

# MATH 60602 - Analyse multidimensionnelle appliquée

Denis Larocque              Léo Belzile

décembre 2022



# Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Survol du cours . . . . .	1
1.1.1	Analyse factorielle exploratoire . . . . .	1
1.1.2	Analyse de regroupements . . . . .	2
1.1.3	Sélection de variables et de modèles . . . . .	3
1.1.4	Régression logistique . . . . .	3
1.1.5	Analyse de survie . . . . .	3
1.1.6	Données manquantes . . . . .	4
<b>2</b>	<b>Analyse exploratoire</b>	<b>5</b>
2.1	Types de variables . . . . .	5
2.2	Validation des données. . . . .	7
2.3	Graphiques . . . . .	7
2.4	Exemple . . . . .	10
2.4.1	Commentaire sur les graphiques . . . . .	17
<b>3</b>	<b>Réduction de la dimension</b>	<b>19</b>
3.1	Introduction . . . . .	19
3.2	Coefficient de corrélation linéaire . . . . .	20
3.3	Présentation des données . . . . .	20
3.4	Analyse en composantes principales . . . . .	23
3.4.1	Choix du nombre de composantes principales . . . . .	27

3.4.2	Formulation mathématique . . . . .	29
3.5	Analyse factorielle exploratoire . . . . .	30
3.5.1	Rotation des facteurs . . . . .	32
3.5.2	Choix du nombre de facteurs . . . . .	35
3.5.3	Construction d'échelles à partir des facteurs . . . . .	36
3.6	Compléments d'information . . . . .	38
3.6.1	Variables ordinaires . . . . .	38
3.6.2	Autres méthodes de rotation des facteurs . . . . .	39
3.6.3	Scores factoriels . . . . .	39
<b>4</b>	<b>Sélection de variables et de modèles</b>	<b>43</b>
4.1	Introduction . . . . .	43
4.2	Sélection de variables et de modèles selon les buts de l'étude . . . . .	44
4.3	Estimation de la performance . . . . .	44
4.3.1	Surajustement . . . . .	46
4.3.2	Principes généraux . . . . .	46
4.3.3	Présentation de l'exemple . . . . .	47
4.3.4	Pénalisation et critères d'information . . . . .	50
4.3.5	Validation externe . . . . .	54
4.3.6	Validation croisée . . . . .	55
4.4	Présentation des données . . . . .	58
4.5	Sélection de variables . . . . .	63
4.5.1	Recherche exhaustive (meilleurs sous-ensembles) . . . . .	63
4.5.2	Méthodes séquentielles de sélection . . . . .	67
4.5.3	Méthodes de régression avec régularisation . . . . .	69
4.5.4	Moyenne de modèles . . . . .	73
4.6	Évaluation de la performance . . . . .	77

<b>5 Régression logistique</b>	<b>81</b>
5.1 Introduction . . . . .	81
5.2 Modèle de régression logistique . . . . .	81
5.2.1 Estimation et interprétation des paramètres . . . . .	83
5.2.2 Méthode du maximum de vraisemblance . . . . .	85
5.2.3 Exemple du <i>Professional Rodeo Cowboys Association</i> . . . . .	86
5.2.4 Modèle avec une seule variable explicative . . . . .	87
5.2.5 Interprétation du paramètre . . . . .	89
5.2.6 Test du rapport de vraisemblance . . . . .	93
5.2.7 Multicolinéarité . . . . .	94
5.3 Classification et prédiction . . . . .	95
5.3.1 Fonction d'efficacité du récepteur . . . . .	100
5.3.2 Classification avec une matrice de gain . . . . .	102
5.3.3 Courbe lift . . . . .	105
5.3.4 Calibration du modèle et détection du surajustement . . . . .	106
5.3.5 Sélection de variables en régression logistique . . . . .	107
5.3.6 Modèle Heckit . . . . .	115
Description du modèle Heckit . . . . .	115
5.4 Modèles pour données multinomiales . . . . .	118
5.4.1 Régression logistique multinomiale . . . . .	118
5.4.2 Régression logistique cumulative à cotes proportionnelles . . . . .	123
<b>6 Analyse de survie</b>	<b>129</b>
6.1 Introduction . . . . .	129
6.1.1 Exemple du temps d'abonnement . . . . .	129
6.1.2 Contexte . . . . .	130
6.2 Fonctions de survie et de risque . . . . .	131
6.3 Estimation d'une courbe de survie et de risque . . . . .	132
6.3.1 Calcul de l'estimateur de Kaplan–Meier . . . . .	135

6.4 Comparaison de deux courbes de survie . . . . .	135
6.5 Modèle à risques proportionnels de Cox . . . . .	139
6.5.1 Description du modèle de Cox . . . . .	139
6.5.2 Estimation de la fonction de survie pour des valeurs particulières des variables explicatives . . . . .	143
6.5.3 Postulat de risques proportionnels . . . . .	145
6.6 Extensions du modèle de Cox . . . . .	147
6.6.1 Variables explicatives dont la valeur change dans le temps . . . . .	147
6.6.2 Modèle à risques compétitifs . . . . .	149
6.7 Risques non proportionnels . . . . .	152
6.7.1 Non-proportionnalité avec un terme d'interaction avec le temps . . . . .	152
6.7.2 Stratification . . . . .	155
<b>7 Analyse de regroupements</b>	<b>159</b>
7.1 Introduction . . . . .	159
7.2 Données . . . . .	160
7.3 Choix des variables . . . . .	161
7.4 Mesures de dissemblance . . . . .	164
7.4.1 Mesures de dissemblance . . . . .	165
7.4.2 Dissemblance et valeurs manquantes . . . . .	168
7.5 Algorithmes pour la segmentation . . . . .	169
7.5.1 $K$ -moyennes . . . . .	170
7.5.2 $K$ -médoides . . . . .	180
7.5.3 Mélange de modèles . . . . .	183
7.5.4 Regroupements hiérarchiques . . . . .	187
7.5.5 Méthodes basées sur la densité . . . . .	191
7.6 Conclusion . . . . .	194

<b>8 Données manquantes</b>	<b>197</b>
8.1 Principes de base . . . . .	197
8.2 Méthodes d'imputation . . . . .	198
8.2.1 Cas complets . . . . .	198
8.2.2 Imputation simple . . . . .	199
8.2.3 Imputation multiple . . . . .	199
8.3 Example d'application de l'imputation . . . . .	201



# Chapitre 1

## Introduction

Ces notes servent de support de cours pour MATH 60602 *Analyse multidimensionnelle appliquée*, un cours du programme de la maîtrise en gestion profil intelligence d'affaires offerte par HEC Montréal. Le cours offre une formation de base en traitement de données multidimensionnelles en axant sur la compréhension intuitive, l'interprétation et l'utilisation de plusieurs techniques statistiques à l'aide de logiciels appropriés.

### 1.1 Survol du cours

#### 1.1.1 Analyse factorielle exploratoire

On dispose de  $p$  variables  $X_1, \dots, X_p$ . Peut-on expliquer les interrelations (la structure de corrélation) entre ces variables à l'aide d'un certain nombre (moins de  $p$ ) de facteurs latents (non observés) ?

L'analyse factorielle est souvent utilisée pour analyser des questionnaires (construction d'échelles) comme dans l'exemple suivant.

**Exemple 1.1.** Pour les besoins d'une enquête, on a demandé à 200 consommateurs adultes de répondre aux questions suivantes par rapport à un certain type de magasin :

Sur une échelle de 1 à 5,

1. pas important
2. peu important
3. moyennement important
4. assez important

### 5. très important

Pour vous, à quel point est-ce important...

1. que le magasin offre de bons prix tous les jours?
2. que le magasin accepte les cartes de crédit majeures (Visa, Mastercard)?
3. que le magasin offre des produits de qualité?
4. que les vendeurs connaissent bien les produits?
5. qu'il y ait des ventes spéciales régulièrement?
6. que les marques connues soient disponibles?
7. que le magasin ait sa propre carte de crédit?
8. que le service soit rapide?
9. qu'il y ait une vaste sélection de produits?
10. que le magasin accepte le paiement par carte de débit?
11. que le personnel soit courtois?
12. que le magasin ait en stock les produits annoncés?

Pouvons-nous identifier un nombre restreint de facteurs (concepts, dimensions) qui pourraient bien rendre compte de la structure de corrélation entre ces 12 variables?

Buts :

- Décrire et comprendre la structure de corrélation d'un ensemble de variables à l'aide d'un nombre restreint de concepts (appelés facteurs).
- Réduire le nombre de variables en créant une nouvelle variable par facteur. Ces nouvelles variables pourront par la suite être utilisées dans d'autres analyses (régression linéaire multiple par exemple).

#### 1.1.2 Analyse de regroupements

On cherche à créer des groupes (« *clusters* ») d'individus homogènes en utilisant  $p$  variables  $X_1, \dots, X_p$ .

Cette méthode est utilisée en marketing pour la **segmentation de marché**, qui consiste en

...définir des sous-groupes réunissant des consommateurs qui partagent les mêmes préférences ou qui réagissent de façon semblable à des variables de marketing<sup>1</sup>

But :

- Combiner des sujets en groupes (interprétables) de telle sorte que les individus d'un même groupe soient les plus semblables possible par rapport à certaines caractéristiques et que les groupes soient les plus différents possible.

---

1. d'Astous, A. (2000). *Le projet de recherche en marketing*, 2e édition. Chenelière/McGraw-Hill.

### 1.1.3 Sélection de variables et de modèles

Dans plusieurs situations, on doit développer un modèle de prévision. Par exemple, on pourrait devoir développer un modèle pour :

- Déetecter les faillites des clients (ou des entreprises)
- Cibler les clients qui seront intéressés par une offre promotionnelle
- Déetecter les fraudes (par carte de crédit ou dans les rapports de revenus)
- Prévoir si un client va nous quitter.

Il y a en général plusieurs variables explicatives potentielles, et aussi plusieurs types de modèles possibles (régression linéaire, réseaux de neurones, arbres de régression ou de classification, etc.). Dans ce chapitre, nous verrons des principes généraux et des outils afin de sélectionner des modèles performants, ou bien un sous-ensemble de variables avec un bon pouvoir prévisionnel.

### 1.1.4 Régression logistique

On cherche à expliquer le comportement d'une variable binaire  $Y$  ( $0 - 1$ ), à l'aide de  $p$  variables quelconques  $X_1, \dots, X_p$ .

**Exemple 1.2.** Une banque offre aux gens la possibilité de faire une demande de carte de crédit en ligne en promettant une approbation (conditionnelle) en quelques minutes seulement. Le tout est basé sur un modèle automatique de classification qui décide d'accorder ou non la carte ( $Y = 1$  ou  $Y = 0$ ) en fonction des réponses fournies par les clients potentiels à différentes questions comme : quel est votre revenu annuel brut ( $X_1$ ), avez-vous d'autres cartes de crédit ( $X_2$ ), êtes-vous locataire ou propriétaire ( $X_3$ ), etc...

Buts :

- Comprendre comment et dans quelle mesure les variables  $X$  influencent la catégorie d'appartenance de  $Y$ .
- Développer un modèle pour faire de la classification, c'est-à-dire, prévoir la catégorie d'appartenance de  $Y$  pour un nouveau sujet à partir des variables  $X$ .

### 1.1.5 Analyse de survie

On s'intéresse au temps avant qu'un événement survienne. Par exemple :

- Temps qu'un client demeure abonné à un service offert par notre compagnie.
- Temps de survie d'un individu après avoir été diagnostiqué avec un certain type de cancer.
- Temps qu'un employé demeure au service de la compagnie.

- Temps qu'une franchise demeure en activité.
- Temps avant la faillite d'une entreprise (ou d'un particulier).
- Temps avant le prochain achat d'un client.

On observe chaque sujet jusqu'à ce que l'une des deux choses suivantes se produise : l'événement survient avant la fin de la période d'observation ou bien l'étude se termine et l'événement n'est toujours pas survenu. Dans le premier exemple, l'événement correspond au fait d'interrompre son abonnement. On dispose donc d'une variable temps  $T$  pour chaque individu qui est soit censurée, soit non censurée. Si l'individu a expérimenté l'événement avant la fin de la période d'observation, la valeur de  $T$  est non censurée. Si l'événement n'est toujours pas survenu à la fin de la période d'observation, la valeur de  $T$  est censurée. Pour chaque individu, on dispose également d'un ensemble de variables explicatives  $X_1, \dots, X_p$ .

But :

- Étudier les effets des variables explicatives sur le temps de survie et obtenir des prévisions du temps de survie.

### 1.1.6 Données manquantes

Il arrive fréquemment d'avoir des valeurs manquantes dans notre échantillon.

Simplement ignorer les sujets avec des valeurs manquantes et faire l'analyse avec les autres sujets conduit généralement à des estimations biaisées et à de l'inférence invalide.

Dans ce chapitre, nous verrons une méthode très générale afin de traiter les données manquantes, l'imputation multiple. Nous verrons comment elle peut être utilisée dans un contexte d'inférence et dans un contexte de prévision.

# Chapitre 2

## Analyse exploratoire

L'analyse exploratoire, comme son nom l'indique, est une étape préliminaire à la modélisation servant à l'acquisition d'une meilleure compréhension des données. L'analyse exploratoire sert à nous assurer que notre analyse ou notre traitement de ces dernières est cohérent. Le but de l'analyse exploratoire graphique est d'extraire des informations utiles, le plus souvent par le biais d'une série de questions qui sont raffinées au fur et à mesure que progresse l'analyse. On s'intéresse particulièrement aux relations et interactions entre différentes variables et la distribution empirique de chaque variable. Les étapes majeures sont :

1. Formuler des questions sur les données
2. Chercher des réponses à ces questions à l'aide de statistiques descriptives, de tableaux de fréquence ou de contingence et de graphiques.
3. Raffiner nos questions, et utiliser les trouvailles pour peaufiner notre analyse

Dans un rapport, un résumé des caractéristiques les plus importantes devrait être inclus pour que le lecteur ou la lectrice puisse valider son interprétation des données.

### 2.1 Types de variables

- Une **variable** représente une caractéristique de la population d'intérêt, par exemple le sexe d'un individu, le prix d'un article, etc.
- une **observation**, parfois appelée donnée, est un ensemble de mesures collectées sous des conditions identiques, par exemple pour un individu ou à un instant donné.

Le choix de modèle statistique ou de test dépend souvent du type de variables collectées. Les variables peuvent être de plusieurs types : quantitatives (discrètes ou continues) si elles prennent

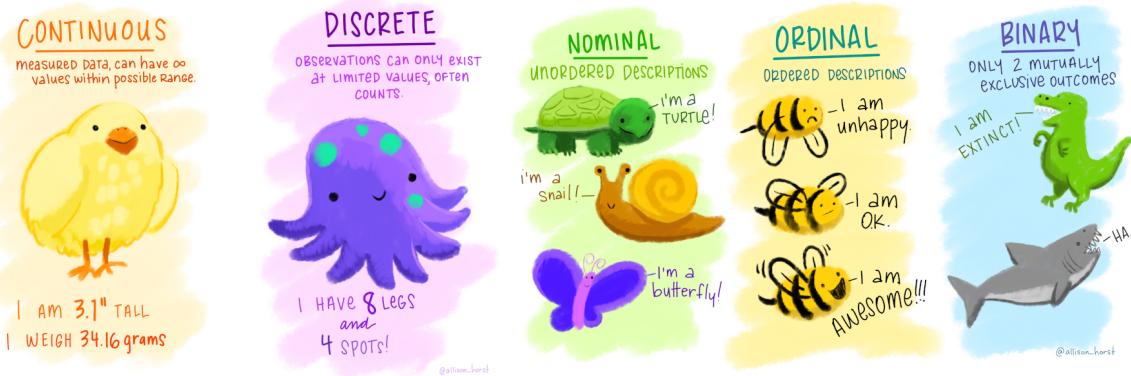


FIGURE 2.1 – Illustrations par Allison Horst de variables numériques (gauche) et catégorielles (droite).

des valeurs numériques, qualitatives (binaires, nominales ou ordinaires) si elles sont décrites par un adjectif; je préfère le terme catégorielle, plus évocateur.

On distingue deux types de variables quantitatives :

- une variable discrète prend un nombre dénombrable de valeurs; ce sont souvent des variables de dénombrement ou des variables dichotomiques.
- une variable continue peut prendre (en théorie) une infinité de valeurs, même si les valeurs mesurées sont arrondies ou mesurées avec une précision limitée (temps, taille, masse, vitesse, salaire). Dans bien des cas, nous pouvons considérer comme continues des variables discrètes si elles prennent un assez grand nombre de valeurs.

Les variables catégorielles représentent un ensemble fini de possibilités. On les regroupe en deux types, pour lesquels on ne fera pas de distinction :

- nominales s'il n'y a pas d'ordre entre les modalités (sexe, couleur, pays d'origine) ou
- ordinales (échelle de Likert, tranche salariale).

La codification des modalités des variables catégorielles est arbitraire; en revanche, on préservera l'ordre lorsqu'on représentera graphiquement les variables ordinaires. Lors de l'estimation, chaque variable catégorielle doit être transformée en un ensemble d'indicateurs binaires : il est donc essentiel de déclarer ces dernières dans votre logiciel statistique, surtout si elles sont encodées dans la base de données à l'aide de valeurs entières.

## 2.2 Validation des données.

Avant de regarder les données, il est souvent utile de se plonger dans la description de la base de données. Il n'est pas rare que cette dernière contienne des informations pertinentes sur la codification des données, par exemple

- telle variable catégorielle est stockée avec des valeurs entières et les étiquettes ne sont disponibles que dans la description.
- des valeurs manquantes sont encodées avec  $-1$  (pour les variables positives) ou  $999$ .
- une variable est une fonction, transformation ou combinaison d'autres variables.

## 2.3 Graphiques

Le principal type de graphique pour représenter la distribution d'une variable catégorielle est le diagramme en bâtons, dans lequel la fréquence de chaque catégorie est présentée sur l'axe des ordonnées ( $y$ ) en fonction de la modalité, sur l'axe des abscisses ( $x$ ), et ordonnées pour des variables ordinaires. Cette représentation est en tout point supérieur au diagramme en camembert, une engeance répandue qui devrait être honnie (notamment parce que l'humain juge mal les différences d'aires, qu'une simple rotation change la perception du graphique et qu'il est difficile de mesurer les proportions) — ce n'est pas de la tarte!

```
library(ggplot2)
library(patchwork)
library(dplyr)
data(renfe, package = "hecmodstat")
g1 <- renfe |>
  count(classe) |>
  mutate(classe =forcats::fct_reorder(classe, n)) |>
ggplot(mapping = aes(y = classe, x = n)) +
  geom_col() +
  labs(subtitle = "classe",
       x = "dénombrément",
       y = "")
g2 <- renfe |>
  count(type) |>
  mutate(type =forcats::fct_reorder(type, n)) |>
ggplot(mapping = aes(y = type, x = n)) +
  geom_col() +
  labs(subtitle = "type de train",
```

```

x = "dénombrément",
y = ""))
g1 + g2

```

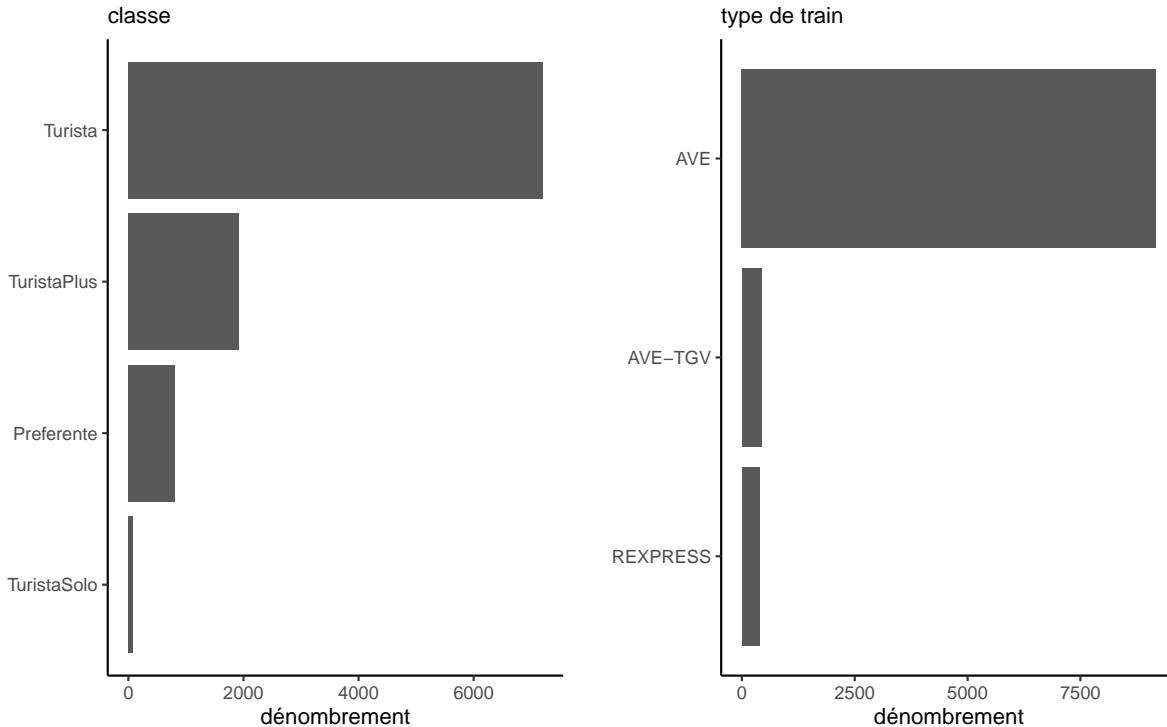


FIGURE 2.2 – Diagramme en bâtons pour la classe des billets de trains du jeu de données Renfe.

Puisque les variables continues peuvent prendre autant de valeurs distinctes qu'il y a d'observations, on ne peut simplement compter le nombre d'occurrence par valeur unique. On regroupera plutôt dans un certain nombre d'intervalle, en discrétilisant l'ensemble des valeurs en classes pour obtenir un histogramme. Le nombre de classes dépendra du nombre d'observations si on veut que l'estimation ne soit pas impactée par le faible nombre d'observations par classe : règle générale, le nombre de classes ne devrait pas dépasser  $\sqrt{n}$ , où  $n$  est le nombre d'observations de l'échantillon. On obtiendra la fréquence de chaque classe, mais si on normalise l'histogramme (de façon à ce que l'aire sous les bandes verticales égale un), on obtient une approximation discrète de la fonction de densité. Faire varier le nombre de classes permet parfois de faire apparaître des caractéristiques de la variable (notamment la multimodalité, l'asymétrie et les arrondis).

Puisque qu'on groupe les observations en classe pour tracer l'histogramme, il est difficile de voir l'étendue des valeurs que prenne la variable : on peut rajouter des traits sous l'histogramme pour représenter les valeurs uniques prises par la variable, tandis que la hauteur de l'histogramme nous

renseigne sur leur fréquence relative.

```
renfe |>
  subset(tarif == "Promo") |>
  ggplot(aes(x = prix)) +
  geom_histogram(aes(y = after_stat(density)),
                 bins = 30) +
  geom_rug(sides = "b") +
  labs(x = "prix de billets au tarif Promo (en euros)",
       y = "densité")
```

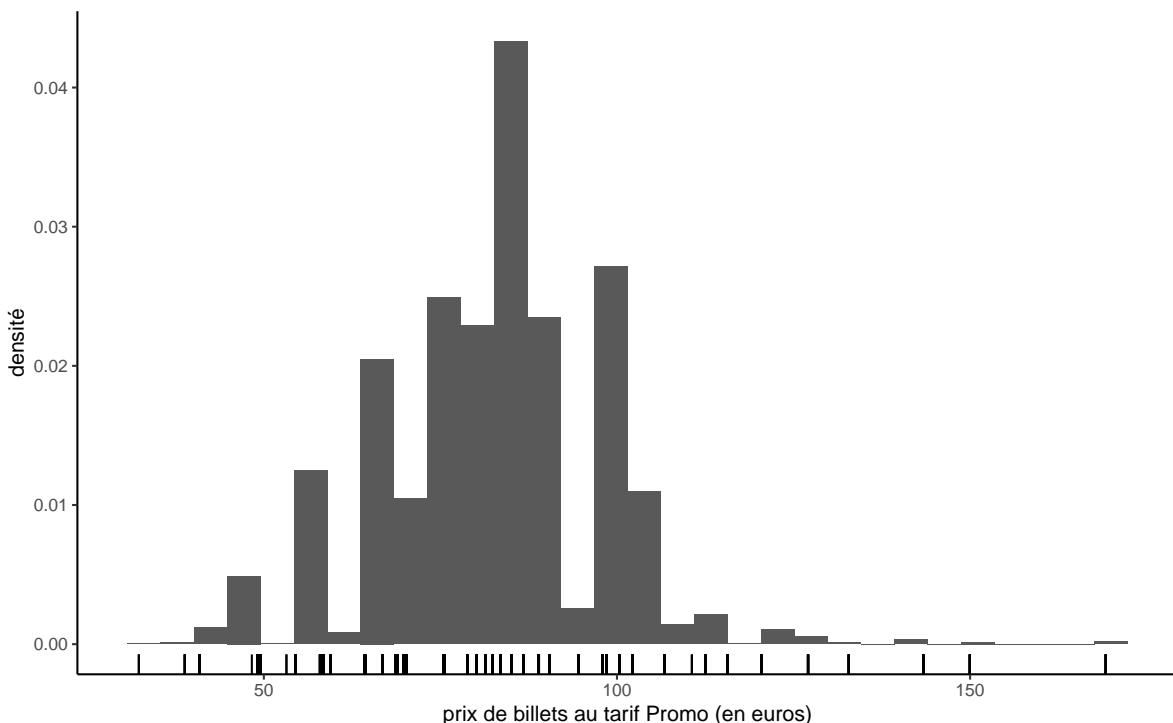


FIGURE 2.3 – Histogramme du prix des billets au tarif Promo de trains du jeu de données Renfe

Une boîte à moustaches représente graphiquement cinq statistiques descriptives.

- La boîte donne les 1e, 2e et 3e quartiles  $q_1, q_2, q_3$ . Il y a donc 50% des observations sont au-dessus/en-dessous de la médiane  $q_2$  qui sépare en deux la boîte.
- La longueur des moustaches est moins de 1.5 fois l'écart interquartile  $q_3 - q_1$  (tracée entre 3e quartile et le dernier point plus petit que  $q_3 + 1.5(q_3 - q_1)$ , etc.)
- Les observations au-delà des moustaches sont encerclées. Notez que plus le nombre d'observations est élevé, plus le nombres de valeurs extrême augmente. C'est un défaut de la boîte à

moustache, qui a été conçue pour des jeux de données qui passeraient pour petits selon les standards actuels.

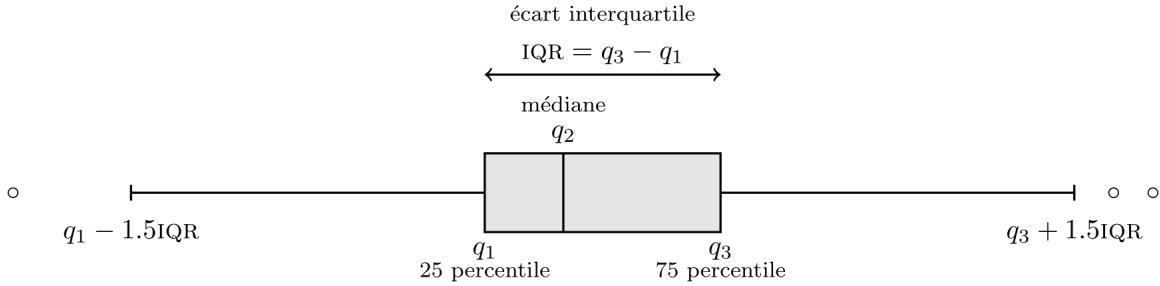


FIGURE 2.4 – Boîte à moustache.

On peut représenter la distribution d'une variable réponse continue en fonction d'une variable catégorielle en traçant une boîte à moustaches pour chaque catégorie et en les disposant côté-à-côte. Une troisième variable catégorielle peut être ajoutée par le biais de couleurs, comme dans la Figure 2.5.

```
renfe |>
  subset(tarif == "Promo") |>
  ggplot(aes(y = prix, x = classe, col = type)) +
  geom_boxplot() +
  labs(y = "prix (en euros)",
       col = "type de train") +
  theme(legend.position = "bottom")
```

Si on veut représenter la covariabilité de deux variables continues, on utilise un nuage de points où chaque variable est représentée sur un axe et chaque observation donne la coordonnée des points. Si la représentation graphique est dominée par quelques valeurs très grandes, une transformation des données peut être utile : vous verrez souvent des données positives à l'échelle logarithmique.

Plutôt que de décrire plus en détail le processus de l'analyse exploratoire, on présente un exemple qui illustre le cheminement habituel sur les données de trains de la Renfe introduites précédemment.

## 2.4 Exemple

La première étape consisterait à lire la description de la base de données. Le jeu de données `renfe` contient les variables suivantes :

- `prix` : prix du billet (en euros) ;

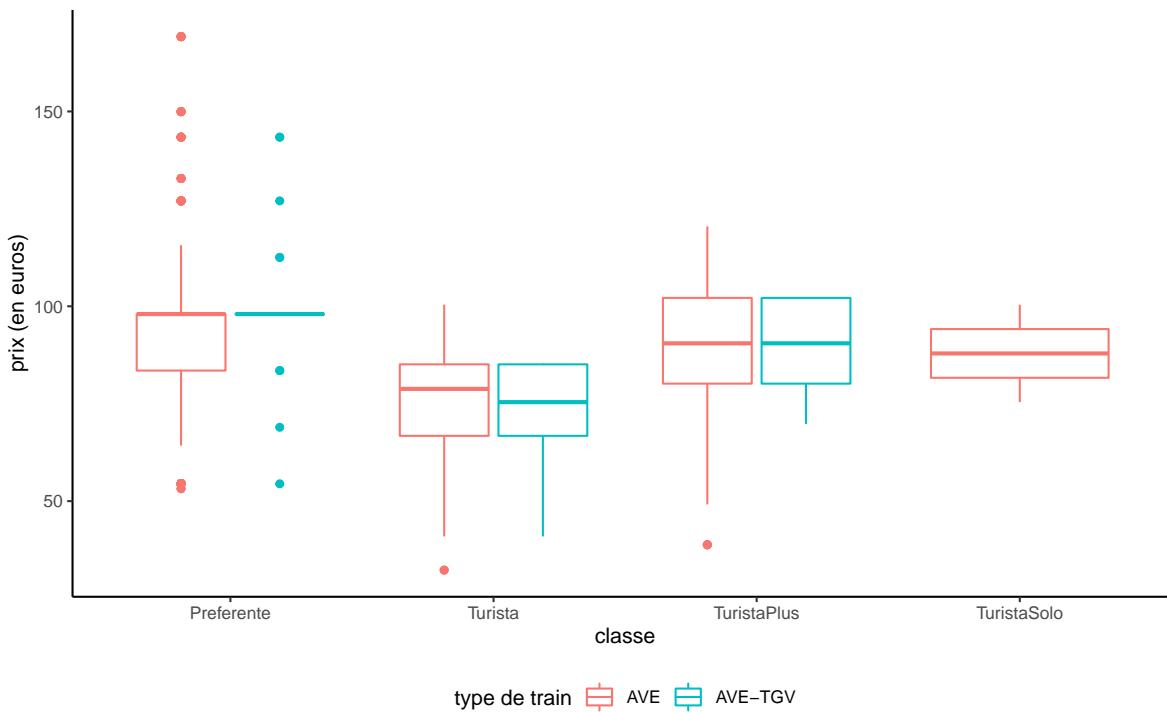


FIGURE 2.5 – Boîte à moustaches du prix des billets au tarif Promo en fonction de la classe pour le jeu de données Renfe.

- dest : indicateur binaire du trajet, soit de Barcelone vers Madrid (0) ou de Madrid vers Barcelone (1);
- tarif : variable catégorielle indiquant le tarif du billet, un parmi AdultoIda, Promo et Flexible;
- classe : classe du billet, soit Preferente, Turista, TuristaPlus ou TuristaSolo;
- type : variable catégorielle indiquant le type de train, soit Alta Velocidad Española (AVE), soit Alta Velocidad Española conjointement avec TGV (un partenariat entre la SNCF et Renfe pour les trains à destination ou en provenance de Toulouse) AVE-TGV, soit les trains régionaux REXPRESS ; seuls les trains étiquetés AVE ou AVE-TGV sont des trains à grande vitesse.
- duree : longueur annoncée du trajet (en minutes);
- jour entier indiquant le jour de la semaine du départ allant de dimanche (1) à samedi (7).

Il n'y a pas de valeurs manquantes et un aperçu des données (`head(renfe)`) montre qu'elles sont en format long, ce qui veut dire que chaque ligne contient une seule valeur pour la variable réponse, ici le prix d'un billet de train. On entame l'analyse exploratoire avec des questions plutôt vagues, par exemple

1. Quels sont les facteurs déterminant le prix et le temps de parcours ?

2. Est-ce que le temps de parcours est le même pour tous les types de train?
3. Quelles sont les caractéristiques distinctives des types de train?
4. Quelles sont les principales différences entre les tarifs?

À l'exception de `prix` et de `duree`, toutes les variables explicatives sont catégorielles. La variable `jour` prend des valeurs entre 1 et 7; s'en souvenir pour éviter les mauvaises surprises ultérieures. En analysant le nombre de trains dans les catégories, on remarque qu'il y a autant de billets de type `REXPRESS` que le nombre de billets au tarif `AdultoIda`. On peut faire le décompte par catégorie avec un tableau de contingence, qui compte le nombre respectif dans chaque sous-catégorie. Dans la base de données Renfe, tous les billets pour les `RegioExpress` sont vendus au tarif `AdultoIda` en classe `Turista`. Le nombre de billets est minime, à peine 397 sur 10000. Cela suggère une nouvelle question : pourquoi ces trains sont-ils si peu populaires?

On remarque également que seulement 17 temps de parcours sont affichés sur les billets. On peut donc penser que la durée affichée sur le billet (en minutes) est le temps de trajet annoncé. La majeure partie (15 sur 17) des temps de parcours sont sous la barre des 3h15, hormis deux qui dépassent les 9h! Selon Google Maps, les deux villes sont distantes de 615km par la route, 500km à vol d'oiseau. Cela implique que, vraisemblablement, certains trains dépassent les 200km/h, tandis que d'autres vont plutôt à 70km/h. Quels sont ces trains plus lents? La variable `type` codifie probablement ce fait, et permet de voir que ce sont les trains `RegioExpress` qui sont dans cette catégorie.

Aller de Madrid à Barcelone à l'aide d'un train régulier prend 18 minutes de plus. Avec plus de 9h de trajet, pas étonnant donc que ces billets soient peu courus. Encore plus frappant, on note que le prix des billets est fixe : 43.25 euros peu importe que le trajet soit aller ou retour. C'est probablement la trouvaille la plus importante jusqu'à maintenant, car les billets de train de type `RegioExpress` ne forment pas un échantillon : il n'y a aucune variabilité! On aurait également pu découvrir cette anomalie en traçant une boîte à moustaches du prix en fonction du type de train.

```
ggplot(data = renfe,
       mapping = aes(x = type, y = prix, col = dest)) +
  geom_boxplot() +
  labs(y = "prix (en euros)",
       x = "type de train",
       color = "destination") +
  theme(legend.position = "bottom")
```

On pourrait soupçonner que les trains étiquetés AVE sont plus rapides, sachant que c'est l'acronyme de *Alta Velocidad Española*, littéralement haute vitesse espagnole. Qu'en est-il des distinctions entre les deux types de trains étiquetés AVE? Selon le site de la SNCF, les trains AVE-TGV sont des partenariats entre la Renfe et la SNCF et effectuent des liaisons entre la France et l'Espagne.

