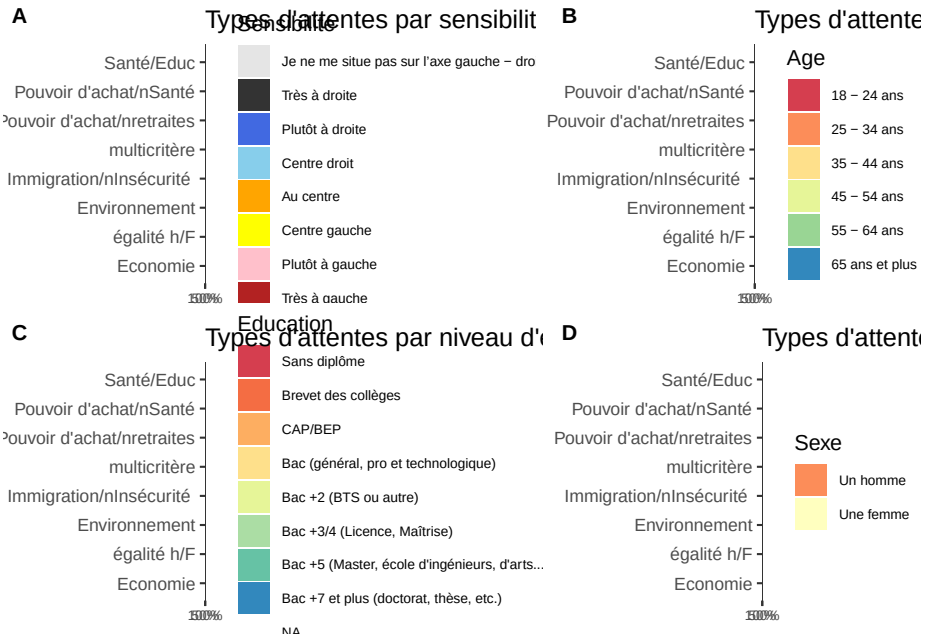
On revient à une approche descriptive, on croisant successivement notre variable typologie avec les critères socio-demo qui ont été mesurés dans l'enquête.

(une boucle simplifierait !)

```
df<-cbind(df,foo5)

df$group[df$memb==1]<-"multicritère"
df$group[df$memb==2]<-"Santé/Educ"
df$group[df$memb==3]<-"Pouvoir d'achat/nretraites"
df$group[df$memb==5]<-"Immigration/nInsécurité "
df$group[df$memb==4]<-"égalité h/F"
df$group[df$memb==6]<-"Pouvoir d'achat/nSanté"
df$group[df$memb==7]<-"Economie"
df$group[df$memb==8]<-"Environnement"


foo<-df %>%
  group_by(group, Sensibilité) %>%
   summarize(n=n())%>%
  mutate(prop=round(n/sum(n),3), cum=1 - (cumsum(prop)-prop/2))

g01<-ggplot(foo,aes(x=group, y=prop, group=Sensibilité))+
  geom_bar(stat="identity",aes(y = prop, fill=Sensibilité)) +
 scale_y_continuous(breaks = brks, labels = scales::percent(brks)) +
  scale_fill_manual(values=SensiP2) +
  geom_text(aes(label = prop, y=cum),size=2,color="white", vjust = 0.5)+
  coord_flip()+
  labs(title = "Types d'attentes par sensibilité politique ",
       x=NULL, y=NULL,)+
  theme_bw()+  theme(axis.text.x = element_text(size = 7), legend.text = element_text(size = 7))



ggsave("./plot/g_segment01.jpg",plot=last_plot(), width = 27, height = 17, units = "cm")


foo<-df %>%
```

```
  group_by(group, Age) %>%
   summarize(n=n())%>%
  mutate(prop=round(n/sum(n),3), cum=1 - (cumsum(prop)-prop/2))

g02<-ggplot(foo,aes(x=group, y=prop, group=Age))+
  geom_bar(stat="identity",aes(y = prop, fill=Age)) +
 scale_y_continuous(breaks = brks, labels = scales::percent(brks)) +
  scale_fill_brewer(palette="Spectral") + geom_text(aes(label = prop, y=cum),size=2,co
  coord_flip()+
  labs(title = "Types d'attentes par classe d'âge ",
       x=NULL, y=NULL,)+theme_bw()  +
  theme(axis.text.x = element_text(size = 7), legend.text = element_text(size = 7))



ggsave("./plot/g_segment02.jpg",plot=last_plot(), width = 27, height = 17, units = "cm


foo<-df %>%
  group_by(group, Sexe) %>%
   summarize(n=n())%>%
  mutate(prop=round(n/sum(n),3), cum=1 - (cumsum(prop)-prop/2))

g03<-ggplot(foo,aes(x=group, y=prop, group=Sexe))+
  geom_bar(stat="identity",aes(y = prop, fill=Sexe)) +
 scale_y_continuous(breaks = brks, labels = scales::percent(brks)) +
  scale_fill_brewer(palette="Spectral") + geom_text(aes(label = prop, y=cum),size=2,co
  coord_flip()+  theme_bw()+
  labs(title = "Types d'attentes par genre ",
       x=NULL, y=NULL,)+
  theme(axis.text.x = element_text(size = 7), legend.text = element_text(size = 7))

ggsave("./plot/g_segment03.jpg",plot=last_plot(), width = 27, height = 17, units = "cm

foo<-df %>%
  group_by(group, Education) %>%
   summarize(n=n())%>%
  mutate(prop=round(n/sum(n),3), cum=1 - (cumsum(prop)-prop/2))

g04<-ggplot(foo,aes(x=group, y=prop, group=Education))+
  geom_bar(stat="identity",aes(y = prop, fill=Education)) +
 scale_y_continuous(breaks = brks, labels = scales::percent(brks)) +
  scale_fill_brewer(palette="Spectral") +
  geom_text(aes(label = prop, y=cum),size=1.5,color="white", vjust = 0.5)+
  coord_flip()+theme_bw()+
```

```
  labs(title = "Types d'attentes par niveau d'éducation ",
       x=NULL, y=NULL,)+
  theme(axis.text.x = element_text(size = 7), legend.text = element_text(size = 7))



ggsave("./plot/g_segment04.jpg",plot=last_plot(), width = 27, height = 17, units = "cm")


plot_grid(g01, g02, g04,g03,  labels = c('A', 'B', 'C', 'D'), label_size = 11, ncol=2,rel_widths
```



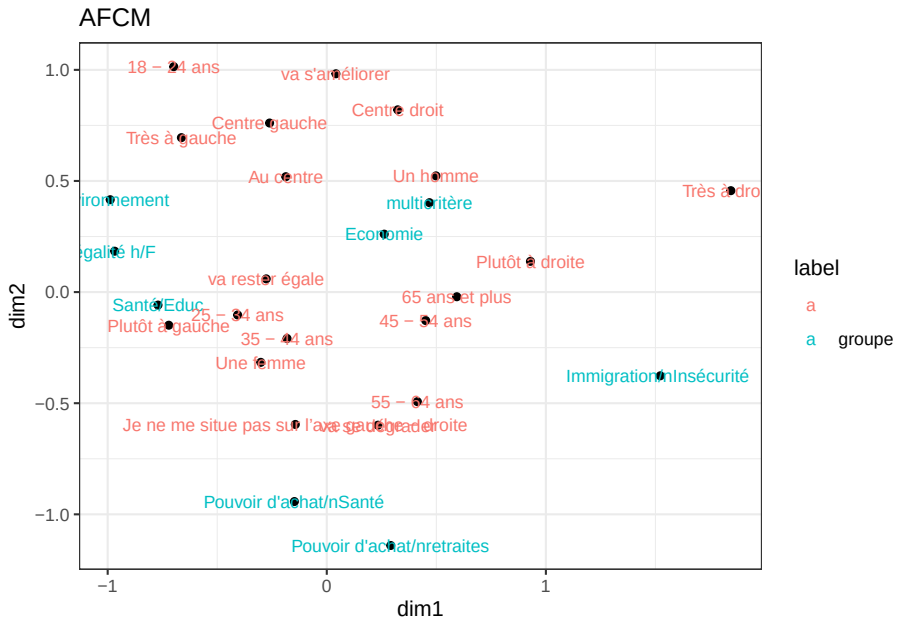

```
ggsave("./plot/g_segment05.jpg",plot=last_plot(), width = 27, height = 17, units = "cm")
```

9.4 AFCM pour une synthèse

C'est le bon moment de donner une seconde illustration de l'utilité de l'AFCM. Pourquoi ne pas synthétiser en une carte l'ensemble des relations statistiques.

```
library(FactoMineR)
library(factoextra)
X<-df %>% dplyr::select( group, Age, Sexe, Sensibilité, Situation2)
res<-MCA(X, graph =FALSE)
```

```
foo<-as.data.frame(res$var$coord) %>%
  rownames_to_column(var="var")%>%
  rename(dim1=2, dim2=3) %>%
  add_rownames(var = "rowname")

foo$rowname<-as.numeric(foo$rowname)
foo<-foo %>%   mutate(label=ifelse(rowname<9, "groupe", ""))



ggplot(foo, aes(x=dim1, y=dim2, label= var) )+
  geom_point()+
  geom_text(aes(label=var, color=label),size=3)+
  theme_bw()+labs( title= "AFCM")
```




```
  theme(legend.position = "none")
```

```
## List of 1
##  $ legend.position: chr "none"
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi FALSE
##  - attr(*, "validate")= logi TRUE
```

9.4.1 Forces et limites

- forces : graphiques, complète
- limites : petite population

9.5 Les méthodes non-hiérarchiques

La première d'entre elles est la méthode k-means dont le principe est très simple : plutôt que de calculer toutes les distances entre tous les objets, on va se concentrer sur les distances en k group supposés et les n individus. L'hyperparamètre est ici le nombre de groupes

9.5.1 principe

9.5.2 Application

9.5.3 Le problème de la détermination du nombre optimal de groupe

- méthode du coude
- méthode silhouette
- gap statistics

9.6 Autres méthodes

de nombreuses variantes sont disponibles

- mediane
- kernel
- les méthodes fuzzy : l'appartenance n'est pas exclusive mais probabilistique
- les méthode de classes latentes
- les méthodes de densités s'appuie sur l'idée que la continuité d'un groupe s'exprime en terme s de densités ** paramètriques ** non - paramètriques http://www.sthda.com/english/wiki/wiki.php?id_contents=7940

: https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-30164-8_211

9.7 Conclusion

Chapter 10

Régression

10.1 Quelques éléments de théorie

En statistique, la régression linéaire multiple est une méthode qui étend la régression linéaire simple pour décrire les variations d'une variable endogène associée aux variations de plusieurs variables exogènes.

Modèle théorique Étant donné un échantillon (Yi, Xi1, ..., Xip) avec i {1, n} , on cherche à expliquer, avec le plus de précision possible, les valeurs prises par Yi, dite variable endogène, à partir d'une série de variables explicatives Xi1, ..., Xip. Le modèle théorique, formulé en termes de variables aléatoires, prend la forme suivante.

$$y_i = a_0 + a_1 x_{i1} + a_2 x_{i2} + ... + a_p x_{ip} + \varepsilon_i, \qquad i = 1, ..., n$$

où i est l'erreur du modèle qui exprime l'information manquante dans l'explication linéaire, c'est la partie stochastique, la relation linéaire capturant la partie déterministe. Les enjeux sont :

- estimer les paramètres a0, a1, ..., ap en exploitant les observations ;
- évaluer la précision de ces estimateurs ;
- mesurer le pouvoir explicatif du modèle ;
- évaluer l'influence des variables dans le modèle :globalement et pour chacune des variables ;
- évaluer la qualité du modèle lors de la prédiction (intervalle de prédiction) ;
- détecter les observations qui peuvent influencer exagérément les résultats (points atypiques).

Soit de manière compacte en notation matricielle, on peut écrire plus simplement :

$$y = Xa + \varepsilon$$

où

y est de dimension (n, 1). C'est le vecteur des valeurs des n individu i que l'on cherche à prédire. X est de dimension (n, p+1) où p est le nombre de variable. (le +1 correspond au terme constant) a est de dimension (p+1, 1), c'est le tableau des paramètres associés que l'on cherche à estimer.La première colonne de la matrice X sert à indiquer que la régression est effectuée avec constante (ici a_0. est de dimension (n, 1); c'est l'écart entre les valeurs observées et les valeur calculée.

Un certain nombre d'hypothèses sont ajoutées pour déterminer les propriétés des estimateurs (biais, convergence) ; et leurs lois de distributions (pour les estimations par intervalle et les tests d'hypothèses).

Il existe principalement deux catégories d'hypothèses :

- Hypothèses stochastiques H1 : Les X_j sont déterminées sans erreurs (pas d'erreur de mesure) H2 : $\mathbb{E}(\varepsilon_i) = 0$ Le modèle est bien spécifié en moyenne H3 : $\mathrm{Var}(\varepsilon_i) = \sigma^2$$ avec
$$\forall i \neq j$$
. C'est l'hypothèse d'homoscédasticité des erreurs (variance constante) H4 : $\mathrm{cov}(\varepsilon_i, \varepsilon_j) = 0$ avec $ i j $. Pas d'autocorrélation des erreurs. H5 : $\mathrm{cov}(X_i, \varepsilon_j) = 0$ avec $\forall i \neq j$ Les erreurs sont linéairement indépendantes des variables exogènes. H6 : $\varepsilon \sim \mathcal{N}_n(0, \sigma^2)$ Les erreurs suivent une loi normale multidimensionnelle

(H6 implique les hypothèses H2, H3 et H4, la réciproque étant fausse car les trois hypothèses réunies n'impliquent pas que soit un vecteur gaussien).

Hypothèses structurelles H7 : absence de colinéarité entre les variables explicatives, H8 : $\frac{1}{n}X^TX$ tend vers une matrice finie non singulière Q lorsque n $\to +\infty$; H9 : $n > p + 1$ Le nombre d'observations est strictement supérieur au nombre de variables + 1 (la constante). S'il y avait égalité, le nombre d'équations serait égal au nombre d'inconnues aj, la droite de régression passerait par tous les points.

La méthode d'estimation des paramètres est celle des moindres carrés ordinaires. Elle consiste à Minimiser la somme des carrés des erreurs la somme des carrés des résidus.

$$\min \sum_{i=1}^{n} \hat{\epsilon}_i^2 = \min_{\hat{a}_0,..,\hat{a}_p} \sum_{i=1}^{n} (y_i - \hat{a}_0 - \hat{a}_1 x_{i,1} - \cdots - \hat{a}_p x_{i,p})^2$$

La solution est trouvée en dérivant cette quantité par rapport à chacun des paramètres et en l'égalant à 0. La solution obtenue est l'estimateur des moindres carrés ordinaires, il s'écrit :

$\hat{a} = (X^T X)^{-1} X^T Y$ est l'estimateur qui minimise la somme des carrés des résidus. Ici la solution est analytique, d'autre méthodes d'estimation demandent des procédures itératives.