Les prix sont beaucoup plus élevés, en moyenne plus de deux fois plus que les trains régionaux. Les

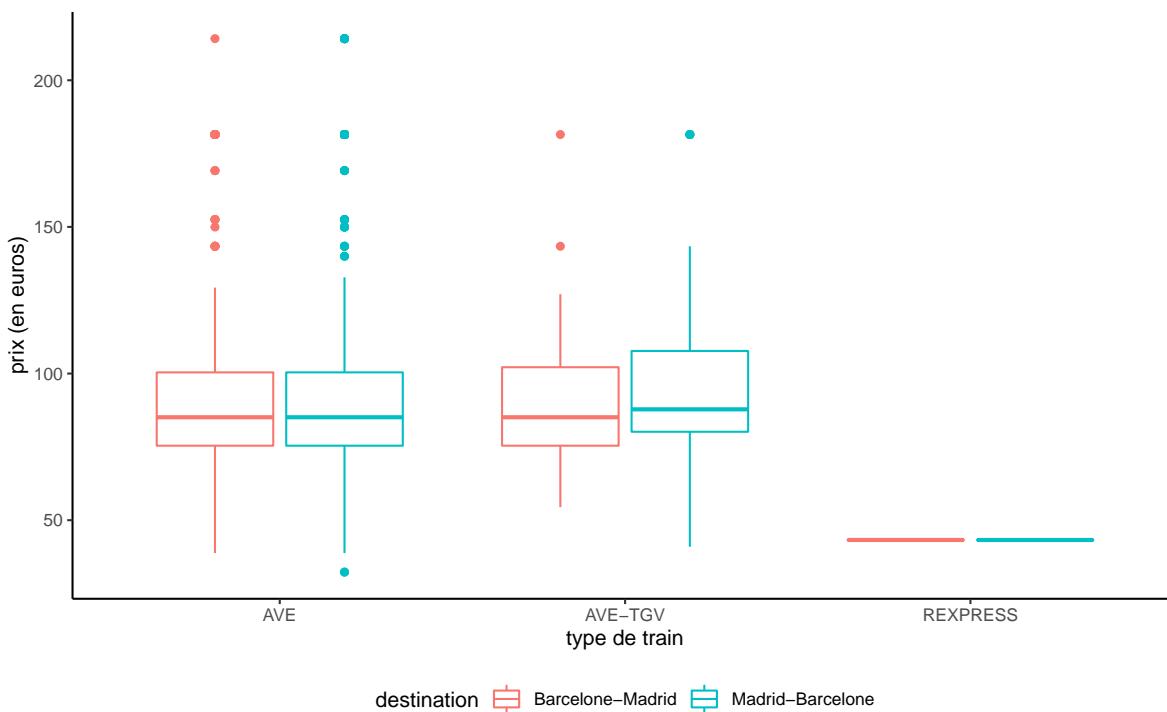


FIGURE 2.6 – Boîte à moustaches du prix de billets de train de Renfe en fonction de la destination et du type de train.

écart de prix importants (l'écart type est de 20 euros) indique qu'il y a peut-être d'autres sources d'hétérogénéité, mais on pourrait soupçonner que la Renfe pratique la tarification dynamique. Il y a un seul temps de parcours prévu pour les trains AVE-TGV. On ne note pas de différence de prix notable selon la direction ou le type de train grande vitesse, mais peut-être que les tarifs ou la classe disponibles diffèrent selon que le train ou non est en partenariat avec la compagnie française.

On n'a pas encore considéré le tarif et la classe des billets, hormis pour les trains RegioExpress. On voit dans la Figure 2.8 une forte différence dans l'hétérogénéité des prix selon le tarif; le tarif Promo prend plusieurs valeurs distinctes, tandis que les tarifs AdultoIda et Flexible semblent ne prendre que quelques valeurs. La première classe (Preferente) est plus chère et il y a moins d'observations dans ce groupe. La classe Turista est la classe la moins dispendieuse et la plus populaire. TuristaPlus offre plus de confort, tandis que TuristaSolo permet d'obtenir un siège individuel.

Côté tarif, Promo et PromoPlus permettent d'obtenir des rabais pouvant aller jusqu'à respectivement 70% et 65%. Les annulations et changements ne sont pas possibles avec Promo, mais disponibles avec PromoPlus moyennant une pénalité équivalente à 30-20% du prix du billet. Le tarif Flexible est disponible au même prix que les billets réguliers, avec des bénéfices additionnels.

```
renfe |>
  subset(tarif != "AdultoIda") |>
  ggplot(aes(y = prix,
              x = classe,
              col = tarif)) +
  geom_boxplot() +
  labs(y = "prix (en euros)",
       x = "classe",
       color = "tarif") +
  theme(legend.position = "bottom")
```

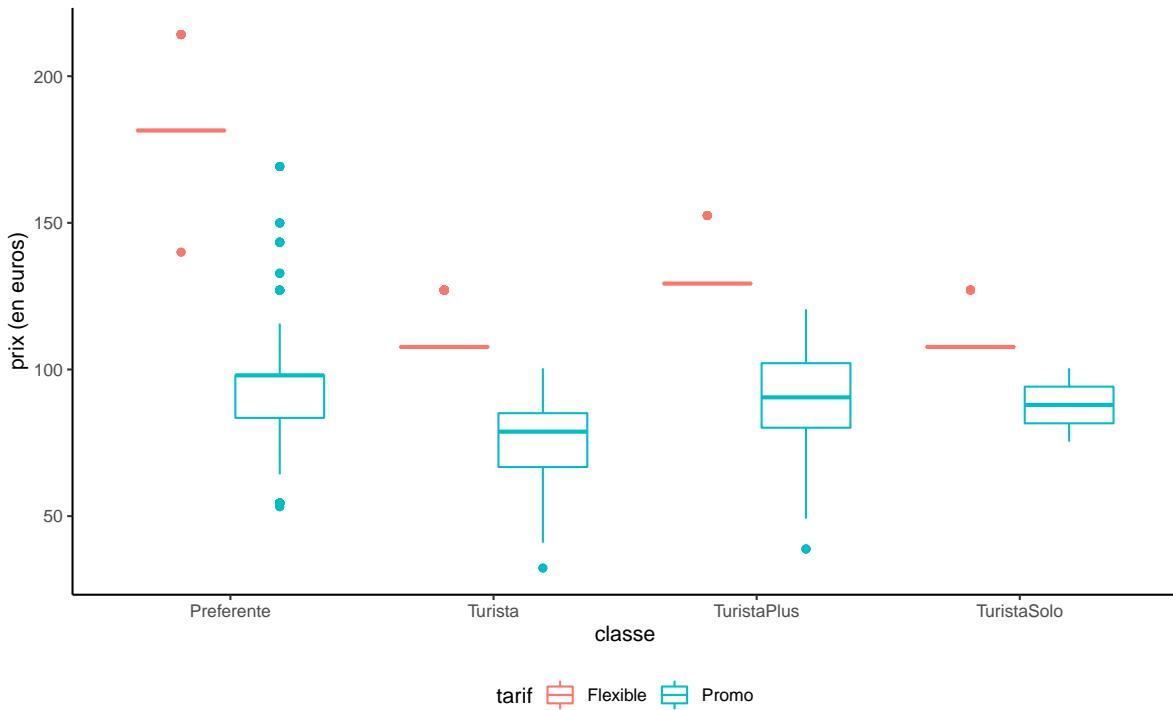
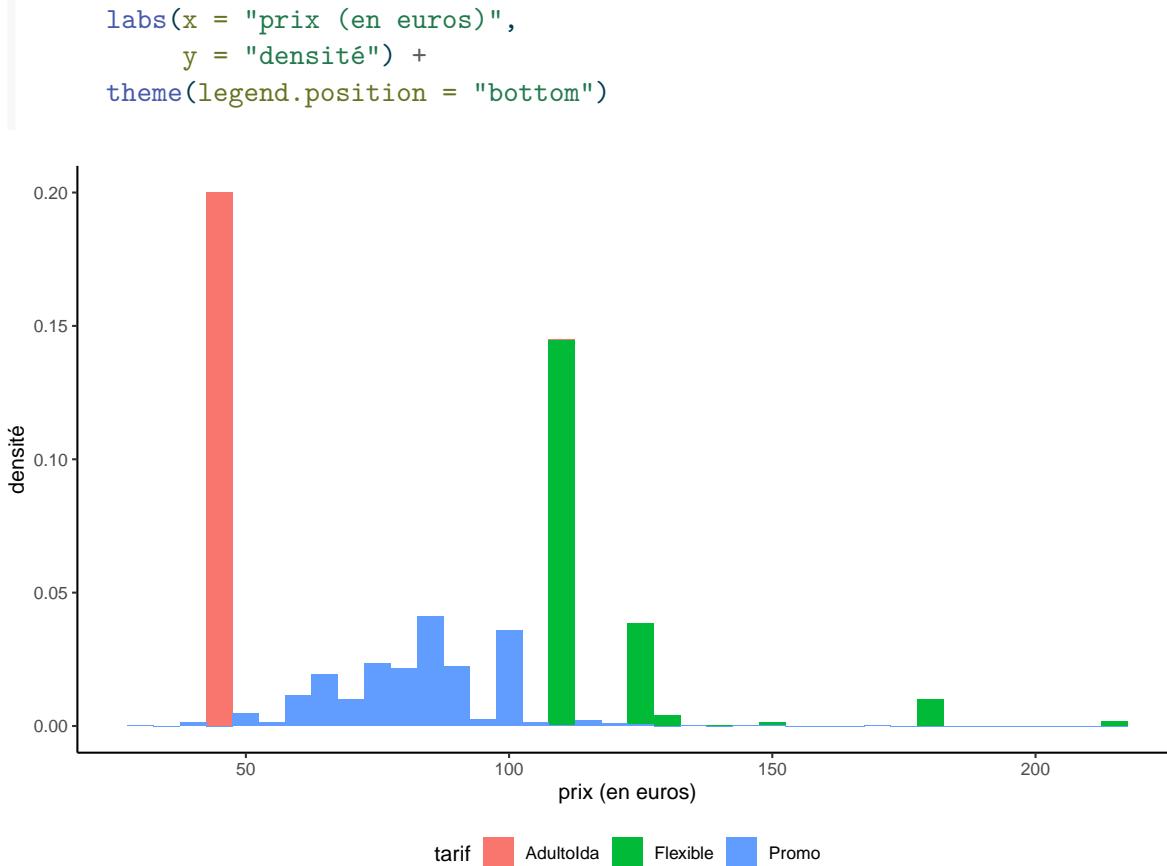


FIGURE 2.7 – Boîte à moustaches du prix en fonction du tarif et de la classe de billets de trains à haute vitesse de la Renfe.

```
ggplot(data = renfe,
       mapping = aes(x = prix,
                     y = after_stat(density),
                     fill = tarif)) +
  geom_histogram(binwidth = 5) +
```



```

renfe |>
  dplyr::subset(tarif == "Flexible") |>
  dplyr::count(prix, classe)

```

On note que la répartition des prix pour les billets de classe Flexible est inhabituelle. Notre boîte à moustaches est écrasée et l'écart interquartile semble nul, même si quelques valeurs inexplicées sont aussi présentes. L'écrasante majorité des billets Flexibles sont en classe Turista, donc ça pourrait être dû à un (trop) faible nombre de billets dans chaque catégorie. On peut rejeter cette hypothèse en calculant le nombre de trains au tarif Flexible pour les différents types de billets, comme dans le Tableau 2.1. Ni la durée, ni le type de train, ni la destination n'expliquent pas pourquoi le prix de certains billets Flexibles est plus faible ou élevés. Le prix des billets Promo est plus faible, et les billets au tarif Preferente (la première classe) sont plus élevés.

On peut résumer notre brève analyse exploratoire :

TABLEAU 2.1 – Nombre de billets au tarif Flexible selon le prix de vente.

prix	classe	n
108	Turista	1050
108	TuristaSolo	67
127	Turista	285
127	TuristaSolo	9
129	TuristaPlus	31
140	Preferente	2
152	TuristaPlus	10
182	Preferente	78
214	Preferente	12

- plus de 91% des trains sont des trains à grande vitesse AVE.
- le temps de trajet dépend du type de train : les trains à grande vitesse mettent 3h20 au maximum pour relier Madrid et Barcelone.
- les temps de trajets sont ceux annoncés (variable discrète avec 17 valeurs uniques, dont 13 pour les trains AVE)
- le prix de trains RegioExpress est fixe (43.25€) ; tous ces billets sont dans la classe Turista et au tarif Adulato Ida. 57% de ces trains vont de Barcelone à Madrid. La durée du trajet pour les RegioExpress est de 9h22 de Barcelona à Madrid, 18 minutes de plus que dans l'autre direction.
- les billets en classe Preferente sont plus chers et moins fréquents. La classe Turista est la classe la moins dispendieuse et la plus populaire. TuristaPlus offre plus de confort, tandis que TuristaSolo permet d'obtenir un siège individuel.
- selon le site web de la Renfe, les billets au tarif Flexible « viennent avec des offres additionnelles qui permettent aux passagers d'échanger leurs billets ou annuler s'ils manquent leurs trains. »; en contrepartie, ces billets sont plus chers et leur tarif est fixe sauf une poignée de billets dont le prix reste inexpliqué.
- la distribution des prix des billets de TGV au tarif Promo est plus ou moins symétrique, tandis que les billets au tarif Flexible apparaissent tronqués à gauche (le prix minimum pour ces billets est 107.7€ dans l'échantillon).
- la Renfe pratique la tarification dynamique pour les billets au tarif promotionnel Promo : ces derniers peuvent être jusqu'à 70% moins chers que les billets à prix régulier lorsqu'achetés via l'agence officielle ou le site de Renfe. Ces billets ne peuvent être ni remboursés, ni échangés.
- il n'y a pas d'indication à effet de quoi les prix varient selon la direction du trajet.

### 2.4.1 Commentaire sur les graphiques

Si vous incluez un graphique (ou un tableau), il est important d'ajouter une légende qui décrit le graphique et le résume, les noms de variables (avec les unités) sur les axes, mais aussi de soigner le rendu et le formatage pour obtenir un produit fini propre, lisible et cohérent : en particulier, votre description devrait coïncider avec le rendu. Votre graphique raconte une histoire, aussi prenez-soin que cette dernière soit nécessaire et attrayante.

#### i En résumé

- Une base de données est normalement constituée de plusieurs variables (colonnes) ; les lignes représentent les différentes observations.
- On classe grossièrement les variables en variables catégorielles (nominales, ordinaires ou binaires) et numériques (continues, entières).
- L'analyse exploratoire est une procédure dynamique qui sert à mieux comprendre les données pour proposer des modèles adéquats. Elle consiste à poser des questions et à raffiner les conclusions à l'aide de tableaux résumés et de graphique.
- Le nettoyage et la validation des données est la première étape de toute analyse. On peut formuler nos attentes et l'utiliser pour vérifier de la conformité de notre base de données (en utilisant des outils comme le paquet `pointblank`).
- Une bonne compréhension des données est nécessaire avant d'envisager la modélisation.
- Il faut déclarer correctement les variables explicatives catégorielles (souvent encodées avec des entiers).
- L'utilisation de graphiques est privilégiée par rapport aux tableaux : une image vaut mille mots.
- Les graphiques usuels employés sont l'histogramme et la boîte à moustaches (données numériques) et le diagramme à bande (données catégorielles). L'utilisation de diagramme circulaires est à proscrire.
- On peut utiliser la couleur, la forme ou la taille comme dimensions additionnelles.
- Toujours utiliser une palette de couleurs pour daltoniens.
- Un graphique devrait toujours inclure une légende globale décrivant la représentation (en plus d'être discuté dans le texte), une légende pour les axes et des unités. Les caractères doivent être lisibles (suffisamment grands).
- Une analyse exploratoire inclut un résumé des principales trouvailles.



## Chapitre 3

# Réduction de la dimension

### 3.1 Introduction

Ce chapitre traite de réduction de la dimensionnalité d'un problème d'analyse multidimensionnelle. On dispose de  $p$  variables  $X_1, \dots, X_p$  : comment résumer cet ensemble avec moins de variables (disons  $k$ ) tout en conservant le plus de variabilité possible ? Nous couvrons deux méthodes dans ce chapitre : la première, intitulée **analyse en composantes principales**, cherche à **réduire le nombre de variables explicatives** tout en préservant le plus possible de variabilité exprimée et en créant de nouvelles variables explicatives qui ne sont pas corrélées les unes avec les autres.

La deuxième, appelée analyse factorielle exploratoire, cherche à expliquer la structure de corrélation entre les  $p$  variables à l'aide d'un nombre restreint de facteurs. Elle répond aux questions suivantes :

- Y a-t-il des groupements de variables ?
- Est-ce que les variables faisant partie d'un groupement semblent mesurer certains aspects d'un facteur commun (non observé) ?

De tels groupements peuvent être détectés si plusieurs variables sont très corrélées entre elles. Une analyse factorielle cherchera à identifier automatiquement ces groupes de variables.

Les facteurs sont des variables latentes qui mesurent des constructions. Par exemple, l'habileté quantitative, habileté sociale, importance accordée à la qualité du service, importance accordée à la loyauté, habileté de leader, etc.

L'analyse factorielle est aussi une méthode de réduction du nombre de variables. En effet, une fois qu'on a identifié les facteurs, on peut remplacer les variables individuelles par un résumé pour chaque facteur (qui est souvent la moyenne des variables qui font partie du facteur).

### 3.2 Coefficient de corrélation linéaire

On veut examiner la relation entre deux variables  $X_j$  et  $X_k$  et on dispose de  $n$  couples d'observations, où  $x_{i,j}$  (respectivement  $x_{i,k}$ ) est la valeur de la variable  $X_j$  ( $X_k$ ) pour la  $i$ e observation.

Le coefficient de corrélation linéaire entre  $X_j$  et  $X_k$ , que l'on note  $r_{j,k}$ , cherche à mesurer la force de la relation linéaire entre deux variables, c'est-à-dire à quantifier à quel point les observations sont alignées autour d'une droite. Le coefficient de corrélation est

$$r_{j,k} = \frac{\widehat{\text{Co}}(X_j, X_k)}{\{\widehat{\text{Va}}(X_j)\widehat{\text{Va}}(X_k)\}^{1/2}}$$

Les propriétés les plus importantes du coefficient de corrélation linéaire  $r$  sont les suivantes :

- 1)  $-1 \leq r \leq 1$ ;
- 2)  $r = 1$  (respectivement  $r = -1$ ) si et seulement si les  $n$  observations sont exactement alignées sur une droite de pente positive (négative). C'est-à-dire, s'il existe deux constantes  $a$  et  $b > 0$  ( $b < 0$ ) telles que  $y_i = a + bx_i$  pour tout  $i = 1, \dots, n$ .

Règle générale,

- Le signe de la corrélation détermine l'orientation de la pente (négative ou positive)
- Plus la corrélation est près de 1 en valeur absolue, plus les points auront tendance à être alignés autour d'une droite.
- Lorsque la corrélation est presque nulle, les points n'auront pas tendance à être alignés autour d'une droite. Il est très important de noter que cela n'implique pas qu'il n'y a pas de relation entre les deux variables. Cela implique seulement qu'il n'y a pas de **relation linéaire** entre les deux variables. La Figure 3.1 montre bien ce point : ces jeux de données ont la même corrélation linéaire (quasi-nulle), mais ne sont pas clairement pas indépendantes puisqu'elles permettent de dessiner un dinosaure ou une étoile.

La matrice de corrélation entre  $X_1, \dots, X_p$ , dont l'entrée  $(i, j)$  contient la corrélation entre  $X_i$  et  $X_j$ , est une matrice symétrique dont les éléments de la diagonale sont égaux à 1. À mesure que le nombre de variables augmente, le nombre de corrélations à estimer augmente : puisque la matrice est  $p \times p$ , ce nombre augmente comme le carré du nombre de variables explicatives. L'estimation ne sera pas fiable à moins que  $n \gg p$ .

### 3.3 Présentation des données

Le questionnaire suivant porte sur une étude dans un magasin. Pour les besoins d'une enquête, on a demandé à 200 consommateurs adultes de répondre aux questions suivantes par rapport à un certain type de magasin sur une échelle de 1 à 5, où

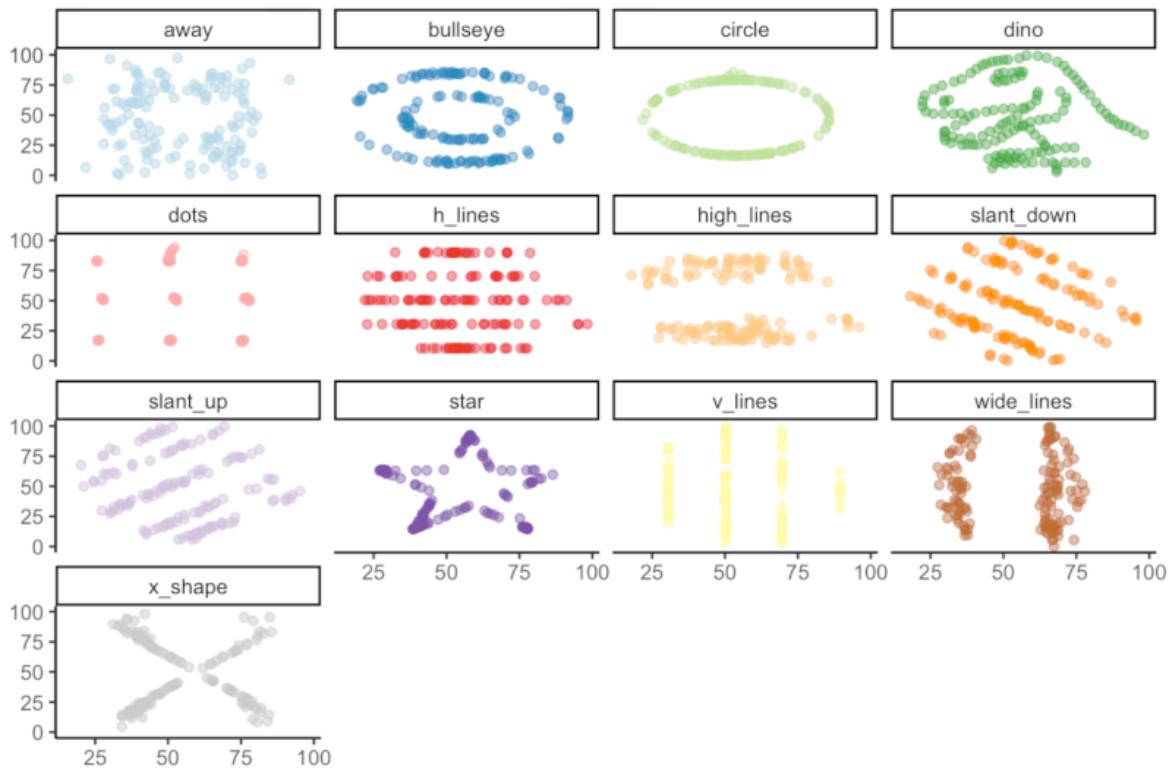


FIGURE 3.1 – Datasaurus (Alberto Cairo) : une douzaine de jeux de données qui ont les mêmes statistiques descriptives (à deux décimales près) et une faible corrélation, mais qui sont visuellement distincts.

1. pas important
2. peu important
3. moyennement important
4. assez important
5. très important

Pour vous, à quel point est-ce important...

1. que le magasin offre de bons prix tous les jours?
2. que le magasin accepte les cartes de crédit majeures (Visa, Mastercard)?
3. que le magasin offre des produits de qualité?
4. que les vendeurs connaissent bien les produits?
5. qu'il y ait des ventes spéciales régulièrement?
6. que les marques connues soient disponibles?
7. que le magasin ait sa propre carte de crédit?

TABLEAU 3.1 – Matrice de corrélation de factor.

	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12
x1		-0.08	-0.14	-0.07	0.38	-0.01	-0.10	-0.13	-0.03	-0.11	-0.12	-0.01
x2			0.04	-0.02	-0.08	0.06	0.50	0.01	-0.01	0.43	-0.12	0.07
x3				0.10	-0.06	0.39	0.00	0.05	0.47	0.08	0.13	0.46
x4					-0.05	0.06	0.08	0.57	0.01	0.09	0.50	0.09
x5						-0.04	-0.04	-0.02	0.03	-0.07	-0.06	-0.07
x6							0.07	0.04	0.32	0.07	-0.04	0.32
x7								0.09	-0.02	0.51	-0.03	0.02
x8									-0.03	0.16	0.55	0.04
x9										0.01	0.02	0.39
x10											0.01	0.02
x11												0.05
x12												

8. que le service soit rapide?
9. qu'il y ait une vaste sélection de produits?
10. que le magasin accepte le paiement par carte de débit?
11. que le personnel soit courtois?
12. que le magasin ait en stock les produits annoncés?

Les statistiques descriptives ainsi que la matrice des corrélations sont obtenues en exécutant les lignes suivantes :

```
data(factor, package = "hecmulti")
# Matrice de corrélation
cor(factor)
# Statistiques descriptives
summary(factor)
```

On voit dans la Figure 3.2 que quelques groupes de variables sont corrélés entre eux. On peut également regrouper certaines questions sous des thèmes manuellement : le but de l'analyse factorielle sera d'automatiser ce regroupement.

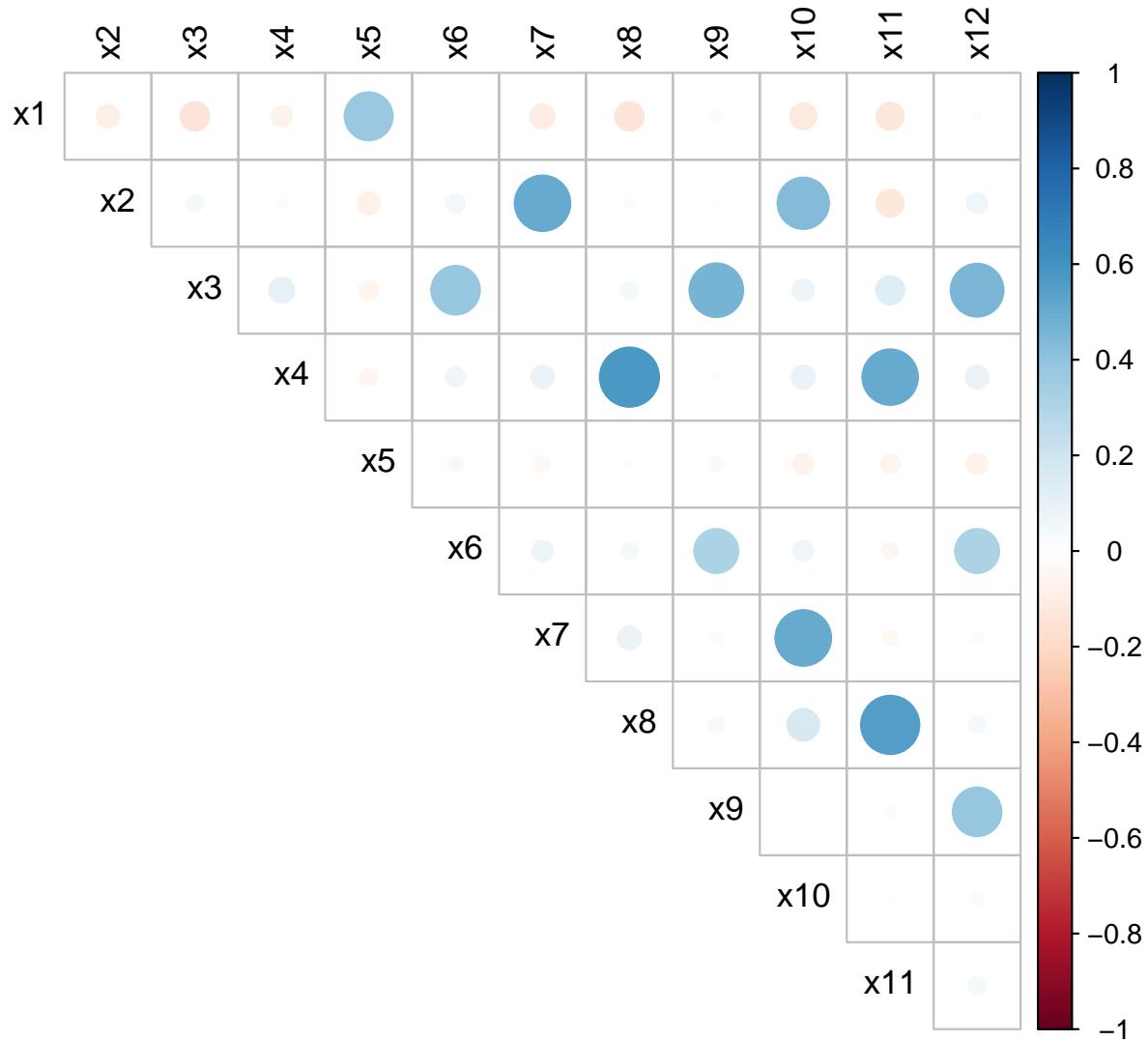


FIGURE 3.2 – Corrélogramme de la base de données factor.

### 3.4 Analyse en composantes principales

Le but de l'analyse en composantes principales est de réduire le nombre de variables explicatives. En partant de  $p$  variables  $X_1, \dots, X_p$ , on forme de nouvelles variables qui sont des combinaisons

TABLEAU 3.2 – Statistiques descriptives des 12 variables du jeu de données factor.

moyenne	écart-type	min	max
2.26	1.13	1	5
2.51	1.24	1	5
3.00	1.19	1	5
2.91	1.33	1	5
3.55	1.17	1	5
2.14	1.14	1	5
1.82	1.06	1	5
2.92	1.32	1	5
3.04	1.12	1	5
2.59	1.32	1	5
2.98	1.33	1	5
3.45	1.16	1	5

linéaires des variables originales,

$$C_j = \underbrace{w_{j1}X_1 + w_{j2}X_2 + \cdots + w_{jp}X_p}_{\text{somme de poids fois variables explicatives}}, \quad (j = 1, \dots, p),$$

$$1 = \underbrace{w_{j1}^2 + \cdots + w_{jp}^2}_{\text{poids standardisés}}$$

de telle sorte que

- La première variable formée,  $C_1$ , appelée première composante principale, possède la variance maximale parmi toutes les combinaisons linéaires sous la contrainte  $w_{11}^2 + \cdots + w_{1p}^2 = 1$ .<sup>1</sup>
- Pour  $j = 2, \dots, p$ , la  $j$ e composante principale  $C_j$  possède la variance maximale parmi toutes les combinaisons linéaires qui sont non corrélées avec  $C_1, \dots, C_{j-1}$  sous la contrainte  $w_{j1}^2 + \cdots + w_{jp}^2 = 1$ .

Ainsi, les composantes principales forment un ensemble de variables non corrélées entre elles, qui récupèrent en ordre décroissant le plus possible de la variance des variables originales. La somme des variances des  $p$  composantes principales est égale à la somme des variances des  $p$  variables originales.

Mathématiquement, les composantes principales correspondent aux vecteurs propres de la matrice

---

1. Les contraintes sur les poids sont nécessaires afin de standardiser le problème car il serait possible d'avoir des variances infinies sinon.

de covariance, mais on peut également utiliser la matrice de corrélation<sup>2</sup>. L'avantage de la matrice de corrélation (ou de la standardisation des variables) est que l'unité de mesure n'impacte pas le résultat; autrement, un poids plus important est attribué aux variables qui ont la plus forte hétérogénéité.

Si on conserve toutes les composantes principales, cela revient à changer le système de coordonnées dans lequel sont exprimées nos observations en effectuant une rotation : avec deux variables, on trouve la direction dans le système 2D dans lequel l'étendue est la plus grande. Si une simple rotation peut sembler inutile, la méthode est fort utile en haute dimension. On espère en général qu'un petit nombre de composantes principales réussira à expliquer la plus grande partie de la variance totale.

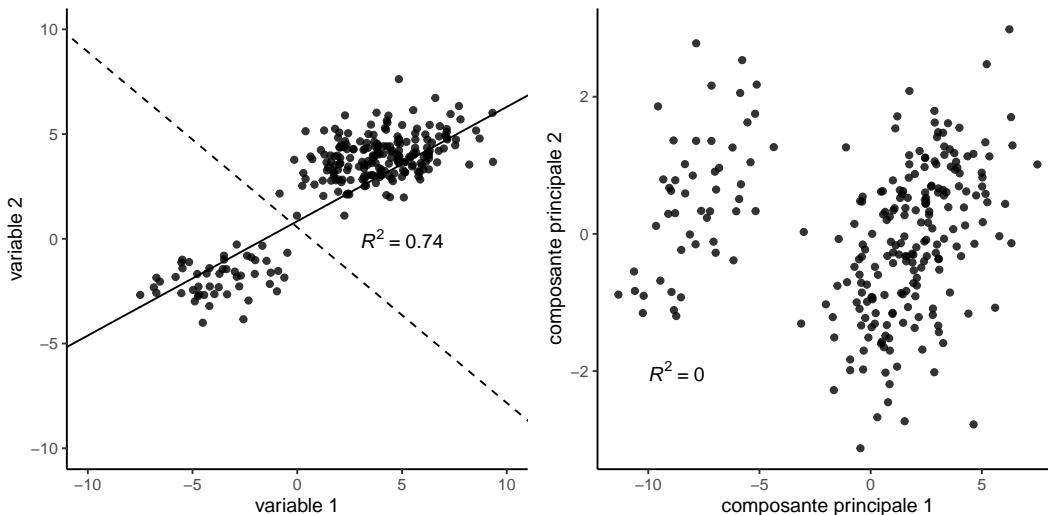


FIGURE 3.3 – Nuage de points avant (gauche) et après (droite) analyse en composantes principales. Les directions des composantes principales (lignes pleines et traitillées), qui forment un angle droit, sont ajoutées au nuage de points à gauche. On peut constater que la corrélation entre les deux composantes principales est nulle.

La Figure 3.3 démontre cette décomposition sur des données bidimensionnelles simulées. La variance des données dans le premier panneau est 13.51 pour l'axe des abscisses et 6.43 pour l'axe des ordonnées avec une corrélation de 0.86, à comparer avec des variances de 18.65 et 1.21 et une corrélation nulle entre les deux composantes principales.

Dans une analyse en composantes principales, on conservera un nombre  $k < p$  de variables explicatives pour résumer les données. Ce outil est utilisé à des fins exploratoires, puisqu'on n'implique pas de variable réponse dans le modèle. L'analyse en composantes principales est

---

2. La fonction `princomp` peut directement utiliser la base de données numériques, ou la matrice de covariance. Dans le premier cas, on peut spécifier que l'on veut la décomposition de la matrice de corrélation à l'aide de l'argument, `cor`

utilisé pour réduire la dimension afin de faire de la classification, de l'analyse de regroupements et aussi réduire les coûts associés à ces méthodes en projetant les données dans un sous-espace de dimension plus faible.

En **R**, on effectue l'analyse en composantes principales avec la fonction `princomp`<sup>3</sup>.

La sortie contient

- l'écart-type de chaque composante, `acp$sdev`
- les poids  $w_{ij}$ , appelés chargements (*loadings*), qui donnent la correspondance entre le système de coordonnées des composantes principales et celui des variables  $X$  originales.
- les nouvelles coordonnées dans l'espace des composantes principales, `acp$scores`

On peut représenter les données à l'aide d'un bigramme : c'est une nuage de points de chaque observations dans l'espace des deux premières composantes principales. Si on couple cela avec les directions offertes par les chargements pour chacune des variables explicatives  $X_1, \dots, X_p$ , il en ressort que certaines variables augmentent/diminuent de pair. Ainsi, on voit dans la Figure 3.4 que les variables  $x_3, x_6, x_9$  et  $x_{12}$  tendent dans la même direction, comme  $x_4, x_8$  et  $x_{11}$ . On reviendra sur ce point dans une section subséquente.

Une fois qu'on a choisi le nombre de composantes, on pourrait ne conserver que les  $k$  premières colonnes de la matrice des composantes principales `acp$scores` pour faire les graphiques ou pour approximer la matrice de covariance. Il faut garder en tête qu'il faudra néanmoins collecter les mêmes questions pour recréer les composantes principales avec de nouvelles observations, ce qui est peu commode si ce qu'on veut est réduire le coût de la collecte.

```
# Analyse en composantes principales
# de la matrice de corrélation
acp <- princomp(factor, cor = TRUE)
loadings(acp) # chargements
biplot(acp) # bigramme
```

On peut étudier la sortie pour vérifier les propriétés de notre décomposition. Le Tableau 3.3 montre la variance de chaque composante principale. Si on additionne l'ensemble des variances (sans arrondir), on obtient une variance cumulative des 12 composantes principales, 12, soit le même que le nombre de variables explicatives puisque les variables standardisées ont variance unitaire. Si on calcule la matrice de corrélation, `cor(acp$scores)`, on remarquera que la corrélation est nulle entre les variables.

---

3. On pourrait aussi utiliser `eigen(cor(factor))` pour extraire directement la décomposition en valeurs propres/vecteurs propres, mais on n'aurait pas accès aux méthodes pour la visualisation.