10.2 Une étude de cas : les offres Blablacar

C'est un jeu de données scrappé sur un site de covoiturage constitué de 13000 offres proposées sur différents types de trajet à un moment donné (au cours de 2016). Le but va être de tester l'effet des signaux de qualité sur les taux de réservation, et de mieux comprendre la nature d'une plateforme qui étant de fait une place de marché se propose d'être collaborative.

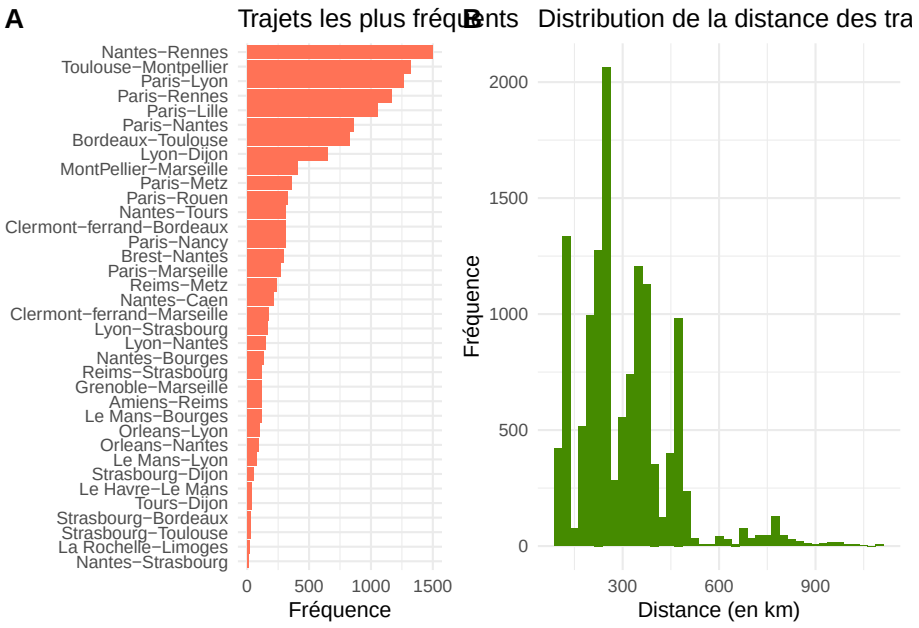
Si elles ne sont plus d'actualité, elle permettent cependant une étude intéressante des facteurs qui encourage la demande dans un moment où le modèle DREAMS de Blablacar était mis en avant pour résoudre le problème de la confiance dans le monde digital : comment échanger avec des inconnus.

```
df <- read_delim("./data/covoit.csv", ";", escape_double = FALSE, trim_ws = TRUE)
df<- df%>% dplyr::rename(Places=`Places Restantes`,
                  Depart=`Ville Depart Capture`,
                  Arrivee=`Ville Arrivee Capture`,
                  Flex=`Flexibilite Horaire`
                  )
df<-df %>%
  filter( Distance<1200)

#on recode le nombre de places restantes par une approximation
#du taux de réservation
df$Occup<- NA
df$Occup[df$Places==0]<-1
df$Occup[df$Places==1]<-.75
df$Occup[df$Places==2]<-.50
df$Occup[df$Places==3]<-.25
df$Occup[df$Places==4]<-0
df$Occup[df$Places>=4]<-0

#l'expérience peut se capter par le nombre de voyages
df$Nombre[is.na(df$Nombre)]<-0
```

Voici la liste des requêtes et le nombre d'offres obtenues pour chacune d'elle. On observe la domination des trajet inter-régionaux avec Nantes-Rennes, Toulouse-Montpellier et Bordeaux-Toulouse. Les trajets ont été échantillonnés pour représenter différents niveaux d'échelle de distance, les courts trajets, les trajets de 100 à 200 km, ceux à 300-400 et une minorité de distances plus longues, ce qui explique la concentration à certains niveaux.

La distribution ne représente pas la distribution de la demande des trajets ou de l'offre, mais un pool d'offres obtenus pour une séries de requête relatives à un trajet donné, un jour donné.

```
trajet<-table(df$Depart, df$Arrivee)
df$trajet<-paste0(df$Depart,"-",df$Arrivee)
foo<-df %>% mutate(n=1)%>%
  group_by(trajet)%>%
  summarise(n=sum(n))%>%filter(n>0)

g01<-ggplot(foo,aes(x=reorder(trajet,n), y=n))+
  geom_bar(stat="identity", fill="coral1")+
  coord_flip()+
  labs(title="Trajets les plus fréquents", x=NULL, y="Fréquence")

g02<-ggplot(df, aes(x=Distance))+
  geom_histogram(fill="Chartreuse4",binwidth = 25) +
  labs(title = "Distribution de la distance des trajets",
       y="Fréquence",
       x= "Distance (en km)")

plot_grid(
  g01, g02,
  labels = "AUTO", ncol = 2
)
```



Les caractérisques de l'offre

l'offre se détermine par plusieurs éléments

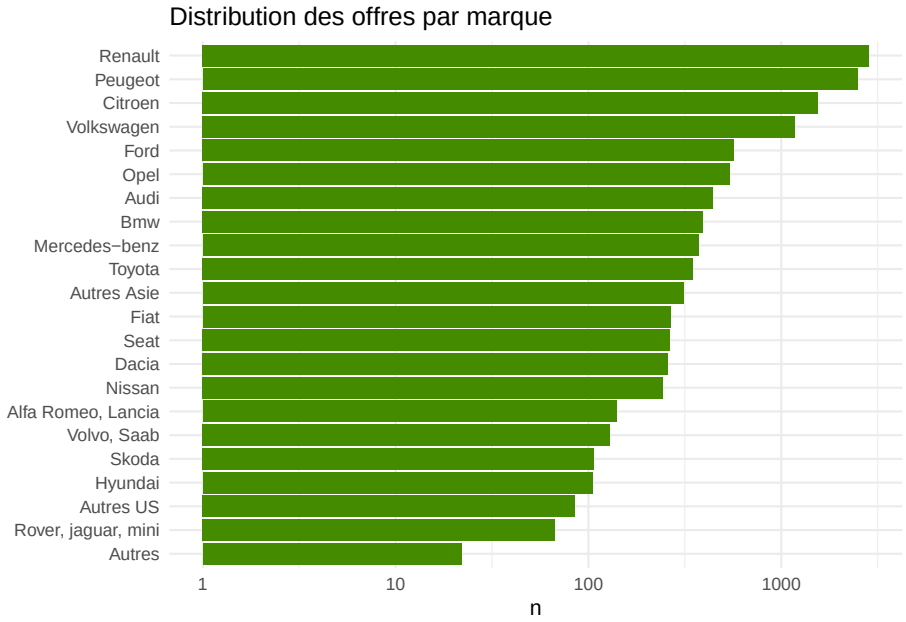
- le véhicule
- le conducteur
- les conditions du trajets

10.2.0.1 Le véhicule

Rôle de la marque et du standing du véhicule mérite plus de recodage. Il est très subjectif, ici on privilégie les origines nationales qui expriment un style, un esprit d'automobile.

```
df$Marque[df$Marque=="Alfa-romeo"]<-"Alfa Romeo, Lancia"
df$Marque[df$Marque=="Alfa Romeo"]<-"Alfa Romeo, Lancia"
df$Marque[df$Marque=="Lancia"]<-"Alfa Romeo, Lancia"
df$Marque[df$Marque=="Volvo"]<-"Volvo, Saab"
df$Marque[df$Marque=="Zx"]<-"Citroen"
df$Marque[df$Marque=="Saab"]<-"Volvo, Saab"
df$Marque[df$Marque=="Audi-quattro"]<-"Audi"
df$Marque[df$Marque=="Mercedes"]<-"Mercedes-benz"
df$Marque[df$Marque=="Vw"]<-"Volkswagen"

df$Marque[df$Marque=="Kia"]<-"Autres Asie"
df$Marque[df$Marque=="Honda"]<-"Autres Asie"
df$Marque[df$Marque=="Mazda"]<-"Autres Asie"
df$Marque[df$Marque=="Suzuki"]<-"Autres Asie"
df$Marque[df$Marque=="Isuzu"]<-"Autres Asie"
df$Marque[df$Marque=="Mitsubishi"]<-"Autres Asie"
df$Marque[df$Marque=="Lexus"]<-"Autres Asie"
df$Marque[df$Marque=="Subaru"]<-"Autres Asie"
df$Marque[df$Marque=="Daewoo"]<-"Autres Asie"
df$Marque[df$Marque=="Huanghai"]<-"Autres Asie"


df$Marque[df$Marque=="Land"]<-"Rover, jaguar, mini"
df$Marque[df$Marque=="Rover"]<-"Rover, jaguar, mini"
df$Marque[df$Marque=="Jaguar"]<-"Rover, jaguar, mini"
df$Marque[df$Marque=="Abarth"]<-"Rover, jaguar, mini"
df$Marque[df$Marque=="Ldv"]<-"Rover, jaguar, mini"
df$Marque[df$Marque=="Austin"]<-"Rover, jaguar, mini"
df$Marque[df$Marque=="Mini"]<-"Rover, jaguar, mini"
df$Marque[df$Marque=="Chevrolet"]<-"Autres US"
df$Marque[df$Marque=="Chrysler"]<-"Autres US"
```

```
df$Marque[df$Marque=="Jeep"]<-"Autres US"
df$Marque[df$Marque=="Dodge"]<-"Autres US"
df$Marque[df$Marque=="Lamborghini"]<-"Sport"
df$Marque[df$Marque=="Maserati"]<-"Sport"
df$Marque[df$Marque=="Porsche"]<-"Sport"
df$Marque[df$Marque=="Ac"]<-"Autres"
df$Marque[df$Marque=="Acura"]<-"Autres"
df$Marque[df$Marque=="eacute"]<-"Autres"
df$Marque[df$Marque=="Iveco"]<-"Autres"
df$Marque[df$Marque=="Camping-car"]<-"Autres"
df$Marque[df$Marque=="Infiniti"]<-"Autres"
df$Marque[df$Marque=="Admiral"]<-"Autres"
df$Marque[df$Marque=="Sport"]<-"Autres"

foo<-df %>%
  group_by(Marque)%>%
  summarise(n=n())%>%
  drop_na()

ggplot(foo, aes(x=reorder(Marque, n), y=n))+
  geom_bar(stat="identity",fill="Chartreuse4") +
    coord_flip() +
    scale_y_log10()+
    labs(title= "Distribution des offres par marque", x=NULL)
```

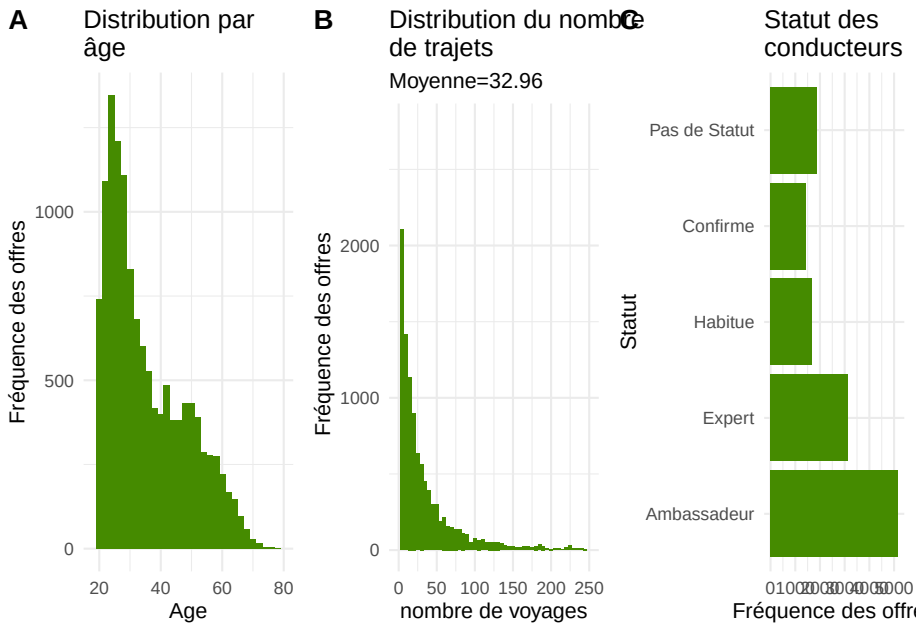


10.2.0.2 L'âge et l'expérience du capitaine

On distingue deux populations en terme d'âge, une en dessous de la trentaine, l'autre de 40 à 50 ans. On Jette un coup d'oeil ensuite sur la relation entre l'âge et la note qui culmine à 30 ans et baisse avec les décades. Est-ce l'effet d'une inadéquation des âges? La demande est-elle plus jeune que l'offre? Cela crée-t-il un biais systématique d'évaluation?

Le conducteur se manifeste au travers de 3 critères : le statut attribué par blablacar, son expérience traduite par le nombre de voyages qu'il a fait (et pour lesquels il a été évalué). La note moyenne obtenue des passagers.

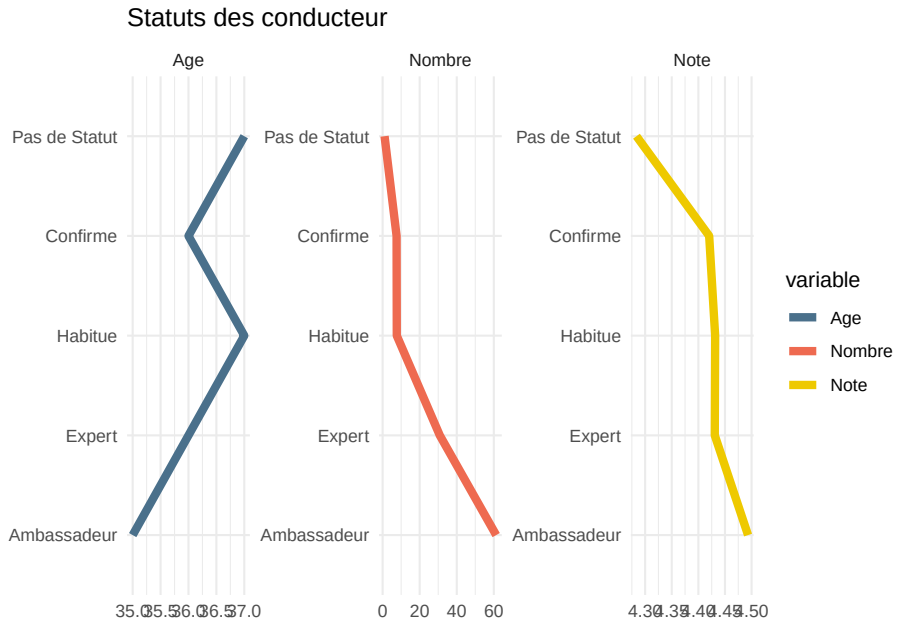
```
g03<-ggplot(df, aes(x=Age))+
  geom_histogram(fill="Chartreuse4", binwidth = 2)+
  labs(title="Distribution par \nâge", y="Fréquence des offres")+xlim (18, 80)

df$Statut<-as.factor(df$Statut)
df$Statut <- factor(df$Statut, ordered = TRUE,
                                levels = c("Ambassadeur", "Expert",  "Habitue", "Confirme","Pas d

g04<-ggplot(df, aes(x=Statut))+
  geom_bar(fill="Chartreuse4") +
  coord_flip()+
  labs(title = "Statut des \nconducteurs", y="Fréquence des offres")

mean<-round(mean(df$Nombre,na.rm=TRUE),2)

g05<-ggplot(df, aes(x=Nombre))+
  geom_histogram(fill="Chartreuse4", binwidth = 5)+
  labs(title = "Distribution du nombre \nde trajets", y=" Fréquence des offres",
       subtitle =paste0("Moyenne=", mean),
       x="nombre de voyages")+
  xlim(0,250)


plot_grid(
  g03, g05,g04,
  labels = "AUTO", ncol = 3
)
```



L'expérience se distribue de manière très inégale, une minorité de conducteurs ayant réalisé plus de 50 voyages, une très grande majorité en ayant fait moins d'une vingtaine de voyages.