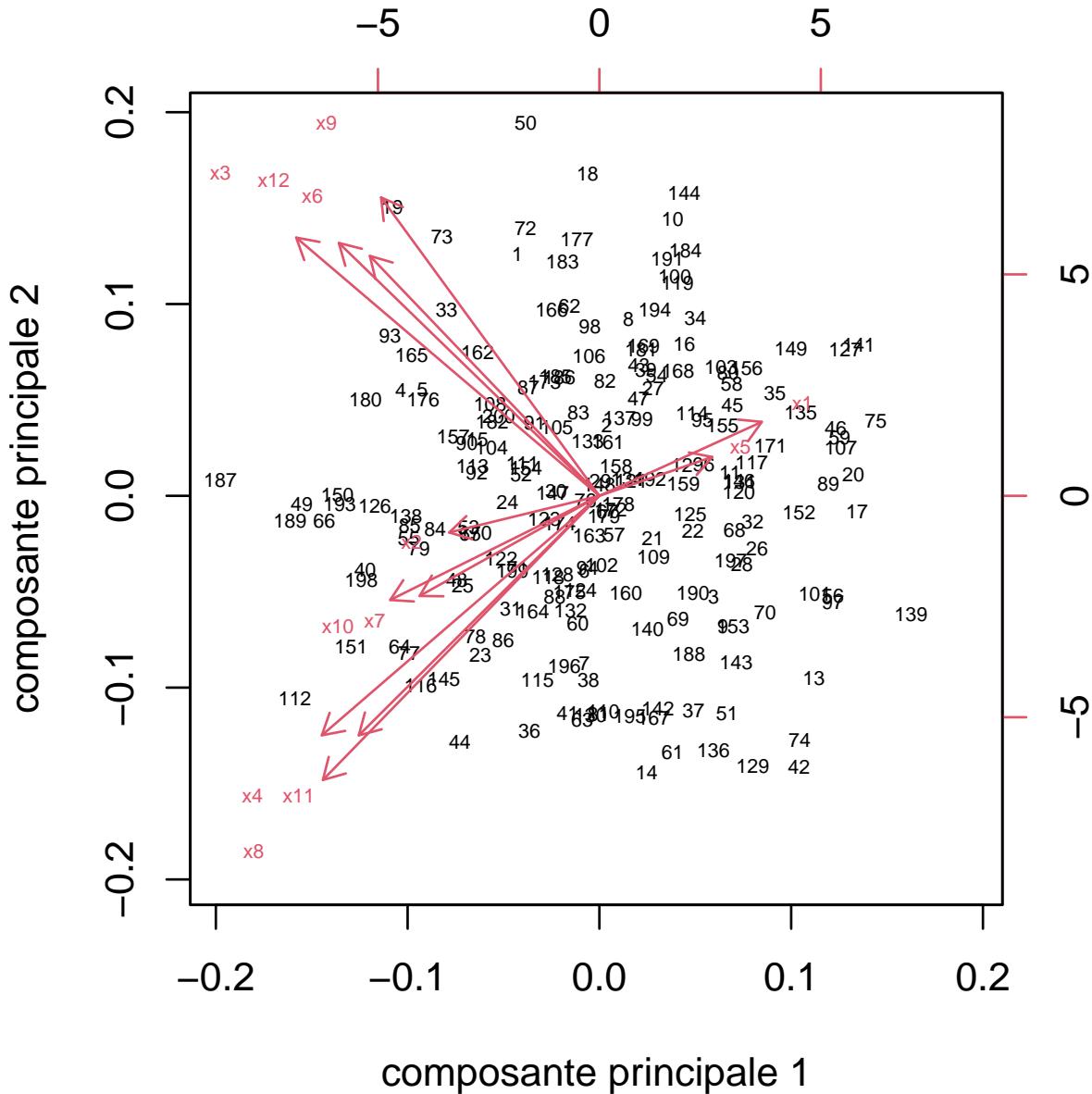


FIGURE 3.4 – Bigramme : nuage de point des coordonnées des deux premières composantes principales et direction selon chargements des variables explicatives originales.

### 3.4.1 Choix du nombre de composantes principales

Si on désire réduire la dimension, il nous faudra choisir  $k \leq p$  variables. Cette section traite du choix du nombre de variables explicatives à retenir. Idéalement, ce nombre devrait être beaucoup plus

TABLEAU 3.3 – Variance des composantes principales

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
2.43	2.00	1.94	1.30	0.74	0.69	0.57	0.54	0.51	0.47	0.46	0.36

petit que le nombre original de variables.

Une première approche est de regarder le pourcentage de la variance totale expliquée. Puisque les composantes principales sont ordonnées en ordre décroissant de variance, on peut étudier la variance cumulative des  $k$  premières composantes et choisir un nombre qui explique le plus possible. Si l'ajout d'une variable augmente peu la variabilité totale expliquée par l'ensemble, alors cette variable est probablement superflue. On pourrait choisir un nombre de composantes pour expliquer un pourcentage prédéfini de la variance totale, disons 70%. Deux autres critères couramment employés sont :

- **critère du coude de Cattell** : ce critère consiste à sélectionner un nombre de composantes dans le diagramme d'éboulis (*screeplot*), un graphique des variances des composantes principales<sup>4</sup>. Habituellement, il y a une décroissance rapide de la variance suivie d'un plateau : on prendra le nombre de composantes qui correspond au  $k$  juste avant l'apparition du plateau (le début du coude, où il a stabilisation apparente). C'est un critère très subjectif, puisqu'il y a souvent plusieurs plateaux et que la variance peut décroître très lentement. On peut utiliser la fonction *screeplot* pour obtenir le diagramme d'éboulis mais il est facile de le créer manuellement et le résultat est esthétiquement plus réussi.
- **critère de Kaiser** : un critère basé sur les valeurs propres de la matrice de corrélation. Le nombre de facteurs choisis est le nombre de composantes principales dont la variance est supérieures à 1. L'idée est de garder seulement les facteurs qui expliquent plus de variance qu'une variable individuelle.

Si on utilise le critère de Kaiser avec les données *factor*, on conservera 4 composantes principales qui expliqueront 63.9 pourcent de la variance totale des variables originales - voir le Tableau 3.3. Le diagramme d'éboulis de la Figure 3.5, qui peut être produit avec la fonction *hecmulti::eboulis(eigen(cor(factor)))* suggère quant à lui cinq composantes.

```
hecmulti::eboulis(eigen(cor(factor)))
```

Une fois qu'on a déterminé le nombre de facteurs, on peut extraire les nouvelles variables explicatives à partir de l'analyse en composantes principales. Les colonnes sont stockées dans *acp\$score* et il suffit de conserver les premières colonnes.

---

4. Soit les valeurs propres de la matrice de covariance ou corrélation

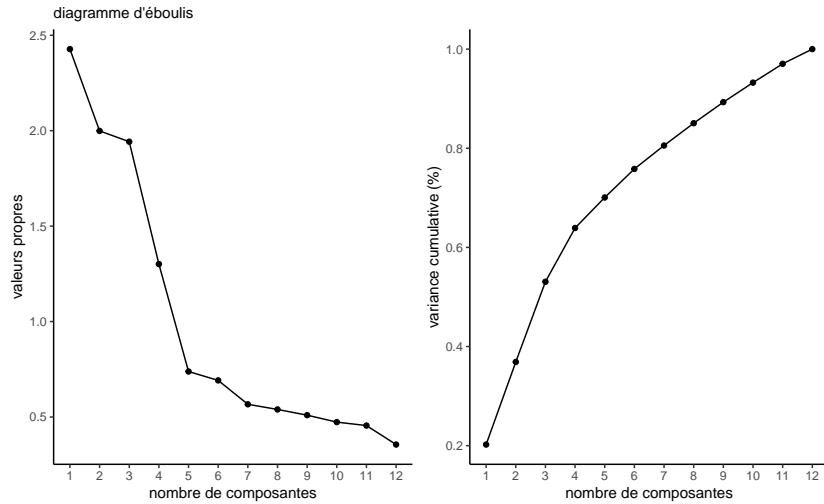


FIGURE 3.5 – Diagramme d'éboulis (gauche) représentant la variance des composantes principales (en ordonnée) en fonction du nombre composantes principales (en abscisse). Variance cumulative en fonction du nombre de composantes principales (droite).

### 3.4.2 Formulation mathématique

Ce complément d'information est optionnel.

Mathématiquement, le problème de l'analyse en composantes principales revient à calculer la décomposition en valeurs propres et vecteurs propres de la matrice de covariance  $\text{Co}(\mathbf{X}) = \boldsymbol{\Sigma}$ . On peut écrire

$$\boldsymbol{\Sigma} = \mathbf{Q} \boldsymbol{\Lambda} \mathbf{Q}^\top$$

où  $\boldsymbol{\Lambda} = \text{diag}\{\lambda_1, \dots, \lambda_p\}$  est une matrice diagonale contenant les valeurs propres en ordre décroissant ( $\lambda_1 \geq \dots \geq \lambda_p > 0$ ) et  $\mathbf{Q}$  est une matrice carrée  $p \times p$  orthogonale contenant les vecteurs propres. La meilleure approximation de rang  $k \leq p$  de  $\boldsymbol{\Sigma}$  est obtenue en spécifiant

$$\tilde{\boldsymbol{\Sigma}}_k = \sum_{j=1}^k \lambda_j \mathbf{q}_j \mathbf{q}_j^\top,$$

une combinaison des vecteurs propres  $\mathbf{q}_1, \dots, \mathbf{q}_k \in \mathbb{R}^p$  non corrélés.

#### i En résumé

- La corrélation mesure la force de la dépendance linéaire entre deux variables : plus elle est élevée, plus les points s'alignent.

- Si le nombre de variables explicatives  $p$  est conséquent par rapport au nombre d'observations  $n$ , on a peu d'information disponible pour estimer de manière fiable les corrélations.
- Une analyse en composante principales fait une décomposition en valeurs propres/vecteurs propres de la matrice de covariance ou de corrélation.
  - Ces nouvelles variables sont orthogonales (corrélation nulle) entre elles.
  - Les composantes principales sont ordonnées en ordre décroissant de variance : si on ne conserve que  $k < p$  de variables, on maximise la variance expliquée.
  - Le choix du nombre de variables est basé sur des règles du pouce : le critère des valeurs propres de Kaiser suggère de prendre autant de composantes principales que de variances supérieures à 1.
  - Un bigramme permet de représenter graphiquement les directions des variables en fonction des deux premières composantes principales.

### 3.5 Analyse factorielle exploratoire

Si le bigramme a permis de faire ressortir quelques orientations communes, on aimerait aller plus loin dans notre exploration. On considère encore une fois la matrice de covariance associée avec  $p$  variables explicatives  $X_1, \dots, X_p$  : le modèle d'analyse factorielle cherche à décrire cette dernière en fonction d'un plus petit nombre de paramètres.

Conceptuellement, le modèle d'analyse factorielle suppose qu'on peut regrouper les variables explicatives numériques (parfois avec quelques variables binaires) à l'aide de concepts communs appelés facteurs. Certaines variables explicatives devraient donc idéalement être fortement corrélées entre elles. Le choix des variables est dicté par le bon sens : on inclut dans le modèle des variables qui peuvent logiquement être associées, par exemple des items de questionnaires excluant les données sociodémographiques.

Le modèle d'analyse factorielle fait l'hypothèse que les variables dépendent linéairement d'un plus petit nombre de variables aléatoires,  $F_1, \dots, F_m$ , appelées facteurs communs. Cette relation n'est pas parfaite, aussi on inclut  $p$  termes d'aléas  $\varepsilon_1, \dots, \varepsilon_p$ , de moyenne zéro et de variance  $Va(\varepsilon_i) = \psi_i$  ( $i = 1, \dots, p$ ). À des fins d'identifiabilité, on suppose que les aléas ne sont pas corrélées aux facteurs  $F$  et entre elles et que les facteurs  $F_1, \dots, F_m$  ont une moyenne nulle et une variance unitaire, donc  $E(F_i) = 0$  et  $Va(F_i) = 1$  ( $i = 1, \dots, p$ ).

Le modèle d'analyse factorielle s'écrit

$$\mathbf{X} = \underbrace{\boldsymbol{\mu}}_{\text{moyenne}} + \underbrace{\Gamma \mathbf{F}}_{\text{combinaison linéaire de facteurs latents}} + \underbrace{\boldsymbol{\varepsilon}}_{\text{aléa}},$$

ou si on écrit le système ligne par ligne,

$$\begin{aligned} X_1 &= \mu_1 + \gamma_{11}F_1 + \gamma_{12}F_2 + \cdots + \gamma_{1m}F_m + \varepsilon_1 \\ X_2 &= \mu_2 + \gamma_{21}F_1 + \gamma_{22}F_2 + \cdots + \gamma_{2m}F_m + \varepsilon_2 \\ &\vdots \\ X_p &= \mu_p + \gamma_{p1}F_1 + \gamma_{p2}F_2 + \cdots + \gamma_{pm}F_m + \varepsilon_p, \end{aligned}$$

où  $\mu_i$  est l'espérance (moyenne théorique) de la variable aléatoire  $X_i$ ,  $\Gamma$  est une matrice  $p \times m$  avec éléments  $\gamma_{ij}$ , qui représentent le chargement (poids) de la variable  $X_i$  sur le facteur  $F_j$  ( $i = 1, \dots, p$ ;  $j = 1, \dots, m$ ).

Les espérances ( $\mu_i$ ), les chargements ( $\gamma_{ij}$ ) et les variances ( $\psi_i$ ) sont des quantités fixes, mais inconnues, tandis que les facteurs communs ( $F_i$ ) et les aléas ( $\varepsilon_i$ ) sont des variables aléatoires non observables.

Selon ce modèle, on obtient

$$\begin{aligned} \text{Va}(\mathbf{X}) &= \mathbf{\Gamma} \text{Va}(\mathbf{F}) \mathbf{\Gamma}^\top + \text{Va}(\boldsymbol{\varepsilon}) \\ &= \mathbf{\Gamma} \mathbf{\Gamma}^\top + \text{diag}(\boldsymbol{\psi}). \end{aligned}$$

Les éléments diagonaux de cette matrice sont  $\text{Va}(X_j) = \sum_{l=1}^k \gamma_{jl}^2 + \psi_j$  : on appelle **communalité** le terme  $h_j = \sum_{l=1}^k \gamma_{jl}^2$ , qui représente la proportion de variance totale de  $X_j$  due à la corrélation entre les facteurs. Le terme  $\psi_j$  est dénommé **unicité**.

Si les variables ont été préalablement standardisées de telle sorte que  $E(X_i) = 0$  et  $\text{Va}(X_i) = 1$  (ce qui revient à utiliser la matrice de corrélation des observations dans l'analyse), alors  $\text{Cor}(X_i, F_j) = \gamma_{ij}$ , c'est-à-dire, le chargement de la variable  $X_i$  sur le facteur  $F_j$  est le coefficient de corrélation entre les deux.

Sans aucune contrainte sur le modèle, la matrice de covariance de  $X_1, \dots, X_p$  possède  $p(p+1)/2$  paramètres, soit  $p$  variances et  $p(p-1)/2$  termes de corrélation. Avec le modèle d'analyse factorielle, on suppose que l'on peut décrire cette structure en utilisant seulement  $p(m+1) - m(m-1)/2$  paramètres<sup>5</sup>. Par exemple, avec  $p = 50$  variables explicatives et  $m = 6$  facteurs, on essaie de décrire la structure de covariance à l'aide de 350 paramètres au lieu de 1275.

Pour faire une analyse factorielle, la taille d'échantillon devrait être quand même conséquente : le nombre d'entrées dans la base de données est  $np$  et ce nombre représente la quantité d'unités (information) disponible pour estimer les covariances. Plusieurs références suggèrent d'avoir une taille d'échantillon entre cinq et 20 fois le fois le nombre de variables, ou bien un nombre minimal de 100 à 1000 observations. Des études de simulations suggèrent que la taille critique dépend

---

5. Soit  $p$  variances spécifiques et  $pm$  chargements, moins les contraintes de diagonalisation dues à l'invariance du modèle à des rotations orthogonales.

des paramètres, communalités, distribution des données, etc. Ces règles du pouce sont donc essentiellement arbitraires.

Il existe plusieurs méthodes pour extraire les facteurs, c'est-à-dire pour estimer les paramètres du modèle (les  $\psi_i$  et les  $\gamma_{ij}$ ). Nous allons discuter de deux d'entre elles : la méthode du maximum de vraisemblance et la méthode des composantes principales. L'avantage de l'estimation par maximum de vraisemblance est qu'elle permet l'utilisation de critères d'information et de statistiques de tests pour guider le choix du nombre de facteurs, en supposant toutefois la normalité des facteurs et des aléas.

### 3.5.1 Rotation des facteurs

Dans le modèle d'analyse factorielle, on peut montrer que, lorsqu'il y a deux facteurs ou plus, il existe plusieurs configurations de facteurs qui donnent la même structure de covariance. En fait, les chargements peuvent seulement être déterminés à une transformation orthogonale près<sup>6</sup>. Si les chargements provenant d'une méthode d'extraction des facteurs ne sont pas uniques, la matrice de corrélation estimée par le modèle est par contre unique.

Il existe plusieurs techniques de rotation de facteurs. Le but de ces techniques est d'essayer de trouver une solution qui fera en sorte que les facteurs seront facilement interprétables. La méthode la plus utilisée est la méthode **varimax** : elle produit une configuration de chargement en maximisant la variance de la somme des carrés des chargements pour les  $m$  facteurs.

La méthode varimax tend à produire une configuration de facteurs tel que les chargements de chaque variable sont dispersés (des chargements élevés positifs ou négatifs et d'autres presque nuls). Il est conseillé de toujours tenter d'interpréter la solution avec une rotation varimax. Si ce n'est pas suffisamment clair, il existe d'autres méthodes de rotation dont certaines (les rotations de type oblique) permettent la présence de corrélation entre les facteurs.

#### 3.5.1.1 Estimation par la méthode des composantes principales

La façon la plus simple d'estimer les chargements est d'utiliser la méthode des composantes principales en prenant comme estimation

$$\hat{\Gamma} = \mathbf{Q}_{1:m} \text{diag}(\lambda_1^{1/2}, \dots, \lambda_m^{1/2}),$$

où  $\lambda_j$  est la  $j$ e plus grande valeur propre de la matrice de covariance empirique  $\mathbf{S}$  et  $\mathbf{Q}_{1:m}$  est la sous-matrice formée par les  $m$  premières colonnes de vecteurs propres de  $\mathbf{Q}$ . On peut estimer les variances des aléas à travers

$$\hat{\psi} = \text{diag}(\mathbf{S} - \hat{\Gamma}\hat{\Gamma}^\top).$$

---

6. Une transformation orthogonale est une transformation qui préserve le produit scalaire; elle préserve ainsi toutes les distances et les angles entre deux vecteurs.

L'avantage de cette approche est que l'on peut utiliser la même décomposition en valeurs propres et vecteurs propres pour chaque valeur de  $m$  : seule la rotation dépend de la dimension choisie. La solution est également toujours valide avec la garantie que  $\hat{\psi}_j > 0$ . On peut utiliser la discussion de la Section 3.4.1 pour choisir le nombre de variables.

```
# Solution (chargements) avec rotation varimax
facto_cp <- hecmulti::factocp(factor, nfact = "kaiser", cor = TRUE)
```

### 3.5.1.2 Estimation par maximum de vraisemblance

Si on suppose que les aléas et les facteurs suivent des lois Gaussiennes, alors on peut obtenir une forme explicite pour la fonction de vraisemblance de la matrice de covariance. L'estimation des paramètres requiert une optimisation numérique qui est souvent difficile et qui mène parfois à des solutions paradoxales. On obtient un cas de quasi-Heywood quand  $h_j = 1$  pour une variable  $j$ , (on parle de cas de Heywood si  $h_j > 1$ ). Si on modélise des variables explicatives centrées réduites,  $V_a(X_j) = 1$ , d'où un problème d'interprétation car le terme  $\psi_j$  serait nul (cas de quasi-Heywood) ou négatif (cas de Heywood) alors même que ce terme représente la variance du  $j$ e aléa. Les cas de quasi-Heywood ont plusieurs causes, lesquelles sont listées dans la documentation **SAS**. Souvent, c'est dû à l'utilisation d'un trop petit ou trop grand nombre de facteurs ou une taille d'échantillon trop petite, etc. Cela complique l'interprétation et nous amène à questionner la validité du modèle d'analyse factorielle comme simplification de la structure de covariance.

Les chargements estimés pour la solution à quatre facteurs, suite à la rotation varimax, sont obtenus avec le code suivant :

```
# Ajuster le modèle factoriel par maximum de vraisemblance
fa4 <- factanal(x = factor,
                  factors = 4L)
# Imprimer les chargements en omettant les valeurs inférieures à 0.3
print(fa4$loadings,
      cutoff = 0.3)
```

On constate à la lecture du Tableau 3.4 des chargements que le chargement associé à la première variable est de 0.992 pour le facteur 4 : cela correspondrait à un facteur avec une corrélation de presque un, donc  $F_4 \approx X_1$ . Le modèle obtenu avec la méthode du maximum de vraisemblance n'est donc pas adéquat puisque l'optimisation a convergé vers un cas de quasi-Heywood et que le facteur n'est pas une variable latente, mais une des variables de la base de données de départ. Pour diagnostiquer le tout, on peut aussi analyser les valeurs d'unicité : la minimum de  $\min(fa4$uniqueness)$  est 0.005, ce qui correspond à la tolérance de l'algorithme (valeur minimale permise dans l'optimisation), voir ?factanal. On retourne à la planche à dessin en réduisant le nombre de variables.

TABLEAU 3.4 – Estimés des chargements (multipliés par 100) pour le modèle à quatres facteurs avec rotation varimax estimé à l'aide de la méthode du maximum de vraisemblance. Les chargements inférieurs à 0.3 sont omis.

	F1	F2	F3	F4
x1				99
x2			67	
x3			75	
x4	71			
x5				37
x6		51		
x7			75	
x8	79			
x9		63		
x10			66	
x11	71			
x12		61		

En général, on associe une variable à un groupe (facteur) si son chargement est supérieur à 0.3 (en valeur absolue), ce qui donne

- Facteur 1 :  $X_4$ ,  $X_8$  et  $X_{11}$
- Facteur 2 :  $X_3$ ,  $X_6$ ,  $X_9$  et  $X_{12}$
- Facteur 3 :  $X_2$ ,  $X_7$  et  $X_{10}$
- Facteur 4 :  $X_1$  et  $X_5$ .

Ce point de coupure est arbitraire et peut être augmenté si on note qu'il y a trop de variables disparates. Le signe des chargements est arbitraires.

Ces facteurs sont interprétables :

- Le facteur 1 représente l'importance accordée au service.
- Le facteur 2 représente l'importance accordée aux produits.
- Le facteur 3 représente l'importance accordée à la facilité de paiement.
- Le facteur 4 représente l'importance accordée aux prix.

Dans cet exemple, les choses se sont bien passées et le nombre de facteurs que nous avons spécifié semble être adéquat (hormis le cas de quasi-Heywood), mais ce n'est pas toujours aussi évident. Il est utile d'avoir des outils pour guider le choix du nombre de facteurs.

### 3.5.2 Choix du nombre de facteurs

Il existe différentes méthodes pour se guider dans le nombre de facteurs,  $m$ , à utiliser. Cependant, le point important à retenir est que, peu importe le nombre choisi, il faut que les facteurs soient **interprétables**. Par conséquent, les méthodes qui suivent ne devraient servir que de guide et non pas être suivies aveuglément. La méthode du maximum de vraisemblance que nous avons utilisée dans l'exemple possède l'avantage de fournir trois critères pour choisir le nombre de facteurs appropriés. Ces critères sont :

- le critère d'information d'Akaike (AIC)
- le critère d'information bayésien de Schwarz (BIC)
- le test du rapport de vraisemblance pour l'hypothèse nulle que le modèle de corrélation décrit le modèle factoriel avec  $m$  facteurs est adéquat, contre l'alternative qu'il n'est pas adéquat.

Les critères d'information servent à la sélection de modèles ; ils seront traités plus en détail dans les chapitres qui suivent. Pour l'instant, il est suffisant de savoir que le modèle avec la valeur du critère AIC (ou BIC) la plus petite est considéré le « meilleur » (selon ce critère).

Le paquet `hecmulti` contient des méthodes pour extraire la log-vraisemblance, les critères d'information pour un modèle d'analyse factorielle (objet de classe `factanal`). On peut extraire la valeur- $p$  pour le test du rapport de vraisemblance comparant le modèle à 12 variables (corrélation empirique) avec le modèle simplifié obtenu en utilisant quatre facteurs : une valeur- $p$  supérieur à un seuil prédéfini (typiquement  $\alpha = 0.05$ ) indique que la simplification est adéquate puisqu'on ne rejette pas l'hypothèse nulle. La sortie suivante dans le Tableau 3.5 présente les diagnostics du modèle en fonction du nombre de facteurs pour les modèles ajustés selon la méthode du maximum de vraisemblance.

```
library(hecmulti)
ajustement_factanal(
  covmat = cov(factor),
  factors = 1:5,
  n.obs = nrow(factor))
```

Le nombre de facteurs à utiliser selon le AIC est 4, versus 3 selon le BIC. Le nombre mimimal de critères selon le test du rapport de vraisemblance est NA. Ainsi, on retient la solution à trois facteurs dans tous les cas : cette adéquation entre les critères est l'exception plutôt que la règle.

Il faut garder en tête que l'estimation par maximum de vraisemblance du modèle d'analyse factorielle est très sensible à l'initialisation : on peut aussi parfois obtenir des valeurs différentes selon les logiciels. Cette fragilité, couplée à la haute fréquence de cas de Heywood, fait en sorte que je préfère utiliser la méthode des composantes principales pour l'estimation.

TABLEAU 3.5 – Ajustement de modèles d'analyse factorielle par la méthode du maximum de vraisemblance pour différent nombres de facteurs : critères d'informations AIC et BIC, valeur-p du test de rapport de vraisemblance, nombre de paramètres estimés et indicateur pour les cas de (quasi)-Heywood

k	AIC	BIC	valeur-p	npar	heywood	
1	1	2267	2346	< 2e-16	24	0
2	2	2138	2253	< 2e-16	35	0
3	3	2017	2166	0.09604	45	0
4	4	2003	2181	0.97262	54	1
5	5	2013	2217	0.97445	62	1

On peut considérer le modèle avec trois facteurs : les chargements (après rotation varimax) sont données dans le Tableau 3.6

Cette solution récupère les trois facteurs *service*, *produits* et *paiement* de la solution précédente à quatre facteurs. Le facteur *prix* (qui était formé de  $X_1$  et  $X_5$ ) n'est plus présent.

On suggère d'utiliser les trois critères découlant de l'utilisation de la vraisemblance et de déterminer le nombre de facteurs à extraire selon différents critères avant d'examiner les modèles avec ce nombre de facteurs et ceux avec un facteur de moins ou de plus. Au final, le plus important est de pouvoir interpréter raisonnablement les facteurs : la configuration de facteurs choisie est logique et compréhensible.

### 3.5.3 Construction d'échelles à partir des facteurs

Si le seul but de l'analyse factorielle est de comprendre la structure de corrélation entre les variables, alors se limiter à l'interprétation des facteurs est suffisant.

Si par contre, le but est de réduire le nombre de variables pour pouvoir par la suite procéder à d'autres analyses statistiques, l'analyse factorielle peut alors servir de guide pour construire de nouvelles variables (échelles). En supposant que l'analyse factorielle a produit des facteurs qui sont interprétables et satisfaisants, la méthode de construction d'échelles la plus couramment utilisée consiste à construire  $m$  nouvelles variables, une par facteur. Pour un facteur donné, la nouvelle variable est simplement la moyenne des variables ayant des chargements élevés sur ce facteur. Une autre méthode, les scores factoriels, sera présentée plus loin. Il est important que les corrélations soient de même signe si on veut regrouper les variables dans les échelles pour que ce regroupement soit logique : certaines questions avec des échelles de Likert ont parfois un encodage inverse comme test d'attention.

Est-il logique de calculer des échelles avec autre chose que des items de questionnaire ramenés sur une plage commune ? Par forcément... Il faut aussi s'assurer que les variables ont la même plage

TABLEAU 3.6 – Estimés des chargements (multipliés par 100) pour le modèle à trois facteurs avec rotation varimax estimé à l'aide de la méthode du maximum de vraisemblance. Les chargements inférieurs à 0.3 sont omis.

	F1	F2	F3
x1			
x2			67
x3		76	
x4	71		
x5			
x6		50	
x7			75
x8	79		
x9		63	
x10			67
x11	72		
x12		60	

ou étendue avant des les combiner, sinon certaines variables seront des poids plumes et seule la variable avec la plus grande étendue ressortira.

Lorsqu'on construit une échelle, il est important d'examiner sa cohérence interne. Ceci peut être fait à l'aide du coefficient alpha de Cronbach. Ce coefficient mesure à quel point chaque variable faisant partie d'une échelle est corrélée avec le total de toutes les variables pour cette échelle. Plus le coefficient est élevé, plus les variables ont tendance à être corrélées entre elles. L'alpha de Cronbach est

$$\alpha = \frac{k}{k-1} \frac{S^2 - \sum_{i=1}^k S_i^2}{S^2},$$

où  $k$  est le nombre de variables dans l'échelle,  $S^2$  est la variance empirique de la somme des variables et  $S_i^2$  est la variance empirique de la  $i$ e variable. En pratique, on voudra que ce coefficient soit au moins égal à 0.6 pour être satisfait de la cohérence interne de l'échelle.<sup>7</sup>

Le paquet `hecmulti` contient une fonction, `alphaC`, pour faire l'estimation du  $\alpha$  de Cronbach

```
# Création des échelles
ech_service <- rowMeans(factor[,c("x4","x8","x11")])
ech_produit <- rowMeans(factor[,c("x3","x6","x9","x12")])
```

7. Bien que ce nombre soit arbitraire.

TABLEAU 3.7 – Coefficient alpha de Cronbach pour les quatre échelles formées.

service	produit	paiement	prix
0.781	0.718	0.727	0.546

```

ech_paiement <- rowMeans(factor[,c("x2","x7","x10")])
ech_prix <- rowMeans(factor[,c("x1","x5")])

# Cohérence interne (alpha de Cronbach)
alphaC(factor[,c("x4","x8","x11")])
alphaC(factor[,c("x3","x6","x9","x12")])
alphaC(factor[,c("x2","x7","x10")])
alphaC(factor[,c("x1","x5")])

```

Ainsi, les  $\alpha$  de Cronbach sont tous satisfaisants (plus grand que 0.6) sauf pour le facteur *prix* (0.546). Tout est donc cohérent. Les échelles provenant des facteurs *service*, *produits* et *paiement*, sont satisfaisantes. Ces facteurs sont identifiés à la fois dans la solution à quatre, mais aussi dans la solution à trois facteurs. Le facteur *prix* est celui qui apparaît en plus dans la solution à quatre facteurs. Il a une interprétation claire (c'est essentiellement *x1*), mais son faible  $\alpha$  ferait en sorte qu'il serait discutable de travailler avec l'échelle *prix* dans d'autres analyses (du moins avec selon l'usage habituel du  $\alpha$ ) plutôt que d'utiliser directement la variable *x1*.

## 3.6 Compléments d'information

### 3.6.1 Variables ordinaires

Théoriquement, une analyse factorielle ne devrait être faite qu'avec des variables continues. Par contre, en pratique, on l'utilise souvent aussi avec des variables ordinaires (comme pour l'exemple portant sur le questionnaire) et même avec des variables binaires (0-1).

Dans ce genre de situation, on peut aussi utiliser d'autres mesures d'associations au lieu du coefficient de corrélation linéaire de Pearson. Par exemple, on peut utiliser la corrélation polychorique, qui est une mesure de corrélation entre deux variables ordinaires. La corrélation tétrachorique correspond au cas spécial de deux variables binaires.

Ma suggestion est d'utiliser la corrélation linéaire ordinaire avec des variables ordinaires (même binaires). Si les résultats ne sont pas satisfaisants, on peut alors essayer avec d'autres mesures d'associations.

### 3.6.2 Autres méthodes de rotation des facteurs

Jusqu'à présent, nous avons utilisé la méthode de rotation orthogonale varimax. Il existe de nombreuses autres méthodes de rotations orthogonales fournies dans le paquet psych. Rappelez-vous que le modèle d'analyse factorielle de base suppose que les facteurs sont non corrélés. Les rotations de type obliques permettent d'introduire de la corrélation entre les facteurs : quelquefois, une telle rotation facilitera davantage l'interprétation des facteurs qu'une rotation orthogonale. Notez qu'il faut être prudent lorsqu'on utilise une méthode de rotation oblique car il y aura trois matrices de chargements après rotation (coefficients de régression normalisés, corrélations semi-partielles ou corrélations). On suggère l'utilisation de la première, soit la représentation avec **coefficients de régression normalisés**. Il s'agit des coefficients de régression si on voulait prédire les variables à l'aide des facteurs. Ils indiquent donc à quel point chaque facteur est associé à chaque variable. Dans le cas d'une rotation orthogonale, ces trois matrices sont les mêmes et il s'agit de trois interprétations valides des chargements.

### 3.6.3 Scores factoriels

Avec les données de l'exemple, en nous basant sur les résultats de l'analyse factorielle, nous avons créé quatre nouvelles échelles (une par facteur) que l'on peut calculer pour chaque individu :

- `service` =  $(X_4 + X_8 + X_{11})/3$ ,
- `produit` =  $(X_3 + X_6 + X_9 + X_{12})/4$ ,
- `paiement` =  $(X_2 + X_7 + X_{10})/3$ ,
- `prix` =  $(X_1 + X_5)/2$ .

Par exemple, la variable `prix` peut donc être vu comme une combinaison linéaire des 12 variables où seulement  $X_1$  et  $X_5$  reçoivent un poids (égal) différent de zéro. Une autre façon de créer de nouvelles variables consiste à calculer des scores factoriels (un pour chaque facteur) pour chaque individu à partir de la matrice de données centrées et réduite  $\mathbf{Z}$  (de telle sorte que la moyenne de chaque colonne soit 0 et la variance 1). Les score factoriel

$$\begin{aligned}\hat{F}_{i,k} &= [\mathbf{Z}\mathbf{R}^{-1}\hat{\Gamma}]_{ik} \\ &= \hat{\beta}_{1,k}z_{i,1} + \cdots + \hat{\beta}_{12,k}z_{i,12}\end{aligned}$$

où  $\hat{\Gamma}$  est la matrice  $p \times m$  des chargements,  $\mathbf{R}$  la matrice  $p \times p$  des corrélations empiriques et où  $z_{i,1}, \dots, z_{i,12}$  est la  $i$  e ligne (observation) de  $\mathbf{Z}$ . La matrice  $p \times m$  de coefficients  $\boldsymbol{\beta} = \mathbf{R}^{-1}\hat{\Gamma}$ .

Ainsi, chacune des 12 variables originales contribue au calcul du score factoriel. Les variables ayant des chargements plus élevés sur ce facteur auront tendance à avoir des poids ( $\hat{\gamma}$ ) plus élevés. Par contre, les scores factoriels ne sont pas uniques car ils dépendent des chargements utilisés (et donc à la fois de la méthode d'estimation et de la méthode de rotation). On peut également utiliser les scores factoriels au lieu des 12 variables originales dans des analyses subséquentes. Il est

TABLEAU 3.8 – Coefficients du score (modèle de régression, maximum de vraisemblance) pour le modèle d'analyse factorielle à quatre facteurs.

	facteur 1	facteur 2	facteur 3	facteur 4
x1	0.03	0.05	0.04	1.01
x2	-0.05	0.01	0.31	0.01
x3	0.01	0.45	-0.02	0.01
x4	0.30	0.01	0.01	0.03
x5	0.00	-0.01	-0.01	0.00
x6	-0.01	0.17	0.02	0.00
x7	-0.01	-0.02	0.44	0.02
x8	0.45	-0.05	0.04	0.04
x9	-0.03	0.27	-0.02	0.00
x10	0.02	0.01	0.30	0.02
x11	0.30	0.00	-0.07	0.02
x12	0.00	0.24	0.00	0.01

suggéré d'utiliser les nouvelles variables (échelles) obtenues en faisant les moyennes des variables identifiées comme faisant partie de chaque facteur pour les raisons suivantes :

- l'interprétation des scores factoriels est moins claire (chaque facteur dépend de toutes les variables)
- les scores factoriels ne sont pas uniques (ils dépendent de la méthode d'estimation et de rotation).
- les coefficients servant au calcul seront différents d'une étude à l'autre.

Les scores factoriels sont obtenus en spécifiant `scores = "regression"` dans les options de la fonction `factanal`. Les poids avec le modèle à quatre facteurs sont rapportés dans le Tableau 3.8. On remarque que

- pour le premier facteur, trois variables ont des poids importants ( $X_4$ ,  $X_8$  et  $X_{11}$ ). Il s'agit donc d'un facteur très proche du facteur *service*.
- pour le deuxième facteur, les variables  $X_3$ ,  $X_6$ ,  $X_9$  et  $X_{12}$  ont des poids importants. Il s'agit donc d'un facteur très proche du facteur *produits*.
- pour le troisième facteur, les variables  $X_2$ ,  $X_7$ ,  $X_{10}$  ont des poids importants. Il s'agit donc d'un facteur très proche du facteur *paiement*.
- pour le quatrième facteur, seule la variable  $X_1$  a un poids important. On aurait pu s'attendre à ce que ce soit également le cas pour  $X_5$ , en lien avec le facteur *prix* — ce facteur était moins clair selon le alpha de Cronbach.

Les corrélations entre les échelles (construites avec les moyennes) et les scores factoriels sont

TABLEAU 3.9 – Corrélation entre quatre premiers scores (modèle d'analyse factorielle à quatre facteurs) et échelles.

	score 1	score 2	score 3	score 4
échelle 1	0.99	0.05	0.03	-0.05
échelle 2	0.05	0.98	0.03	-0.04
échelle 3	0.03	0.05	0.99	-0.07
échelle 4	-0.07	-0.04	-0.08	0.82

données dans le Tableau 3.9. On remarque la forte corrélation entre le score factoriel et les échelles construites avec les moyennes pour les facteurs *service*, *produits* et *paiement*. Cela veut dire qu'utiliser les échelles ou les scores factoriels ne devrait pas faire de différence dans des analyses subséquentes. Par contre, cette corrélation est plus faible (0.82) pour le facteur *prix*.

#### 💡 Étapes de l'analyse factorielle exploratoire

- Déterminer les variables à utiliser dans l'analyse
- Vérifier les prérequis
- Sélectionner une méthode d'estimation et extraire les facteurs
- Déterminer le nombre de facteurs
- Effectuer une rotation des facteurs
- Interpréter les facteurs
- Évaluer la validité du modèle

#### ℹ️ En résumé

- L'analyse factorielle exploratoire fournit un modèle pour la matrice de corrélation
- La solution du problème n'est pas unique : on choisit celle qui permet de mieux séparer les variables.
- Seules les variables numériques pour lesquelles on suspecte une dimension commune sont incluses dans l'analyse.
- On doit avoir beaucoup d'observations (au moins 100, 10 fois plus que de variables) pour estimer le modèle.
- On estime le modèle à l'aide de la méthode des composantes principales (modèle toujours valide et moins coûteux en calcul, mais critères pour la sélection du nombre de facteurs arbitraires) ou du maximum de vraisemblance (optimisation numérique avec solutions fréquemment problématique, critères d'information pour choix du nombre de facteurs).
- Le nombre de facteurs retenu doit donner des regroupements logiques (facteur

*wow).*

- On utilise toujours une rotation orthogonale pour faciliter l'interprétation (varimax par défaut).
- L'interprétation se fait à partir des chargements (corrélation entre variables et facteurs).
- On crée des échelles en prenant la moyenne des variables qui ont un chargement élevés en lien avec un facteur donné (de même signe).
- Les échelles sont cohérentes si le  $\alpha$  de Cronbach est supérieur à 0.6, faute de quoi elles sont rejetées.

## Chapitre 4

# Sélection de variables et de modèles

### 4.1 Introduction

Ce chapitre présente des principes, outils et méthodes très généraux pour choisir un « bon » modèle. Nous allons principalement utiliser la régression linéaire pour illustrer les méthodes en supposant que tout le monde connaît ce modèle de base. Les méthodes présentées sont en revanche très générales et peuvent être appliquées avec n'importe quel autre modèle (régression logistique, arbres de classification et régression, réseaux de neurones, analyse de survie, etc.)

L'expression « sélection de variables » fait référence à la situation où l'on cherche à sélectionner un sous-ensemble de variables à inclure dans notre modèle à partir d'un ensemble de variables  $X_1, \dots, X_p$ . Le terme variable ici inclut autant des variables distinctes que des transformations d'une ou plusieurs variables.

Par exemple, supposons que les variables `age`, `sex` et `revenu` soient trois variables explicatives disponibles. Nous pourrions alors considérer choisir entre ces trois variables. Mais aussi, nous pourrions considérer inclure  $\text{age}^2$ ,  $\text{age}^3$ ,  $\log(\text{age})$ , etc. Nous pourrions aussi considérer des termes d'interactions entre les variables, comme  $\text{age} \cdot \text{revenu}$  ou  $\text{age} \cdot \text{revenu} \cdot \text{sex}$ . Le problème est alors de trouver un bon sous-ensemble de variables parmi toutes celles considérées.

L'expression « sélection de modèle » est un peu plus générale. D'une part, elle inclut la sélection de variables car, pour une famille de modèles spécifiques (régression linéaire par exemple), choisir un sous-ensemble de variables revient à choisir un modèle. D'autre part, elle fait référence à la situation où l'on cherche à trouver le meilleur modèle parmi des modèles de natures différentes. Par exemple, on pourrait choisir entre une régression linéaire, un arbre de régression, une forêt aléatoire, un réseau de neurones, etc.

## 4.2 Sélection de variables et de modèles selon les buts de l'étude

Nous disposons d'une variable réponse  $Y$  et d'un ensemble de variables explicatives  $X_1, \dots, X_p$ . L'attitude à adopter dépend des buts de l'étude.

- **1e situation** : On veut développer un modèle pour faire des prédictions sans qu'il soit important de tester formellement les effets des paramètres individuels.

Dans ce cas, on désire seulement que notre modèle soit performant pour prédire des valeurs futures de  $Y$ . On peut alors baser notre choix de variable (et de modèle) en utilisant des outils qui nous guideront quant aux performances prédictives futures du modèle (voir AIC, BIC et validation croisée plus loin). On pourra enlever ou rajouter des variables et des transformations de variables au besoin afin d'améliorer les performances prédictives. Les méthodes que nous allons voir concernent essentiellement ce contexte.

- **2e situation** : On veut développer un modèle pour estimer les effets de certaines variables sur notre  $Y$  et tester des hypothèses de recherche spécifiques concernant certaines variables.

Dans ce cas, il est préférable de spécifier le modèle dès le départ selon des considérations scientifiques et de s'en tenir à lui. Faire une sélection de variables dans ce cas est dangereux car on ne peut pas utiliser directement les valeurs- $p$  des tests d'hypothèses (ou les intervalles de confiance sur les paramètres) concernant les paramètres du modèle final car elles ne tiennent pas compte de la variabilité due au processus de sélection de variables.

Une bonne planification de l'étude est alors cruciale afin de collecter les bonnes variables, de spécifier le ou les bons modèles, et de s'assurer d'avoir suffisamment d'observations pour ajuster le ou les modèles désirés.

Si procéder à une sélection de variables est quand même nécessaire dans ce contexte, il est quand même possible de le faire en divisant l'échantillon en deux. La sélection de variables pourrait être alors effectuée avec le premier échantillon. Une fois qu'un modèle est retenu, on pourrait alors réajuster ce modèle avec le deuxième échantillon (sans faire de sélection de variables cette fois-ci). L'inférence sur les paramètres (valeurs- $p$ , etc.) sera alors valide. Le désavantage ici qu'il faut avoir une très grande taille d'échantillon au départ afin d'être en mesure de le diviser en deux.

## 4.3 Estimation de la performance

Il est préférable d'avoir un modèle un peu trop complexe qu'un modèle trop simple. Plaçons-nous dans le contexte de la régression linéaire et supposons que le vrai modèle est inclus dans le modèle qui a été ajusté. Il y a donc des variables en trop dans le modèle qui a été ajusté : ce dernier est dit surspécifié.

Par exemple, supposons que le vrai modèle est  $Y = \beta_0 + \beta_1 X_1 + \varepsilon$  mais que c'est le modèle  $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$  qui a été ajusté. Dans ce cas, règle générale, les estimateurs des paramètres et

les prédictions provenant du modèle sont sans biais. Mais leurs variances estimées seront un peu plus élevées car on estime des paramètres pour des variables superflues.

Supposons à l'inverse qu'il manque des variables dans le modèle ajusté et que le modèle ajusté est sous-spécifié. Par exemple, supposons que le vrai modèle est  $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \varepsilon$ , mais que c'est le modèle  $Y = \beta_0 + \beta_1 X_1 + \varepsilon$  qui est ajusté. Dans ce cas, généralement, les estimateurs des paramètres et les prédictions sont biaisés.

```
# Simuler des observations d'un modèle linéaire
n <- 200L
# Variables explicatives bidons
X1 <- rpois(n = n, lambda = 3)
X2 <- rbinom(n = n, size = 1, prob = c(0.7, 0.3))
# Modèle de régression X*beta + aléas
y_a <- 20 + 2*X1 + 5*X2 + rnorm(n)
y_b <- 20 + 2*X1 + rnorm(n)
# Ajuster un modèle linéaire
# lm(réponse ~ variables explicatives)
mod_1a <- lm(y_a ~ X1) # sous-spécifié
mod_2a <- lm(y_a ~ X1 + X2) # correct

mod_1b <- lm(y_b ~ X1) # correct
mod_2b <- lm(y_b ~ X1 + X2) # surspécifié

# Comparer l'estimation des coefficients
# aux vraies valeurs
# Vérifier si ces dernières se trouvent
# dans l'intervalle de confiance
coef(mod_1a) #coefficients
## (Intercept)          X1
##      22.6            2.0
confint(mod_1a) #intervalles de confiance
##                   2.5 % 97.5 %
## (Intercept) 21.9  23.36
## X1           1.8   2.21
coef(mod_2a)
## (Intercept)          X1          X2
##      19.98            2.03            5.01
confint(mod_2a)
##                   2.5 % 97.5 %
## (Intercept) 19.66  20.3
```

```
## X1      1.95    2.1
## X2      4.73    5.3
```

Ainsi, il est généralement préférable d'avoir un modèle légèrement surspécifié qu'un modèle sous-spécifié. Plus généralement, il est préférable d'avoir un peu trop de variables dans le modèle que de prendre le risque d'omettre une ou plusieurs variables importantes. Il faut faire attention et ne pas tomber dans l'excès et avoir un modèle trop complexe (avec trop de variables inutiles) car il pourrait souffrir de surajustement (*over-fitting*). Les exemples qui suivent illustreront ce fait.

### 4.3.1 Surajustement

Cette section traite de l'optimisme de l'évaluation d'un modèle (*trop beau pour être vrai*) lorsqu'on utilise les mêmes données qui ont servies à l'ajuster pour évaluer sa performance. Un principe fondamental lorsque vient le temps d'évaluer la performance prédictive d'un modèle est le suivant : si on utilise les mêmes observations pour évaluer la performance d'un modèle que celles qui ont servi à l'ajuster (à estimer le modèle et ses paramètres), on va surestimer sa performance. Autrement dit, notre estimation de l'erreur que fera le modèle pour prédire des observations futures sera biaisée à la baisse. Ainsi, il aura l'air meilleur que ce qu'il est en réalité. C'est comme si on demandait à un cinéaste d'évaluer son dernier film. Comme c'est son film, il n'aura généralement pas un regard objectif. C'est pourquoi on aura tendance à se fier à l'opinion d'un critique.

On cherchera donc à utiliser des outils et méthodes qui nous donneront l'heure juste (une évaluation objective) quant à la performance prédictive d'un modèle.

### 4.3.2 Principes généraux

Les idées présentées ici seront illustrées à l'aide de la régression linéaire. Par contre, elles sont valides dans à peu près n'importe quel contexte de modélisation.

Plaçons-nous d'abord dans un contexte plus général que celui de la régression linéaire. Supposons que l'on dispose de  $n$  observations indépendantes sur  $(Y, X_1, \dots, X_p)$  et que l'on a ajusté un modèle  $\hat{f}(X_1, \dots, X_p)$ , avec ces données, pour prédire une variable continue  $Y$ .

Ce modèle peut être un modèle de régression linéaire,

$$\hat{f}(X_1, \dots, X_p) = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \dots + \hat{\beta}_p X_p$$

mais il pourrait aussi avoir été construit selon d'autres méthodes (réseau de neurones, arbre de régression, forêt aléatoire, etc.) Une manière de quantifier la performance prédictive du modèle est l'erreur quadratique moyenne (*mean squared error*),

$$\text{EQM} = E \left[ \{(Y - \hat{f}(X_1, \dots, X_p))^2\} \right]$$

lorsque  $(Y, X_1, \dots, X_p)$  est choisi au hasard dans la population. Cette quantité mesure l'erreur théorique (la différence au carré entre la vraie valeur de  $Y$  et la valeur prédictive par le modèle) que fait le modèle en moyenne pour l'ensemble de la population. Plus cette quantité est petite, meilleur est le modèle. Le problème est que l'on ne peut pas la calculer car on n'a pas accès à toute la population. Tout au plus peut-on essayer de l'estimer ou bien d'estimer une fonction qui, sans l'estimer directement, classifiera les modèles dans le même ordre qu'elle.

Une première idée est d'estimer l'erreur quadratique moyenne de l'échantillon d'apprentissage (*training mean squared error*),

$$\widehat{\text{EQM}}_a = \frac{1}{n} \sum_{i=1}^n \{Y_i - \hat{f}(X_{i1}, \dots, X_{ip})\}^2.$$

Malheureusement, selon le principe fondamental de la section précédente, cette quantité n'est pas un bon estimateur de l'EQM. En effet, comme on utilise les mêmes observations que celles qui ont estimé le modèle, l' $\widehat{\text{EQM}}_a$  aura tendance à toujours diminuer lorsqu'on augmente la complexité du modèle (par exemple, lorsqu'on augmente le nombre de paramètres). L' $\widehat{\text{EQM}}_a$  tend à surestimer la qualité du modèle en sous-estimant l'EQM et le modèle a l'air meilleur qu'il ne l'est en réalité.

#### 4.3.3 Présentation de l'exemple

Cet exemple simple sur le choix d'un modèle polynomial en régression linéaire servira à illustrer le fait qu'on ne peut utiliser directement les mêmes données qui ont servi à ajuster un modèle pour évaluer sa performance.

Nous disposons de 100 observations sur une variable cible  $Y$  et d'une seule variable explicative  $X$  dans la base de données `selection1_train`. Nous voulons considérer des modèles polynomiaux (en  $X$ ) afin d'en trouver un bon pour prédire  $Y$ . Un modèle polynomial est un modèle de la forme  $Y = \beta_0 + \beta_1 X + \dots + \beta_k X^k + \varepsilon$ . Le cas  $k = 1$  correspond à un modèle linéaire simple,  $k = 2$  à un modèle cubique,  $k = 3$  à un modèle cubique, etc. Notre but est de déterminer l'ordre ( $k$ ) du polynôme qui nous donnera un bon modèle. Voici d'abord le graphe de ces 100 observations de l'échantillon d'apprentissage.

Ces données ont été obtenues par simulation et le vrai modèle sous-jacent (celui qui a généré les données) est le modèle cubique, c'est-à-dire le modèle d'ordre  $k = 3$ .

J'ai ajusté tour à tour à tour les modèles polynomiaux jusqu'à l'ordre 10, avec l'échantillon d'apprentissage de taille 100. C'est-à-dire, le modèle linéaire avec un polynôme d'ordre  $k = 1$  (linéaire),  $k = 2$  (quadratique), etc., jusqu'à  $k = 10$ . J'ai ensuite obtenu la valeur de l'erreur quadratique moyenne d'apprentissage pour chacun de ces modèles. En pratique, on ne pourrait pas calculer l'erreur quadratique moyenne de généralisation puisqu'on ne connaît pas le vrai modèle. J'ai fait une approximation de cette dernière en simulant 100 000 observations du vrai modèle (`selection1_test`), en

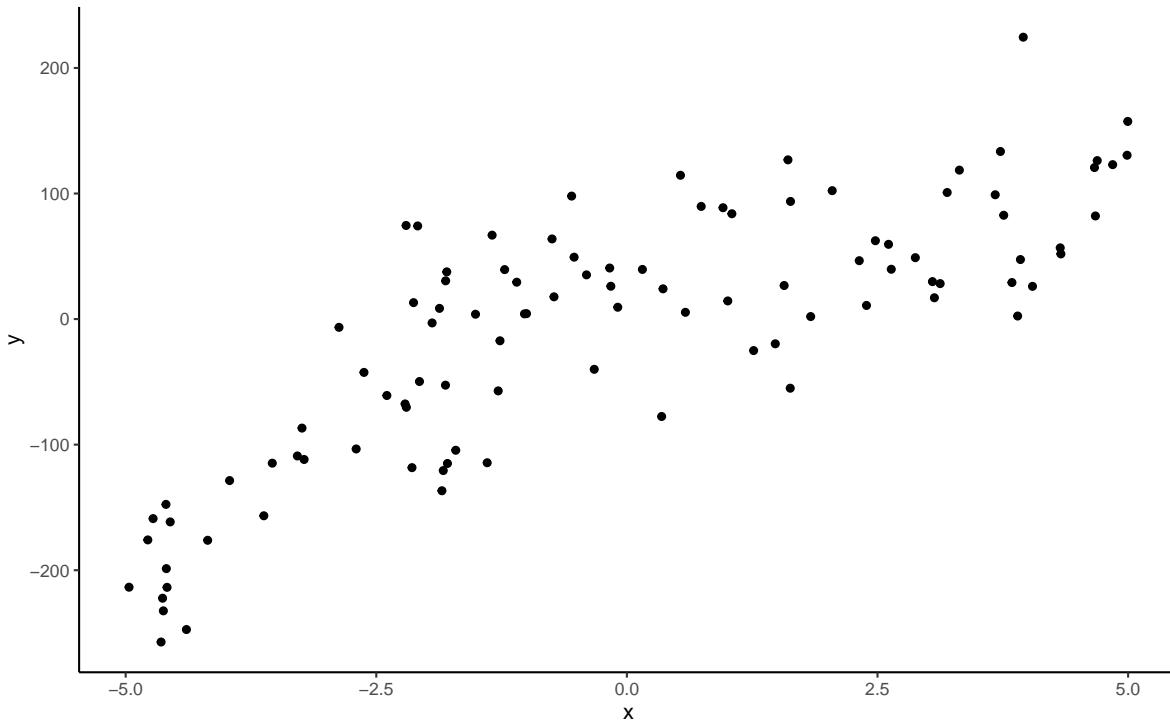


FIGURE 4.1 – Nuage de points de 100 observations simulées d'un modèle polynomial de degré inconnu.

obtenant la prédiction pour chacune de ces 100 000 observations en utilisant le modèle d'ordre  $k$  ajusté sur les données d'apprentissage et en calculant l'erreur quadratique moyenne par la suite.

On voit clairement dans la Figure 4.2 que l' $\widehat{\text{EQM}}_a$  diminue en fonction de l'ordre sur l'échantillon d'apprentissage : plus le modèle est complexe, plus l'erreur observée sur l'échantillon d'apprentissage est petite. La courbe EQM donne l'heure juste, car il s'agit d'une estimation de la performance réelle des modèles sur de nouvelles données. On voit que le meilleur modèle est donc le modèle cubique ( $k = 3$ ), ce qui n'est pas surprenant puisqu'il s'agit du modèle que utilisé pour générer les données. On peut aussi remarquer d'autres éléments intéressants. Premièrement, on obtient un bon gain en performance (EQM) en passant de l'ordre 2 à l'ordre 3. Ensuite, la perte de performance en passant de l'ordre 3 à 4, et ensuite à des ordres supérieurs n'est pas si sévère, même si elle est présente. Cela illustre empiriquement qu'il est préférable d'avoir un modèle un peu trop complexe que d'avoir un modèle trop simple. Il serait beaucoup plus grave pour la performance de choisir le modèle avec  $k = 2$  que celui avec  $k = 4$ .

En pratique par contre, on n'a pas accès à la population : les 100 000 observations qui ont servi à estimer l'EQM théorique ne seront pas disponible. Si on a seulement l'échantillon d'apprentissage, soit 100 observations dans notre exemple, comment faire alors pour choisir le bon modèle? C'est

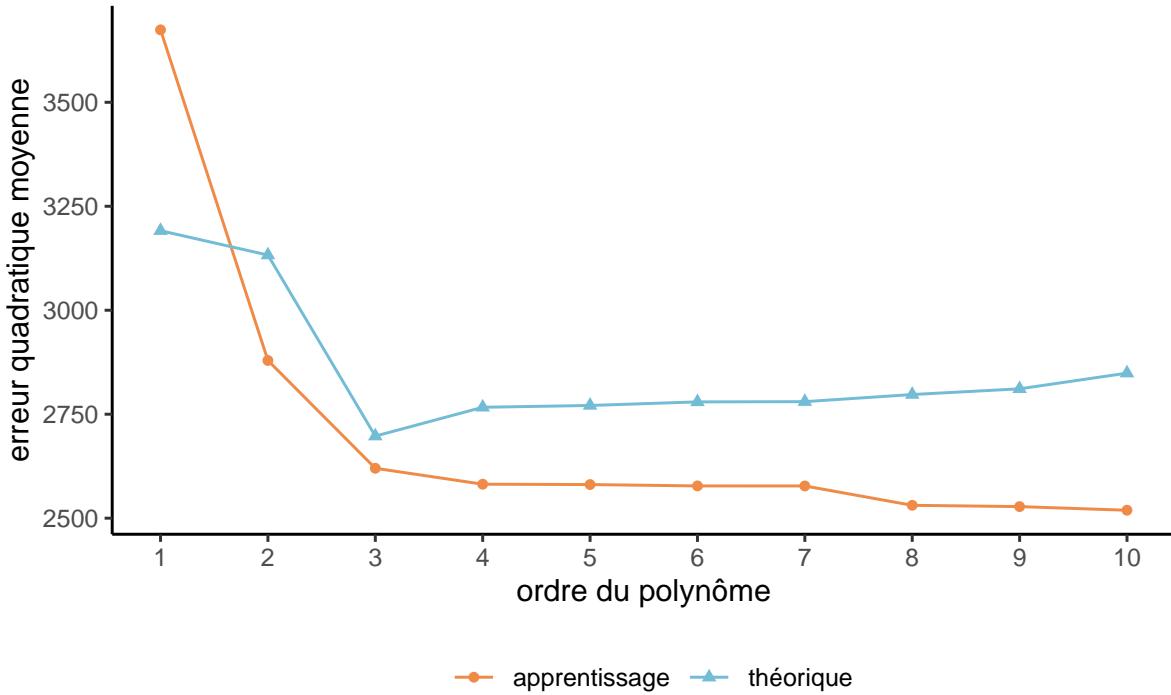


FIGURE 4.2 – erreur quadratique moyenne d'apprentissage ( $\widehat{\text{EQM}}_a$ ) et erreur quadratique moyenne théorique ( $\text{EQM}$ ) en fonction de l'ordre ( $k$ ) du polynôme ajusté.

ce que nous verrons à partir de la section suivante.

Mais avant cela, nous allons discuter un peu plus en détail au sujet de la régression linéaire et d'une mesure très connue, le coefficient de détermination ( $R^2$ ). Supposons que l'on a ajusté un modèle de régression linéaire

$$\hat{f}(X_1, \dots, X_p) = \hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \dots + \hat{\beta}_p X_p.$$

La somme du carré des erreurs (SCE) pour notre échantillon est

$$\text{SCE} = \sum_{i=1}^n (Y_i - \hat{\beta}_0 - \hat{\beta}_1 X_1 - \dots - \hat{\beta}_p X_p)^2 = \sum_{i=1}^n (Y_i - \hat{Y}_i)^2.$$

On peut démontrer que si on ajoute une variable quelconque au modèle, la valeur de la somme du carré des erreurs va nécessairement baisser. Il est facile de se convaincre de cela. En régression linéaire, les estimations sont obtenues par la méthode des moindres carrés qui consiste justement à minimiser la SCE. Ainsi, en ajoutant une variable  $X_{p+1}$  au modèle, la SCE ne peut que baisser car, dans le pire des cas, le paramètre de la nouvelle variable sera  $\hat{\beta}_{p+1} = 0$  et on retombera sur le modèle sans cette variable. C'est pourquoi, la quantité  $\widehat{\text{EQM}}_a = \text{SCE}/n$  ne peut être utilisée comme outil de sélection de modèles en régression linéaire.

Nous venons d'ailleurs d'illustrer cela avec notre exemple sur les modèles polynomiaux. En effet, augmenter l'ordre du polynôme de 1 revient à ajouter une variable. Le coefficient de détermination ( $R^2$ ) est souvent utilisé, à tort, comme mesure de qualité du modèle. Il peut s'interpréter comme étant la proportion de la variance de  $Y$  qui est expliquée par le modèle.

Le coefficient de détermination est

$$R^2 = \{\text{cor}(\mathbf{y}, \hat{\mathbf{y}})\}^2 = 1 - \frac{\text{SCE}}{\text{SCT}},$$

où  $\text{SCT} = \sum_{i=1}^n (Y_i - \bar{Y})^2$  est la somme des carrés totale calculée en centrant les observations. La somme des carrés totale,  $\text{SCT}$ , ne varie pas en fonction du modèle. Ainsi, on voit que le  $R^2$  va mécaniquement augmenter lorsqu'on ajoute une variable au modèle (car la  $\text{SCE}$  diminue). C'est pourquoi on ne peut pas l'utiliser comme outil de sélection de variables.

Le problème principal que nous avons identifié jusqu'à présent afin d'être en mesure de bien estimer la performance d'un modèle est le suivant : si on utilise les mêmes observations pour évaluer la performance d'un modèle que celles qui ont servi à l'ajuster, on va surestimer sa performance.

Il existe deux grandes approches pour contourner ce problème lorsque le but est de faire de la sélection de variables ou de modèle :

- utiliser les données de l'échantillon d'apprentissage (en échantillon) et pénaliser la mesure d'ajustement (ici  $\widehat{\text{EQM}}_a$ ) pour tenir compte de la complexité du modèle (par exemple, à l'aide de critères d'informations).
- tenter d'estimer l'EQM directement sur d'autres données (hors échantillon) en utilisant des méthodes de rééchantillonnage, notamment la validation croisée ou la validation externe (division de l'échantillon).

#### 4.3.4 Pénalisation et critères d'information

Plaçons-nous dans le contexte de la régression linéaire pour l'instant. Nous avons déjà utilisé les critères AIC et BIC en analyse factorielle. Il s'agit de mesures qui découlent d'une méthode d'estimation des paramètres, la méthode du maximum de vraisemblance.

Il s'avère que les estimateurs des paramètres obtenus par la méthode des moindres carrés en régression linéaire sont équivalents à ceux provenant de la méthode du maximum de vraisemblance si on suppose la normalité des termes d'erreurs du modèle. Ainsi, dans ce cas, nous avons accès aux AIC et BIC, deux critères d'information définis pour les modèles dont la fonction objective est la vraisemblance (qui mesure la probabilité des observations sous le modèle postulé suivant une loi choisie par l'utilisateur). La fonction de vraisemblance  $\mathcal{L}$  et la log-vraisemblance  $\ell$  mesurent l'adéquation du modèle.

Supposons que nous avons ajusté un modèle avec  $p$  paramètres en tout (**inclus** l'ordonnée à l'origine). En régression linéaire, le critère d'information d'Akaike, AIC, est

$$\text{AIC} = -2\ell(\hat{\beta}, \hat{\sigma}^2) + 2p = n \ln(\text{EQM}) + 2p + \text{constante},$$

tandis que le critère d'information bayésien de Schwartz, BIC, est défini par

$$\text{BIC} = -2\ell(\hat{\beta}, \hat{\sigma}^2) + p \ln(n) = n \ln(\text{EQM}) + p \ln(n) + \text{constante}.$$

Plus la valeur du AIC (ou du BIC) est petite, meilleur est l'adéquation. Que se passe-t-il lorsqu'on ajoute un paramètre à un modèle? D'une part, la somme du carré des erreurs va mécaniquement diminuer tout comme l'erreur quadratique moyenne  $\text{EQM} = \text{SCE}/n$ , donc la quantité  $n \ln(\text{EQM})$  va diminuer. D'autre part, la valeur de  $p$  augmente de 1. Ainsi, le AIC peut soit augmenter, soit diminuer, lorsqu'on ajoute un paramètre; idem pour le BIC. Par exemple, le AIC va diminuer seulement si la baisse de la somme du carré des erreurs est suffisante pour compenser le fait que le terme  $2p$  augmente à  $2(p+1)$ .

Ces critères pénalisent l'ajout de variables afin de se prémunir contre le surajustement. De plus, le BIC pénalise plus que le AIC. Par conséquent, le critère BIC va choisir des modèles contenant soit le même nombre, soit moins de paramètres que le AIC.

Les critères AIC et BIC peuvent être utilisés comme outils de sélection de variables en régression linéaire mais aussi beaucoup plus généralement avec d'autres méthodes basées sur la vraisemblance (analyse factorielle, régression logistique, etc.) En fait, n'importe quel modèle dont les estimateurs proviennent de la méthode du maximum de vraisemblance produira ces quantités. Nous donnerons des formules générales pour le AIC et le BIC dans le chapitre sur la régression logistique.

Le critère BIC est le seul de ces critères qui est convergent. Cela veut dire que si l'ensemble des modèles que l'on considère contient le vrai modèle, alors la probabilité que le critère BIC choisissent le bon modèle tend vers 1 lorsque  $n$  tend vers l'infini. Il faut mettre cela en perspective : il est peu vraisemblable que  $Y$  ait été généré exactement selon un modèle de régression linéaire, car le modèle de régression n'est qu'une approximation de la réalité. Certains auteurs trouvent que le BIC est quelquefois trop sévère (il choisit des modèles trop simples) pour les tailles d'échantillons finies. Dans certaines applications, cette parcimonie est utile, mais il n'est pas possible de savoir d'avance lequel de ces deux critères (AIC et BIC) sera préférable pour un problème donné.

Il est facile d'obtenir le AIC et BIC avec les méthodes AIC et BIC. On illustre ceci avec le modèle cubique :

```
data(polynome, package = "hecmulti")
# Ajuster un polynôme de degré trois (modèle cubique)
mod_cub <- lm(y ~ poly(x, 3),
                data = polynome)
summary(mod_cub) # Tableau résumé des coefficients
```

```

## 
## Call:
## lm(formula = y ~ poly(x, 3), data = polynome)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
## -111.91 -34.11   2.08  36.69 142.48 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) -9.77      5.22   -1.87   0.0645 .  
## poly(x, 3)1  824.81    52.24   15.79 < 2e-16 ***
## poly(x, 3)2 -281.95    52.24   -5.40  4.9e-07 ***
## poly(x, 3)3  160.99    52.24    3.08   0.0027 ** 
## --- 
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
## 
## Residual standard error: 52.2 on 96 degrees of freedom 
## Multiple R-squared:  0.75,   Adjusted R-squared:  0.742 
## F-statistic: 96 on 3 and 96 DF,  p-value: <2e-16 

AIC(mod_cub)
## [1] 1081
BIC(mod_cub)
## [1] 1094

```

Le Tableau 4.1 résume ces quantités pour tous les modèles de l'ordre 1 à l'ordre 10.

On voit dans le Tableau 4.1 que l'erreur quadratique moyenne des données d'apprentissage,  $\widehat{EQM}_a$ , diminue toujours à mesure qu'on ajoute des variables (c'est-à-dire, qu'on augmente l'ordre du polynôme); ces valeurs sont représentées dans la Figure 4.2. Les critères d'information, AIC et BIC, ne sont pas sur la même échelle, mais le graphique de la Figure 4.3 illustre un comportement semblable à la vraie courbe de l'erreur quadratique moyenne théorique et suggèrent que le meilleur modèle est le modèle cubique ( $k = 3$ ), c'est-à-dire le vrai modèle. N'oubliez pas que ces critères sont calculés avec l'échantillon d'apprentissage ( $n = 100$ ), mais en pénalisant l'ajout de variables. On est ainsi en mesure de contrecarrer le problème provenant du fait qu'on ne peut pas utiliser directement le  $\widehat{EQM}_a$ .

Le AIC et le BIC sont des critères très utilisés et très généraux. Ils sont disponibles dès qu'on utilise la méthode du maximum de vraisemblance comme méthode d'estimation.

TABLEAU 4.1 – Mesures de la qualité de l'ajustement d'un modèle polynomial aux données en fonction de l'ordre du polynôme.

	EQM	$\widehat{EQM}_a$	$R^2$	AIC	BIC	$VC_{10}$
1	3191	3674	0.65	1111	1119	3675
2	3133	2879	0.73	1088	1099	2898
3	2697	2620	0.75	1081	1094	2676
4	2767	2582	0.75	1081	1097	2666
5	2771	2581	0.75	1083	1102	2711
6	2780	2578	0.75	1085	1106	2757
7	2780	2577	0.75	1087	1111	2788
8	2797	2531	0.76	1087	1113	2846
9	2811	2528	0.76	1089	1118	2896
10	2849	2519	0.76	1091	1122	2976

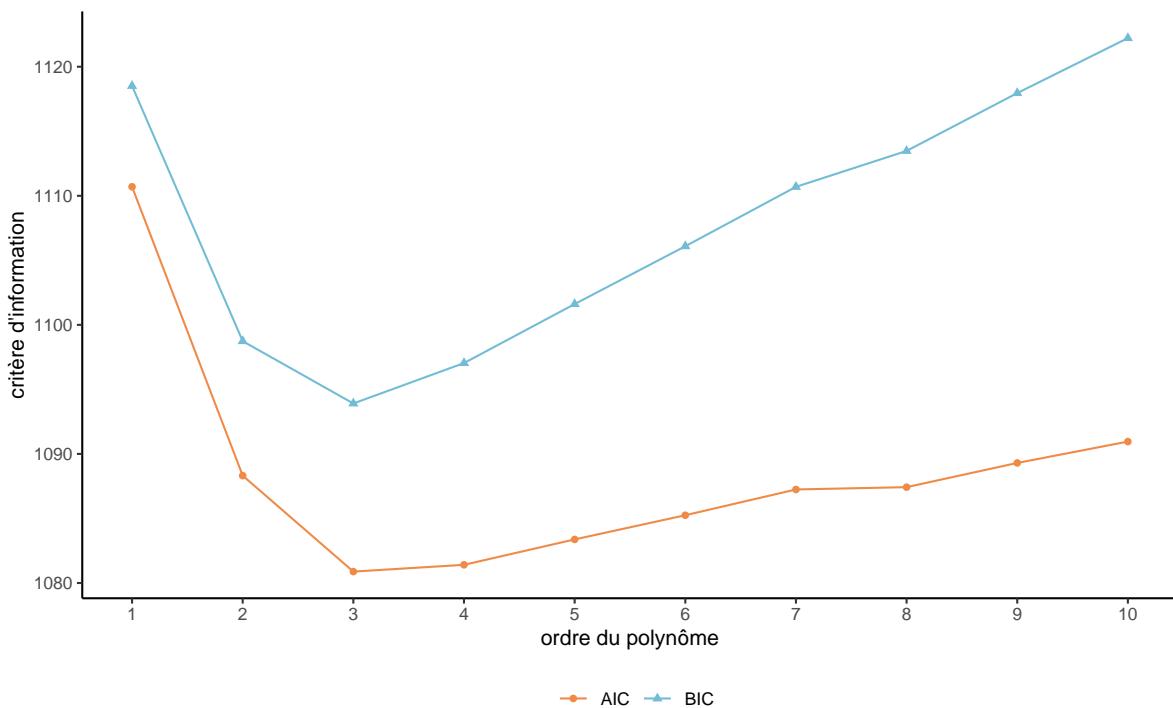


FIGURE 4.3 – Critères d'information en fonction de l'ordre du polynôme.

### 4.3.5 Validation externe

La deuxième grande approche après celle consistant à pénaliser le  $\widehat{\text{EQM}}_a$  consiste à tenter d'estimer le EQM directement sans utiliser deux fois les mêmes données. Nous allons voir deux telles méthodes ici, la validation externe (division de l'échantillon) et la validation croisée (*cross-validation*).

Ces deux méthodes s'attaquent directement au problème qu'on ne peut utiliser (sans ajustement) les mêmes données qui ont servi à estimer les paramètres d'un modèle pour estimer sa performance. Pour ce faire, l'échantillon de départ est divisé en deux, ou plusieurs parties, qui vont jouer des rôles différents.

L'idée de la validation externe est simple. Nous avons un échantillon de taille  $n$  que nous pouvons diviser au hasard en deux parties de tailles respectives  $n_1$  et  $n_2$  ( $n_1 + n_2 = n$ ), soit

- un échantillon d'apprentissage (*training*) de taille  $n_1$  et
- un échantillon de validation (*test*) de taille  $n_2$ .