Examinons le statut et notamment en comparant l'expérience et l'évaluation.

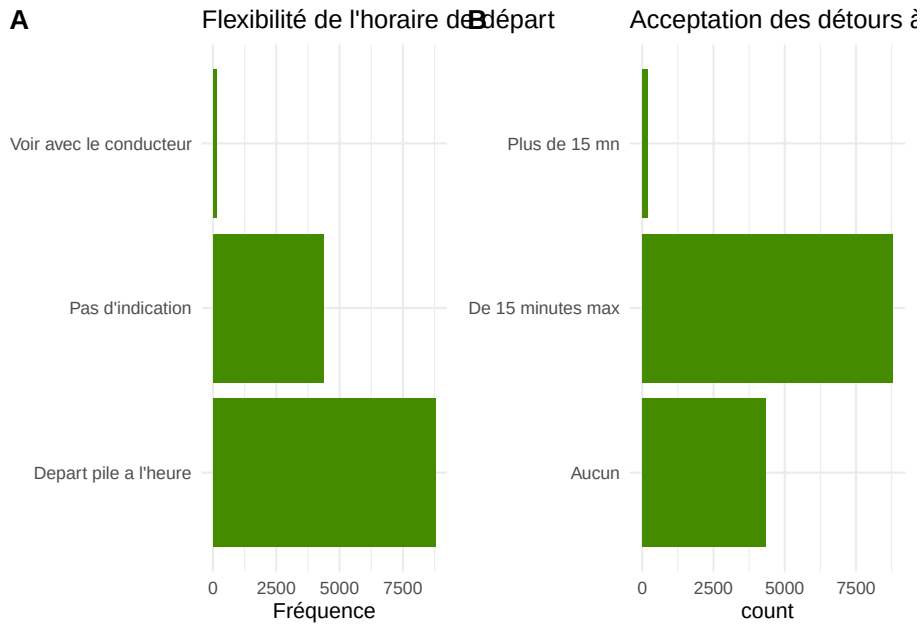
```
foo<-df %>%
  group_by(Statut)%>%
  summarise(Note=mean(Note,na.rm=TRUE),
            Nombre=mean(Nombre,na.rm=TRUE),
            Age=round(mean(Age,na.rm=TRUE),0))%>%
  pivot_longer(-Statut,names_to = "variable",values_to = "Moyenne" )


ggplot(foo, aes(x=Statut, y=Moyenne,group=variable))+
  geom_line(aes(color=variable), size=2)+
  coord_flip()+
  facet_wrap(vars(variable),scales ="free", ncol=3)+
  labs(title = "Statuts des conducteur", x=NULL, y=NULL)+
  scale_color_manual(values=c("skyblue4", "coral2", "Gold2"))
```



10.2.0.3 Le trajet

Le conducteur peut être flexible dans l'horaire de départ ou pas. Un recodage est cependant nécessaire.Le trajet peut être strict ou comporter des détours. Là aussi besoin d'un peu de recodage.

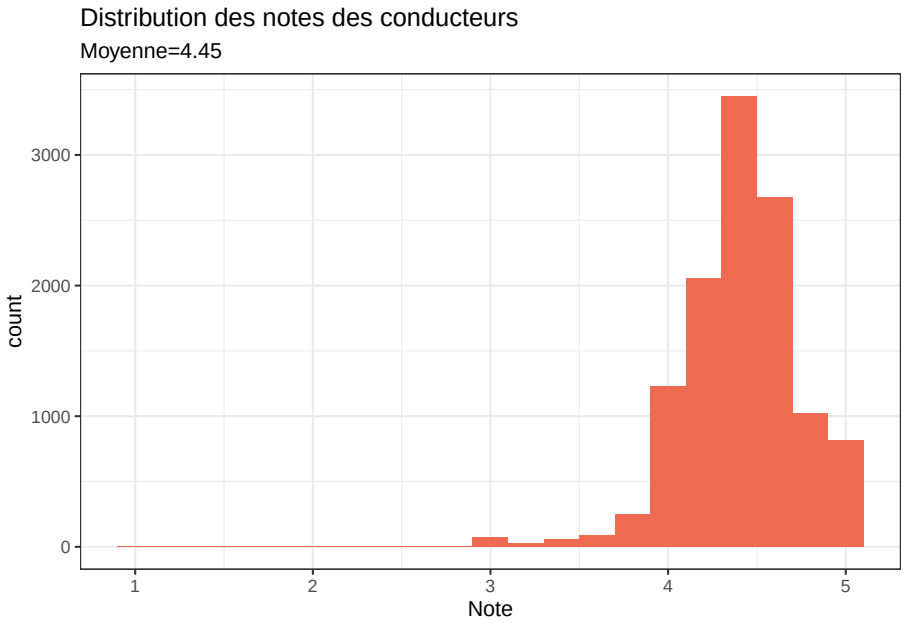
```
df$Flex[df$Flex!="Depart pile a l'heure" & df$Flex!="Pas d'indication"]<-"Voir avec le conducteur
df$Flex[is.na(df$Flex)]<-"Pas d'indication"
g07<-ggplot(df, aes(x=Flex))+
  geom_bar(fill="Chartreuse4")+
  coord_flip()+
  labs(title="Flexibilité de l'horaire de départ", x=NULL, y="Fréquence")

df$Detour[df$Detour=="De 30 minutes max"]<-"Plus de 15 mn"
df$Detour[df$Detour=="Autant que possible"]<-"Plus de 15 mn"
g08<-ggplot(df, aes(x=Detour))+geom_bar(fill="Chartreuse4")  +
  coord_flip()+labs(title="Acceptation des détours à l'arrivée", x=NULL)

plot_grid(
  g07, g08,
  labels = "AUTO", ncol = 2
)
```



10.3 Notes, prix et taux d'occupations

Les notes sont en moyenne de 4.45 et fortement déviées à droite, réduisant la capacité de discrimination.

```
mean<-round(mean(df$Note,na.rm=TRUE),2)
ggplot(df, aes(x=Note))+
  geom_histogram(fill="coral2",binwidth = .2)+
  theme_bw() +
  labs(title = "Distribution des notes des conducteurs", subtitle =paste0("Moyenne=", m
```



Analyse des prix

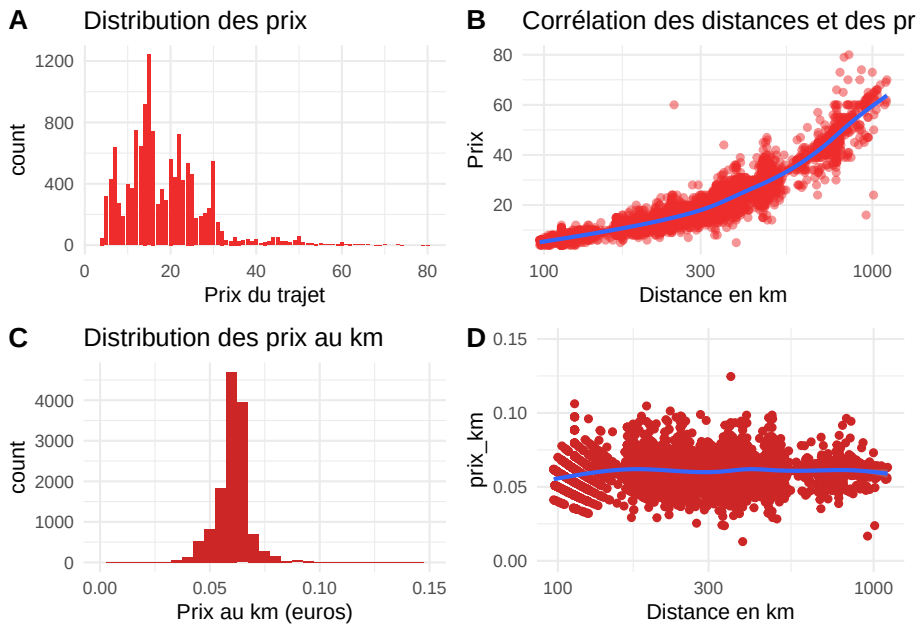
Au premier examen la distribution des prix semble être multimodale, elle est étroitement associée à la distribution des distance et de notre stratégie d'échantillonage des paires départ/destination.

Le prix est une multiplication de la distance par un tarif kilométrique, même si la relation ne semble pas tout à faire linéaire. La convexité de la courbe signale une sorte de rendement croissant avec la distance (incorporation du prix des péages ou effet de rareté ?).

C'est pourquoi on calcule un tarif au km, qui lui n'est plus corrélé ou à peine à la distance parcourue. Voici les résultats principaux.

```
g20<-ggplot(df, aes(x=Prix))+geom_bar(fill="firebrick2")  +
  labs(title = "Distribution des prix",
       x="Prix du trajet")

g21<-ggplot(df, aes(x=Distance, y=Prix))+
  geom_point(color="firebrick2", alpha =0.5) +
  geom_smooth(method="gam")+scale_x_log10()+
  labs(title = "Corrélation des distances et des prix", x= "Distance en km")

df$prix_km<-df$Prix/df$Distance

g22<-ggplot(df, aes(x=prix_km))+
  geom_histogram(fill="firebrick3", binwidth = 0.005)  +
```

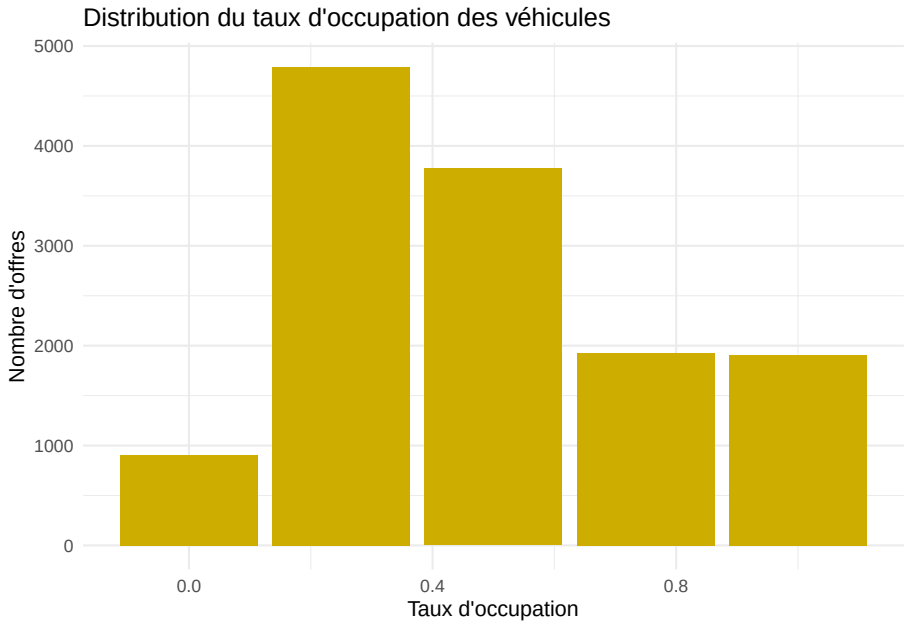
```
  labs(title = "Distribution des prix au km", x="Prix au km (euros)")+xlim(0,0.15)
g23<-ggplot(df, aes(x=Distance, y=prix_km))+
  geom_point(color="firebrick3") +
  geom_smooth(method="gam")+
  scale_x_log10()+labs(x="Distance en km")+
  ylim(0,0.15)

plot_grid(
  g20, g21,g22,g23,
  labels = "AUTO", ncol = 2
)
```



10.4 Analyser la demande : qu'est ce qui détermine le taux d'occupation ?

On utilise la variable nombre de place restante qu'on traduit par un indicateur codé de 0 (tout est libre) à 1 (la voiture est pleine). En voici la distribution.

```
ggplot(df, aes(x=Occup))+
  geom_bar(fill="gold3")  +
  labs(title="Distribution du taux d'occupation des véhicules",
       y="Nombre d'offres", x= "Taux d'occupation")
```



10.4.1 Un modèle OLS

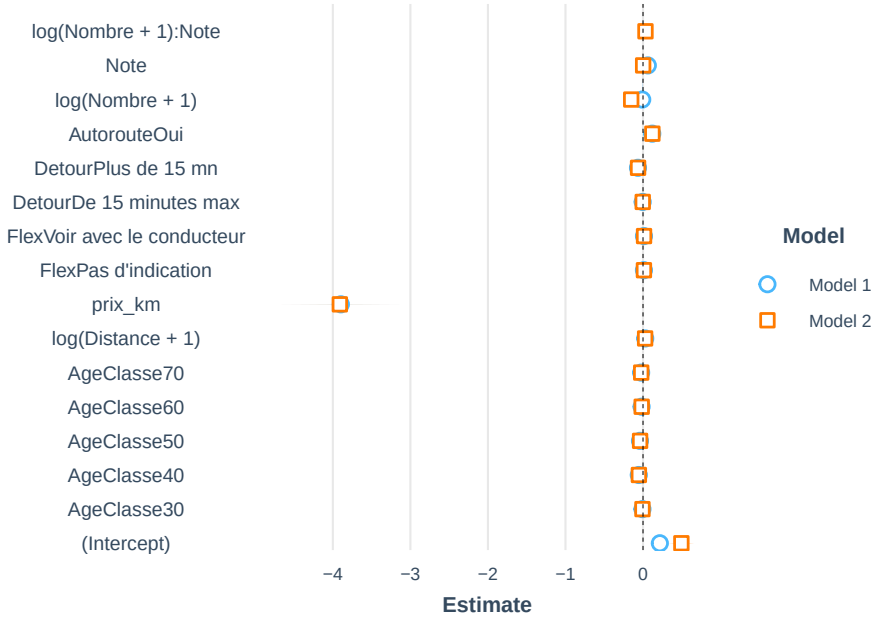
On commence par un modèle simple et linéaire où l'on cherche à expliquer, prédire, le taux d'occupation du véhicule en fonction des variables dont nous disposons. La flexibilité de l'horaire et la possibilité de détour n'affecte pas vraiment le taux de réservation. Si les autres variables ont des relations significatives (avec des valeur t très élevées), on notera que la variance expliquée est faible.