L'échantillon d'apprentissage servira à estimer les paramètres du modèle. L'échantillon de validation servira à estimer la performance prédictive (par exemple estimer l'EQM) du modèle. Comme cet échantillon n'a pas servi à estimer le modèle lui-même, il est formé de « nouvelles » observations qui permettent d'évaluer d'une manière réaliste la performance du modèle. Comme il s'agit de nouvelles observations, on n'a pas à pénaliser la complexité du modèle et on peut directement utiliser le critère de performance choisi, par exemple, l'erreur quadratique moyenne, c'est-à-dire, la moyenne des erreurs au carré pour l'échantillon de validation. Cette quantité est une estimation valable de l'EQM de ce modèle. On peut faire la même chose pour tous les modèles en compétition et choisir celui qui a la meilleure performance sur l'échantillon de validation.

Cette approche possède plusieurs avantages. Elle est facile à implanter. Elle est encore plus générale que les critères AIC et BIC. En effet, ces critères découlent de la méthode d'estimation du maximum de vraisemblance. Plusieurs autres types de modèles ne sont pas estimés par la méthode du maximum de vraisemblance (par exemple, les arbres, les forêts aléatoires, les réseaux de neurones, etc.) La performance de ces modèles peut toujours être estimée en divisant l'échantillon. Cette méthode peut donc servir à comparer des modèles de familles différentes. Par exemple, choisit-on un modèle de régression linéaire, une forêt aléatoire ou bien un réseau de neurones?

Cette approche possède tout de même un désavantage. Elle nécessite une grande taille d'échantillon au départ. En effet, comme on divise l'échantillon, on doit en avoir assez pour bien estimer les paramètres du modèle (l'échantillon d'apprentissage) et assez pour bien estimer sa performance (l'échantillon de validation).

La méthode consistant à diviser l'échantillon en deux (apprentissage et validation) afin de sélectionner un modèle est valide. Par contre, si on veut une estimation sans biais de la performance du modèle choisi (celui qui est le meilleur sur l'échantillon de validation), on ne peut pas utiliser directement la valeur observée de l'erreur de ce modèle sur l'échantillon de validation car elle risque de sous-évaluer l'erreur. En effet, supposons qu'on a 10 échantillons et qu'on ajuste 10 fois

le même modèle séparément sur les 10 échantillons. Nous aurons alors 10 estimations différentes de l'erreur du modèle. Il est alors évident que de choisir la plus petite d'entre elles sous-estimerait la vraie erreur du modèle. C'est un peu ce qui se passe lorsqu'on choisit le modèle qui minimise l'erreur sur l'échantillon de validation. Le modèle lui-même est un bon choix, mais l'estimation de son erreur risque d'être sous-évaluée.

Une manière d'avoir une estimation de l'erreur du modèle retenu consiste à diviser l'échantillon de départ en trois (plutôt que deux). Aux échantillons d'apprentissage et de validation, s'ajoute un échantillon « test ». Cet échantillon est laissé de côté durant tout le processus de sélection du modèle qui est effectué avec les deux premiers échantillons tel qu'expliqué plus haut. Une fois un modèle retenu (par exemple celui qui minimise l'erreur sur l'échantillon de validation), on peut alors évaluer sa performance sur l'échantillon test qui n'a pas encore été utilisé jusque là. L'estimation de l'erreur du modèle retenu sera ainsi valide. Il est évident que pour procéder ainsi, on doit avoir une très grande taille d'échantillon au départ.

#### 4.3.6 Validation croisée

Si la taille d'échantillon n'est pas suffisante pour diviser l'échantillon en deux et procéder comme nous venons de l'expliquer, la validation croisée est une bonne alternative. Cette méthode permet d'imiter le processus de division de l'échantillon.

Voici les étapes à suivre pour faire une validation croisée à  $K$  groupes ( *$K$ -fold cross-validation*) :

1. Diviser l'échantillon au hasard en  $K$  parties  $P_1, P_2, \dots, P_K$  contenant toutes à peu près le même nombre d'observations.
2. Pour  $j = 1$  à  $K$ ,
  - i. Enlever la partie  $j$ .
  - ii. Estimer les paramètres du modèle en utilisant les observations des  $K - 1$  autres parties combinées.
  - iii. Calculer la mesure de performance (par exemple la somme du carré des erreurs) de ce modèle pour le groupe  $P_j$ .
3. Combiner les  $K$  estimations de performance pour obtenir une mesure de performance finale.<sup>1</sup>

Pour l'erreur quadratique moyenne, cette dernière étape revient à additionner la somme du carré des erreurs avant de diviser par la taille de l'échantillon totale.

On recommande habituellement de prendre  $K = \min\{n^{1/2}, 10\}$  groupes (le choix de cinq ou 10 groupes sont ceux qui reviennent le plus souvent en pratique). Si on prend  $K = 10$  groupes, alors

---

1. Le fait d'utiliser  $K \neq n$  mène à une estimation biaisée de la quantité d'intérêt et ce biais peut être important si  $n$  est petit; un ajustement simple est possible pour réduire ce dernier et présenté dans Davison et Hinkley (1997), *Bootstrap Methods and their Application*, Cambridge University Press à l'équation 6.48.

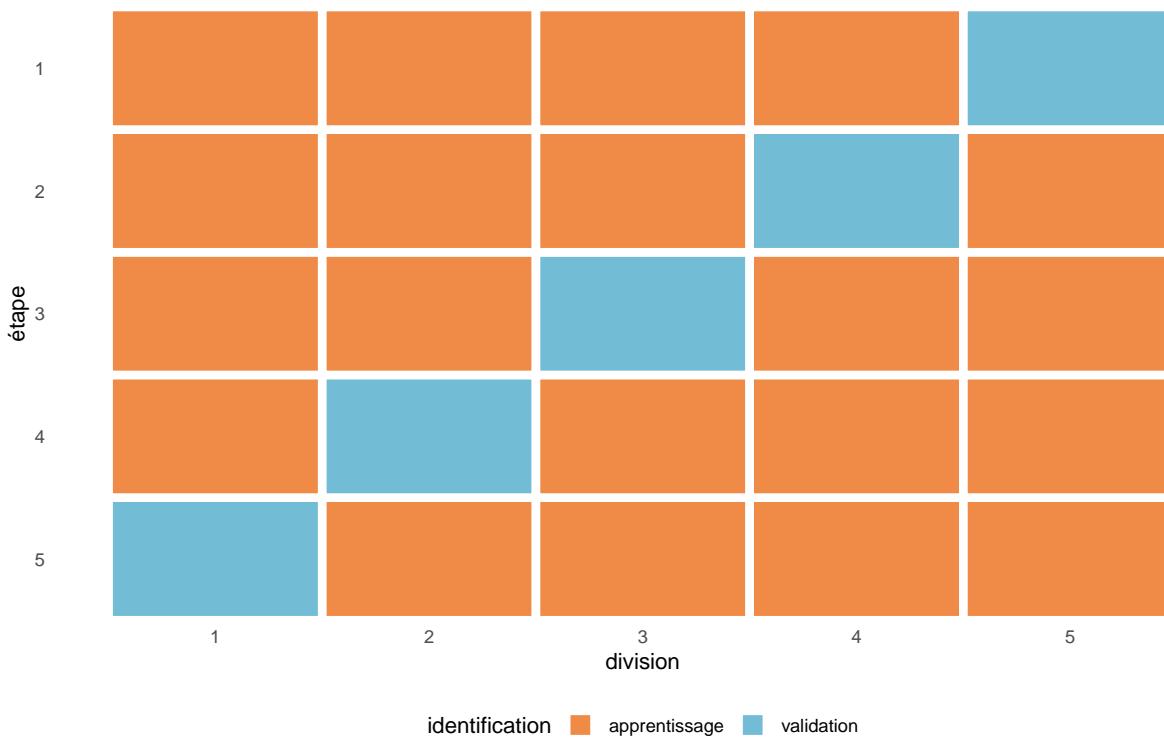


FIGURE 4.4 – Illustration de la validation croisée : on scinde l'échantillon d'apprentissage en cinq groupes (abscisse) et à chaque étape, une portion différente des données est mise de côté et ne sert que pour la validation.

chaque modèle est estimé avec 90% des données et on prédit ensuite le 10% restant. Comme on passe en boucle les 10 parties, chaque observation est prédite une et une seule fois à la fin. Il est important de souligner que les groupes sont formés de façon aléatoire et donc que l'estimé que l'on obtient peut être très variable, surtout si la taille de l'échantillon d'apprentissage est petite. Il arrive également que le modèle ajusté sur un groupe ne puisse pas être utilisé pour prédire les observations mises de côté, notamment si des variables catégorielles sont présentes mais qu'une modalité n'est présente que dans un des groupes ; ce problème se présente en pratique si certaines classes ont peu d'observations. Un échantillonnage stratifié permet de pallier à cette lacune et de s'assurer d'une répartition plus homogène des variables catégorielles.

```
# Validation croisée avec k groupes
lmkfold <- function(formula, data, k, ...){
  # Créer un accumulateur pour le calcul de l'EQM
  accu <- 0
```

```

k <- as.integer(k) # nombre de groupes
n <- nrow(data) # nombre d'observations
# Permuter les indices des observations
gp <- sample.int(n, n, replace = FALSE)
# Créer une liste de k éléments avec les nos d'observations
folds <- split(gp, cut(seq_along(gp), k, labels = FALSE))
for(i in seq_len(k)){
    # Extraire les indices des observations de la portion validation
    g <- as.integer(unlist(folds[i]))
    # Ajuster le modèles à toutes les données, moins celles de la portion validation
    fitlm <- lm(formula, data = data[-g,])
    # ajouter l'erreur quadratique du pli de validation
    accu <- accu + sum((data[g, all.vars(formula)[1]] - predict(fitlm, newdata=data[g,])))
}
# Diviser par la taille de l'échantillon
# pour obtenir la moyenne
return(accu/n)
}

# Le paquet 'caret' a une fonction
# pour faire la validation croisée
cv_caret <-
  caret::train(form = formula,
               data = data,
               method = "lm",
               trControl = caret::trainControl(
                 method = "cv",
                 number = 10))
eqm_cv <- cv_caret$results$RMSE^2

```

Le cas particulier  $K = n$  (en anglais *leave-one-out cross validation*, ou LOOCV) consiste à enlever une seule observation, à estimer le modèle avec les  $n - 1$  autres et à valider à l'aide de l'observation laissée de côté : on répète cette procédure pour chaque observation. Pour les modèles linéaires, il existe des formules explicites qui nous permettent d'éviter d'ajuster  $n$  régressions par moindre carrés. Cette forme de validation croisée tend à être trop optimiste.

Revenons à notre exemple où une seule variable explicative est disponible et où l'on cherche à déterminer un bon modèle polynomial. La dernière colonne de Tableau 4.1,  $VC_{10}$ , donne les moyennes de 100 répliques de estimations de l'EQM obtenues avec la validation croisée à 10 groupes. Notez que si vous exécutez le programme, vous n'obtiendrez pas les mêmes valeurs car il y a un élément aléatoire dans ce processus.

Le modèle cubique (ordre 3) est aussi choisi par la validation croisée, en moyenne (comme il l'était par le AIC et le BIC). Le graphe qui suit trace les valeurs de l'estimation par validation croisée (courbe de validation croisée) et aussi le EQM. On voit que l'estimation par validation croisée suit assez bien la forme du EQM (qu'il est supposé estimer). Les boîtes à moustache permettent d'apprécier la variabilité des estimés de l'erreur quadratique moyenne telles qu'estimée par validation croisée avec 10 groupes.

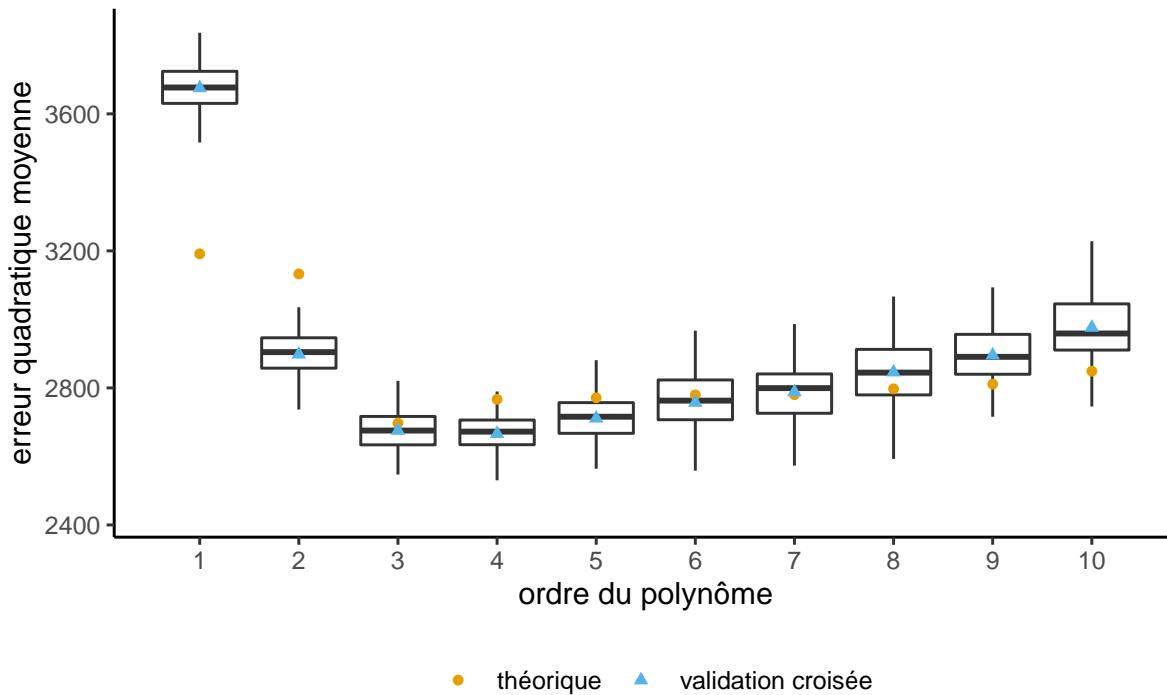


FIGURE 4.5 – Boîtes-à-moustaches des 100 réplications des valeurs de l'erreur quadratique moyenne estimées par validation croisée à 10 plis pour chaque ordre du polynôme.

#### 4.4 Présentation des données

Nous allons présenter un exemple classique de commercialisation de bases de données qui nous servira à illustrer la sélection de modèles, la régression logistique et la gestion de données manquantes. Le but est de cibler les clients pour l'envoi d'un catalogue.

Le contexte est le suivant : une entreprise possède une grande base de données client. Elle désire envoyer un catalogue à ses clients mais souhaite maximiser les revenus d'une telle initiative. Il est évidemment possible d'envoyer le catalogue à tous les clients mais ce n'est probablement pas optimal. La stratégie envisagée est la suivante :

1. Envoyer le catalogue à un échantillon de clients et attendre les réponses. Le coût de l'envoi d'un catalogue est de 10\$.
  2. Construire un modèle avec cet échantillon afin de décider à quels clients (parmi les autres) le catalogue devrait être envoyé, afin de maximiser les revenus.

Plus précisément, on s'intéresse aux clients de 18 ans et plus qui ont au moins un an d'historique avec l'entreprise et qui ont effectué au moins un achat au cours de la dernière année. Dans un premier lieu, on a envoyé le catalogue à un échantillon de 1000 clients. Un modèle sera construit avec ces 1000 clients afin de cibler lesquels des clients restants seront choisis pour recevoir le catalogue.

Pour les 1000 clients de l'échantillon d'apprentissage, les deux variables cibles suivantes sont disponibles :

- *yachat*, une variable binaire qui indique si le client a acheté quelque chose dans le catalogue égale à 1 si oui et 0 sinon.
  - *ymontant*, le montant de l'achat si le client a acheté quelque chose.

Les 10 variables suivantes sont disponibles pour tous les clients et serviront de variables explicatives pour les deux variables cibles. Il s'agit de :

- x1 : sexe de l'individu, soit homme (0) ou femme (1);
  - x2 : l'âge (en année);
  - x3 : variable catégorielle indiquant le revenu, soit moins de 35 000\$ (1), entre 35 000\$ et 75 000\$ (2) ou plus de 75 000\$ (3);
  - x4 : variable catégorielle indiquant la région où habite le client (de 1 à 5);
  - x5 : couple : la personne est elle en couple (0=non, 1=oui);
  - x6 : nombre d'année depuis que le client est avec la compagnie;
  - x7 : nombre de semaines depuis le dernier achat;
  - x8 : montant (en dollars) du dernier achat;
  - x9 : montant total (en dollars) dépensé depuis un an;
  - x10 : nombre d'achats différents depuis un an.

Les données se trouvent dans le fichier dbm. Voici d'abord des statistiques descriptives pour l'échantillon d'apprentissage.

TABLEAU 4.2 – Tableaux de fréquence pour les variables catégorielles de la base de données marketing.

sexe	décompte	revenu	décompte	couple	décompte	région	décompte
		1	397			1	216
0	534	2	337	0	575	2	185
1	466	3	266	1	425	3	216
						4	191
						5	192

TABLEAU 4.3 – Statistiques descriptives des variables numériques de la base de données marketing.

variable	description	moyenne	écart-type	min	max
x2	âge	37.06	9.27	20	70
x6	nombre d'année comme client	6.01	2.92	1	11
x7	nombre de semaines depuis le dernier achat	9.97	9.34	1	52
x8	montant du dernier achat	48.41	28.27	20	252
x9	montant total dépensé sur un an	229.27	173.97	22	1407
x10	nombre d'achats différents sur un an	5.64	2.31	1	14

```

## $ x6      : num [1:101000] 8.6 8.6 1.4 10.7 9.1 9.4 10.6 4.8 4 10.3 ...
## $ x7      : num [1:101000] 8 9 9 42 5 1 6 5 48 9 ...
## $ x8      : num [1:101000] 49 70 120 31 30 28 59 70 73 55 ...
## $ x9      : num [1:101000] 159 123 434 110 55 102 593 298 83 90 ...
## $ x10     : num [1:101000] 5 5 8 3 3 8 10 6 2 3 ...
## $ yachat  : int [1:101000] 0 0 0 0 0 0 1 1 1 ...
## $ ymontant: num [1:101000] NA NA NA NA NA NA 52 79 77 ...
## $ test    : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 ...
```

Il y a 46.6% de femmes parmi les 1000 clients de l'échantillon. De plus, 39.7% ont un revenu de moins de 35 000\$, 33.7% sont entre 35 000\$ et 75 000\$ et 26.6% ont plus de 75 000\$. 42.5% de ces clients qui ont un conjoint.

Le nombre d'achats différents depuis un an par ces clients varie entre 1 et 14. Un peu plus de la moitié (51.4%) ont fait cinq achats ou moins. Parmi les 1000 clients de l'échantillon d'apprentissage, 210 ont acheté quelque chose dans le catalogue. La variable yachat sera l'une des variables que nous allons chercher à modéliser en vue d'obtenir des prédictions.

L'âge des 1000 clients de l'échantillon d'apprentissage varie entre 20 et 70 avec une moyenne de 37.1 ans. En moyenne, ces clients ont acheté pour 229.30\$ depuis un an. Le dernier achat de ces

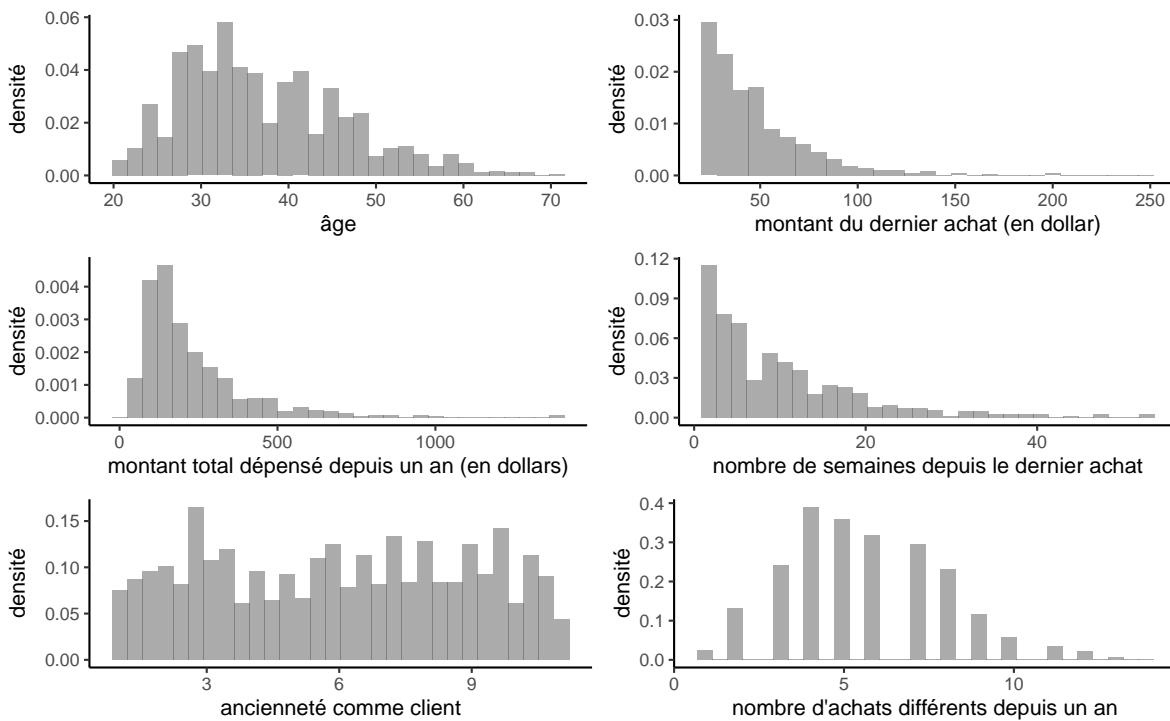


FIGURE 4.6 – Histogrammes des variables continues de la base de données dbm pour les 1000 clients par intention d'achat.

clients remonte, en moyenne, à 10 semaines.

Dans cette section, nous modéliserons le montant d'achat,  $ymontant$ . Seuls 210 clients ont acheté quelque chose dans le catalogue et les statistiques rapportées correspondent seulement à ces derniers, car la variable  $ymontant$  est manquante si le client n'a rien acheté dans le catalogue. On pourrait également remplacer ces valeurs par des zéros et les modéliser, mais nous aborderons cet aspect ultérieurement. Les clients qui ont acheté quelque chose ont dépensé en moyenne 67.3\$, et au minimum 25\$. La Figure 4.6 présente les histogrammes de quelques unes de ces variables.

Il y a plusieurs façons d'utiliser l'échantillon d'apprentissage afin de mieux cibler les clients à qui envoyer le catalogue et maximiser les revenus. En voici quelques unes.

- On pourrait développer un modèle afin d'estimer la probabilité qu'un client achète quelque chose si on lui envoie un catalogue. Plus précisément, on peut développer un modèle pour  $\text{Pr}(yachat = 1)$ . Comme la variable  $yachat$  est binaire, un modèle possible est la régression logistique, que nous décrirons au chapitre suivant. Ainsi, en appliquant le modèle aux 100 000 clients restant, on pourra cibler les clients susceptibles d'acheter (ceux avec une probabilité élevée).

- b) Une autre façon serait de tenter de prévoir le montant d'argent dépensé. Nous venons de voir la distribution de la variable  $y_{montant}$ . Il y a deux situations, ceux qui ont acheté et ceux qui n'ont pas achetés. En conditionnant sur le fait d'avoir acheté quelque chose, il est possible de décomposer le problème de la manière suivante :

$$\begin{aligned} E(y_{montant}) &= E(y_{montant} | y_{achat} = 1)P(y_{achat} = 1) \\ &\quad + E(y_{montant} | y_{achat} = 0)P(y_{achat} = 0) \\ &= E(y_{montant} | y_{achat} = 1)P(y_{achat} = 1), \end{aligned}$$

puisque le terme  $E(y_{montant} | y_{achat} = 0)$  est zéro : les gens qui n'ont pas acheté n'ont rien dépensé.

On peut donc estimer  $E(y_{montant} | y_{achat} = 1)$  et  $P(y_{achat} = 1)$ , pour ensuite les combiner et avoir une estimation de  $E(y_{montant})$ . Le développement du modèle pour  $E(y_{montant} | y_{achat} = 1)$  peut se faire avec la régression linéaire, en utilisant seulement les clients qui ont acheté dans l'échantillon d'apprentissage, car  $y_{montant}$  est une variable continue dans ce cas. Le développement du modèle pour  $P(y_{achat} = 1)$  peut se faire avec la régression logistique, tel que mentionné plus haut, en utilisant tous les 1000 clients de l'échantillon d'apprentissage. En fait, nous verrons plus loin qu'il est possible d'estimer conjointement les deux modèles avec un modèle Tobit. En appliquant le modèle aux 100 000 clients restants, on pourra cibler les clients qui risquent de dépenser un assez grand montant.

Comme nous n'avons pas encore vu la régression logistique, nous allons nous limiter à illustrer les méthodes qui restent à voir dans ce chapitre avec la régression linéaire en cherchant à développer un modèle pour  $E(y_{montant} | y_{achat} = 1)$ , le montant d'argent dépensé par les clients qui ont acheté quelque chose.

La base de donnée contient deux variables explicatives catégorielles. Il s'agit de revenu ( $x_3$ ) et région ( $x_4$ ). Il faut coder d'une manière appropriée afin de pouvoir les incorporer dans les modèles. La manière habituelle est de créer des variables indicatrices (binaires) qui indiquent si la variable prend ou non une valeur particulière dans **R** est de transformer la variable en facteur (**factor**). En général, si une variable catégorielle possède  $K$  valeurs possibles, il est suffisant de créer  $K - 1$  indicatrices, en laissant une modalité comme référence. Par exemple, pour  $x_3$ , nous allons créer deux variables,

- $x_{31}$  : variable binaire égale à 1 si  $x_3$  égale 1 et 0 sinon,
- $x_{32}$  : variable binaire égale à 1 si  $x_3$  égale 2 et 0 sinon.

Ainsi, la valeur 3 est celle de référence. Ces deux indicatrices sont suffisantes pour récupérer toute l'information comme le démontre le Tableau 4.4.

TABLEAU 4.4 – Valeur des indicateurs en fonction du niveau de la variable catégorielle

x3	x31	x32
1	1	0
2	0	1
3	0	0

Il est important de noter que, si le modèle qui inclut toutes les modalités (ordonnée à l'origine, x31 et x32) possibles ne dépend pas de la catégorie de référence, ce ne sera plus le cas si on permet lors de la sélection de variables de ne conserver que certains niveaux de la variable catégorielle. Par exemple, si on inclut uniquement x31 comme variable explicative, l'ordonnée à l'origine englobera toutes les autres valeurs de x3, à savoir {2, 3}.<sup>2</sup>

#### ⚠ Danger de surajustement avec variables catégorielles

La principale cause de mauvaise performance est le surajustement sélectif. Dans l'exemple que l'on considère avec la base de données marketing, la plupart des modalités des variables catégorielles semblent à première vues suffisantes pour estimer des coefficients. Si on s'intéresse par contre aux interactions, on se rendra rapidement compte qu'il y a trop peu de valeurs pour certaines combinaisons (par exemple, x3\*x5) pour estimer de manière fiable l'effet combiné. Si on a une valeur aberrante dans un groupe avec de faibles modalités, les indicateurs donneront systématiquement préférence à l'inclusion d'un terme pour l'accommoder (au détriment de la généralisation). Cela a pour effet de fausser la sélection et donner une grande erreur quadratique moyenne de validation. Si certaines modalités ont des effectifs trop petits, on peut envisager de les regrouper avec d'autres similaires.

## 4.5 Sélection de variables

### 4.5.1 Recherche exhaustive (meilleurs sous-ensembles)

Lorsque nous voulons comparer un petit nombre de modèles, il est relativement aisé d'obtenir les critères (AIC, BIC ou autre) pour tous les modèles et de choisir le meilleur. C'était le cas dans l'exemple du choix de l'ordre du polynôme où il y avait seulement 10 modèles en compétitions. Mais lorsqu'il y a plusieurs variables en jeu, le nombre de modèles potentiel augmente très rapidement.

En fait, supposons qu'on a  $p$  variables distinctes disponibles. Avant même de considérer les transformations des variables et les interactions entre elles, il y a déjà trop de modèles possibles. En

2. La fonction MASS::stepAIC ne segmente pas les variables catégorielles : tous les niveaux sont inclus à la fois. La fonction leaps::regsubsets va quant à elle créer des indicateurs binaires.

TABLEAU 4.5 – Nombres de modèles en fonction du nombre de paramètres.

$p$	nombre de paramètres
5	32
10	1024
15	32768
20	1048576
25	33554432
30	1073741824

effet, chaque variable est soit incluse ou pas (deux possibilités) et donc il y a  $2^p = 2 \times 2 \times \dots \times 2$  ( $p$  fois) modèles en tout à considérer. Ce nombre augmente très rapidement comme en témoigne le Tableau 4.5.

Ainsi, si le nombre de variables est restreint, il est possible de comparer tous les modèles potentiels et de choisir le meilleur (selon un critère). Il existe même des algorithmes très efficaces qui permettent de trouver le meilleur modèle sans devoir examiner tous les modèles possibles. Le nombre de variables qu'il est possible d'avoir dépend de la puissance de calcul et augmente d'année en année. Par contre, dans plusieurs applications, il ne sera pas possible de comparer tous les modèles et il faudra effectuer une recherche limitée. Faire une recherche exhaustive parmi tous les modèles possibles s'appelle sélection de tous les sous-ensembles (*best subsets*).

On veut trouver un bon modèle pour prévoir la valeur de l'montant des clients qui ont acheté quelque chose. On a vu qu'il y a 210 clients qui ont acheté dans l'échantillon d'apprentissage. Nous allons chercher à développer un « bon » modèle avec ces 210 clients. Dans ce premier exemple, nous allons seulement utiliser les 10 variables explicatives de base (14 variables avec les indicatrices).

Pour un nombre de variables fixé, le meilleur modèle selon le  $R^2$  est aussi le meilleur selon les critères d'information AIC et BIC, pour ce nombre fixé de variables. Pour vous convaincre de cette affirmation, fixons le nombre de variables et restreignons-nous seulement aux modèles avec ce nombre de variables. Comme  $R^2 = 1 - \text{SCE}/\text{SCT}$  et que SCT est une constante indépendante du modèle, le modèle avec le plus grand coefficient de détermination,  $R^2$ , est aussi celui avec la plus petite somme du carré des erreurs (SCE). Comme  $\text{AIC} = n \ln(\text{EQM}) + 2p$ , ce sera aussi celui avec le plus petit AIC car la pénalité  $2p$  est la même si on fixe le nombre de variables; la même remarque est valide pour le BIC.

Ainsi, pour trouver le meilleur modèle globalement (sans fixer le nombre de variables), il suffit de trouver le modèle à  $k$  variables explicatives ayant le coefficient de détermination le plus élevé pour tous les nombres de variables fixés et d'ensuite de trouver celui qui minimise le AIC (ou le BIC) parmi ces modèles. Ainsi, le modèle linéaire simple qui a le plus grand  $R^2$  est celui qui inclut l'indicateur de couple (x5). Le meilleur modèle (selon le  $R^2$ ) parmi tous les modèles avec deux variables est celui avec x5 et x6.

TABLEAU 4.6 – Modèle parmi les candidats ayant la plus grande valeur de coefficient de détermination selon le nombre de régresseurs, avec valeurs des critères d'informations.

variables	BIC	AIC
x5	-96	-103
x5 x6	-196	-206
x31 x5 x6	-311	-324
x31 x5 x6 x10	-351	-368
x1 x31 x5 x6 x10	-369	-389
x1 x31 x5 x6 x7 x10	-387	-411
x1 x31 x5 x6 x7 x8 x10	-392	-419
x1 x31 x44 x5 x6 x7 x8 x10	-391	-421
x1 x2 x31 x44 x5 x6 x7 x8 x10	-390	-424
x1 x2 x31 x44 x5 x6 x7 x8 x9 x10	-387	-424
x1 x2 x31 x43 x44 x5 x6 x7 x8 x9 x10	-383	-423
x1 x2 x31 x41 x42 x43 x44 x5 x6 x7 x8 x10	-378	-422
x1 x2 x31 x41 x42 x43 x44 x5 x6 x7 x8 x9 x10	-375	-422

Un algorithme par séparation et évaluation permet d'effectuer cette recherche de manière efficace sans essayer tous les candidats pour ces sous-ensembles. Dans l'exemple, on voit que le modèle avec les variables x1 x2 x31 x44 x5 x6 x7 x8 x9 et x10 est celui qui minimise le AIC globalement (AIC de -423.754). Le modèle choisi par le BIC contient seulement sept variables explicatives (plutôt que 10), soit x1 x31 x5 x6 x7 x8 x10.

```

data(dbm, package = "hecmulti")
dbm_a <- dbm |>
  dplyr::filter(test == 0,
                !is.na(ymontant))
# Conserver données d'entraînement (test == 0)
# des personnes qui ont acheté ymontant > 0

rec_ex <- leaps::regsubsets(
  x = ymontant ~ x1+x2+x3+x4+x5+x6+x7+x8+x9+x10,
  nvmax = 13L,
  method = "exhaustive",
  data = dbm_a)
resume_rec_ex <- summary(rec_ex,
                           matrix.logical = TRUE)

```

```
# Trouver le modèle avec le plus petit BIC
min_BIC <- which.min(resume_rec_ex$bic)
# Nom des variables dans le modèle retenu
rec_ex$xnames[resume_rec_ex$which[min_BIC,]]
# Coefficients
# coef(rec_ex, id = min_BIC)
```

Nous avons seulement inclus les variables de base pour ce premier essai. Il est possible qu'ajouter des variables supplémentaires améliore la performance du modèle. Pour cet exemple, nous allons considérer les variables suivantes<sup>3</sup> :

- les variables continues au carré, comme `age`<sup>2</sup>.
- toutes les interactions d'ordre deux entre les variables de base, comme `sexe · age`.

Aux variables de base (10 variables explicatives, mais 14 avec les indicatrices pour les variables catégorielles), s'ajoutent ainsi 90 autres variables. Il y a donc 104 variables explicatives potentielles si on inclut les interactions et les termes quadratiques. Notez qu'il y a des interactions entre chacune des variables indicatrices et chacune des autres variables, mais il ne sert à rien de calculer une interaction entre deux indicatrices d'une même variable (car une telle variable est zéro pour tous les individus). De même, il ne sert à rien de calculer le carré d'une variable binaire codée {0, 1}.

Dans la mesure où on aura un ratio d'environ un paramètre pour deux observations, Le modèle à 104 variables servira uniquement à illustrer le surajustement. Pensez à la taille de votre échantillon comme à un budget et aux paramètres comme à un nombre d'items : plus vous achetez d'items, moins votre budget est élevé pour chacun et leur qualité en pâtit. Réalistement, un modèle avec plus d'une vingtaine de variables ici serait difficilement estimable de manière fiable et l'inclusion d'interactions et de termes quadratiques sert surtout à augmenter la flexibilité et les possibilités lors de la sélection de variables.

```
# (...)^2 crée toutes les interactions d'ordre deux
# I(x^2) permet de créer les termes quadratiques
formule <-
  formula(ymontant ~
    (x1 + x2 + x3 + x4 + x5 +
     x6 + x7 + x8 + x9 + x10)^2 +
    I(x2^2) + I(x6^2) + I(x7^2) +
    I(x8^2) + I(x9^2) + I(x10^2))
mod_complet <- lm(formule, data = dbm_a)
```

---

3. Pourquoi prendre ces variables en particulier? Si on suppose que la vrai moyenne de la variable réponse `ymontant` centrée et réduite est une fonction lisse inconnue et qu'on utilise les bonnes variables explicatives centrées, le modèle ajusté précédemment capture l'approximation de degré 1 (série de Taylor) de la vraie fonction de moyenne, tandis que le modèle avec termes quadratiques (incluant les interactions) représente l'approximation de degré 2.

Lancer une sélection exhaustive de tous les sous-modèles avec 104 variables risque de prendre un temps énorme. Que faire alors? Il y a plusieurs possibilités. Nous pourrions faire une recherche limitée avec les méthodes que nous allons voir à partir de la section suivante. Nous pourrions aussi combiner les deux approches. Supposons que notre ordinateur permet de faire une recherche exhaustive de tous les sous-modèles avec 40 variables. Nous pourrions alors commencer avec une recherche limitée pour trouver un sous-ensemble de 40 « bonnes » variables et faire une recherche exhaustive, mais en se restringant à ces 40 variables.

#### 4.5.2 Méthodes séquentielles de sélection

Les méthodes de sélection ascendante, descendante et séquentielle sont des algorithmes gloutons. Elles ont été développées à une époque où la puissance de calcul était bien moindre, et où il était impossible de faire une recherche exhaustive des sous-modèles. La procédure `leaps::regsubsets` permet une sélection de modèle avec une approche séquentielle, ascendante ou descendante en choisissant le meilleur modèle (côté ajustement) avec  $k$  variables ( $k = 1, \dots, k_{\max}$ ). La procédure `MASS::stepAIC` permet de faire cette sélection en utilisant un critère d'information.