Le remplissage des voiture est une affaire de loto. Ce qui se comprend, car la probabilité qu'une offre et une demande coïncide est relativement faible si on retient une plage horaire étroite pour le passager qui l'élargira pour accroître le choix au prix d'un effort de recherche supplémentaire.

```
df$AgeClasse<- round(df$Age/10,0)*10
df$AgeClasse[df$AgeClasse==80]<-70
df$AgeClasse[df$AgeClasse==90]<-70
df$AgeClasse[df$AgeClasse==100]<-70
df$AgeClasse<-as.factor(df$AgeClasse)

fit0<-lm(Occup~AgeClasse+log(Distance+1)+prix_km+
           Flex+Detour+Autoroute+log(Nombre+1)+Note, data=df)
fit1<-lm(Occup~AgeClasse+log(Distance+1)+prix_km+
           Flex+Detour+Autoroute+log(Nombre+1)*Note, data=df)
```

```
fit_logit<-lm(Occup~AgeClasse+ Marque+log(Distance+1)+prix_km+
                Flex+Detour+Autoroute+log(Nombre+1)*Note, data=df)

export_summs(fit0, fit1,plot.distributions = TRUE,scale = FALSE, digits = 3)
```

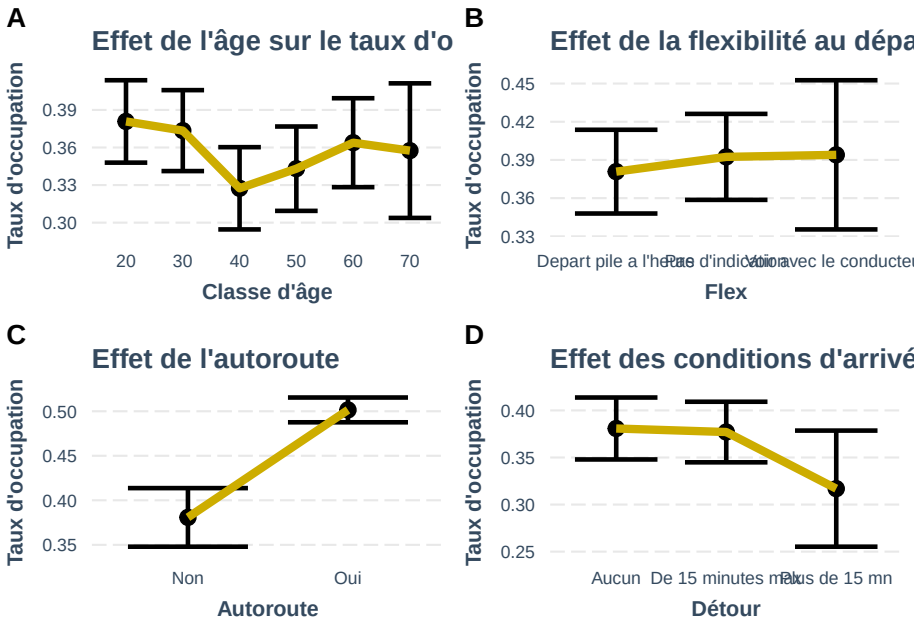
```
plot_summs(fit0, fit1,plot.distributions = TRUE,
           omit.coefs=c("MarqueAudi",
                        "MarqueAutres",
                        "MarqueAutres Asie" ,
                        "MarqueAutres US",
                        "MarqueBmw",
                        "MarqueCitroen" , "MarqueDacia","MarqueFiat",
                        "MarqueFord",
                        "MarqueHyundai",
                        "MarqueMercedes-benz","MarqueNissan","MarqueOpel",
                        "MarquePeugeot","MarqueRenault ",
                        "MarqueRover, jaguar, mini",
                        "MarqueSeat",
                        "MarqueSkoda","MarqueToyota",
                        "MarqueVolkswagen","MarqueVolvo", "Saab"))
```



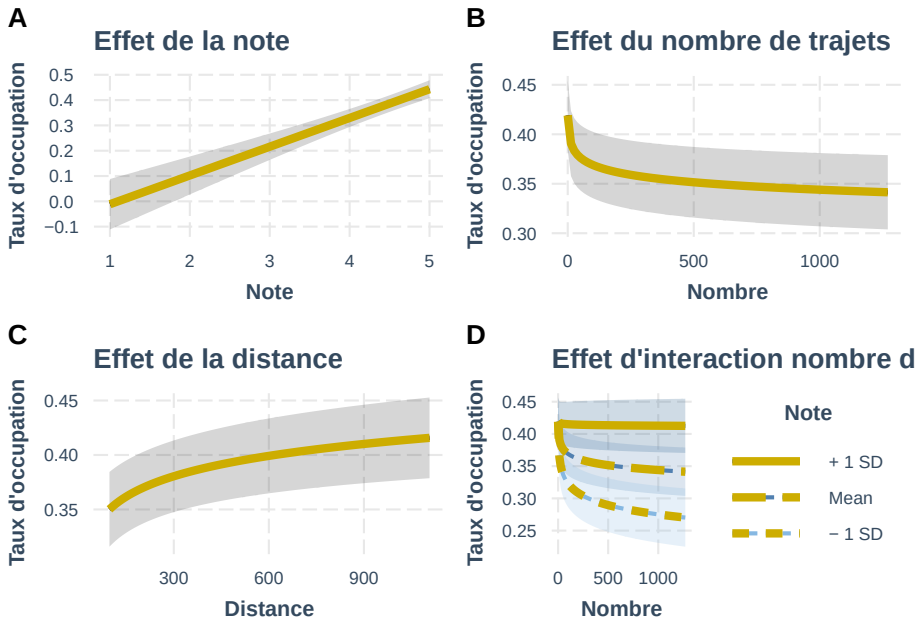
Le modèle tient même s'il explique très peu de variance. Les tests sont cependant significatifs et même si l'effet est faible il est notable.

Voici des diagrammes d'effets qui représentent la valeur de la variable dépendante pour une plage de variation des variables indépendantes prises une à une, en fonction du modèle. On voit ainsi plus clairement que le taux d'occupation passe de 20% environ quand la note est de 3 (les notes inférieures sont rares!) à 45% quand la moyenne du conducteur frôle les 5.

```
g10<-effect_plot(fit1, pred=AgeClasse,interval = TRUE ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de l'âge sur le taux d'occupation",
       x="Classe d'âge",
       y="Taux d'occupation"
       )
g11<-effect_plot(fit1, pred=Flex,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de la flexibilité au départ",
       y="Taux d'occupation")
g12<-effect_plot(fit1, pred=Autoroute,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de l'autoroute",
       y="Taux d'occupation")

g13<-effect_plot(fit1, pred=Detour, interval = TRUE ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet des conditions d'arrivée", x="Détour",
       y="Taux d'occupation")

plot_grid(
  g10, g11,g12,g13,
  labels = "AUTO", ncol = 2
)
```




```
g16<-effect_plot(fit1, pred=Note,interval = TRUE ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de la note",
       y="Taux d'occupation")
g17<-effect_plot(fit1, pred=Nombre,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet du nombre de trajets",
       y="Taux d'occupation")
g18<-effect_plot(fit1, pred=Distance,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de la distance",
       y="Taux d'occupation")
g19<-interact_plot(fit1, pred=Nombre,modx=Note,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet d'interaction nombre de trajets x note",
       y="Taux d'occupation")

plot_grid(
  g16, g17,g18,g19,
  labels = "AUTO", ncol = 2
)
```



La particularité du modèle est le terme d'interaction. L'expérience du conducteur traduite par le nombre de trajets qu'il a effectué est gage de sécurité, de confiance, pourvu que ses notes soient bonnes. On s'attend à ce que si elle sont moins bonnes, le nombre de voyages réalisés en amplifiera l'impact négatif. L'expérience signale aussi la crédibilité de la note, et on peut raisonnablement penser que plus ce nombre est grand et plus le signal, positif ou négatif est crédible. Ici la note module l'effet de crédibilité.

L'analyse de l'interaction est claire : quand les contenus sont négatifs , l'expérience du conducteur aggrave la réticence et le taux d'occupation se réduit. Quand les notes vont meilleures que la moyenne.L'amplification par l'expérience de l'effet des notes sur la réservation est moindre. L'interaction est significative (t=8.76). Reste à comprendre pourquoi la tendance générale reste à une relation négative entre le taux de remplissage et le nombre de trajet effectué . Est-ce le résultat d'une participation fréquente à la plateforme, à un comportement particulier des conducteurs très actifs (opportunisme) ? Une explication complémentaire peut venir de ceux qui pratique occasionnellement le covoiturage, leur activité étant plus rare , elle est peut être plus planifiée, et les offres sont présentes depuis plus longtemps que celles des utilisateurs fréquents qui publieraient leurs annonces de la veille au lendemain. Ceci mérite une analyse plus approfondie que nous ne mènerons pas ici.

10.5 Autres modèles

Dans la pratique la construction d'un modèle de régression dépend de trois éléments :

- la spécification fonctionnelle qui définit f(y, X) autrement dit ce qui relie ce qu'on observe à ce qui peut lui être corrélé de manière linéaires ou moins linéaires. La forme fonctionnelle dépend largement de la variable dépendante et de sa distribution :

** normale : ** log-normale voire exponentielle ** binaire ** proportionnelle ** dénombrement. Il s'agit des données de comptages : un nombre d'achat au cours d'une période par exemple. poisson, NBD ** Les durées sont positives,

Dans ce cas on va utiliser une méthode de modèle linéaire généralisé (GLM) qui reposent sur 3 éléments:

- Un prédicteur linéaire

  $$\eta$$

  qui décrit la combinaison linéaire des variables explicatives.

$$\eta = \sum_{n=1}^{10} \beta_0 + \beta_i x_i$$

- Une fonction de lien : Contrairement aux modèles linéaires classiques, les valeurs prédites par le prédicteur linéaire ne correspondent pas à la prédiction moyenne d'une observation, mais à une transformation mathématique. Les beta sont estimés après transformation des réponses selon la fonction de lien choisie.

$$g(\mu_y) = \eta$$

Par exemple, pour les données de comptage :

$$log(\mu_y) = \eta$$

ou dans le cas de données binaire (modèle de régression logistique)

$$log(\frac{\mu_y}{1-\mu_y}) = \eta$$

Le but de la fonction de lien est de contraindre les valeurs prédites à être dans l'échelle des valeurs observées. Ainsi, dans le cas des données de comptage, qui sont obligatoirement positives, ou nulles, la fonction de lien log contraint les valeurs prédites par le prédicteur linéaire à devenir également positives ou nulles après l'emploi de la fonction inverse du log.

- La structure d'erreur : A une fonction de lien donnée, correspond généralement une structure d'erreur particulière.Il s'agit d'une famille de distribution des erreurs. Par exemple, pour les données de comptage, la fonction de lien est le log et la structure d'erreur correspondante est la distribution de Poisson. Cette structure d'erreur, permet notamment de spécifier correctement la relation entre la moyenne et la variance. Cette relation est utilisée par l'approche de maximum de vraisemblance pour estimer les coefficients des paramètres (les beta) du GLM.

Ici un tableau récapitulatif des structures d'erreurs, fonctions de lien, fonctions de moyennes et fonctions de variance des données de type numériques continues non bornée, de comptage et binaire.

Type des réponses (et des erreurs)	Domaine de définition des réponses	Distribution des erreurs (et des réponses)	Nom de la fonction de lien	Fonction de Lien	Fonction de la moyenne	Fonction de la variance
Quantitatif continu	Réel $]-\infty\,;\,+\infty[$	Gaussienne	Identité	$\sum_{j=1}^{p} x_{ij}\beta_j = \mu$	$\mu = \sum_{j=1}^{p} x_{ij}\beta_j$	$var(y_i) = cste$
Comptage	Entier $[0\,;\,+\infty[$	Poisson	Log	$\sum_{j=1}^{p} x_{ij}\beta_j = \ln(\mu)$	$\mu = exp\left(\sum_{j=1}^{p} x_{ij}\,\beta_j\right)$	$var(y_i) = \propto \mu$
Binaire (oui/non)	Entier $[0\,;1]$	Binomiale	Logit	$\sum_{j=1}^{p} x_{ij}\beta_j = \ln\left(\frac{\mu}{1-\mu}\right)$	$\mu = \frac{1}{1+\exp\left(-\sum_{j=1}^{p} x_{ij}\beta_j\right.}$	$var(y_i) = \propto \frac{\mu\,(n-\mu)}{n}$

* Estimation par une méthode de maximum de vraisemblance et de déviance qui est en quelque sorte une généralisation de la variance.

10.5.1 Régression logistique

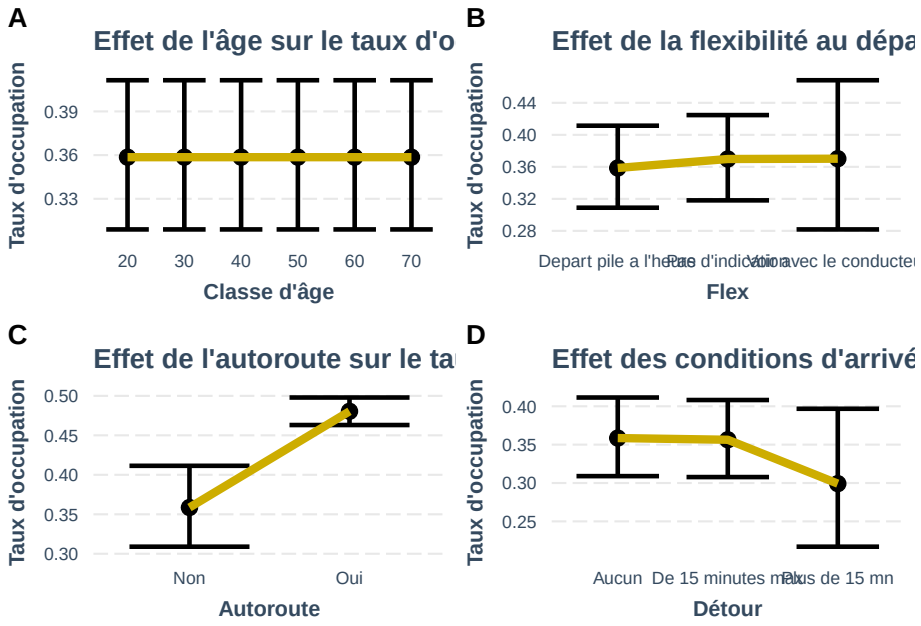
Sur les même données mais en considérant le taux d'occupation plus sérieusement : il est compris entre 0 et 1 et se prête don à un modèle logistique, car le taux d'occupation va de 0 à 100% même si nous n'avons que 4 niveaux. ça fait un modèle plus réaliste qui ajuste la valeur prédite entre 0 et 1.