L'idée de la **sélection ascendante** est d'ajouter à chaque étape au modèle précédent la variable qui améliore le plus l'ajustement. Le modèle de départ est celui qui n'inclut que l'ordonnée à l'origine (aucune variable explicative). À chaque étape, on ajoute la variable qui améliore le plus le critère d'ajustement jusqu'à ce qu'aucune amélioration ne soit résultante.

Un algorithme glouton résoud un problème d'optimisation étape par étape : après  $k$  étapes, le modèle construit par la procédure n'est pas nécessairement le meilleur modèle (si on essayait toutes les combinaisons). Si on commence avec  $p$  variables, on regarde  $p$  choix à la première étape de la procédure ascendante, puis on choisit une variable parmi les  $p - 1$  restantes à la deuxième étape, etc. La procédure exhaustive essaiera toutes les  $\binom{p}{k}$  combinaisons possibles<sup>4</sup> : puisque plus de modèles sont essayés, la solution finale est nécessairement meilleure côté performance évaluée sur l'échantillon d'apprentissage.

La **sélection descendante** est similaire, sauf qu'on part avec le modèle qui inclut toutes les variables explicatives. À chaque étape, on retire la variable qui contribue le moins à l'ajustement jusqu'à ce que le critère d'ajustement ne puisse plus être amélioré ou jusqu'à ce qu'on recouvre le modèle sans variables explicatives, selon le scénario. C'est l'inverse de la méthode ascendante : on va tester le retrait de chaque variable individuellement et retirer celle qui est la moins significative.

La **méthode de sélection séquentielle** est un hybride entre les méthodes de sélection ascendantes et descendante. On débute la recherche à partir du modèle ne contenant que l'ordonnée à l'origine. À chaque étape, on fait une étape ascendante suivie de une (ou plusieurs) étapes descendantes. On

---

4. En pratique, il existe des algorithmes d'optimisation qui permettront de faire cette exploration de manière astucieuse sans ajuster les modèles sous-optimaux.

continue ainsi tant que le modèle retourné par l'algorithme n'est pas identique à celui de l'étape précédente (dépendant de notre critère). Le dernier modèle est celui retenu.

Avec la méthode séquentielle, une fois qu'on entre une variable (étape ascendante), on fait autant d'étapes descendante afin de retirer toutes les variables qui satisfont le critère de sortie (il peut ne pas y en avoir). Une fois cela effectué, on refait une étape ascendante pour voir si on peut ajouter une nouvelle variable.

Avec la méthode ascendante, une fois qu'une variable est dans le modèle, elle y reste. Avec la méthode descendante, une fois qu'une variable est sortie du modèle, elle ne peut plus y entrer. Avec la méthode séquentielle, une variable peut entrer dans le modèle et sortir plus tard dans le processus. Par conséquent, parmi les trois, la méthode séquentielle est généralement préférable aux méthodes ascendante et descendante, car elle inspecte potentiellement un plus grand nombre de modèles.

```
# Cette procédure séquentielle retourne
# la liste de modèles de 1 variables à
# nvmax variables.
rec_seq <-
  leaps::regsubsets(
    x = formule,
    data = dbm_a,
    method = "seqrep",
    nvmax = length(coef(mod_complet)))
which.min(summary(rec_seq)$bic)

# Alternative avec procédure séquentielle
# qui utilise le critère AIC pour déterminer
# l'inclusion ou l'exclusion de variables
#
# Procédure plus longue à rouler
# (car les modèles linéaires sont ajustés)
#
# On ajoute ou retire la variable qui
# améliore le plus le critère de sélection
# à chaque étape.
seq_AIC <- MASS::stepAIC(
  lm(ymontant ~ 1, data = dbm_a),
  # modèle initial sans variables explicative
  scope = formule, # modèle maximal possible
  direction = "both", #séquentielle
```

```

trace = FALSE, # ne pas imprimer le suivi
keep = function(mod, AIC, ...){
  # autres sorties des modèles à conserver
  list(bic = BIC(mod),
       coef = coef(mod)),
  k = 2) #
# Remplacer k=2 par k = log(nrow(dbm_a)) pour BIC

# L'historique des étapes est disponible via
# seq_AIC$anova

```

La procédure exhaustive est préférable aux méthodes séquentielles si le nombre de variables n'est pas trop élevé. S'il y a trop de variables, rien ne nous empêche de combiner plusieurs méthodes : on pourrait par exemple faire une procédure descendante pour ne conserver que 40 variables. En utilisant seulement ce sous-ensemble de variables, on choisit le meilleur modèle selon le AIC ou le BIC en faisant une recherche exhaustive de tous les sous-modèles. On pourrait également faire une recherche séquentielle avec le AIC et choisir le modèle parmi l'historique avec le plus petit BIC.

On peut voir sur la Figure 4.7 l'historique des valeurs de AIC et BIC à mesure qu'on augmente le nombre de variables dans le modèle obtenu par une procédure séquentielle : les mêmes variables sont enlevées à chaque étape, mais la valeur optimale du critère est différente pour la sélection finale. Sur l'axe des abscisses, j'ai ajouté l'erreur quadratique moyenne de l'échantillon de validation pour les clients avec ymontant positif. Cet exemple n'est pas réaliste puisqu'on regarde la solution, mais il permet de nous comparer et de voir à quel point ici le critère d'information bayésien suit la même tendance que l'erreur quadratique moyenne de validation. L'erreur quadratique moyenne obtenue par validation croisée est trop optimiste (mais aléatoire!), comme le AIC. Pour éviter le surentraînement dans une région où le critère est quasi constant, on peut utiliser la règle d'une erreur-type. Puisque on a plusieurs réplications, on peut estimer ce dernier avec la validation croisée en même temps que l'EQM et choisir le modèle le plus simple à distance une erreur-type du modèle avec la plus petite erreur de validation croisée.

#### 4.5.3 Méthodes de régression avec régularisation

Une façon d'éviter le surajustement est d'ajouter une pénalité sur les coefficients : ce faisant, on introduit un biais dans nos estimés, mais dans l'espoir de réduire leur variabilité et ainsi d'obtenir une meilleure erreur quadratique moyenne.

L'avantage des moindres carrés est que les valeurs ajustées et les prédictions ne changent pas si on fait une transformation affine (de type  $Z = aX + b$ ). Peu importe le choix d'unité (par exemple, exprimer une distance en centimètres plutôt qu'en mètres, ou la température en Fahrenheit plutôt qu'en Celcius), on obtient le même ajustement. En revanche, une fois qu'on introduit un terme de

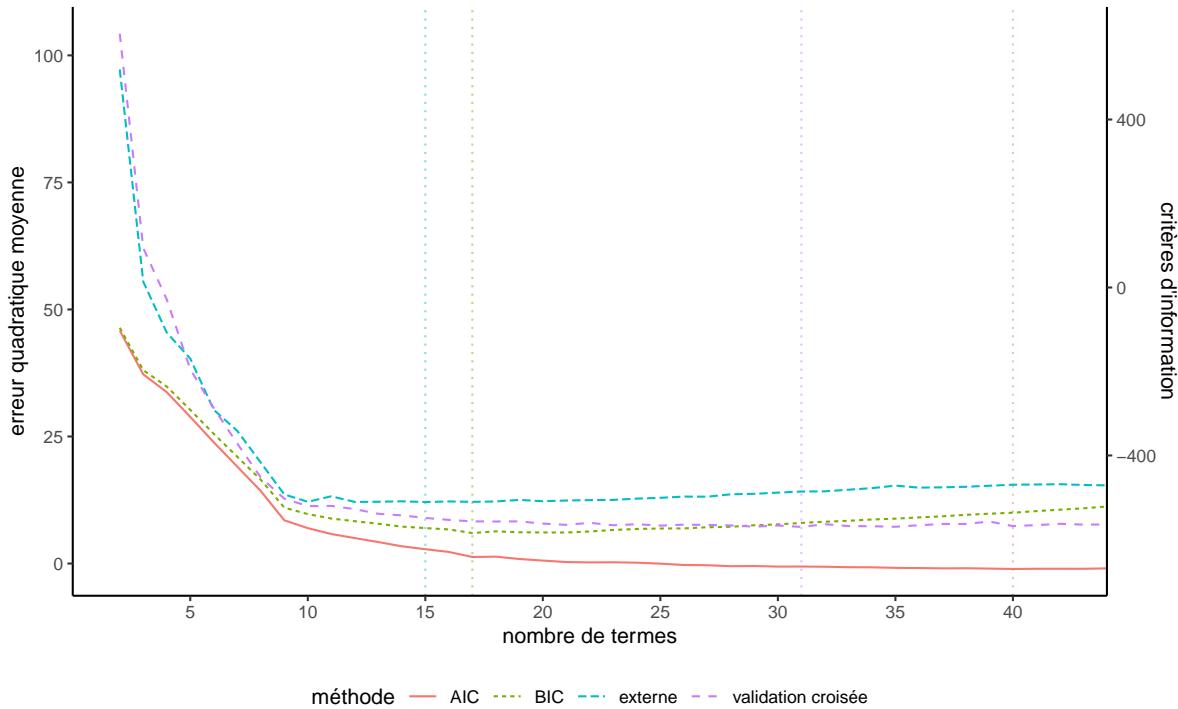


FIGURE 4.7 – Critères d’information et estimation de l’erreur quadratique moyenne de validation externe et de validation croisée (10 groupes) pour les 40 premiers modèles de la procédure descendante, selon le nombre de termes inclus dans la régression linéaire. Les traitillés verticaux indiquent le nombre de terme du modèle avec la meilleure valeur du critère pour chaque méthode.

pénalité, notre solution dépendra de l’unité de mesure, d’où l’importance d’utiliser les données centrées et réduites pour que la solution reste la même.

Les estimateurs des moindres carrés ordinaires pour la régression linéaire représentent la combinaison qui minimise la somme du carré des erreurs,

$$SCE = \sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p X_{ij} \beta_j \right)^2.$$

On peut ajouter à cette fonction objective SCE un terme additionnel de pénalité qui va contraindre les paramètres à ne pas être trop grand. On considère une pénalité additionnelle pour la valeur absolue des coefficients,

$$q_1(\lambda) = \lambda \sum_{j=1}^p |\beta_j|.$$

Pour chaque valeur de  $\lambda$  donnée, on obtiendra une solution différente pour les estimés car on minimisera désormais  $SCE + q_1(\lambda)$ . On ne pénalise pas l’ordonnée à l’origine  $\beta_0$ , parce que ce

coefficient sert à recentrer les observations et a une signification particulière : si on standardise les données, de manière à ce que leur moyenne empirique soit zéro et leur écart-type un, alors  $\hat{\beta}_0 = \bar{y}$ .

La pénalité  $q_1(\lambda)$  a un rôle particulier parce qu'elle a deux effets : elle réduit la taille des paramètres, mais elle force également certains paramètres très proches de zéro à être exactement égaux à zéro, ce qui fait que la régression pénalité agit également comme outil de sélection de variables. Des algorithmes efficaces permettent de trouver la solution du problème d'optimisation

$$\min_{\boldsymbol{\beta}} \{SCE + q_1(\lambda)\} = \min_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^n \left( y_i - \beta_0 - \sum_{j=1}^p X_{ij} \beta_j \right)^2 + \lambda \sum_{j=1}^p |\beta_j| \right\}$$

laquelle est appelée LASSO. La Figure 4.8 montre la fonction objective dans le cas où on a deux paramètres,  $\beta_1$  et  $\beta_2$ . La solution des moindres carrés ordinaires, qui minimisent l'erreur quadratique moyenne, est au centre des ellipses de contour et correspond à la solution du modèle avec  $\lambda = 0$ . À mesure que l'on augmente la pénalité  $\lambda$ , les coefficients rétrécissent vers  $(0, 0)$ . On peut interpréter la pénalité  $l_1$  comme une contraire budgétaire : les coefficients estimés pour une valeur de  $\lambda$  donnée sont ceux qui minimisent la somme du carré des erreurs, mais doivent être à l'intérieur d'un budget alloué (losange). La forme de la région fait en sorte que la solution, qui se trouve sur la bordure du losange, intervient dans un coin avec certaines coordonnées nulles.

Plusieurs variantes existent dans la littérature qui généralisent le modèle à des contextes plus compliqués. Le choix des variables à inclure dans la sélection dépend du choix de la pénalité  $\lambda$ , qui est règle générale estimée par validation croisée à cinq ou 10 groupes.

```
# Sélection par LASSO
library(glmnet)
# Paramètre de pénalité déterminé par
# validation croisée à partir d'un vecteur
# de valeurs candidates
lambda_seq <- seq(from = 0.1,
                   to = 10,
                   by = 0.01)
cv_output <-
  glmnet::cv.glmnet(x = as.matrix(dbm_a[, 1:10]),
                     y = dbm_a$ymontant,
                     alpha = 1,
                     lambda = lambda_seq)
plot(cv_output)

# On réestime le modèle avec la pénalité
lasso_best <-
```

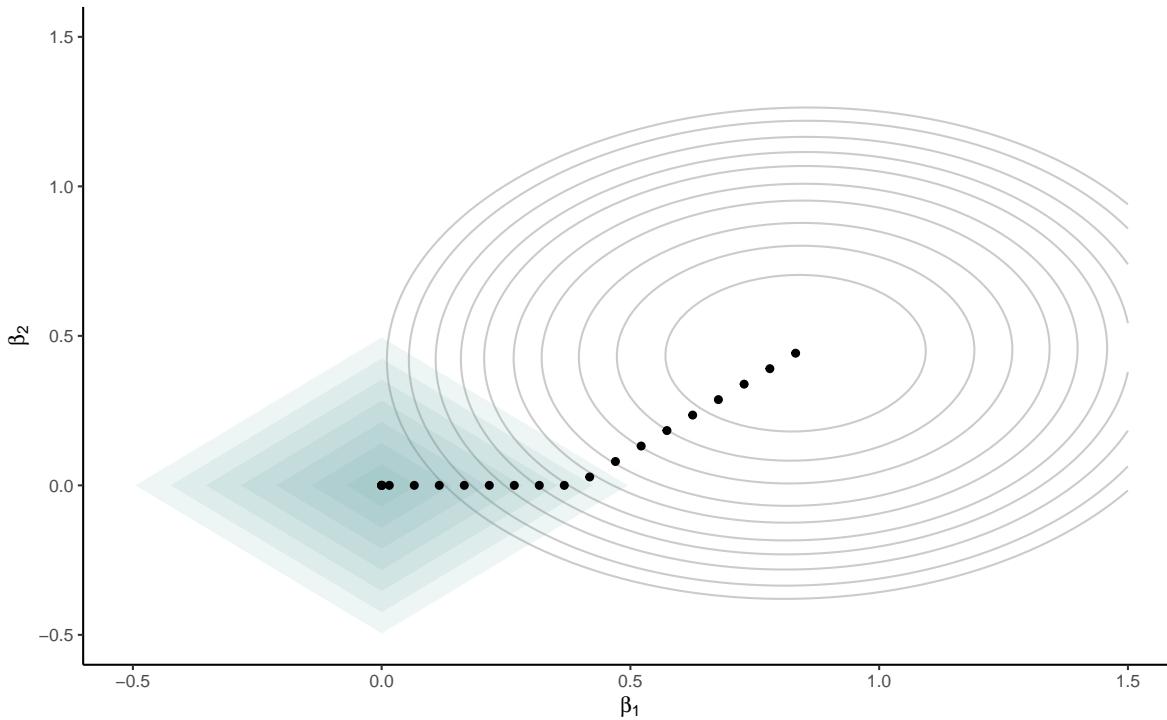


FIGURE 4.8 – Courbes de contour du critère de l'erreur quadratique moyenne (ellipses) et fonction de pénalité (losanges) pour différentes valeurs de  $\lambda$ . Les points dénotent des solutions différentes et intersectent les contours du losange.

```
glmnet::glmnet(
  x = as.matrix(dbm_a[,1:10]),
  y = dbm_a$ymontant,
  alpha = 1,
  lambda = lambopt)
# Prédictions et calcul de l'EQM
# On pourrait remplacer `newx` par
# d'autres données (validation externe)
pred <- predict(lasso_best,
                 s = lambopt,
                 newx = as.matrix(dbm_a[,-1]))
eqm_lasso <- mean((pred - dbm_a$ymontant)^2)
```

Le graphique de la Figure 4.9 montre l'évolution de l'erreur quadratique moyenne estimée en fonction du logarithme naturel de la pénalité (axe des abscisses). Comme plusieurs pénalités sont

dans la marge d'erreurs, on choisit le première modèle à un erreur-type de la valeur minimale

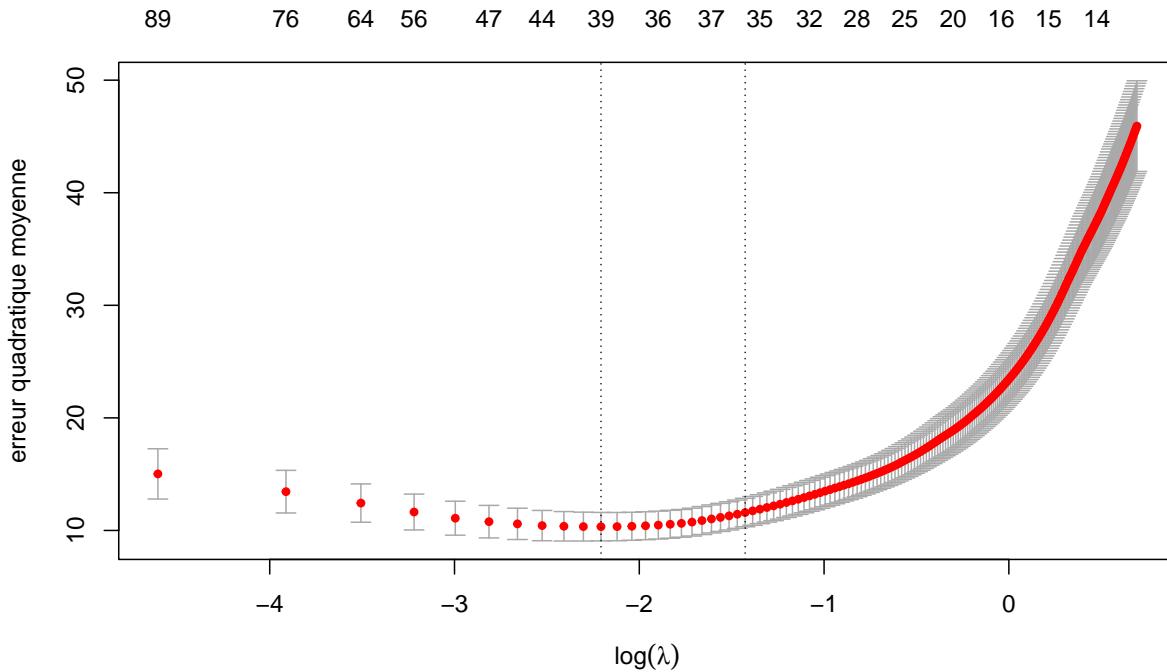


FIGURE 4.9 – Estimation de l'erreur quadratique moyenne (validation croisée à 10 groupes) pour les modèles avec pénalité LASSO en fonction de la pénalité (échelle log).

#### 4.5.4 Moyenne de modèles

Il est souvent préférable de combiner plusieurs modèles plutôt que d'en choisir un seul. La technique des forêts aléatoires (*random forests*) est une des meilleures techniques de prédiction disponibles de nos jours. Elle est basée sur cette idée, en combinant plusieurs arbres de classification (ou de régression) individuels. C'est une des techniques de base en exploitation de données.

Ici, nous allons voir comment cette idée peut être appliquée à notre contexte. Toutes les méthodes que nous avons vues jusqu'à maintenant font une sélection « rigide » de variables, dans le sens que chaque variable est soit sélectionnée pour faire partie du modèle, soit elle ne l'est pas. C'est donc tout ou rien pour chaque variable. Il y a beaucoup de variabilité associée à une telle forme de sélection. Une variable peut avoir été très près d'être choisie, mais elle ne l'a pas été et est éliminée complètement. Construire plusieurs modèles et en faire la moyenne permet d'adoucir le processus de sélection car une variable peut alors être partiellement sélectionnée.

Supposons qu'on dispose de deux échantillons et qu'on fasse une sélection de variables séparément pour les deux échantillons, avec l'une des approches que nous avons vues jusqu'à maintenant. Il

est alors très probable qu'on ne va pas avoir exactement les mêmes variables sélectionnées pour les deux échantillons. Supposons ensuite qu'on fasse la moyenne des coefficients pour les deux modèles. Si une variable, disons  $X_1$ , a été choisie les deux fois, alors la moyenne des deux coefficients devrait estimer en quelque sorte un effet global pour cette variable. Si une autre variable, disons  $X_2$ , n'a pas été choisie du tout pour les deux échantillons, alors la moyenne de ses deux coefficients est nulle. Mais si une variable, disons,  $X_3$ , a été choisie pour seulement l'un des deux échantillons, alors la moyenne de ses deux coefficients est la moitié du coefficient pour le modèle dans lequel elle a été choisie (car l'autre est zéro). Ainsi, cette variable est donc représentée par une « moitié » d'effet dans la moyenne des modèles. Donc au lieu d'être totalement là ou totalement absente, elle est présente en fonction de sa probabilité d'être sélectionnée. Ceci diminue de beaucoup la variabilité engendrée par une sélection « rigide » de variables et permet souvent de produire un modèle fort raisonnable.

Le problème est que l'on n'a pas plusieurs échantillons mais un seul. Une solution possible est de générer nous-mêmes des échantillons différents à partir de l'échantillon original. Cela peut être fait avec l'autoamorçage (*bootstrap*). Un échantillon d'autoamorçage est tout simplement un échantillon choisi au hasard et **avec remise** dans l'échantillon original. Ainsi, une même observation peut être sélectionnée plus d'une fois tandis qu'une autre peut ne pas être sélectionnée du tout.

L'idée est alors la suivante :

- 1) Générer plusieurs échantillons par autoamorçage nonparamétrique à partir de l'échantillon original.
- 2) Faire une sélection de variables pour chaque échantillon.
- 3) Faire la moyenne des paramètres de ces modèles.

```
# Moyenne de modèles
moyenne_modeles <- function(
  data,
  form,
  aic = FALSE,
  B = 100L,
  ks = 2){
  B <- as.integer(B)
  stopifnot(is.logical(aic),
            length(aic) == 1L,
            inherits(form, "formula"),
            B > 1,
            ks >= 0,
            inherits(data, "data.frame"))
  N <- nrow(data)
  # Faire une expansion pour obtenir colonnes
```

```

matmod <- model.matrix(form, data = data)
# Nombre de variables explicatives
p <- ncol(matmod) - 1L
# Formule du modèle complet
fmod <- formula(paste0("y ~", paste0("x", seq_len(p), collapse = "+")))
# Sauvegarder les noms
noms <- colnames(matmod)
xnom <- paste0("x", seq_len(p))
# Extraire le nom de la variable réponse
nom_reponse <- all.vars(form)[attr(terms(form), "response")]
# Créer une base de données avec la réponse
# moins l'ordonnée à l'origine
matmod <- data.frame(cbind(
  y = get(nom_reponse, data),
  matmod[,-1]))
colnames(matmod) <- c("y", xnom)
# Contenant pour params/ nb de sélections
params <- nselect <- rep(0, p + 1)
names(params) <-
  names(nselect) <-
  c("(Intercept)", xnom)

# Boucle
for(b in seq_len(B)){
  # Procédure de sélection avec AIC ou BIC
  modselect <- MASS::stepAIC(
    # Valeurs de départ
    object = lm(formula = y ~ 1,
    # Rééchantillonner données (avec remplacement)
    data = matmod[sample.int(n = N,
      size = N,
      replace = TRUE),]),
    # Modèle maximal additif considéré
    scope = fmod,
    # pénalité pour critère d'information
    # k = ifelse(aic, 2, log(N)),
    direction = "both",
    trace = FALSE,
    keep = function(mod, AIC, ...){
      # autres sorties des modèles à conserver
}

```

```

        list(IC = AIC(mod, k = ifelse(aic, 2, log(N))),
             coef = coef(mod))},
    k = ks)
min_IC <- which.min(unlist(modselect$keep['IC',]))
coefsv <- modselect$keep[2,min_IC]$coef
# Trouver quelles colonnes représentent
# un coefficient non-nul
colind <- match(names(coefsv),
                  names(params))
# Incrémenter paramètres non-nuls
params[colind] <- params[colind] +
  as.numeric(coefsv)
nselect[colind] <- nselect[colind] + 1L
}

names(nselect) <- noms
names(params) <- noms
return(list(coefs = params / B,
            nselect = nselect[-1] / B))
}

# Moyenne de modèles
# procédure séquentielle ascendante (AIC)
# sélection de modèle selon BIC
mmodeles <-
  moyenne_modeles(
    data = dbm_a,
    form = formule,
    B = 10L,
    aic = FALSE)

# Proportion des variables
# sélectionnées dans plus d'un modèle
sum(mmodeles$nselect > 0)
# Nombre moyen de coefficients
sum(mmodeles$nselect)
# variables retenues plus de 20% du temps
names(which(mmodeles$nselect > 0.2))

```

```
# moyenne des coefficients
mmodeles$coefs
```

Chaque modèle est construit à l'aide d'un échantillon aléatoire avec remise. Utilisez `set.seed` pour fixer le générateur de nombre aléatoire et permettre la reproductibilité

Toutes les méthodes employées jusqu'à maintenant utilise une méthode de pénalisation pour déterminer le meilleur modèle. Une alternative avec serait de répéter la sélection en utilisant directement l'erreur quadratique moyenne estimée à l'aide de la validation croisée comme critère de sélection : pour cela, il faudrait ajuster l'ensemble des modèles candidats retournés par une procédure exhaustive ou séquentielle.

## 4.6 Évaluation de la performance

La direction de la compagnie a décidé de passer outre vos recommandations et d'envoyer le catalogue aux 100 000 clients restants ; nous pouvons donc faire un post-mortem afin de voir ce que chaque modèle aurait donné comme profit, comparativement à la stratégie de référence. Les 100 000 autres clients serviront d'échantillon de validation pour évaluer la performance des modèles et, plus précisément, afin d'évaluer les revenus (ou d'autres mesures de performance) si ces modèles avaient été utilisés. L'échantillon de validation nous donnera donc l'heure juste quant aux mérites des différentes approches que nous allons comparer. En pratique, nous ne pourrions pas faire cela car la valeur de la variable cible ne serait pas connue pour ces clients et nous utiliserions plutôt les modèles pour obtenir des prédictions pour déterminer quels clients cibler avec l'envoi. Parmi, les 100 000 clients restants, il y en a 23 179 qui auraient acheté quelque chose si on leur avait envoyé le catalogue. Ces 23 179 observations vont nous servir pour estimer l'erreur quadratique moyenne (théorique) des modèles retenus par nos critères.

Commençons par l'estimation de l'erreur quadratique moyenne (moyenne des carrés des erreurs) pour les deux modèles retenus par le AIC et le BIC avec les variables de base. Le Tableau 4.7 contient aussi l'estimation de l'erreur quadratique moyenne si on utilise toutes les variables (14 en incluant les indicatrices) sans faire de sélection. On voit que le modèle choisi par le BIC est le meilleur des trois. Ces deux méthodes font mieux que le modèle qui inclut toutes les variables sans faire de sélection, mais nous verrons que leur performance est exécrable : les variables de base ne permettent pas de capturer les effets présents dans les données et ce manque de flexibilité coûte cher.

TABLEAU 4.7 – Estimation de l'erreur quadratique moyenne sur l'échantillon test avec les variables de base. Les meilleurs modèles selon les critères d'informations découlent d'une recherche exhaustive de tous les sous-ensembles.

nombre de variables	EQM	méthode
15	25.69	toutes les variables
12	25.53	exhaustive - AIC
10	25.04	exhaustive - BIC

TABLEAU 4.8 – Comparaison des méthodes selon l'erreur quadratique moyenne avec les variables de base, les interactions et les termes quadratiques.

nombre de variables	EQM	méthode
104	19.63	toutes les variables
21	12	séquentielle ascendante, choix selon AIC
15	12.31	séquentielle ascendante, choix selon BIC
23	12.75	séquentielle ascendante avec critère AIC
20	12.4	séquentielle descendante avec critère BIC
30	12	LASSO, validation croisée avec 10 groupes

Le Tableau 4.8 présente la performance de toutes les méthodes avec les autres variables. On voit d'abord qu'utiliser toutes les 104 variables sans faire de sélection fait mieux (EQM de 19.63) que les modèles précédents basés sur les 10 variables originales. Mais faire une sélection permet une amélioration très importante de la performance (EQM jusqu'à 12 dans l'exemple). Utiliser les 104 variables mène à du surajustement (*over-fitting*).

Les méthodes séquentielles avec un critère d'information (qui pénalisent davantage que les tests d'hypothèse classique) mènent à des modèles plus parcimonieux qui ont une erreur quadratique moyenne de validation plus faible. Le LASSO performe très bien dans ce cas de figure. Les coefficients sont tous rétrécis vers zéro (donc le nombre de coefficients non-nuls n'est pas évoquant), ce qui engendre du biais et peut affecter négativement la performance si le rapport signal-bruit est élevé.

Il faut bien comprendre qu'il ne s'agit que d'un seul exemple : il ne faut surtout pas conclure que la méthode séquentielle sera toujours la meilleure. En fait, il est impossible de prévoir quelle méthode donnera les meilleurs résultats.

Il y aurait plusieurs autres approches/combinatoires qui pourraient être testées. Le but de ce chapitre était simplement de présenter les principes de base en sélection de modèles et de variables ainsi que certaines approches pratiques. Il y a d'autres approches intéressantes, tels le filet élastique.

Ces méthodes sont dans la même mouvance moderne que celle qui consiste à faire la moyenne de plusieurs modèles, en permettant à la fois une sélection de variables et en permettant d'avoir des parties d'effet par le rétrécissement (*shrinkage*). De récents développements théoriques permettent aussi de corriger les valeurs-*p* pour faire de l'inférence post-sélection avec le LASSO.

### i En résumé

- En présence de nombreuses variables explicatives, choisir un modèle **prédictif** est compliqué : le nombre de modèles possibles augmente rapidement avec le nombre de prédicteurs,  $p$ .
- Si un modèle est mal spécifié (variables importantes manquantes), alors les estimations sont biaisées. Si le modèle est surspécifié, les coefficients correspondants aux variables superflues incluses sont en moyenne nuls, mais contribuent à l'augmentation de la variance ( compromis *bias/variance*).
- La taille du modèle ( $p$ , le nombre de variables explicatives) est restreinte par le nombre d'observations disponibles,  $n$ .
  - En général, il faut s'assurer d'avoir suffisamment d'observations pour estimer de manière fiable les coefficients (le rapport  $n/p$  donne le budget moyen par paramètre).
  - Porter une attention particulière aux variables binaires et aux interactions avec ces dernières : si les effectifs de certaines modalités sont faibles, il y a possible de surajustement.
- Le principal critère pour juger de la qualité d'un modèle linéaire est l'erreur quadratique moyenne.
  - L'estimation de l'erreur quadratique moyenne obtenue à partir de l'échantillon d'apprentissage (qui sert à estimer les paramètres) est trompeuse et mène au surajustement :
    - plus le modèle est compliqué, plus cette erreur décroît.
    - cette performance n'est pas répétée sur de nouvelles données.
- Critères de sélection : Plusieurs stratégies existent pour pallier à cet excès d'optimisme
  - validation externe : diviser le jeu de données aléatoirement au préalable en deux ou trois. Nécessite une grande base de données, potentiellement sous-optimal.
  - validation croisée : diviser aléatoirement le jeu de données en plis et varier les échantillons d'apprentissage en conservant un pli en réserve à chaque fois comme validation. Plus coûteux en calcul (il faut réajuster plusieurs fois les modèles), applicable avec des petites bases de données.
  - pénalisation a posteriori : ajouter une pénalité fonction du nombre de paramètres qui compense pour l'augmentation constante de l'ajustement (par ex., critères d'information).
  - rétrécissement des coefficients : inclure dans la fonction objective qui est maximisée

une pénalité qui constraint les paramètres et les force à demeurer petit. Cela introduit du biais pour réduire la variance.

- Une pénalité particulière (LASSO) constraint certains paramètres à être exactement nuls, ce qui correspond implicitement à une sélection de variables.
- En pratique, on cherche à essayer plusieurs modèle pour trouver un choix optimal de variables.
  - Une recherche exhaustive garantie le survol du plus grand nombre de modèles possibles, mais est coûteuse et limitée à moins de 50 variables.
  - Les algorithmes gloutons de recherche séquentielle sont sous-optimaux, mais rapides
- On applique le critère de sélection sur la liste de modèles candidats pour retenir celui qui donne la meilleure performance.
- Pour éviter une sélection rigide, on peut perturber les données et répéter la procédure pour calculer une moyenne de modèles. Cette approche est très coûteuse en calcul.

# Chapitre 5

## Régression logistique

### 5.1 Introduction

En régression linéaire, on cherche à expliquer le comportement d'une variable quantitative  $Y$  que l'on peut traiter comme étant continue (elle peut prendre suffisamment de valeurs différentes).

Supposons à présent que l'on veut expliquer le comportement d'une variable  $Y$  prenant seulement deux valeurs que l'on va noter 0 et 1.

Exemples :

- Est-ce qu'un client potentiel va répondre favorablement à une offre promotionnelle ?
- Est-ce qu'un client est satisfait du service après-vente ?
- Est-ce qu'un client va faire faillite ou non au cours des trois prochaines années.

En général, on cherchera à expliquer le comportement d'une variable binaire  $Y$  en utilisant un modèle basé sur  $p$  variables explicatives  $X_1, \dots, X_p$ .

Notre but sera de faire de l'inférence, de la prédiction, ou les deux à la fois, soit

- 1) **Inférence** : comprendre comment et dans quelles mesures les variables  $\mathbf{X}$  influencent  $Y$  (ou bien la probabilité que  $Y = 1$ ).
- 2) **Prédiction** : développer un modèle pour prévoir des valeurs de  $Y$  futures à partir des variables  $\mathbf{X}$ .

### 5.2 Modèle de régression logistique

Avec une variable réponse continue, le modèle de régression linéaire,

$$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \varepsilon,$$

avec  $E(\varepsilon | \mathbf{X}) = 0$  et  $Va(\varepsilon | \mathbf{X}) = \sigma^2$ , peut être écrit de manière équivalente comme  $E(Y | \mathbf{X}) = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p$  et  $Va(Y | \mathbf{X}) = \sigma^2$ .

Si  $Y$  est binaire (0/1), on peut facilement vérifier que

$$E(Y | \mathbf{X}) = \Pr(Y = 1 | \mathbf{X}),$$

soit la probabilité que  $Y$  égale 1 étant donné les valeurs des variables explicatives. Pour simplifier la notation, posons la probabilité de succès  $p = \Pr(Y = 1 | \mathbf{X})$  en se rappelant que  $p$  est une fonction des variables explicatives.

À première vue, on peut se demander pourquoi ne pas utiliser le même modèle que la régression linéaire, c'est-à-dire

$$\eta = \beta_0 + \beta_1 X_1 + \dots + \beta_p X_p.$$

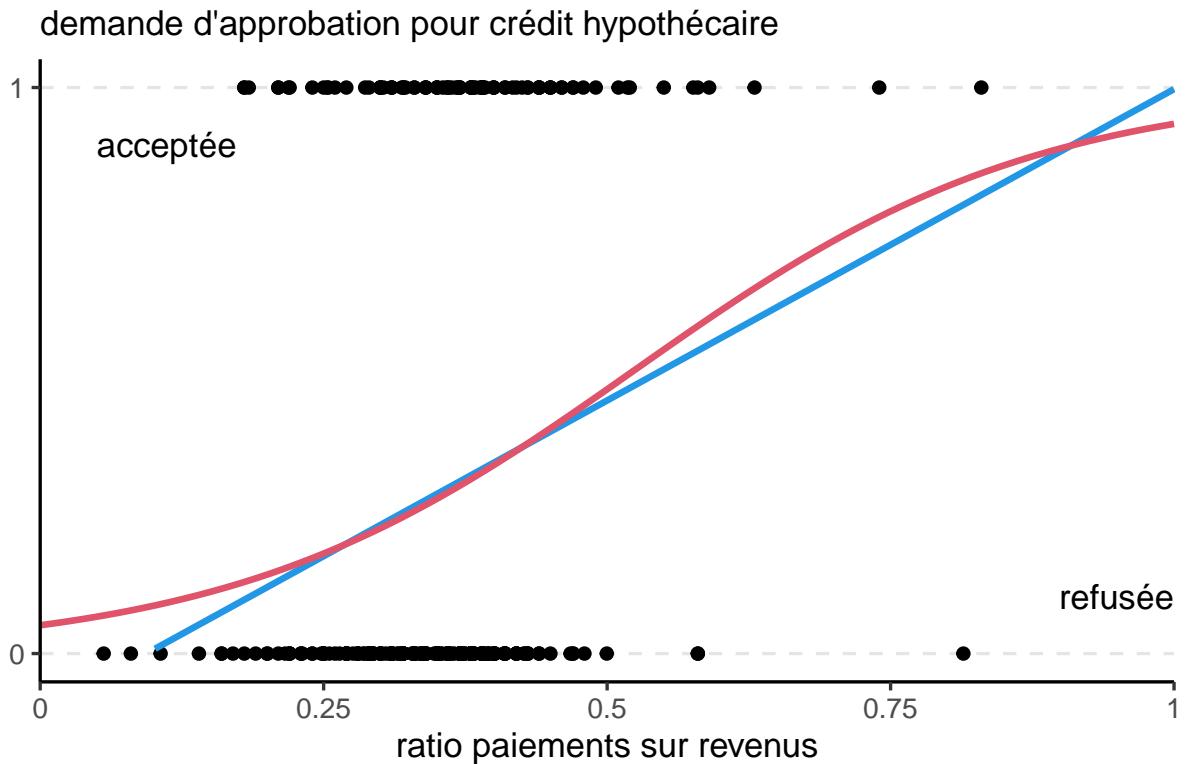


FIGURE 5.1 – Données de la réserve de Boston sur l'approbation de prêts hypothécaires (1990); données tirées de Stock et Watson (2007).