Pour mieux prendre en compte l'hétérogénéité du set de données (composée par un échantillon de requêtes) on introduit les trajets comme composante aléatoire. l'estimation est réalisée avec `lme4`

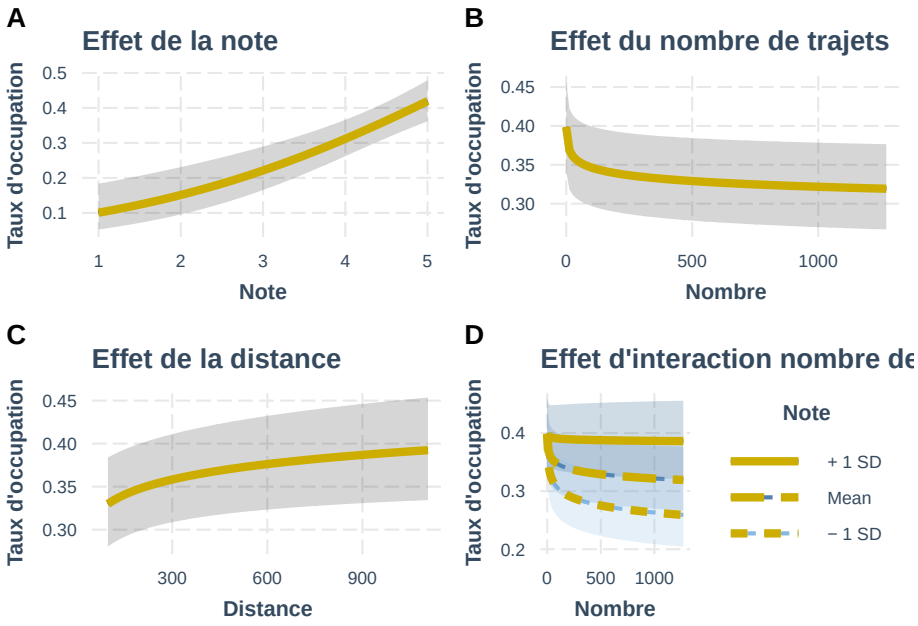
Les effets semblent être ici plus pertinent et sensibles mais surtout l'effet d'interaction semble renforcé. L'effet de la note est là clairement amplifié pour les notes négatives mais aussi pour les notes positives.

```
fit_logit<-glm(Occup~Age+log(Distance+1)+prix_km+Flex+Detour+Autoroute+log(Nombre+1)*No

summ(fit_logit)
```

```
g10<-effect_plot(fit_logit, pred=AgeClasse,interval = TRUE ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de l'âge sur le taux d'occupation",x="Classe d'âge",
       y="Taux d'occupation")

g11<-effect_plot(fit_logit, pred=Flex,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de la flexibilité au départ sur le taux d'occupation",
       y="Taux d'occupation")


g12<-effect_plot(fit_logit, pred=Autoroute,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de l'autoroute sur le taux d'occupation",
       y="Taux d'occupation")

g13<-effect_plot(fit_logit, pred=Detour, interval = TRUE ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet des conditions d'arrivée sur le taux d'occupation", x="Détour",
       y="Taux d'occupation")

plot_grid(
  g10, g11,g12,g13,
  labels = "AUTO", ncol = 2
)
```




```
g16<-effect_plot(fit_logit, pred=Note,interval = TRUE ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de la note",
       y="Taux d'occupation")
g17<-effect_plot(fit_logit, pred=Nombre,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet du nombre de trajets",
       y="Taux d'occupation")
g18<-effect_plot(fit_logit, pred=Distance,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet de la distance",
       y="Taux d'occupation")
g19<-interact_plot(fit_logit, pred=Nombre,modx=Note,interval = TRUE  ,data=df)+
  geom_line(color="gold3",size=2)+
  labs(title="Effet d'interaction nombre de trajets x note",
       y="Taux d'occupation")

plot_grid(
  g16, g17,g18,g19,
  labels = "AUTO", ncol = 2
)
```



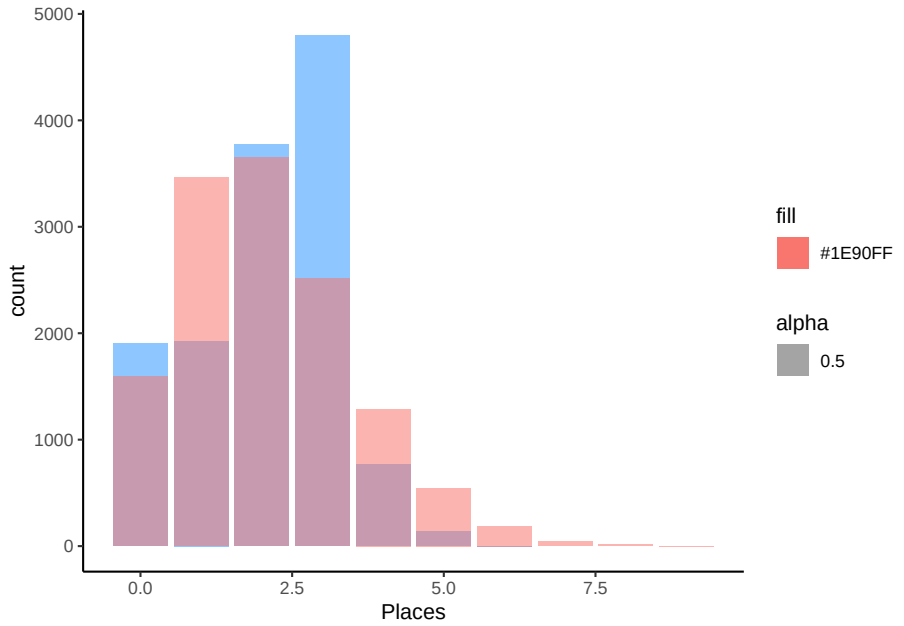
10.5.2 Modèle de comptage

et un dernier qui revient aux données originelles : le comptage des place disponibles. Dans ce type de situation (données de comptage), on considère que ce type de variable se distribue selon une loi de poisson, dont la propriété est l'égalité de la moyenne et de l'écart-type.

On s'assure qu'il n'y ait pas de sur-dispersion en calculant le ratio de la variance résiduelle par le nombre de degré de liberté. Il est ici de l'ordre de 1 et donc on conclura à une absence de sur-dispersion, même si en comparant la distribution empiriques du nombre de places restantes à la distribution théorique d'une loi de poisson, l'ajustement n'est pas parfait.

```
mean_places<-mean(df$Places)
mean_sd<-sd(df$Places)

set.seed(1234) # permet de simuler toujours les mêmes comptages.
theoretic_count <-rpois(nrow(df),mean_places)

# on incorpore ces comptages théoriques dans un data frame
tc_df <-data.frame(theoretic_count)


ggplot(df,aes(Places))+
```

```
  geom_bar(fill="#1E90FF", alpha=0.5)+
  geom_bar(data=tc_df, aes(theoretic_count,fill="#1E90FF", alpha=0.5))+
  theme_classic()#+   theme(legend.position="none")
```




```
df$Occup[df$Occup==1]<-.999
df$Occup[df$Occup==0]<-.001

fit_poisson <- glm(Places~Age+log(Distance+1)+prix_km+Flex+Detour+Autoroute+log(Nombre+1)*Note, f

summary(fit_poisson)
```

```
##
## Call:
## glm(formula = Places ~ Age + log(Distance + 1) + prix_km + Flex +
##     Detour + Autoroute + log(Nombre + 1) * Note, family = "poisson",
##     data = df)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.51367  -0.74448   0.03874   0.60289   2.13066
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)                  0.6879101  0.1897188   3.626 0.000288 ***
## Age                          0.0019560  0.0005095   3.839 0.000123 ***
## log(Distance + 1)           -0.0513430  0.0137967  -3.721 0.000198 ***
## prix_km                      7.6872057  0.9123308   8.426  < 2e-16 ***
## FlexPas d'indication        -0.0236138  0.0140880  -1.676 0.093704 .
## FlexVoir avec le conducteur -0.0109685  0.0595949  -0.184 0.853974
## DetourDe 15 minutes max      0.0023592  0.0137864   0.171 0.864125
## DetourPlus de 15 mn          0.1349013  0.0620044   2.176 0.029580 *
## AutorouteOui                -0.2434932  0.0347369  -7.010 2.39e-12 ***
## log(Nombre + 1)              0.2883459  0.0713807   4.040 5.36e-05 ***
## Note                        -0.0088294  0.0362270  -0.244 0.807444
## log(Nombre + 1):Note        -0.0592471  0.0159474  -3.715 0.000203 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 11105  on 11718  degrees of freedom
## Residual deviance: 10889  on 11707  degrees of freedom
##   (1586 observations deleted due to missingness)
## AIC: 38248
##
## Number of Fisher Scoring iterations: 5
```

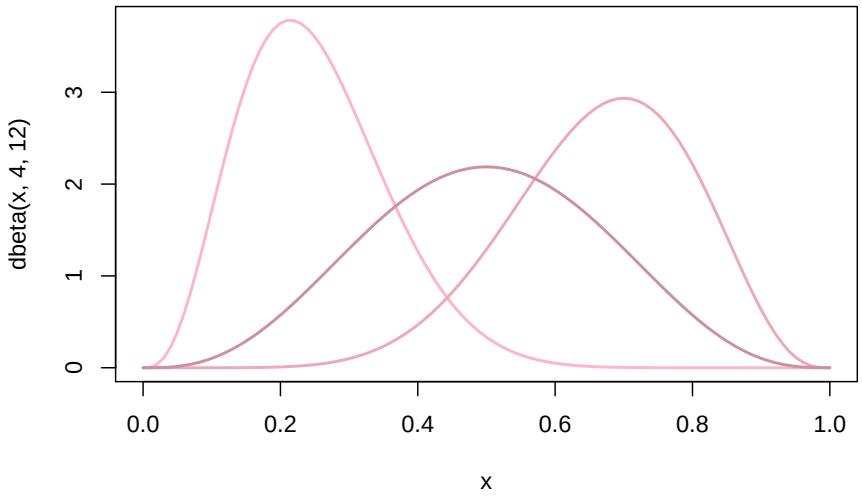
```
fit_poisson2 <- glm(Places~Age+log(Distance+1)+prix_km+Flex+Detour+Autoroute+log(Nombre

export_summs(fit_logit,fit_poisson, fit_poisson2)
```

10.5.3 Modèle de régression beta

Encore une autre variation avec le modèle de regression beta qui s'attache à modéliser des variables de proportion par une loi de distribution beta dont la caractéristiques est d'être souple et de s'adapter à toute forme de distribution en respectant la contrainte de varier entre presque 0 et presque 1. Elle dépend de deux paramètres α et β

```
curve(dbeta(x, 4, 12),
      col = "pink1",
      lwd = 2)
curve(dbeta(x, 8, 4),
      add = TRUE, col = "pink2",
      lwd = 2)
curve(dbeta(x, 4, 4),
```

```
      add = TRUE, col = "pink3",
      lwd = 2)
```



La spécificité de la régression beta est de modéliser les deux paramètre de la fonction de distribution par deux équation linéraire. L'une ajustant la moyenne, l'autre ajustant la dispersion.

```
#recodage
df$Occup[df$Occup==1]<-.999
df$Occup[df$Occup==0]<-.001

fit_beta1 <- betareg(Occup~AgeClasse+log(Distance+1)+prix_km+Flex+Detour+Autoroute+log(Nombre+1)*
summary(fit_beta1)
```

```
##
## Call:
## betareg(formula = Occup ~ AgeClasse + log(Distance + 1) + prix_km + Flex +
##     Detour + Autoroute + log(Nombre + 1) * Note, data = df)
##
## Standardized weighted residuals 2:
##     Min      1Q  Median      3Q     Max
## -2.5648 -0.4721 -0.1841  0.1925  2.5672
##
## Coefficients (mean model with logit link):
```

```
##                                Estimate Std. Error z value Pr(>|z|)
## (Intercept)                     0.477618   0.346880   1.377  0.16854
## AgeClasse30                    -0.055681   0.030938  -1.800  0.07189 .
## AgeClasse40                    -0.305290   0.035616  -8.572  < 2e-16 ***
## AgeClasse50                    -0.196017   0.039618  -4.948 7.51e-07 ***
## AgeClasse60                    -0.134592   0.044980  -2.992  0.00277 **
## AgeClasse70                    -0.220373   0.099999  -2.204  0.02754 *
## log(Distance + 1)               0.119285   0.024997   4.772 1.82e-06 ***
## prix_km                       -17.809578   1.675785 -10.628  < 2e-16 ***
## FlexPas d'indication            0.029596   0.025342   1.168  0.24285
## FlexVoir avec le conducteur    -0.060868   0.108458  -0.561  0.57465
## DetourDe 15 minutes max        -0.008982   0.024985  -0.360  0.71922
## DetourPlus de 15 mn            -0.284942   0.119009  -2.394  0.01665 *
## AutorouteOui                    0.540004   0.068146   7.924 2.30e-15 ***
## log(Nombre + 1)                -0.846267   0.132436  -6.390 1.66e-10 ***
## Note                           -0.049875   0.066261  -0.753  0.45163
## log(Nombre + 1):Note            0.179977   0.029493   6.102 1.04e-09 ***
##
## Phi coefficients (precision model with identity link):
##       Estimate Std. Error z value Pr(>|z|)
## (phi)  1.08213    0.01123   96.36   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Type of estimator: ML (maximum likelihood)
## Log-likelihood:  2788 on 17 Df
## Pseudo R-squared: 0.0308
## Number of iterations: 25 (BFGS) + 2 (Fisher scoring)
```

Dans une régression bêta, a variable dépendante est distribuée selon la loi bêta avec une espérance et une variance (1-)/(1+). Ainsi, est un paramètre de précision : plus est élevé, plus la variance est faible pour une moyenne donnée. Le paramètre de précision peut dépendre des régresseurs comme la moyenne.