La Figure 5.1 montre le modèle de régression linéaire (bleu) et le modèle logistique. La pente pour la ligne bleue correspond à l'augmentation (réputée constante) de la probabilité d'approbation de crédit, de l'ordre de 11% par augmentation de 0.1 du rapport paiements hypothécaires sur revenu.

Il y a quelques problèmes avec le modèle linéaire. D'abord, les données binaires ne respectent pas le postulat d'égalité des variances, ce qui rend les tests d'hypothèses caducs. Le problème principal est que  $p$  est une probabilité. Par conséquent  $p$  prend seulement des valeurs entre 0 et 1 alors que rien n'empêche  $\eta$  de prendre des valeurs dans  $\mathbb{R} = (-\infty, \infty)$  : par exemple, on voit que la droite de la Figure 5.1 retourne des prédictions négatives dès que le ratio paiements/revenus est en dessous de 0.094 : on peut évidemment tronquer ces prédictions à zéro, mais cela sous-tend que la probabilité d'acceptation est nulle, alors que certaines personnes dans l'échantillon ont reçu un prêt.

Une façon de résoudre ce problème consiste à appliquer une transformation à  $p$  de telle sorte que la quantité transformée puisse prendre toutes les valeurs entre  $-\infty$  et  $\infty$ . Le modèle de régression logistique est défini à l'aide de la transformation logit,

$$\text{logit}(p) = \ln\left(\frac{p}{1-p}\right) = \eta = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p,$$

où  $\ln$  est le logarithme naturel.

En régression linéaire, on suppose que l'espérance de  $Y$  étant donné les valeurs des variables explicatives est une combinaison linéaire de ces dernières. En régression logistique, on suppose que le logit de la probabilité de succès est une combinaison linéaire des variables explicatives.

Une simple manipulation algébrique permet d'exprimer ce modèle en terme de la probabilité  $p$ ,

$$p = \text{expit}(\eta) = \frac{\exp(\eta)}{1 + \exp(\eta)} = \frac{1}{1 + \exp(-\eta)}.$$

On peut voir qu'à mesure que le prédicteur linéaire  $\eta = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$  augmente, la probabilité augmente. Si le coefficient  $\beta_j$  est négatif,  $p$  diminuera à mesure que  $X_j$  augmente.

Pour une variable binaire  $Y$ , le rapport  $p/(1-p)$  est appelé **cote** et représente le ratio de la probabilité de succès ( $Y = 1$ ) sur la probabilité d'échec ( $Y = 0$ ),

$$\text{cote}(p) = \frac{p}{1-p} = \frac{\Pr(Y = 1 | \mathbf{X})}{\Pr(Y = 0 | \mathbf{X})}.$$

Par exemple, une cote de 4 veut dire qu'il y a 4 fois plus de chance que  $Y$  soit égale à 1 par rapport à 0. Une cote de 0.25 veut dire le contraire, il y a 4 fois moins de chance que  $Y = 1$  par rapport à 0 ou bien, de manière équivalente, il y a 4 fois plus de chance que  $Y = 0$  par rapport à 1. Le Tableau 5.1 donne un aperçu de cotes pour quelques probabilités  $p$ .

### 5.2.1 Estimation et interprétation des paramètres

Supposons qu'on dispose d'un échantillon de taille  $n$  sur les variables  $(Y, X_1, \dots, X_p)$ . À l'aide de ces observations, on peut estimer les paramètres  $\boldsymbol{\beta} = (\beta_0, \beta_1, \dots, \beta_p)$  du modèle de régression logistique

$$\text{logit}(p) = \ln\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p.$$

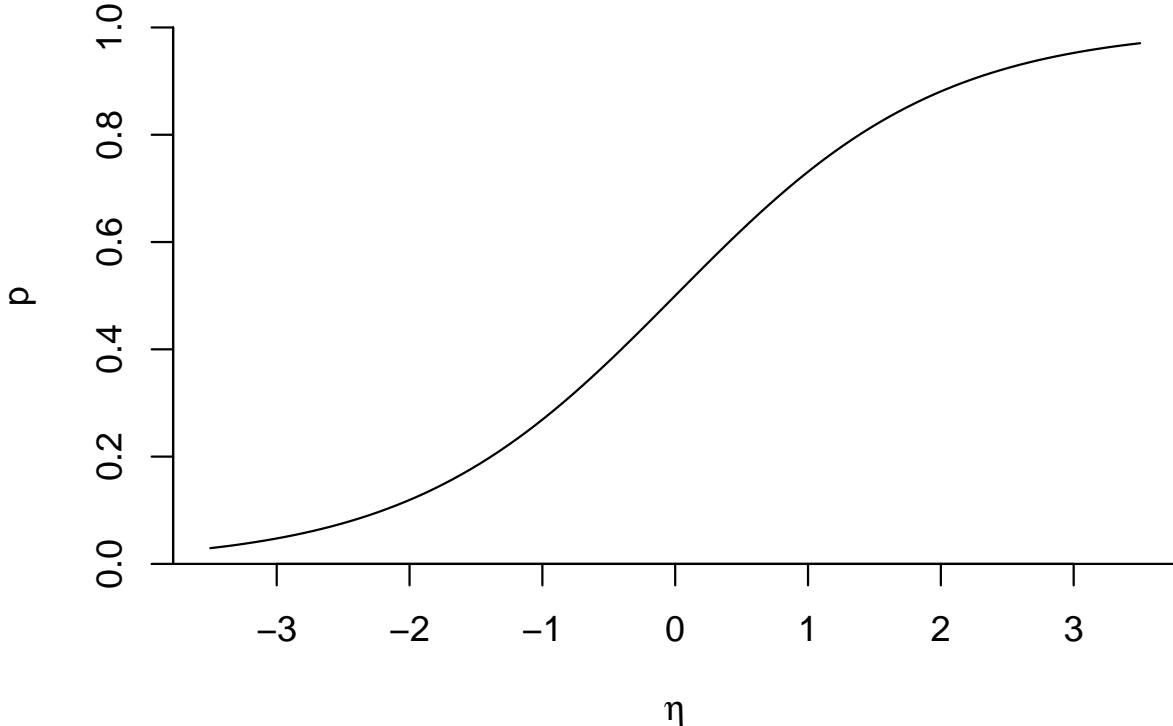


FIGURE 5.2 – Valeurs ajustées du modèle de régression logistique en fonction du prédicteur linéaire  $\eta$ .

On obtient ainsi les estimés des paramètres  $\hat{\beta}$ , desquels découle une estimation de  $\Pr(Y = 1)$  pour les valeurs  $X_1 = x_1, \dots, X_p = x_p$  d'un individu donné,

$$\hat{p} = \text{expit}(\hat{\beta}_0 + \dots + \hat{\beta}_p X_p).$$

Un modèle ajusté peut ensuite être utilisé pour faire de la classification (prédiction) pour de nouveaux individus pour lesquels la variable réponse  $Y$  n'est pas observée. Pour ce faire, on choisit un point de coupure  $c$  (souvent  $c = 0.5$  mais pas toujours) et on classe les observations en deux groupes :

TABLEAU 5.1 – Cote et probabilité de succès

$\Pr(Y = 1)$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
cote	0.11	0.25	0.43	0.67	1	1.5	2.3	4	9
	$\frac{1}{9}$	$\frac{1}{4}$	$\frac{3}{7}$	$\frac{2}{3}$	1	$\frac{3}{2}$	$\frac{7}{3}$	4	9

- Si  $\hat{p} < c$ , alors  $\hat{Y} = 0$  (c'est-à-dire, on assigne cette observation à la catégorie 0 ou échec).
- Si  $\hat{p} \geq c$ , alors  $\hat{Y} = 1$  (c'est-à-dire, on assigne cette observation à la catégorie 1 ou succès).

On reviendra en détail sur cet aspect dans une section suivante.

La méthode d'estimation des paramètres usuelle est la méthode du maximum de vraisemblance. Pour les applications, il est suffisant de savoir manipuler trois quantités importantes : la log-vraisemblance, le AIC et le BIC. Les deux critères d'information, que nous avons couvert dans les chapitres précédents, servent à la sélection de modèles tandis que la log-vraisemblance  $\ell$  servira à construire un test d'hypothèse.

### 5.2.2 Méthode du maximum de vraisemblance

Cette sous-section est facultative. Elle donne plus de détails sur la méthode du maximum de vraisemblance et les quantités en découlant, soit AIC, BIC et  $\ell(\hat{\beta})$ .

La méthode du maximum de vraisemblance (*maximum likelihood*) est possiblement la méthode d'estimation la plus utilisée en statistique. En général, pour un échantillon donné et un modèle avec des paramètres inconnus  $\theta$ , on peut calculer la « probabilité » d'avoir obtenu les observations de notre échantillon selon les paramètre. Si on traite cette « probabilité » comme étant une fonction des paramètres du modèle,  $\theta$ , on l'appelle alors la vraisemblance (*likelihood*). La méthode du maximum de vraisemblance consiste à trouver les valeurs des paramètres qui maximisent la vraisemblance. On cherche donc les estimations qui sont les plus vraisemblables étant donné nos observations.

En pratique, il est habituellement plus simple de chercher à maximiser le log de la vraisemblance (ce qui revient au même car le logarithme naturel est une fonction croissante) et on nomme cette fonction la log-vraisemblance (*log-likelihood*).

Vous connaissez déjà des exemples d'estimateurs du maximum de vraisemblance. La moyenne d'un échantillon est l'estimateur du maximum de vraisemblance pour la moyenne de la population  $\mu$  si les observations représentent un échantillon aléatoire simple tiré d'une loi normale.

Dans le cas d'un modèle de régression linéaire multiple de la forme  $Y_i \sim \text{No}(\beta_0 + \sum_{j=1}^p \beta_j X_{ij}, \sigma^2)$  des termes indépendants et de même loi, la log-vraisemblance du modèle pour un échantillon de taille  $n$  est

$$\ell(\boldsymbol{\beta}, \sigma^2) = -\frac{n}{2} \ln(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{i=1}^n (Y_i - \beta_0 - \beta_1 X_{1i} - \cdots - \beta_p X_{pi})^2.$$

Puisque le premier terme ne dépend pas des paramètres  $\boldsymbol{\beta}$ , il est clair que maximiser cette fonction de  $\boldsymbol{\beta}$  revient à minimiser  $\sum_{i=1}^n (Y_i - \beta_0 - \beta_1 X_{1i} - \cdots - \beta_p X_{pi})^2$  : ce critère est exactement le même que celui des moindres carrés. Par conséquent, les estimations des paramètres  $\boldsymbol{\beta}$  provenant de la méthode des moindres carrés peuvent être vues comme étant des estimateurs du maximum de vraisemblance sous l'hypothèse de normalité et d'homoscédasticité des observations ; il est même possible d'obtenir une formule explicite pour le calcul des estimateurs.

Dans le cas de la régression logistique, la fonction de log-vraisemblance s'écrit

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^n Y_i (\beta_0 + \beta_1 X_{i1} + \cdots + \beta_p X_{ip}) - \sum_{i=1}^n \ln \{1 + \exp(\beta_0 + \cdots + \beta_p X_{ip})\}$$

Contrairement au cas de la régression linéaire, on ne peut trouver une solution explicite pour les valeurs des paramètres qui maximisent cette fonction. Des méthodes numériques doivent être utilisées pour l'optimisation. Une fois la maximisation accomplie, on obtient les estimés du maximum de vraisemblance,  $\hat{\boldsymbol{\beta}}$ . On peut alors calculer la valeur maximale (numérique) de la log-vraisemblance,  $\ell(\hat{\boldsymbol{\beta}})$ . Par analogie avec la régression linéaire la valeur de la log-vraisemblance évaluée à  $\hat{\boldsymbol{\beta}}$ ,  $\ell(\hat{\boldsymbol{\beta}})$ , augmente toujours lorsqu'on ajoute des régresseurs et c'est pourquoi on ne pourra pas l'utiliser comme outil de sélection de variables.

Les critères d'information sont des fonctions de la log-vraisemblance, mais incluent une pénalité pour le nombre de coefficients  $\boldsymbol{\beta}$ ,

$$\begin{aligned} AIC &= -2\ell(\hat{\boldsymbol{\beta}}) + 2(p+1) \\ BIC &= -2\ell(\hat{\boldsymbol{\beta}}) + \ln(n)(p+1) \end{aligned}$$

Ces définitions sont utilisables dans plusieurs situations lorsque le modèle est ajusté par la méthode du maximum de vraisemblance. Tout comme en régression linéaire et en analyse factorielle, ces deux critères pourront être utilisés pour faire de la sélection de modèles si on calcule les estimateurs du maximum de vraisemblance.

### 5.2.3 Exemple du *Professional Rodeo Cowboys Association*

L'exemple suivant est inspiré de l'article

Daneshvary, R. et Schwer, R. K. (2000) The Association Endorsement and Consumers' Intention to Purchase. *Journal of Consumer Marketing* 17, 203-213.

Dans cet article, les auteurs cherchent à voir si le fait qu'un produit soit recommandé par le *Professional Rodeo Cowboys Association* (PRCA) a un effet sur les intentions d'achats. On dispose de 500 observations sur les variables suivantes :

- $Y$  : seriez-vous intéressé à acheter un produit recommandé par le PRCA
  - 0 : non
  - 1 : oui
- $X_1$  : quel genre d'emploi occupez-vous?

- 1 : à la maison
- 2 : employé
- 3 : ventes/services
- 4 : professionnel
- 5 : agriculture/ferme
- $X_2$  : revenu familial annuel
  - 1 : moins de 25 000
  - 2 : 25 000 à 39 999
  - 3 : 40 000 à 59 999
  - 4 : 60 000 à 79 999
  - 5 : 80 000 et plus
- $X_3$  : sexe
  - 0 : homme
  - 1 : femme
- $X_4$  : avez-vous déjà fréquenté une université?
  - 0 : non
  - 1 : oui
- $X_5$  : âge (en années)
- $X_6$  : combien de fois avez-vous assisté à un rodéo au cours de la dernière année?
  - 1 : 10 fois ou plus
  - 2 : entre six et neuf fois
  - 3 : cinq fois ou moins

Le but est d'examiner les effets de ces variables sur l'intentions d'achat ( $Y$ ). Les données se trouvent dans la base de données `logit1`.

#### 5.2.4 Modèle avec une seule variable explicative

Faisons tout d'abord une analyse en utilisant seulement  $X_5$  (âge) comme variable explicative. L'ajustement du modèle de régression incluant uniquement  $X_5$  sera effectuée en exécutant le programme

```
data(logit1, package = "hecmulti")
# Nombre d'observations par groupe
with(logit1, table(y))
# Ajustement du modèle avec une
# seule variable explicative
modele1 <- glm(formula = y ~ x5,
                 family = binomial(link = "logit"),
                 data = logit1)
```

```
# Tableau résumé avec coefficients
summary(modele1)
# Cote
cote <- exp(modele1$coefficients)
# Intervalles de confiance profilés
# pour les paramètres betas
confbeta <- confint(modele1)
# Intervalles de confiance pour la cote
exp(confbeta)
# Tester la significativité globale
# à l'aide du rapport de vraisemblance
anova(modele1, test = 'Chisq')
# Critères d'information
np <- length(coef(modele1))
n <- nrow(logit1)
AIC(modele1)
# -2*logLik(modele1) + 2*np
BIC(modele1)
# -2*logLik(modele1) + log(n)*np
```

Par défaut, pour des variables 0/1, le modèle décrit la probabilité de succès. On peut transformer la variable réponse en facteur (`factor`) et changer la catégorie de référence via `relevel` pour obtenir le modèle  $\text{Pr}(Y = 0 | X_5)$ .

```
nlogit1 <- logit1 |>
  dplyr::mutate(y = relevel(factor(y), "1"))
glm(formula = y ~ x5,
    family = binomial(link = "logit"),
    data = nlogit1)
```

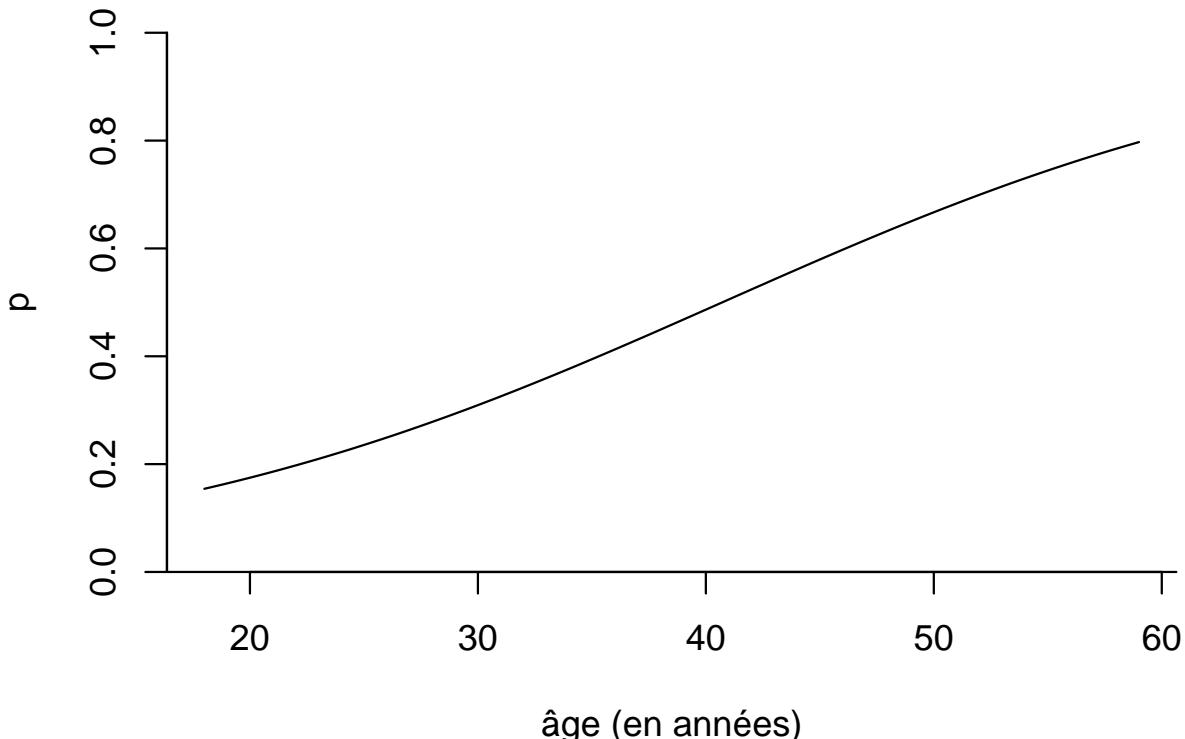
Quelques observations sur l'échantillon et les données :

- On voit qu'il y a 272 personnes (0) qui ne sont pas intéressées à acheter un produit recommandé par le PRCA et 228 personnes (1) qui le sont.
- Les estimés des paramètres sont  $\hat{\beta}_0 = -3.05$  et  $\hat{\beta}_{\text{age}} = 0.0749$ .
- Un intervalle de confiance de niveau 95% pour l'effet de l'âge est [0.0465; 0.1043].
- Le modèle ajusté est  $\text{logit}\{\text{Pr}(Y = 1 | X_5 = x_5)\} = -3.05 + 0.0749x_5$ . On peut également exprimer ce résultat sous forme de probabilité :  $\text{Pr}(Y = 1 | X_5 = x_5) = \exp(-3.05 + 0.0749x_5) / (1 + \exp(-3.05 + 0.0749x_5))$ .

mer ce modèle directement en terme de la probabilité de succès,

$$\begin{aligned}\Pr(Y = 1 | X_5 = x_5) &= \text{expit}(-3.05 + 0.0749x_5) \\ &= \frac{1}{1 + \exp(-3.05 - 0.0749x_5)}\end{aligned}$$

Le graphe de cette fonction pour  $X_5$  allant de 18 à 59 ans, respectivement les valeurs minimales et maximales observées dans l'échantillon, montre que le lien entre l'âge et  $p$  est presque linéaire entre 20 et 60 ans. On décèle tout de même la forme sigmoïde de la fonction logit aux deux extrémités.



- La valeur- $p$  pour  $\hat{\beta}_{\text{age}}$  correspondant au test des hypothèses  $\mathcal{H}_0 : \beta_{\text{age}} = 0$  versus  $\mathcal{H}_1 : \beta_{\text{age}} \neq 0$ , est plus petite que  $10^{-4}$  et donc l'effet de la variable âge est statistiquement différent de zéro. Plus l'âge augmente, plus la probabilité d'être intéressé à acheter un produit recommandé par le PRCA augmente.

### 5.2.5 Interprétation du paramètre

Si une variable est modélisée à l'aide d'un seul paramètre (pas de terme quadratique et pas d'interaction avec d'autre covariables), une valeur positive du paramètre indique une association positive avec  $p$  alors qu'une valeur négative indique le contraire.

Ainsi, le signe du paramètre donne le sens de l'association. Si le coefficient  $\beta_j$  de la variable  $X_j$  est positif, alors plus la variable augmente, plus  $\Pr(Y = 1)$  augmente. Inversement, si le coefficient  $\beta_j$  est négatif, plus la variable augmente, plus  $\Pr(Y = 1)$  diminue.

En régression linéaire, l'interprétation de coefficient  $\beta_j$  est simple : lorsque la variable  $X_j$  augmente de un, la variable  $Y$  augmente en moyenne de  $\beta_j$ , toute chose étant égale par ailleurs. Cette interprétation ne dépend pas de la valeur de  $X_j$ . En régression logistique, comme le modèle est nonlinéaire en fonction de  $\Pr(Y = 1)$  (courbe sigmoïde), l'augmentation ou la diminution de  $\Pr(Y = 1 | \mathbf{X})$  pour un changement d'une unité de  $X_j$  dépend de la valeur de cette dernière. C'est pourquoi il est parfois plus utile d'utiliser la cote pour interpréter globalement l'effet d'une variable.

Dans notre exemple, on peut exprimer le modèle ajusté en termes de cote,

$$\frac{\Pr(Y = 1 | X_5 = x_5)}{\Pr(Y = 0 | X_5 = x_5)} = \exp(-3.05) \exp(0.0749x_5).$$

Ainsi, lorsque  $X_5$  augmente d'une année, la cote est multipliée par  $\exp(0.0749) = 1.078$  peu importe la valeur de  $x_5$ . Pour deux personnes dont la différence d'âge est un an, la cote de la personne plus âgée est 7.8% plus élevée. On peut aussi quantifier l'effet d'une augmentation d'un nombre d'unités quelconque. Par exemple, pour chaque augmentation de 10 ans de  $X_5$ , la cote est multiplié par  $1.078^{10} = 2.12$ , soit une augmentation de 112%.

L'interprétation des coefficients du modèle logistique se fait au niveau du rapport de cote, à savoir  $\exp(\beta)$ .

Un des avantages d'utiliser la vraisemblance comme fonction objective est que les intervalles de confiance et les estimateurs basés sur la vraisemblance (profilée) sont invariant aux reparamétrisations : l'intervalle de confiance à niveau 95% pour  $\exp(\beta_{\text{age}})$  est obtenu en prenant l'exponentielle des bornes de l'intervalle pour  $\beta_{\text{age}}$ ,  $[\exp(0.0465); \exp(0.1043)]$ , soit  $[1.048; 1.110]$  tel que rapporté dans la sortie. Ce n'est **pas** le cas des intervalles usuels de Wald qui ont la forme  $\hat{\beta} \pm 1.96\text{se}(\hat{\beta})$ .

Comme l'exponentielle est une transformation monotone croissante, on a  $\beta > 0$  si et seulement si  $\exp(\beta) > 1$ , etc. On peut ainsi utiliser les intervalles de confiance pour tester l'hypothèse  $\mathcal{H}_0 : \beta_j = 0$  ou de façon équivalente  $\mathcal{H}_0 : \exp(\beta_j) = 1$  à niveau 95%.

Ajustons à présent le modèle avec toutes les variables explicatives. Rappelez-vous que la variable  $X_1$  (quel genre d'emploi occupez-vous) a cinq catégories,  $X_2$  (revenu familial annuel) a cinq catégories, et  $X_6$  (combien de fois avez-vous assisté à un rodéo au cours de la dernière année) a trois catégories. Notez qu'on pourrait aussi traiter  $X_2$  comme continue car elle est ordinaire et possède tout de même cinq modalités, mais on la traitera comme variable nominale.

Les variables de type `factor` sont modélisées par défaut à l'aide d'un ensemble de variables indicatrices, la catégorie de référence étant celle qui apparaît en dernier en ordre alphabétique.

```

str(logit1)
modele2 <- glm(
  y ~ x1 + x2 + x3 + x4 + x5 + x6,
  data = logit1,
  family = binomial(link = "logit"))
)
summary(modele2)
ic <- confint(modele2)
# Tests de rapport de vraisemblance
car::Anova(modele2, type = "3")

```

TABLEAU 5.2 – Coefficients (cote), intervalles de confiance profilée de 95 pourcent et valeurs-p pour le test de rapport de vraisemblance pour le modèle logistique avec toutes les variables catégorielles.

variables	cote <sup>1</sup>	IC 95% <sup>1</sup>	valeur-p
x1			0.4
1	—	—	
2	0.44	0.18, 1.06	
3	0.51	0.21, 1.21	
4	0.51	0.21, 1.25	
5	0.70	0.27, 1.80	
x2			<0.001
1	—	—	
2	0.83	0.38, 1.82	
3	0.57	0.25, 1.31	
4	0.09	0.03, 0.25	
5	0.26	0.08, 0.84	
x3			<0.001
0	—	—	
1	3.85	2.34, 6.50	
x4			<0.001
0	—	—	
1	6.24	3.53, 11.4	
x5	1.12	1.08, 1.16	<0.001
x6			<0.001
1	—	—	
2	0.25	0.13, 0.49	
3	0.09	0.04, 0.18	

TABLEAU 5.3 – Mesures d'ajustement du modèle avec toutes les variables explicatives, et du modèle nul (pour lequelle la probabilité de succès est la proportion de 1).

	AIC	BIC	log.vrais.
modèle ajusté	544.20	603.20	-258.1
modèle nul	691.27	695.48	-258.1

<sup>1</sup>cote = rapport de cote, IC = intervalle de confiance

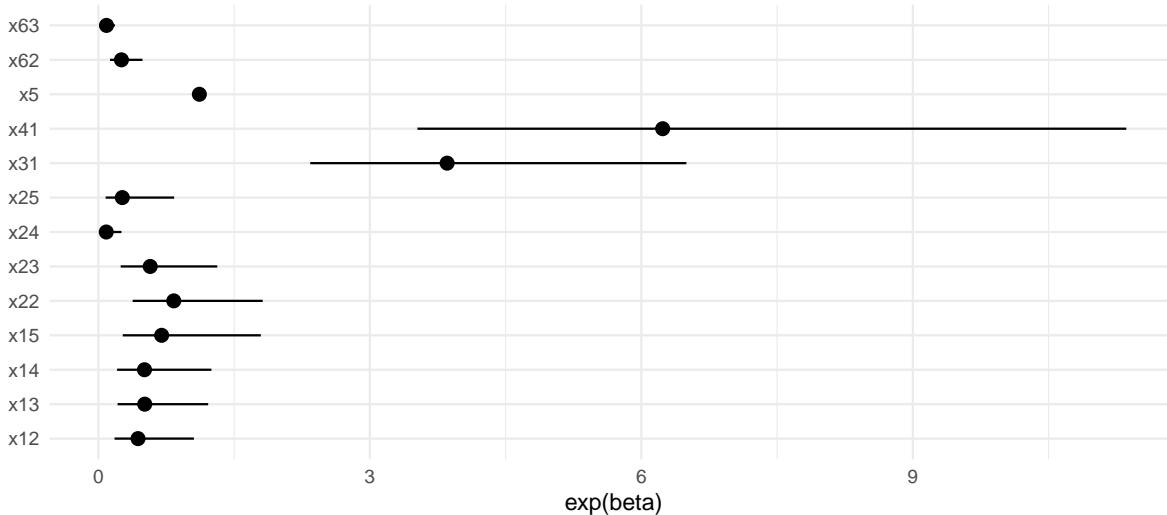


FIGURE 5.3 – Intervalles de confiance profilés de niveau 95% pour les coefficients du modèle logistique (échelle exponentielle).

L'interprétation se fait comme en régression linéaire multiple puisqu'il n'y a pas ni terme quadratique, ni interaction. Les paramètres estimés représentent donc l'effet de la variable correspondante sur le logit une fois que les autres variables sont dans le modèle, et demeurent fixes.

Prenons le coefficient associé à l'âge ( $X_5$ ) comme exemple. Le paramètre estimé est  $\hat{\beta}_{\text{age}} = 0.109$  et il est significativement différent de zéro. Ainsi, plus l'âge augmente, plus  $\Pr(Y = 1 | \mathbf{X})$  augmente, toutes autres choses étant égales par ailleurs. Pour chaque augmentation d'un an de  $X_5$ , la cote est multipliée par  $\exp(0.109) = 1.116$ , lorsque les autres variables demeurent fixes.

N'oubliez pas la nuance suivante concernant l'interprétation d'un test lorsque plusieurs variables explicatives font partie du modèle. Si un paramètre n'est pas significativement différent de zéro, cela ne veut pas dire qu'il n'y a pas de lien entre la variable correspondante et  $Y$ . Cela veut seulement dire qu'il n'y a pas de lien significatif une fois que les autres variables sont dans le modèle.

L'interprétation des variables catégorielles est analogue à celle faite en régression linéaire. On peut

aussi interpréter individuellement les paramètres des indicatrices : pour  $I(X_6 = 2)$ , lorsque les autres variables demeurent fixes, les personnes ayant assisté entre six et 10 fois à un rodéo au cours de la dernière année voient leur cote multipliée par  $\exp(-1.370) = 0.255$  par rapport aux personnes ayant assisté plus de di fois. Ce paramètre est significativement différent de zéro car sa valeur- $p$  est négligeable ; l'intervalle de confiance à 95% pour le rapport de cotes, basé sur la vraisemblance profilée, est [0.13; 0.49] et la valeur 1 (qui correspond à un rapport de cote constant) n'est pas dans l'intervalle. Ainsi, il y a une différence significative entre les gens qui ont assisté à 10 rodéos ou plus et les gens qui ont assisté à 5 rodéos ou moins, pour ce qui est de l'intérêt à acheter un produit recommandé par le PRCA.

Si on désire comparer les deux modalités  $X_6 = 2$  et  $X_6 = 3$ , il suffit de changer la modalité de référence de  $x_6$  et d'exécuter le modèle à nouveau. Une alternative est de calculer le rapport de cotes pour ces deux modalités.

### 5.2.6 Test du rapport de vraisemblance

Les tests rapportés d'ordinaire dans le tableau avec les coefficients (et correspondants aux valeurs- $p$ ) sont des tests de Wald, à savoir  $W = \hat{\beta}/\text{se}(\hat{\beta})$ . Ces tests feront l'affaire dans la plupart des applications. Par contre, il existe un autre test qui est généralement plus puissant, c'est-à-dire qu'il sera meilleur pour détecter que  $\mathcal{H}_0$  n'est pas vraie lorsque c'est effectivement le cas. Ce test est le test du rapport de vraisemblance (*likelihood ratio test*). Il découle de la méthode d'estimation du maximum de vraisemblance et est donc généralement applicable lorsqu'on estime les paramètres avec cette méthode. Il est basé sur la quantité  $\ell$  que nous avons vue plus tôt.

La procédure consiste à ajuster deux modèles **emboîtés** :

- Le premier modèle, le modèle complet, contient tous les paramètres et l'estimateur du maximum de vraisemblance  $\hat{\beta}$ .
- Le deuxième modèle correspondant à l'hypothèse nulle  $\mathcal{H}_0$ , le modèle réduit, contient tous les paramètres avec les restrictions imposées sous  $\mathcal{H}_0$ ; on dénote l'estimateur du maximum de vraisemblance  $\hat{\beta}_0$

Le test est basé sur la statistique

$$D = -2\{\ell(\hat{\beta}_0) - \ell(\hat{\beta})\}.$$

Cette différence  $D$ , lorsque l'hypothèse  $\mathcal{H}_0$  est vraie suit approximativement une loi khi-deux avec un nombre de degrés de liberté égal au nombre de paramètre testé (le nombre de restrictions sous  $\mathcal{H}_0$ ). On peut donc calculer la valeur- $p$  en utilisant la distribution du khi-deux.

Prenons comme exemple le test de la significativité de  $X_6$ , qui est modélisée à l'aide deux variables binaires et dont les paramètres correspondants sont  $\beta_{6_2}$  et  $\beta_{6_3}$ . Pour effectuer le test du rapport de vraisemblance, il suffit de retirer la variable  $X_6$  et de réajuster le modèle à nouveau avec toutes les autres variables.

```

# Ajuster modèle sous H0 (sans X6)
modeleH0 <- update(modele2, formula. = ". ~ . - x6")
anova(modeleH0, modele2, test = "LRT")
# Le modèle 'modeleH0' est équivalent à
# glm(y ~ x1 + x2 + x3 + x4 + x5,
# data = logit1,
# family = binomial(link = "logit"))

## Calculer statistique du test manuellement
## Deviance = -2*log vraisemblance
rvrais <- modeleH0$deviance - modele2$deviance
# Valeur-p
pchisq(rvrais, df = 2, lower.tail = FALSE)

```

Considérons maintenant la variable  $X_6$ , qui représente le nombre de fois où l'individu a assisté à un rodéo au cours de la dernière année. Cette variable est modélisée à l'aide de deux variables indicatrices,  $I(X_6 = 2)$  égale à un si  $X_6 = 2$  et zéro autrement, et  $I(X_6 = 3)$  égale à un si  $X_6 = 3$  et zéro sinon. La catégorie de référence est  $X_6 = 1$ , c'est-à-dire les personnes ayant assisté plus de 10 fois à un rodéo au cours de la dernière année. Pour tester la significativité globale d'une variable catégorielle qui est modélisée avec plusieurs indicatrices, il faut utiliser un test qui compare l'ajustement du modèle avec ou sans toutes les variables binaires associées à  $X_6$ ; l'hypothèse nulle  $\mathcal{H}_0 : \beta_{6_2} = \beta_{6_3} = 0$  versus l'alternative qu'au moins un de ces deux paramètres est différent de zéro. La statistique du test de rapport de vraisemblance  $D$  de 50.251 et la valeur- $p$  peut-être obtenue de la loi du khi-deux avec 2 degrés de liberté via le code suivant permet d'imprimer la valeur- $p$ , qui est  $1.22 \times 10^{-11}$ .

### 5.2.7 Multicolinéarité

Rappelez-vous que le terme multicolinéarité fait référence à la situation où les variables explicatives sont très corrélées entre elles ou bien, plus généralement, à la situation où une (ou plusieurs) variable(s) explicative(s) est (sont) très corrélée(s) à une combinaison linéaire des autres variables explicatives.

L'effet potentiellement néfaste de la multicolinéarité est le même qu'en régression linéaire, c'est-à-dire, elle peut réduire la précision des estimations des paramètres (augmenter leurs écarts-types estimés).

En pratique, le problème est qu'il devient difficile de départager l'effet individuel d'une variable explicative lorsqu'elle est fortement corrélée avec d'autres variables explicatives.