C'est ce qui est spécifié ci-dessous, en reliant le px au km à ce paramètre de dispersion. La valeur est élevée, la valeur p extrêmement faible, on en déduit qu'effectivement, plus le prix au kilomètre est élevé et plus faible est la variance de la distribution.

```
fit_beta2 <- betareg(Occup~AgeClasse+log(Distance+1)+prix_km+Flex+Detour+Autoroute+log

summary(fit_beta2)
```

```
##
```

```
## Call:
## betareg(formula = Occup ~ AgeClasse + log(Distance + 1) + prix_km + Flex +
##     Detour + Autoroute + log(Nombre + 1) * Note | prix_km, data = df)
##
## Standardized weighted residuals 2:
##     Min      1Q  Median      3Q     Max
## -2.8774 -0.4693 -0.1836  0.1952  4.4017
##
## Coefficients (mean model with logit link):
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                   0.36456    0.34639   1.052  0.29259
## AgeClasse30                  -0.04898    0.03093  -1.583  0.11332
## AgeClasse40                  -0.30011    0.03558  -8.436  < 2e-16 ***
## AgeClasse50                  -0.19509    0.03957  -4.930 8.20e-07 ***
## AgeClasse60                  -0.13058    0.04488  -2.909  0.00362 **
## AgeClasse70                  -0.21690    0.09991  -2.171  0.02992 *
## log(Distance + 1)             0.12737    0.02497   5.100 3.39e-07 ***
## prix_km                     -16.90573    1.66364 -10.162  < 2e-16 ***
## FlexPas d'indication          0.03339    0.02530   1.320  0.18690
## FlexVoir avec le conducteur  -0.06949    0.10821  -0.642  0.52076
## DetourDe 15 minutes max      -0.00998    0.02496  -0.400  0.68922
## DetourPlus de 15 mn          -0.30605    0.11831  -2.587  0.00969 **
## AutorouteOui                  0.52620    0.06900   7.627 2.41e-14 ***
## log(Nombre + 1)              -0.84903    0.13213  -6.426 1.31e-10 ***
## Note                         -0.04752    0.06600  -0.720  0.47155
## log(Nombre + 1):Note          0.18104    0.02943   6.152 7.64e-10 ***
##
## Phi coefficients (precision model with log link):
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.6573     0.0877  -7.495 6.63e-14 ***
## prix_km      12.1320     1.4288   8.491  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Type of estimator: ML (maximum likelihood)
## Log-likelihood:  2826 on 18 Df
## Pseudo R-squared: 0.03075
## Number of iterations: 29 (BFGS) + 3 (Fisher scoring)
```

Table 10.1

	Model 1	Model 2
(Intercept)	0.217 ***	0.495 ***
	(0.053)	(0.080)
AgeClasse30	-0.007	-0.007
	(0.007)	(0.007)
AgeClasse40	-0.054 ***	-0.053 ***
	(0.008)	(0.008)
AgeClasse50	-0.039 ***	-0.038 ***
	(0.009)	(0.009)
AgeClasse60	-0.018	-0.017
	(0.010)	(0.010)
AgeClasse70	-0.024	-0.023
	(0.023)	(0.023)
log(Distance + 1)	0.029 ***	0.027 ***
	(0.006)	(0.006)
prix_km	-3.896 ***	-3.910 ***
	(0.386)	(0.385)
FlexPas d'indication	0.012 *	0.012 *
	(0.006)	(0.006)
FlexVoir avec le conducteur	0.012	0.013
	(0.025)	(0.025)
DetourDe 15 minutes max	-0.004	-0.004
	(0.006)	(0.006)
DetourPlus de 15 mn	-0.065 *	-0.064 *
	(0.027)	(0.027)
AutorouteOui	0.119 ***	0.121 ***
	(0.016)	(0.016)
log(Nombre + 1)	-0.010 ***	-0.152 ***
	(0.002)	(0.030)
Note	0.062 ***	0.002

Observations	11719 (1586 missing obs. deleted)
Dependent variable	Occup
Type	Generalized linear model
Family	binomial
Link	logit

$\chi^2(11)$	106.04
Pseudo-R^2 (Cragg-Uhler)	0.02
Pseudo-R^2 (McFadden)	0.01
AIC	15784.58
BIC	15873.00

	Est.	S.E.	z val.	p
(Intercept)	0.07	0.56	0.13	0.90
Age	-0.00	0.00	-2.61	0.01
log(Distance + 1)	0.11	0.04	2.79	0.01
prix_km	-16.30	2.72	-6.00	0.00
FlexPas d'indication	0.05	0.04	1.20	0.23
FlexVoir avec le conducteur	0.05	0.17	0.29	0.77
DetourDe 15 minutes max	-0.01	0.04	-0.24	0.81
DetourPlus de 15 mn	-0.27	0.19	-1.39	0.16
AutorouteOui	0.50	0.11	4.50	0.00
log(Nombre + 1)	-0.64	0.21	-2.97	0.00
Note	0.00	0.11	0.01	0.99
log(Nombre + 1):Note	0.13	0.05	2.77	0.01

Standard errors: MLE

Table 10.2

	Model 1	Model 2	Model 3
(Intercept)	0.07	0.69 ***	0.69 ***
	(0.56)	(0.19)	(0.15)
Age	-0.00 **	0.00 ***	0.00 ***
	(0.00)	(0.00)	(0.00)
log(Distance + 1)	0.11 **	-0.05 ***	-0.05 ***
	(0.04)	(0.01)	(0.01)
prix_km	-16.30 ***	7.69 ***	7.69 ***
	(2.72)	(0.91)	(0.75)
FlexPas d'indication	0.05	-0.02	-0.02 *
	(0.04)	(0.01)	(0.01)
FlexVoir avec le conducteur	0.05	-0.01	-0.01
	(0.17)	(0.06)	(0.05)
DetourDe 15 minutes max	-0.01	0.00	0.00
	(0.04)	(0.01)	(0.01)
DetourPlus de 15 mn	-0.27	0.13 *	0.13 **
	(0.19)	(0.06)	(0.05)
AutorouteOui	0.50 ***	-0.24 ***	-0.24 ***
	(0.11)	(0.03)	(0.03)
log(Nombre + 1)	-0.64 **	0.29 ***	0.29 ***
	(0.21)	(0.07)	(0.06)
Note	0.00	-0.01	-0.01
	(0.11)	(0.04)	(0.03)
log(Nombre + 1):Note	0.13 **	-0.06 ***	-0.06 ***
	(0.05)	(0.02)	(0.01)
N	11719	11719	11719
AIC	15784.58	38248.46	
BIC	15873.00	38336.88	
Pseudo R2	0.02	0.02	0.02

*** $p < 0.001$; ** $p < 0.01$; * $p < 0.05$.

Chapter 11

Modèle de survie

voir étude de cas CartedeFidelité.

Chapter 12

Les modèles linéaires hiérarchiques (HLM)

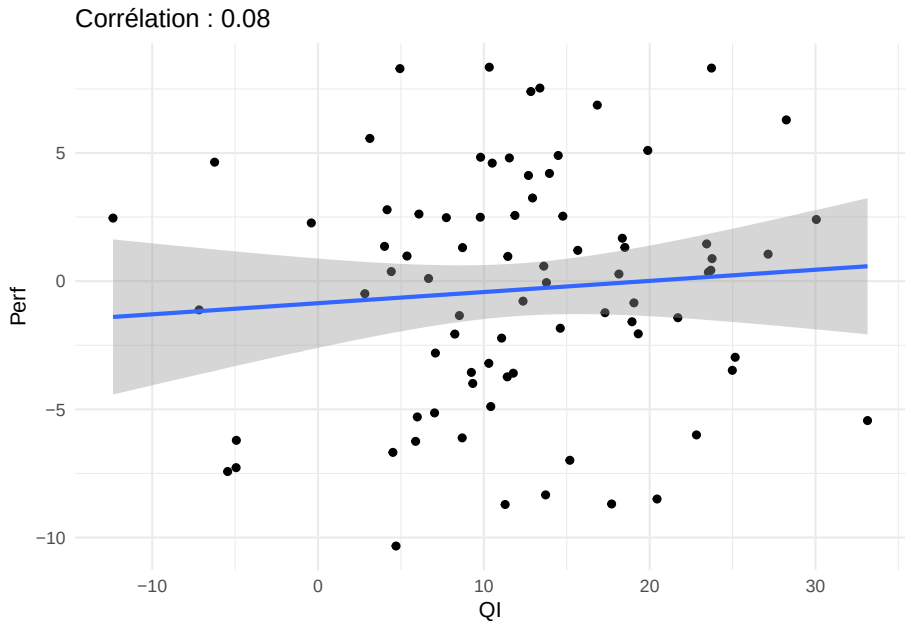
Les modèles de panels en économies, ou multi-niveaux en sociologie, sont caractérisés par le fait que les données sont un empilement de différents échantillons correspondant à une stratification.

12.1 en guise d'introduction

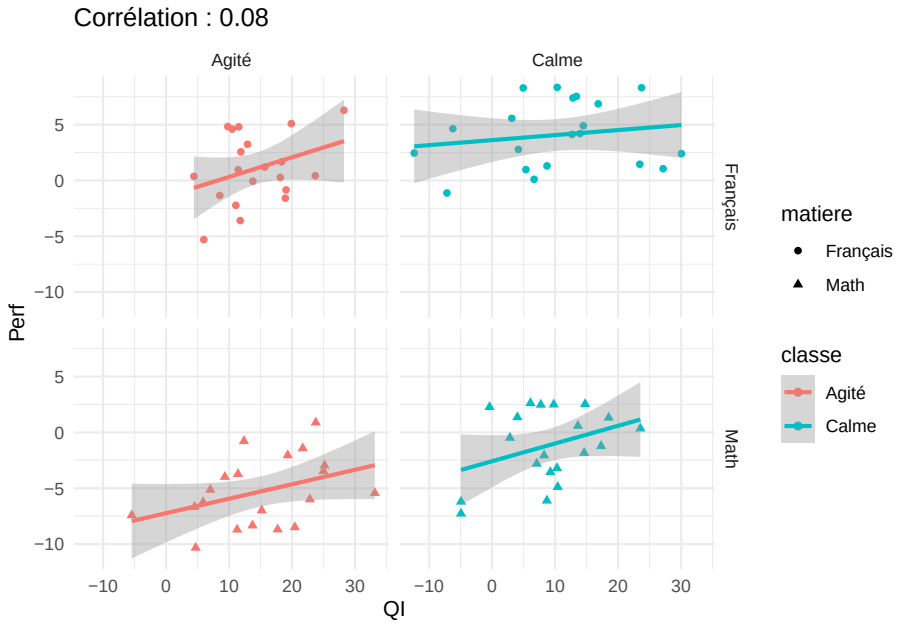
L'exemple de la performance scolaire va nous éclairer. Supposons que l'on veuille établir l'effet d'une mesure d'aptitude intellectuelle (par le exemple le score de QI) sur les notes obtenues. On va mesurer cette relation en recueillant les données deux classes et dans deux matières. La classe A est calme, la classe B est agitée, les matières sont les maths et le français.

```
a=.1
foo_MC<-as.data.frame(rnorm(20, mean=12, sd=10))%>%
  rename(QI=1) %>% group_by(row_number()) %>%
  mutate(e=-5+10*runif(1))%>% ungroup() %>%
  mutate(Perf=a*QI-3+1+e,matiere="Math", classe="Calme")

foo_MA<-as.data.frame(rnorm(20, mean=12, sd=10))%>%
  rename(QI=1) %>% group_by(row_number()) %>%
  mutate(e=-5+10*runif(1))%>% ungroup() %>%
  mutate(Perf=a*QI-3-3+e,matiere="Math", classe="Agité")

foo_FC<-as.data.frame(rnorm(20, mean=12, sd=10))%>%
  rename(QI=1) %>% group_by(row_number()) %>%
```

```
  mutate(e=-5+10*runif(1))%>% ungroup() %>%
  mutate(Perf=a*QI+2+1+e,matiere="Français", classe="Calme")

foo_FA<-as.data.frame(rnorm(20, mean=12, sd=10))%>%
  rename(QI=1) %>% group_by(row_number()) %>%
  mutate(e=-5+10*runif(1))%>% ungroup() %>%
  mutate(Perf=a*QI+2-4+1+e,matiere="Français", classe="Agité")



foo<-rbind(foo_MA, foo_MC, foo_FA,foo_FC)
r =round(cor(foo$QI, foo$Perf),2)

ggplot(foo, aes(x=QI, y=Perf))+
  geom_point()+geom_smooth(method="lm")+
  labs(title=paste("Corrélation :",r))
```




```
ggplot(foo, aes(x=QI, y=Perf, shape=matiere, color=classe))+
  geom_point()+geom_smooth(method="lm")+facet_grid(matiere~classe)+
  labs(title=paste("Corrélation :",r))
```




```
fit01<-lm(Perf~QI, data=foo)
fit02<-lm(Perf~QI+matiere+classe, data=foo)
fit03<-lm(Perf~QI+matiere*classe, data=foo)

export_summs(fit01,fit02, fit03, number_format = "%.3g")
```

Dans ce petit exemple, les niveaux sont peu nombreux. On pourrait imaginer qu'ils soient bien plus nombreux, par exemple en réalisant l'enquête sur des dizaines de classes pour lesquelles le degré d'agitation est variable et se distribue certainement de manière normale. Ne tenons plus en compte la matière pour le moment. On ne va prendre que la moyenne des maths.

On peut écrire le modèle où

$$\beta_k$$

représente le terme constant de chacune des k classes.

$$y_{ik} = \beta_k + \beta_1 Aptitude_i + \epsilon_{ik}$$

en supposant que les

$$\beta_k$$

se distribue de manière normale avec une moyenne

$$\overline{\beta}$$

Table 12.1

	Model 1	Model 2	Model 3
(Intercept)	-0.859	-0.723	-0.426
	(0.875)	(0.823)	(0.875)
QI	0.0435	0.102 *	0.105 **
	(0.0585)	(0.0395)	(0.0396)
matiereMath		-5.78 ***	-6.44 ***
		(0.667)	(0.944)
classeCalme		4.08 ***	3.43 ***
		(0.696)	(0.957)
matiereMath:classeCalme			1.33
			(1.34)
N	80	80	80
R2	0.00703	0.594	0.599

*** p < 0.001; ** p < 0.01; * p < 0.05.

et une variance

$$\mu_k$$

, on peut réécrire l'équation de la manière suivante

$$y_{ik} = \overline{\beta} + \beta_1 Aptitude_i + \mu_k + \epsilon_{ik}$$

C'est un modèle à composantes d'erreur où μ_k représente l'effet de la classe.

bbbba

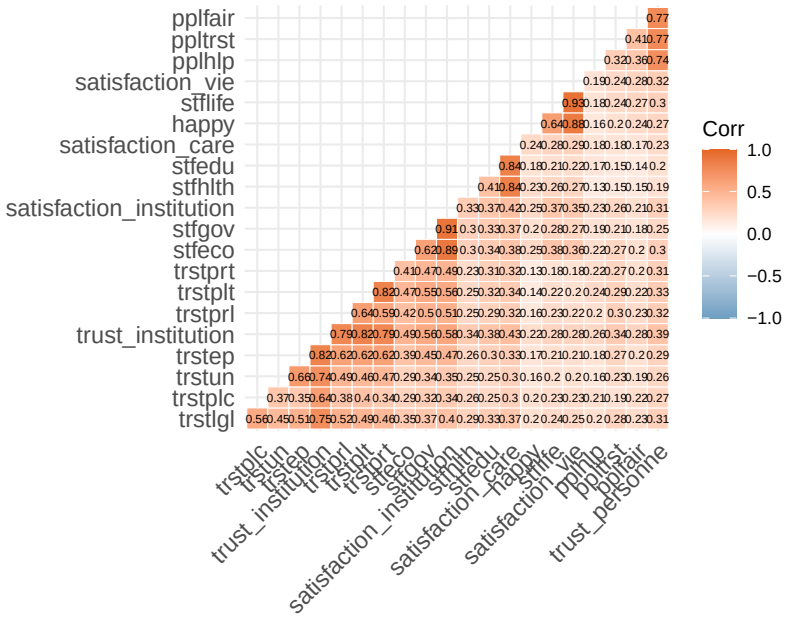
sffs

12.2 Une application

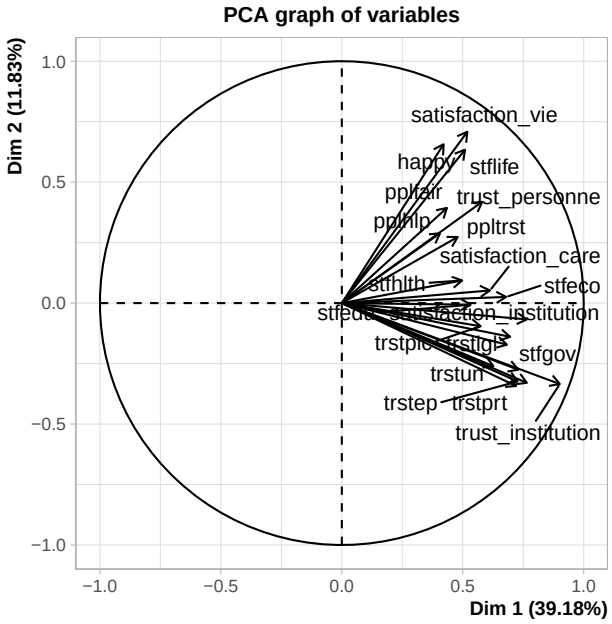
voir données ESS

```
df<-readRDS("./data/ESS10fr.rds")
```

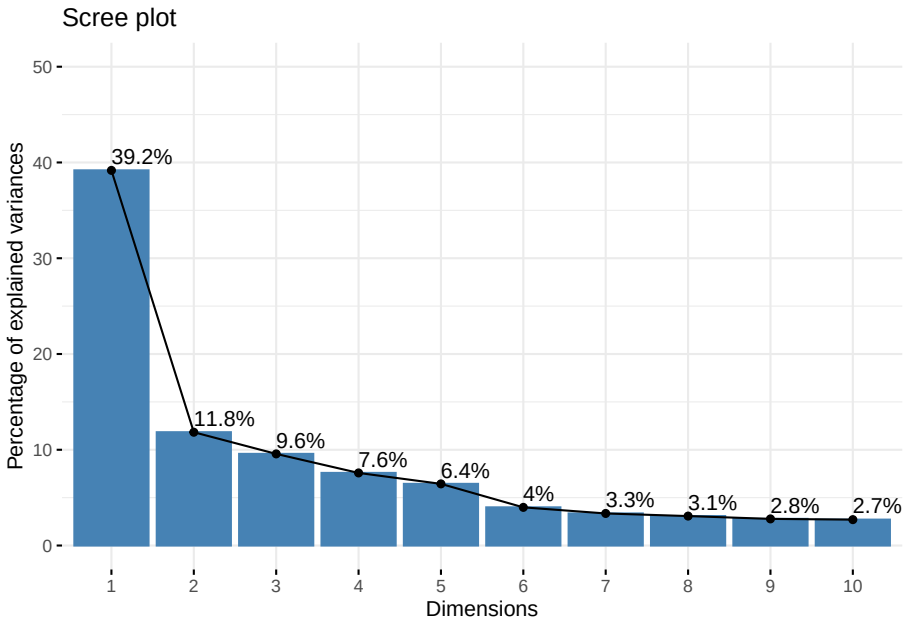
```
library(ggcorrplot)
foo<-cbind(df[,5:26]) %>%
  dplyr::select(-stfmjob)%>%
  drop_na()
r<-cor(foo)
ggcorrplot(r, hc.order = TRUE, type = "lower",
   outline.col = "white",
   colors = c("#6D9EC1", "white", "#E46726"), lab=TRUE, lab_size=2)
```



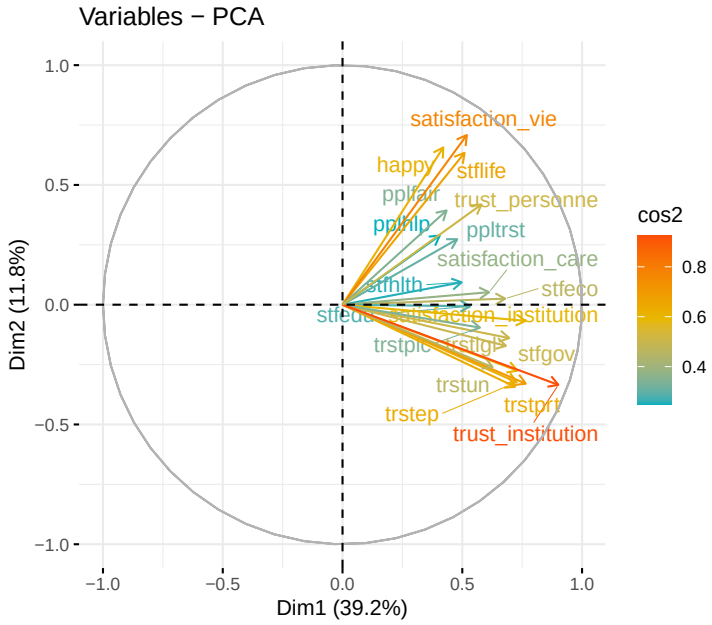

```
res.pca <- PCA(foo, scale.unit = TRUE, ncp = 3, graph = TRUE)
```




```
fviz_eig(res.pca, addlabels = TRUE, ylim = c(0, 50))
```

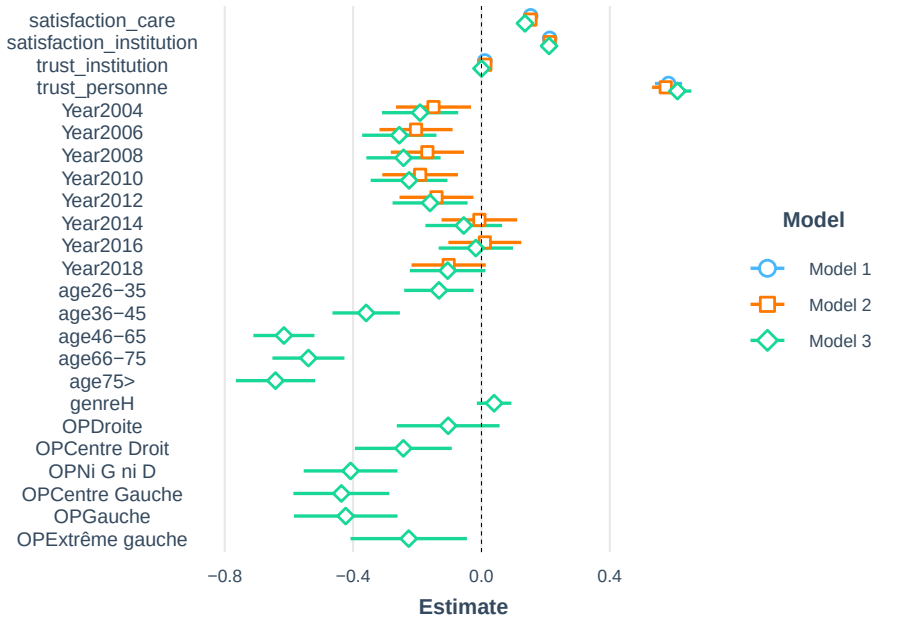


```
fviz_pca_var(res.pca, col.var = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE # Avoid text overlapping
             )
```




```
foo<-df%>%
  dplyr::select(satisfaction_vie, satisfaction_care, satisfaction_institution, trust_institution,
  drop_na()

fit0<-lm(satisfaction_vie~satisfaction_care+satisfaction_institution+trust_institution+trust_pers

fit1<-lm(satisfaction_vie~satisfaction_care+satisfaction_institution+trust_institution+trust_pers

fit2<-lm(satisfaction_vie~age+genre+OP+satisfaction_care+satisfaction_institution+trust_instituti



export_summs(fit0, fit1,fit2, digits=3)
```

```
plot_summs(fit0, fit1,fit2)
```




```
fit3<-lmer(satisfaction_vie~satisfaction_care+satisfaction_institution+trust_institutio

summary(fit3)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: satisfaction_vie ~ satisfaction_care + satisfaction_institution +
##     trust_institution + trust_personne + (1 | Year) + (1 | age) +
##     (1 | OP) + (1 | genre)
##    Data: df
##
## REML criterion at convergence: 58389.3
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -4.3979 -0.5826  0.0737  0.6521  3.4952
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Year     (Intercept) 0.008060 0.08978
##  OP       (Intercept) 0.025631 0.16010
##  age      (Intercept) 0.068338 0.26141
##  genre    (Intercept) 0.000441 0.02100
##  Residual             2.795825 1.67207
## Number of obs: 15082, groups:  Year, 9; OP, 7; age, 6; genre, 2
```

```
##
## Fixed effects:
##                             Estimate Std. Error t value
## (Intercept)                4.1081843  0.1403198   29.28
## satisfaction_care          0.1364299  0.0088328   15.45
## satisfaction_institution   0.2109299  0.0093703   22.51
## trust_institution         -0.0005267  0.0104985   -0.05
## trust_personne             0.6105600  0.0220637   27.67
##
## Correlation of Fixed Effects:
##             (Intr) stsfctn_c stsfctn_n trst_n
## satsfctn_cr -0.204
## stsfctn_nst  0.023 -0.227
## trst_nstttn -0.066 -0.224    -0.439
## trust_prsnn -0.217 -0.058    -0.101    -0.242
```

12.3 Sem avec Lavaan

On reste sur le jeu de données précédent

Table 12.2

	Model 1	Model 2	Model 3
(Intercept)	3.909 ***	4.028 ***	4.883 ***
	(0.056)	(0.070)	(0.109)
satisfaction_care	0.153 ***	0.153 ***	0.136 ***
	(0.009)	(0.009)	(0.009)
satisfaction_institution	0.212 ***	0.213 ***	0.211 ***
	(0.009)	(0.009)	(0.009)
trust_institution	0.010	0.012	0.000
	(0.010)	(0.010)	(0.011)
trust_personne	0.583 ***	0.574 ***	0.611 ***
	(0.022)	(0.022)	(0.022)
Year2004		-0.150 *	-0.191 **
		(0.060)	(0.061)
Year2006		-0.204 ***	-0.256 ***
		(0.058)	(0.059)
Year2008		-0.169 **	-0.243 ***
		(0.058)	(0.059)
Year2010		-0.191 **	-0.226 ***
		(0.060)	(0.061)
Year2012		-0.140 *	-0.160 **
		(0.059)	(0.060)
Year2014		-0.007	-0.055
		(0.060)	(0.061)
Year2016		0.010	-0.018
		(0.058)	(0.059)
Year2018		-0.102	-0.105
		(0.059)	(0.060)
age26-35			-0.133 *
			(0.055)
age36-45			-0.359 ***

Chapter 13

Arbres de Décision

L'objectif de cette note est double. Le premier est une introduction aux méthodes d'arbres de décision et leur généralisation récente par les random forests. Le second est d'introduire à l'approche d'apprentissage et de test, autrement aux machine learning avec le package caret qui facilite la condition des opérations d'échantillonngage, de découpage des échanges et de production des indicateurs.

13.1 Construire un arbre de décision

Les origines et le principe

C'est une approche qui remonte à Morgan and Sonquist (1963)

généralisés aux variables qualitatives avec Chaid (Kass (1980)) :

Le principe général suis le pseudo algorithme suivant :

pour chaque variable potentiellement explicative, trouver le meilleur découpage (dichotomique), c'est à dire celui qui va différencier au mieux la variable de réponse. Choisir parmi les variables et leur dichotomitsation celle qui répond au même critère que précedemment recommencer l'opération à 1 Il peut s'appliquer à une variable quantitative (regression) ou qualitative (chaid)

puis Cart avec breiman. Breiman (1998)

13.2 Mise en oeuvre avec Partykit

Le package partykit a pour objectif de représenter les arbres de décisions. Il inclue cependant plusieurs méthodes d'arbres de decisions, en en particulier une

approche ctree Hothorn et al. (2006) dont le principe est. La méthode est incluse dans partykit Hothorn and Zeileis (2015)

Avec partykit on contrôle la construction de l'arbre sur différents critères, par exemple : * le type de test employé pour prendre la décision * le nombre minimum d'individus dans une feuille terminale

https://apiacoa.org/blog/2014/02/initiation-a-rpart.fr.html

https://apiacoa.org/blog/2014/02/initiation-a-rpart.fr.html

```
knitr::opts_chunk$set(echo = TRUE, include=TRUE, cache=TRUE, message=FALSE, warning=FA

library(partykit)

library(tidyverse)
#lecture du fichier
df<-readRDS("./data/last.rds") %>%drop_na()
df$Age<-as.factor(df$Age)
df$Sexe<-as.factor(df$Sexe)
df$Education<-factor(df$Education, ordered = FALSE )
df$Situation2<-as.factor(df$Situation2)

df$Situation3<-as.factor(ifelse(df$Situation<5,"degradation"," Amelioration"))
table(df$Situation3)
```

```
##
##  Amelioration   degradation
##         20327         19071
```

```
fit <-ctree(Situation3 ~ Age+Sexe+Education, data=df)
print(fit)
```

```
##
## Model formula:
## Situation3 ~ Age + Sexe + Education
##
## Fitted party:
## [1] root
## |   [2] Age in 18 - 24 ans, 25 - 34 ans
## |   |   [3] Age in 18 - 24 ans
## |   |   |   [4] Sexe in Un homme
## |   |   |   |   [5] Education in Sans diplôme, Brevet des collèges, Bac +5 (Master,
## |   |   |   |   [6] Education in CAP/BEP, Bac (général, pro et technologique), Bac +
## |   |   |   [7] Sexe in Une femme:  Amelioration (n = 3963, err = 38.3%)
## |   |   [8] Age in 25 - 34 ans
```

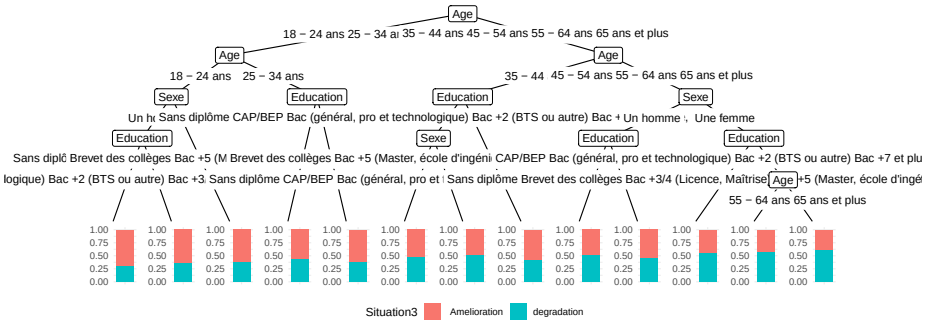
```
## |   |   |   [9] Education in Sans diplôme, CAP/BEP, Bac (général, pro et technologique), Bac +
## |   |   |   [10] Education in Brevet des collèges, Bac +5 (Master, école d'ingénieurs, d'arts.
## |   [11] Age in 35 - 44 ans, 45 - 54 ans, 55 - 64 ans, 65 ans et plus
## |   |   [12] Age in 35 - 44 ans
## |   |   |   [13] Education in Sans diplôme, CAP/BEP, Bac (général, pro et technologique), Bac
## |   |   |   |   [14] Sexe in Un homme:  Amelioration (n = 2588, err = 47.6%)
## |   |   |   |   [15] Sexe in Une femme: degradation (n = 4926, err = 48.8%)
## |   |   |   [16] Education in Brevet des collèges, Bac +5 (Master, école d'ingénieurs, d'arts.
## |   |   [17] Age in 45 - 54 ans, 55 - 64 ans, 65 ans et plus
## |   |   |   [18] Sexe in Un homme
## |   |   |   |   [19] Education in Sans diplôme, CAP/BEP, Bac (général, pro et technologique),
## |   |   |   |   [20] Education in Brevet des collèges, Bac +5 (Master, école d'ingénieurs, d'a
## |   |   |   [21] Sexe in Une femme
## |   |   |   |   [22] Education in Sans diplôme, Brevet des collèges, Bac +3/4 (Licence, Maîtri
## |   |   |   |   [23] Education in CAP/BEP, Bac (général, pro et technologique), Bac +2 (BTS ou
## |   |   |   |   |   [24] Age in 45 - 54 ans: degradation (n = 2929, err = 43.3%)
## |   |   |   |   |   [25] Age in 55 - 64 ans, 65 ans et plus: degradation (n = 3724, err = 39.4
##
## Number of inner nodes:    12
## Number of terminal nodes: 13
```

```
pred <- predict(fit, df)
library(caret)
cm = confusionMatrix(df$Situation3, pred)
print(cm)
```

```
## Confusion Matrix and Statistics
##
##                Reference
## Prediction       Amelioration degradation
##    Amelioration         10507        9820
##   degradation            7387       11684
##
##                Accuracy : 0.5633
##                  95% CI : (0.5583, 0.5682)
##     No Information Rate : 0.5458
##     P-Value [Acc > NIR] : 1.769e-12
##
##                   Kappa : 0.129
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.5872
##             Specificity : 0.5433
##          Pos Pred Value : 0.5169
```

```
##          Neg Pred Value : 0.6127
##              Prevalence : 0.4542
##          Detection Rate : 0.2667
##    Detection Prevalence : 0.5159
##       Balanced Accuracy : 0.5653
##
##        'Positive' Class :  Amelioration
##
```

```
library(ggparty)
autoplot(fit)
```




```
#library(irks)
#rules<-ct_rules(fit)

#rules %>%
#  kable() %>%
#  kable_styling(bootstrap_options = "striped", font_size = 10)
```

https://topepo.github.io/caret/

13.3 forêts aléatoires

voire le cas

Chapter 14

Premiers éléments de Machine Learning

Le Machine Learning ou en français l'apprentissage automatique désigne un processus par lequel on construit un modèle prédictif. C'est en général un modèle qui permet de classifier des observations à parti de leurs caractéristiques. Distinguer les spam du ham, les mauvais emprunteurs des bons emprunteurs, des cookies brulés de ceux bien cuits sur chaine de production. Trois éléments essentiels le caractérise.

- un modèle qui asssocie à une variable y que l'on cherche à prédire un vecteur de caractéristiques x. c'est une fonction où y=f(x|theta)+e. Aujourd'hui des dizaines de fonction sont disponibles

- un processus de validation. Su le modèle est entrainé sur un set de données dit d'apprentissage, sa qualité prédictive est établie sur un set de données qui n'a pas contribué à l'estimation des paramètres.

-

14.1 une typologie de modèles

14.1.1 le modèle linéaire

14.1.2 le modèle logit

14.1.3 les modèles à régularisation

14.1.4 les random forest

14.2 forêts aléatoires

voire le cas

et pour le texte :

et surtout celui-ci

Chapter 15

20 Annexes

15.1 Données Eric-ESS

Extraction France

Le fichier est disponible sous format ESS10fr.rds

```
df<-readRDS("./data/trustFrAll.rds")
write.csv(df, "./data/dfTrust.rds")
df<-read.csv("./data/dfTrust.rds")

#quelques recodages
#on renomme pour plus de clarté

names(df)[names(df)=="trstun"] <- "NationsUnies"
names(df)[names(df)=="trstep"] <- "ParlementEurop"
names(df)[names(df)=="trstlgl"] <- "Justice"
names(df)[names(df)=="trstplc"] <- "Police"
names(df)[names(df)=="trstplt"] <- "Politiques"
names(df)[names(df)=="trstprl"] <-"Parlement"
names(df)[names(df)=="trstprt"] <- "Partis"
names(df)[names(df)=="pplhlp"] <- "help"
names(df)[names(df)=="pplfair"] <- "fair"
names(df)[names(df)=="ppltrst"] <- "trust"

#on construit les scores de confiance
df<-df %>%
  mutate(trust_institut=(Partis+Parlement+Politiques+Police+Justice+NationsUnies+ParlementEurop)*
df$Year<-2000
#recodage des variables independantes
```

```
df$Year[df$essround==1]<-2002
df$Year[df$essround==2]<-2004
df$Year[df$essround==3]<-2006
df$Year[df$essround==4]<-2008
df$Year[df$essround==5]<-2010
df$Year[df$essround==6]<-2012
df$Year[df$essround==7]<-2014
df$Year[df$essround==8]<-2016
df$Year[df$essround==9]<-2018
df$Year<-as.factor(df$Year)

df$OP<-" "
#ggplot(df,aes(x=lrscale))+geom_histogram()
df$OP[df$lrscale==0] <- "Extrême gauche"
df$OP[df$lrscale==1] <- "Gauche"
df$OP[df$lrscale==2] <- "Gauche"
df$OP[df$lrscale==3] <- "Centre Gauche"
df$OP[df$lrscale==4] <- "Centre Gauche"
df$OP[df$lrscale==5] <- "Ni G ni D"
df$OP[df$lrscale==6] <- "Centre Droit"
df$OP[df$lrscale==7] <- "Centre Droit"
df$OP[df$lrscale==8] <- "Droite"
df$OP[df$lrscale==9] <- "Droite"
df$OP[df$lrscale==10] <- "Extrême droite"
#la ligne suivante est pour ordonner les modalités de la variables
df$OP<-factor(df$OP,levels=c("Extrême droite","Droite","Centre Droit","Ni G ni D","Cent


df$revenu<-" "
df$revenu[df$hincfel>4] <- NA
df$revenu[df$hincfel==1] <- "Vie confortable"
df$revenu[df$hincfel==2] <- "Se débrouille avec son revenu"
df$revenu[df$hincfel==3] <- "Revenu insuffisant"
df$revenu[df$hincfel==4] <- "Revenu très insuffisant"
df$revenu<-factor(df$revenu,levels=c("Vie confortable","Se débrouille avec son revenu"

df$habitat<-" "

df$habitat[df$domicil==1]<- "Big city"
df$habitat[df$domicil==2]<-"Suburbs"
df$habitat[df$domicil==3]<-"Town"
df$habitat[df$domicil==4]<-"Village"
df$habitat[df$domicil==5]<-"Countryside"
df$habitat<-factor(df$habitat,levels=c("Big city","Suburbs","Town","Village","Countrysi
```

```
df$genre<-" "

df$genre[df$gndr==1]<-"H"
df$genre[df$gndr==2]<-"F"

df$age<-" "

df$age[df$agea<26]<-"25<"
df$age[df$agea>25 & df$agea<36]<-"26-35"
df$age[df$agea>35 & df$agea<46]<-"36-45"
df$age[df$agea>45 & df$agea<66]<-"46-65"
df$age[df$agea>65 & df$agea<76]<-"66-75"
df$age[df$agea>75]<-"75>"
df$age<-factor(df$age,levels=c("25<","26-35","36-45","46-65","66-75", "75>"))

saveRDS(df, "./data/dfTrust.rds")


######
##########
df <- read_csv("Data/ESS-Data-Wizard-subset-2022-10-06.csv")

df$Year<-2000
#recodage des variables independantes
df$Year[df$essround==1]<-2002
df$Year[df$essround==2]<-2004
df$Year[df$essround==3]<-2006
df$Year[df$essround==4]<-2008
df$Year[df$essround==5]<-2010
df$Year[df$essround==6]<-2012
df$Year[df$essround==7]<-2014
df$Year[df$essround==8]<-2016
df$Year[df$essround==9]<-2018
df$Year<-as.factor(df$Year)

df$OP<-" "
#ggplot(df,aes(x=lrscale))+geom_histogram()
df$OP[df$lrscale==0] <- "Extrême gauche"
df$OP[df$lrscale==1] <- "Gauche"
df$OP[df$lrscale==2] <- "Gauche"
df$OP[df$lrscale==3] <- "Centre Gauche"
df$OP[df$lrscale==4] <- "Centre Gauche"
df$OP[df$lrscale==5] <- "Ni G ni D"
df$OP[df$lrscale==6] <- "Centre Droit"
df$OP[df$lrscale==7] <- "Centre Droit"
```

```
df$OP[df$lrscale==8] <- "Droite"
df$OP[df$lrscale==9] <- "Droite"
df$OP[df$lrscale==10] <- "Extrême droite"
#la ligne suivante est pour ordonner les modalités de la variables
df$OP<-factor(df$OP,levels=c("Extrême droite","Droite","Centre Droit","Ni G ni D","Cent

df$genre<-" "
df$genre[df$gndr==1]<-"H"
df$genre[df$gndr==2]<-"F"

df$age<-" "
df$age[df$agea<26]<-"25<"
df$age[df$agea>25 & df$agea<36]<-"26-35"
df$age[df$agea>35 & df$agea<46]<-"36-45"
df$age[df$agea>45 & df$agea<66]<-"46-65"
df$age[df$agea>65 & df$agea<76]<-"66-75"
df$age[df$agea>75]<-"75>"
df$age<-factor(df$age,levels=c("25<","26-35","36-45","46-65","66-75", "75>"))

df_confiance<-df%>%
  dplyr::select(trstep, trstlgl,trstplc,trstplt,trstprl,trstprt,trstun,pplfair,pplhlp,
  mutate(trstep=ifelse(trstep==77 |trstep==88| trstep==66,NA,trstep),
         trstlgl=ifelse(trstlgl==77 |trstlgl==88| trstlgl==66,NA,trstlgl),
         trstplc=ifelse(trstplc==77 |trstplc==88| trstplc==66,NA,trstplc),
         trstplt=ifelse(trstplt==77 |trstplt==88| trstplt==66,NA,trstplt),
         trstprt=ifelse(trstprt==77 |trstprt==88| trstprt==66,NA,trstprt),
         trstun=ifelse(trstun==77 |trstun==88| trstun==66,NA,trstun),
         trstprl=ifelse(trstprl==77 |trstprl==88| trstprl==66,NA,trstprl),
         pplfair=ifelse(pplfair==77 |pplfair==88| pplfair==66,NA,pplfair),
         pplhlp=ifelse(pplhlp==77 |pplhlp==88| pplhlp==66,NA,pplhlp),
         ppltrst=ifelse(ppltrst==77 |ppltrst==88| ppltrst==66,NA,ppltrst),
         trust_institution= (trstep+trstlgl+trstplc+trstplt+trstprl+trstprt+trstun)/7,
         trust_personne= (pplfair+pplhlp+ppltrst)/7
         )



df_satisfaction<-df%>%
  dplyr::select(stfeco,stfedu,stfgov, stfhlth, stflife,stfmjob, happy) %>%
  mutate(stfeco=ifelse(stfeco==77 |stfeco==88| stfeco==66,NA,stfeco),
         stfedu=ifelse(stfedu==77 |stfedu==88 | stfedu==66,NA,stfedu),
         stfgov=ifelse(stfgov==77 |stfgov==88 | stfgov==66, NA, stfgov),
         stfhlth=ifelse(stfhlth==77 |stfhlth==88|stfhlth==66,NA,stfhlth),
         stflife=ifelse(stflife==77 |stflife==88|stflife==66,NA,stflife),
         happy=ifelse(happy==77 |happy==88|happy==66,NA,happy),
```

```
          stfmjob=ifelse(stfmjob==77 |stfmjob==88|stfmjob==66,NA,stfmjob),
          satisfaction_vie=(stflife+happy)/2, #à examiner
          satisfaction_institution=(stfeco+stfgov)/2,
          satisfaction_care=(stfhlth+stfedu)/2
)

df_sample<-df%>%
  dplyr::select(Year, age, OP, genre)


df<-cbind(df_sample, df_satisfaction, df_confiance)

saveRDS(df, "./data/ESS10fr.rds")
```

15.2 fichier Airbnb Bruxelles

La source est InsideAirbnb. L'extraction est de 2020.

15.3

Bibliography

Schwartz, S. H. (2006). Les valeurs de base de la personne : théorie, mesures et applications:. *Revue française de sociologie*, Vol. 47(4):929–968.

Sneath, P. H. A. and Sokal, R. R. (1973). *Numerical taxonomy: the principles and practice of numerical classification*. A Series of books in biology. W. H. Freeman, San Francisco.

Thurstone, L. L. (1931). Multiple factor analysis. *Psychological Review*, 38(5):406–427.