Comme la multicolinéarité est une propriété des variables explicatives (le  $Y$  n'intervient pas) on peut utiliser les mêmes outils qu'en régression linéaire pour tenter de la détecter, par exemple,

le facteur d'inflation de la variance (*variance inflation factor*), accessible via `car::vif` pour un modèle de régression. Cette quantité ne dépend que des variables explicatives  $\mathbf{X}$ , pas du modèle ou de la variable réponse.

La multicolinéarité est surtout un problème lorsque vient le temps d'interpréter et tester l'effet des paramètres individuels. Si le but est seulement de faire de la classification (prédiction) et que l'interprétation des paramètres individuels n'est pas cruciale alors il n'y a pas lieu de se soucier de la multicolinéarité. Il faut alors plutôt comparer correctement la performance de classification des modèles en utilisant des méthodes permettant d'obtenir un bon modèle tout en se protégeant contre le surajustement. Certaines de ces méthodes (division de l'échantillon, validation croisée) ont déjà été présentées.

### 5.3 Classification et prédiction

La finalité du modèle de régression logistique est fréquemment l'obtention de prédictions. Une fois qu'on a ajusté un modèle, on peut l'utiliser pour prévoir la valeur de  $Y$  pour de nouvelles observations. Ceci consiste à assigner une classe (0 ou 1) à ces observations (pour lesquels  $Y$  est inconnue) à partir des valeurs prises par  $X_1, \dots, X_p$ .

Le modèle ajusté nous fournit une estimation de  $\Pr(Y = 1 | \mathbf{X} = \mathbf{x})$  pour des valeurs  $X_1 = x_1, \dots, X_p = x_p$  données. Cet estimé est

$$\hat{p} = \frac{1}{1 + \exp\{-(\hat{\beta}_0 + \hat{\beta}_1 x_1 + \dots + \hat{\beta}_p x_p)\}}.$$

Classification de base : pour classifier des observations, il suffit de choisir un point de coupure  $c$ , souvent  $c = 0.5$ , et de classifier une observation de la manière suivante :

- Si  $\hat{p} < c$ , on assigne cette observation à la catégorie zéro et  $\hat{Y} = 0$ .
- Si  $\hat{p} \geq c$ , on assigne cette observation à la catégorie un et  $\hat{Y} = 1$ .

Si on prend  $c = 0.5$  comme point de coupure, cela revient à assigner l'observation à la classe (catégorie) la plus probable, un choix fort raisonnable. Nous verrons dans une section suivante que, lorsque les conséquences de faussement classifier une observation (succès, mais échec prédit et vice-versa) ne sont pas les mêmes, il peut être avantageux d'utiliser un autre point de coupure.

Dans un cadre de prédiction, il nous faudra un critère pour juger de la qualité de l'ajustement du modèle. Rappelez-vous que pour une réponse continue, nous avons utilisé l'erreur quadratique moyenne,  $\text{EQM} = E\{(Y - \hat{Y})^2\}$ , où  $\hat{Y} = E(Y | \mathbf{X})$ , pour juger de la performance d'un modèle. Comme la réponse  $Y$  est binaire ici, nous allons utiliser des critères différents.

Voyons d'abord un premier critère pour juger de la qualité d'un modèle de prédiction. Soit  $Y$  la vraie valeur de la réponse binaire et  $\hat{Y}$  (soit 0 ou 1) la valeur de  $Y$  prédite par un modèle pour une

observation choisie au hasard dans la population. Un premier critère pour juger de la performance d'un modèle est le **taux de mauvaise classification**, un estimé de la probabilité de mal classifier une observation choisie au hasard dans la population,  $\Pr(Y \neq \hat{Y})$ . Plus  $\Pr(Y \neq \hat{Y})$  est petite, meilleure est la capacité prédictive du modèle.

Tout comme l'erreur quadratique moyenne, on ne peut qu'estimer  $\Pr(Y \neq \hat{Y})$ . Pour les raisons vues au chapitre précédent, l'estimer en calculant le taux de mauvaise classification des observations ayant servi à l'ajustement du modèle sans aucune correction n'est pas une bonne approche. Les approches couvertes dans le dernier chapitre pour l'estimation de l'erreur quadratique moyenne, telles la validation-croisée et la division de l'échantillon, peuvent être utilisées pour estimer le taux de mauvaise classification  $\Pr(Y \neq \hat{Y})$ .

Cette utilisation d'un modèle de régression logistique sera illustrée avec l'exemple que nous avons traité au chapitre précédent : notre objectif final est de construire un modèle avec les 1000 clients de l'échantillon d'apprentissage et cibler ensuite lesquels des 100 000 clients restants seront choisis pour recevoir le catalogue. Les variables cibles sont :

- *yachat* : variable binaire égale à un si le client a acheté quelque chose dans le catalogue et zéro sinon.
- *ymontant* : le montant de l'achat si le client a acheté quelque chose

Les 10 variables suivantes sont disponibles pour tous les clients et serviront de variables explicatives,

- *x1* : sexe de l'individu, soit homme (0) ou femme (1);
- *x2* : l'âge (en année);
- *x3* : variable catégorielle indiquant le revenu, soit moins de 35 000\$ (1), entre 35 000\$ et 75 000\$ (2) ou plus de 75 000\$ (3);
- *x4* : variable catégorielle indiquant la région où habite le client (de 1 à 5);
- *x5* : conjoint : le client a-t-il un conjoint, soit oui (1) ou non (0);
- *x6* : nombre d'année depuis que le client est avec la compagnie;
- *x7* : nombre de semaines depuis le dernier achat;
- *x8* : montant (en dollars) du dernier achat;
- *x9* : montant total (en dollars) dépensé depuis un an;
- *x10* : nombre d'achats différents depuis un an.

Dans le chapitre précédent, nous avons cherché à développer un modèle pour prévoir *ymontant*, le montant dépensé, étant donné que le client achète quelque chose. Cette fois-ci, nous allons travailler avec la variable *yachat*, qui est binaire, à l'aide de la régression logistique.

Afin d'introduire différentes notions, nous allons, dans un premier temps, utiliser les 10 variables de base. À partir de la section suivante, nous chercherons à optimiser le modèle en considérant les interactions d'ordre deux.

```

data(dbm, package = "hecmulti")
formule <- formula("yachat ~ x1 + x2 + x3 +
                     x4 + x5 + x6 + x7 + x8 + x9 + x10")
dbm_class <- dbm |>
  dplyr::filter(test == 0) |>
  dplyr::mutate(yachat = factor(yachat))
set.seed(202209)
predprob <- hecmulti::predvc(
  modele = glm(formula = formule,
                data = dbm_class,
                family = binomial),
  K = 10, #nombre de plis
  nrep = 1,
  type = "response")
classif <- with(dbm, yachat[test == 0])
# Tableau de la performance
hecmulti::perfo_logistique(
  prob = predprob,
  resp = classif)

```

Le modèle utilise seulement les 10 variables de base. Des prévisions pour les clients restants seront exportées dans le fichier. La méthode predict permet d'obtenir les prédictions des probabilités et la fonction maison hecmulti::perfo\_logistique retourne un tableau de classification. Tel que nous l'avons vu au chapitre précédent, il y a 210 clients qui ont acheté quelque chose parmi les 1000.

TABLEAU 5.4 – Tableau de classification avec le nombre de vrais positifs (VP), vrais négatifs (VN), faux positifs (FP) et faux négatifs (FN), le taux de bonne classification, la sensibilité, la spécificité, le taux de vrais positifs (TVP) et le taux de vrais négatifs (TVN).

coupe	VP	VN	FP	FN	correct (%)	sensibilité (%)	spécificité (%)	TVP	TVN
0.02	210	209	581	0	41.9	100.0	26.5	26.5	100.0
0.04	207	320	470	3	52.7	98.6	40.5	30.6	99.1
0.06	201	398	392	9	59.9	95.7	50.4	33.9	97.8
0.08	199	451	339	11	65.0	94.8	57.1	37.0	97.6
0.10	193	480	310	17	67.3	91.9	60.8	38.4	96.6
0.12	191	512	278	19	70.3	91.0	64.8	40.7	96.4
0.14	184	547	243	26	73.1	87.6	69.2	43.1	95.5
0.16	176	572	218	34	74.8	83.8	72.4	44.7	94.4
0.18	172	598	192	38	77.0	81.9	75.7	47.3	94.0

0.20	164	611	179	46	77.5	78.1	77.3	47.8	93.0
0.22	162	626	164	48	78.8	77.1	79.2	49.7	92.9
0.24	158	639	151	52	79.7	75.2	80.9	51.1	92.5
0.26	153	645	145	57	79.8	72.9	81.6	51.3	91.9
0.28	150	657	133	60	80.7	71.4	83.2	53.0	91.6
0.30	143	667	123	67	81.0	68.1	84.4	53.8	90.9
0.32	138	679	111	72	81.7	65.7	85.9	55.4	90.4
0.34	134	695	95	76	82.9	63.8	88.0	58.5	90.1
0.36	130	699	91	80	82.9	61.9	88.5	58.8	89.7
0.38	126	708	82	84	83.4	60.0	89.6	60.6	89.4
0.40	120	715	75	90	83.5	57.1	90.5	61.5	88.8
0.42	115	723	67	95	83.8	54.8	91.5	63.2	88.4
0.44	112	731	59	98	84.3	53.3	92.5	65.5	88.2
0.46	109	736	54	101	84.5	51.9	93.2	66.9	87.9
0.48	106	739	51	104	84.5	50.5	93.5	67.5	87.7
0.50	100	744	46	110	84.4	47.6	94.2	68.5	87.1
0.52	98	748	42	112	84.6	46.7	94.7	70.0	87.0
0.54	92	750	40	118	84.2	43.8	94.9	69.7	86.4
0.56	87	753	37	123	84.0	41.4	95.3	70.2	86.0
0.58	83	761	29	127	84.4	39.5	96.3	74.1	85.7
0.60	80	766	24	130	84.6	38.1	97.0	76.9	85.5
0.62	77	769	21	133	84.6	36.7	97.3	78.6	85.3
0.64	74	771	19	136	84.5	35.2	97.6	79.6	85.0
0.66	68	772	18	142	84.0	32.4	97.7	79.1	84.5
0.68	62	774	16	148	83.6	29.5	98.0	79.5	83.9
0.70	54	775	15	156	82.9	25.7	98.1	78.3	83.2
0.72	51	777	13	159	82.8	24.3	98.4	79.7	83.0
0.74	49	778	12	161	82.7	23.3	98.5	80.3	82.9
0.76	46	778	12	164	82.4	21.9	98.5	79.3	82.6
0.78	41	781	9	169	82.2	19.5	98.9	82.0	82.2
0.80	35	783	7	175	81.8	16.7	99.1	83.3	81.7
0.82	33	783	7	177	81.6	15.7	99.1	82.5	81.6
0.84	32	783	7	178	81.5	15.2	99.1	82.1	81.5
0.86	28	784	6	182	81.2	13.3	99.2	82.4	81.2
0.88	25	786	4	185	81.1	11.9	99.5	86.2	80.9
0.90	21	787	3	189	80.8	10.0	99.6	87.5	80.6
0.92	18	787	3	192	80.5	8.6	99.6	85.7	80.4

TABLEAU 5.5 – Matrice de confusion avec point de coupure 0.465.

		$Y = 1$		$Y = 0$				
		$\hat{Y} = 1$	109	52				
		$\hat{Y} = 0$	101	738				
0.94	14	788	2	196	80.2	6.7	99.7	87.5
0.96	6	788	2	204	79.4	2.9	99.7	75.0
0.98	2	790	0	208	79.2	1.0	100.0	100.0
								79.2

Le Tableau 5.4 contient des estimations de plusieurs quantités intéressantes rattachées à la classification, en faisant varier le point de coupure. Pour chaque point de coupure, ces estimations ont été obtenues par validation croisée à  $n$  groupes (en anglais, *leave-one-out cross-validation*, ou LOOCV). Ainsi, ces estimations sont meilleures que les estimés sans ajustement aucun car elles ne sont pas obtenues en utilisant les mêmes observations que celles qui ont servi à estimer le modèle.

La colonne **Correct** donne le taux de bonne classification,  $\Pr(Y = \hat{Y})$  : avec un point de coupure de 0, on classifie toutes les observations à la classe achat (1), car  $\hat{p}$  est forcément plus grande que zéro. Le taux de bonne classification dans ce cas de figure sera de 21%, puisque 210 individus ont acheté un produit dans le catalogue dans l'échantillon d'apprentissage. L'autre extrême, avec un point de coupure  $c = 1$ , donne un taux de bonne classification de 79%.

On peut chercher dans le tableau les points de coupure qui donnent le meilleur taux de bonne classification. Ce dernier, à savoir 84.6%, est atteint par trois points de coupure, soit 0.52, soit 0.6, soit 0.62. Une recherche plus fine donne 0.465 comme point de coupure optimal, avec un taux de mauvaise classification de 15.3%.

Avec une variable réponse binaire, il y a deux classifications possibles et le tableau de confusion contient, en partant du coin supérieur gauche et dans le sens des aiguilles d'une montre, le nombre de vrai positif ( $Y = 1, \hat{Y} = 1$ ), de faux positif ( $Y = 0, \hat{Y} = 1$ ), de vrai négatif ( $Y = 0, \hat{Y} = 0$ ) et finalement de faux négatif ( $Y = 1, \hat{Y} = 0$ ). La **matrice de confusion**, qui compare les vraies valeurs avec les prédictions, peut être construite à partir des colonnes VP, VN, FP et FN. Ces nombres proviennent de la validation croisée à  $n$  groupes et ne sont pas ceux qu'on obtiendrait si on appliquait directement le modèle ajusté à notre échantillon. Le taux de mauvaise classification est  $(FP + FN)/n$ ; une estimation plus fiable serait obtenue en utilisant la validation croisée à 10 groupes.

Quatre autres quantités, dérivées à partir de la matrice de confusion, sont parfois utilisées :

- la **sensibilité** (*sensitivity*),  $\Pr(\hat{Y} = 1 | Y = 1)$ , ou  $VP/(VP + FN)$ ;
- la **spécificité** (*specificity*),  $\Pr(\hat{Y} = 0 | Y = 0)$ , ou  $VN/(VN + FP)$ ;
- le **taux de vrais positifs**,  $\Pr(Y = 1 | \hat{Y} = 1)$ , ou  $VP/(VP + FP)$ ;
- le **taux de vrais négatifs**,  $\Pr(Y = 0 | \hat{Y} = 0)$ , ou  $VN/(VN + FN)$ .

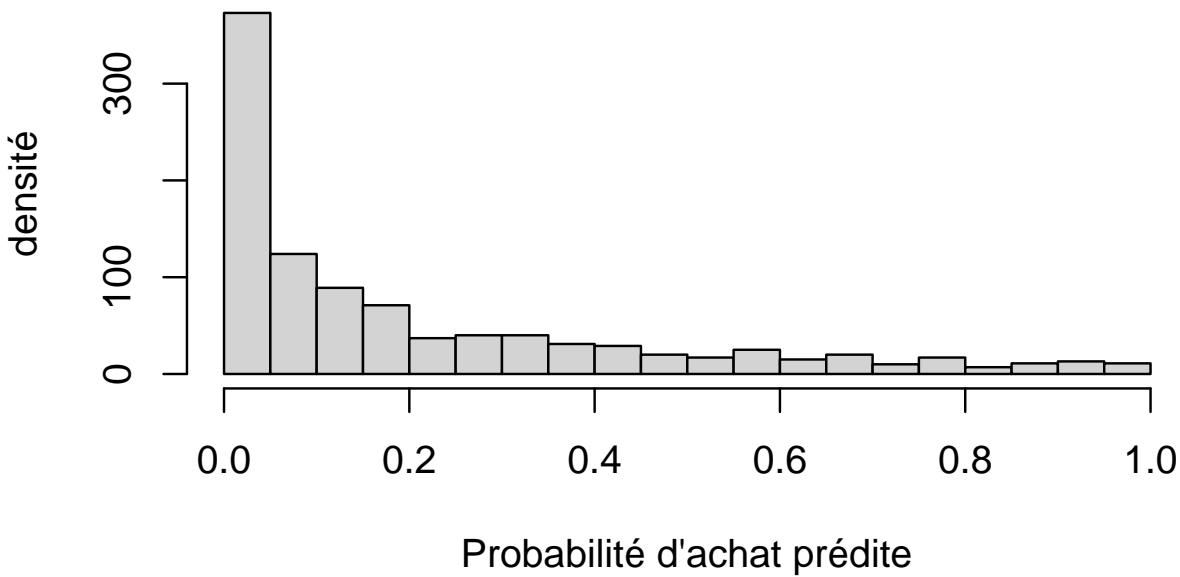


FIGURE 5.4 – Répartition des probabilités de succès prédites par validation croisée à  $n$  groupes.

Les estimés empiriques sont simplement obtenus en calculant les rapports du nombre d'observations dans chaque classe.

La sensibilité mesure à quel point notre modèle est performant pour détecter un vrai positif (classe 1). La spécificité mesure à quel point notre modèle est performant pour détecter un résultat négatif (classe 0). Plus le point de coupure augmente, plus la sensibilité et le taux de faux positifs diminuent mais plus la spécificité et le taux de faux négatifs augmentent.

### 5.3.1 Fonction d'efficacité du récepteur

La **fonction d'efficacité du récepteur**, parfois appelée courbe ROC (*receiver operating characteristic*) est parfois utilisée pour représenter globalement la performance du modèle. Il s'agit du graphe de la sensibilité en fonction de un moins la spécificité, en faisant varier le point de coupure. Un modèle parfait aurait une sensibilité et une spécificité égales à 1 (correspondant au coin supérieur gauche de la fonction d'efficacité du récepteur). Ainsi, plus le couple (1–spécificité, sensibilité) est près de (0, 1), meilleur est le modèle. Par conséquent, plus la courbe ROC tend vers (0, 1) meilleur est le pouvoir prévisionnel des variables. L'**aire sous la courbe** (*area under the curve*, ou AUC) est souvent utilisée en parallèle comme mesure de la qualité : comme son nom l'indique, c'est l'aire sous la courbe de la fonction d'efficacité du récepteur.

La fonction `courbe_roc` permet de tracer la courbe et de calculer l'aire sous la courbe. Plus cette valeur est près de 1, mieux c'est : une probabilité de 0.5 correspond à une allocation aléatoire, représentée sur la fonction d'efficacité du récepteur par la ligne diagonale.

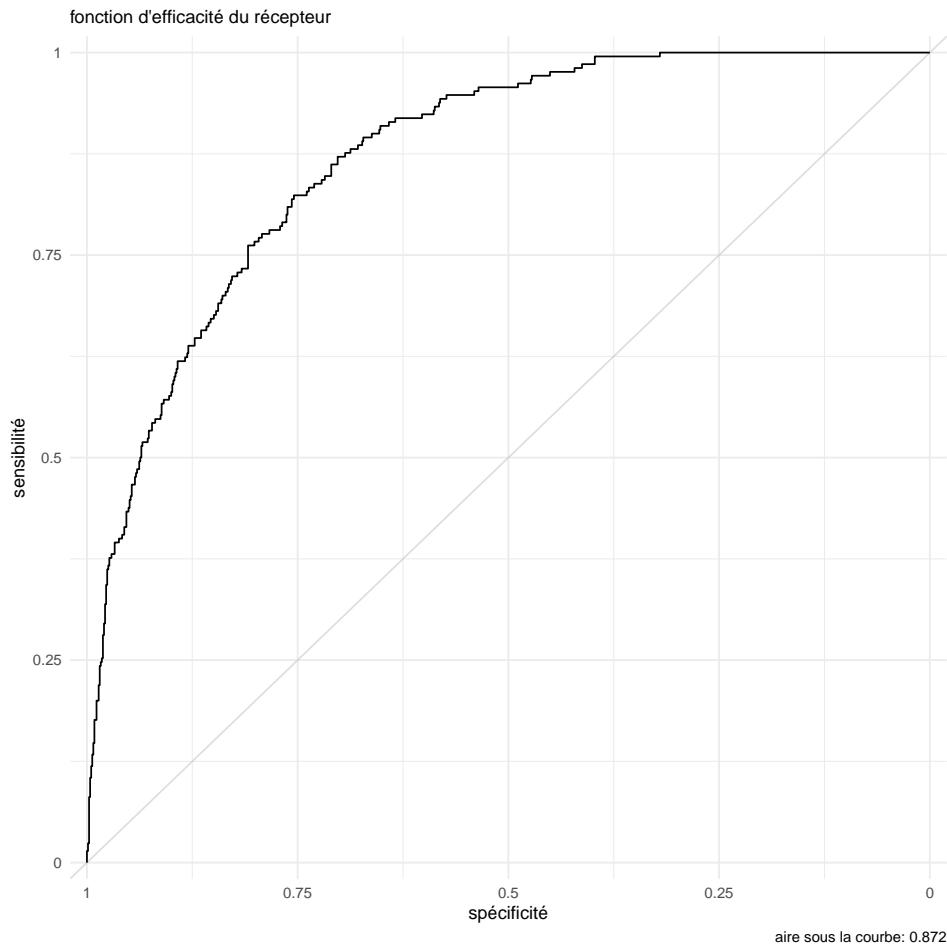


FIGURE 5.5 – Fonction d'efficacité du récepteur avec probabilités de succès issues de la validation croisée à  $n$  groupes.

```
# Fonction d'efficacité du récepteur
roc <- hecmulti::courbe_roc(
  resp = classif,
  prob = predprob,
  plot = TRUE)
print(roc)
## Pour extraire l'aire sous la courbe,
# roc$aire
```

### 5.3.2 Classification avec une matrice de gain

Utiliser le taux de mauvaise classification  $\Pr(Y \neq \hat{Y})$ , comme critère de performance, revient au même que d'utiliser le taux de bonne classification  $\Pr(Y = \hat{Y})$ , car  $\Pr(Y \neq \hat{Y}) = 1 - \Pr(Y = \hat{Y})$ . On veut un modèle avec un haut taux de bonne classification (ou un faible taux de mauvaise classification).

Lorsqu'on utilise  $\Pr(Y \neq \hat{Y})$  comme critère pour juger de la qualité d'un modèle prévisionnel, on fait l'hypothèse que le gain associé à bien classifier une observation dans la catégorie 0 lorsqu'elle est réellement dans la catégorie 0 est le même que celui associé à classifier une observation dans la catégorie 1 lorsqu'elle est réellement dans la catégorie 1 : cela correspond à la matrice de gain.

TABLEAU 5.6 – Matrice de gain correspondant au taux de bonne classification

		observation	
		gain	
prédition	$\hat{Y} = 1$		$Y = 1$
	$\hat{Y} = 0$		$Y = 0$

Le gain vaut 1 lorsque la prévision est bonne (les deux cas sur la diagonale) et 0 lorsque le modèle se trompe (les deux autres cas). L'unité de mesure du gain n'est pas importante pour l'instant. Le gain total est

$$\begin{aligned} \text{gain} &= 1 \Pr(\hat{Y} = 1, Y = 1) + 1 \Pr(\hat{Y} = 0, Y = 0) \\ &\quad + 0 \Pr(\hat{Y} = 1, Y = 0) + 0 \Pr(\hat{Y} = 0, Y = 1) \\ &= \Pr(Y = \hat{Y}). \end{aligned}$$

Maximiser le gain total revient donc à maximiser le taux de bonne classification.

Dans certaines situations, les gains (ou la perte si le gain est négatif) associés aux bonnes décisions et aux erreurs ne sont pas équivalents.

Supposons que le gain de classer une observation à  $i$  ( $i \in \{0, 1\}$ ) lorsqu'elle vaut  $j$  ( $j \in \{0, 1\}$ ) en réalité est de  $c_{ij}$ . La matrice de gain est alors

TABLEAU 5.7 – Matrice de gain pondérée en fonction d'un coût

		observation	
		gain	
prédition	$\hat{Y} = 1$		$Y = 1$
	$\hat{Y} = 0$		$Y = 0$

TABLEAU 5.8 – Statistiques descriptives des montants d'achats pour la base de données marketing (échantillon d'apprentissage).

n	moyenne	écart-type	minimum	maximum
210	67.29	13.24	25	109

En pratique, l'une de ces quatre quantités peut être fixée à 1 car seulement les poids relatifs (les ratios) des gains sont importants. Dans ce cas, le gain moyen est

$$\begin{aligned} \text{gain} &= c_{11} \Pr(\hat{Y} = 1, Y = 1) + c_{00} \Pr(\hat{Y} = 0, Y = 0) \\ &\quad + c_{10} \Pr(\hat{Y} = 1, Y = 0) + c_{01} \Pr(\hat{Y} = 0, Y = 1) \end{aligned}$$

Le meilleur modèle est alors celui qui maximise le gain moyen.

Nous allons encore une fois seulement utiliser les 10 variables de base. Mais nous allons intégrer des revenus et coûts afin de trouver le meilleur point de coupure. Rappelez-vous que le coût de l'envoi d'un catalogue est de 10\$. Le tableau des variables descriptives qui suit montre que, pour les 210 clients qui ont acheté quelque chose, le revenu moyen est de 67.29\$ (moyenne de la variable *ymontant*).

Nous allons travailler en termes de revenu net. Nous pouvons donc spécifier la matrice de gain du Tableau 5.9 pour notre problème. Si on n'envoie pas de catalogue, notre gain est nul. Si on envoie le catalogue à un client qui n'achète pas, on perd 10\$ (le coût de l'envoi). En revanche, notre revenu net est de 57\$ (revenu moyen moins coût de l'envoi).

TABLEAU 5.9 – Matrice de gain pour l'envoi de catalogue

		observation	
		gain	
prédiction	$\hat{Y} = 1$	57	-10
	$\hat{Y} = 0$	0	0

On peut calculer la performance du modèle et le gain moyen en faisant varier le point de coupure. Pour avoir une mesure fidèle, on utilise la validation croisée à  $K = 10$  groupes (la mesure affichée correspondant à la moyenne de 10 réplications)

```
data(dbm, package = "hecmulti")
donnees <- dbm |>
  dplyr::filter(test == 0)
formule = formula(yachat ~ x1 + x2 + x3 +
  x4 + x5 + x6 + x7 +
```

```

x8 + x9 + x10)
modele <- glm(formule,
               family = binomial,
               data = donnees)
coupe <- hecmulti::select_pcoupe(
  modele = modele,
  c00 = 0,
  c01 = 0,
  c10 = -10,
  c11 = 57,
  plot = TRUE)
coupe

```

Point de coupure optimal: 0.01

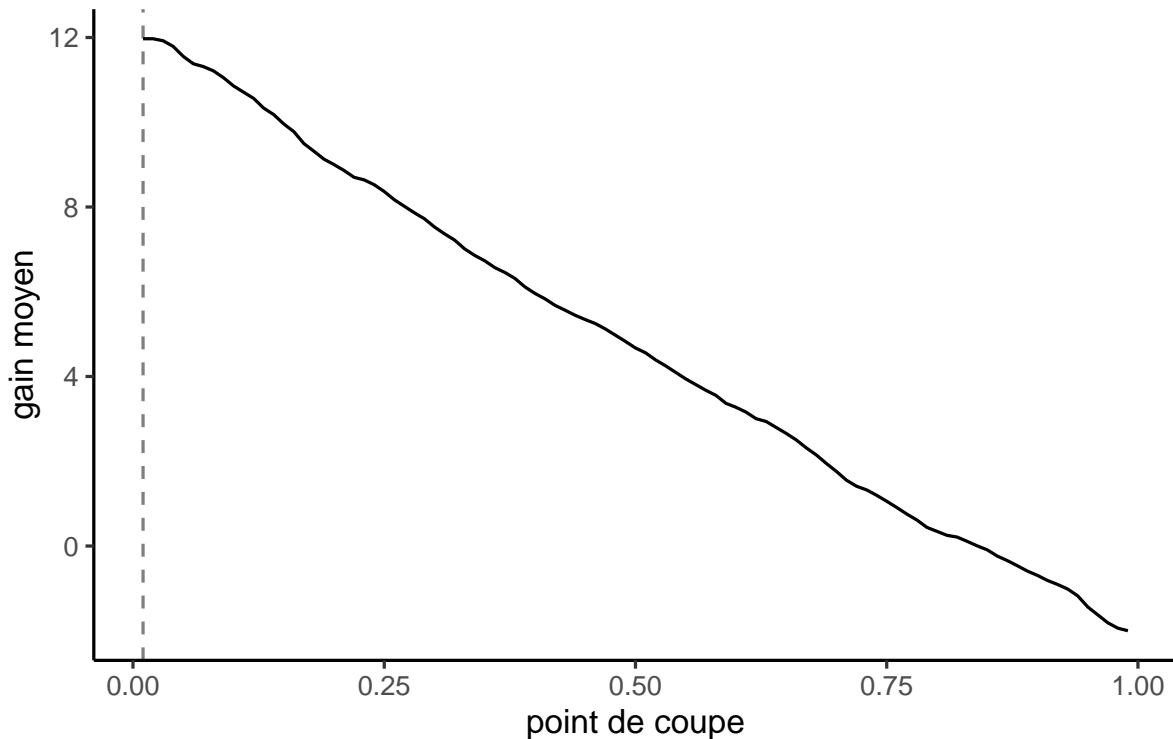


FIGURE 5.6 – Estimation du gain moyen en fonction du point de coupure pour l'exemple de base de données marketing.

La fonction `select_pcoup` donne l'estimation du gain moyen (`gain`) pour différents points de coupures (`pcoupe`). Cette estimation provient d'une validation-croisée avec  $K$  groupes (`ncv`) dans la fonction), répétée `nrep` fois. On a effectué ici la validation croisée avec 10 groupes et fait la moyenne des 10 répétitions afin d'avoir plus de précision.

On voit dans la Figure 5.6 que le meilleur point de coupure, celui qui maximise le gain est 0.01. Avec ce point de coupure, et selon le Tableau 5.4, on estime que le taux de bonne classification est de 70.3 et que la sensibilité est de 90.95. Ainsi, on estime qu'on va détecter environ 91% des clients qui achètent.

Comme il est très coûteux de rater un client qui aurait acheté quelque chose, il est préférable d'envoyer le catalogue à plus de clients, quitte à ce que plusieurs d'entre eux n'achètent rien. Bien que le point de coupure de 0.5 donne un meilleur taux de bonne classification, il correspond à un gain moyen plus faible car on rate trop de clients qui achètent (la sensibilité est de seulement 47.62%). Travailler avec la matrice de gain permet de trouver le point de coupure optimal en incorporant des notions de coûts et profits.

### 5.3.3 Courbe lift

Un autre type de graphe qui est souvent utilisé dans des contextes de gestion est la courbe lift (sic) (en anglais, *lift chart*). Cette courbe est obtenue en ordonnant les probabilités de succès estimées par le modèle,  $\hat{p}$ , en ordre croissant et en regardant quelle pourcentage de ces derniers seraient bien classifiés (le nombre de vrais positifs sur le nombre de succès).

```
tab_lift <- hecmulti::courbe_lift(
  prob = predprob, # probabilité de succès (Y=1)
  resp = classif, # variable binaire réponse 0/1
  plot = TRUE)
tab_lift
```

Le Tableau 5.10 présente les 10 déciles. Si on classifiait comme acheteurs les 10% qui ont la plus forte probabilité estimée d'achat, on détecterait 81 des 210 clients (37.6%). En comparaison, on s'attend que 21 clients soient sélectionnés en moyenne si on prend un échantillon aléatoire de 100 personnes. Le ratio 81/21 (dernière colonne) est le *lift* du modèle : il permet de détecter 3.86 fois plus de succès que le hasard.

La Figure 5.7 présente le pourcentage d'observations bien classées parmi les variables (pourcentage des probabilités prédites qui correspondent à un succès parmi les  $k$  plus susceptibles selon le modèle). La référence est la ligne diagonale, qui correspond à une détection aléatoire.

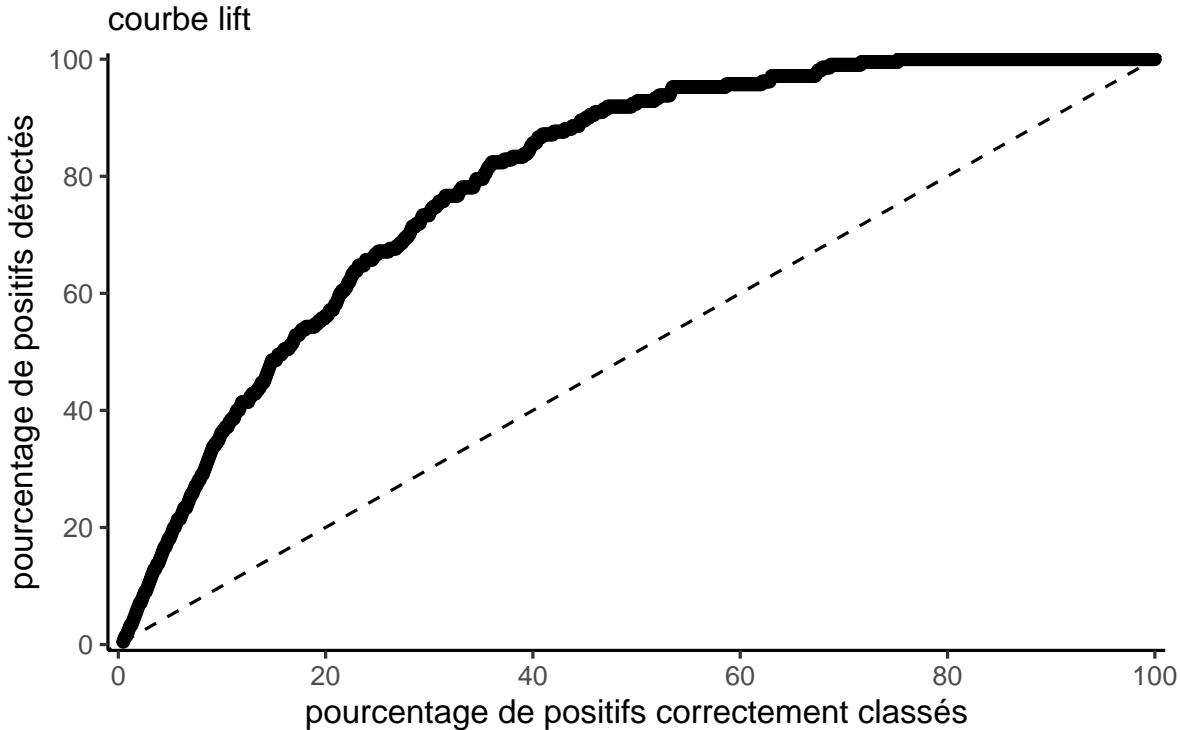


FIGURE 5.7 – Courbe lift

### 5.3.4 Calibration du modèle et détection du surajustement

Il peut être intéressant de vérifier la **calibration** de notre modèle, et une statistique simple proposée par Spiegelhalter (1986) peut être utile à cette fin. Pour une variable binaire  $Y \in \{0, 1\}$ , l'erreur quadratique moyenne s'écrit

$$\bar{B} = \frac{1}{n} \sum_{i=1}^n (Y_i - p_i)^2 = \frac{1}{n} \sum_{i=1}^n (Y_i - p_i)(1 - 2p_i) + \frac{1}{n} \sum_{i=1}^n p_i(1 - p_i).$$

Le premier terme représente le manque de calibration du modèle, tandis que le deuxième correspond à la séparation entre variables. Si notre modèle était parfaitement calibré, alors  $E_0(Y_i) = p_i$  et  $Va_0(Y_i) = p_i(1 - p_i)$ . On peut utiliser ce fait pour construire une statistique de test de la forme  $Z = \{\bar{B} - E_0(\bar{B})\} / \sqrt{Va_0(\bar{B})}$ , où

$$E_0(\bar{B}) = \frac{1}{n} \sum_{i=1}^n p_i(1 - p_i)$$

$$Va_0(\bar{B}) = \frac{1}{n^2} \sum_{i=1}^n p_i(1 - p_i)(1 - 2p_i)^2$$

TABLEAU 5.10 – Tableau du lift (déciles).

	pourcent	hasard	modele	lift
10%	10	21	78	3.714286
20%	20	42	120	2.857143
30%	30	63	157	2.492063
40%	40	84	180	2.142857
50%	50	105	195	1.857143
60%	60	126	201	1.595238
70%	70	147	208	1.414966
80%	80	168	210	1.250000
90%	90	189	210	1.111111

Sous l'hypothèse nulle de calibration parfaite,  $Z \sim \text{No}(0, 1)$  en grand échantillon. Pour le modèle simple avec toutes les covariables, la valeur- $p$  approximative calculée avec les probabilités de succès obtenues par validation-croisée et les données de l'échantillon d'apprentissage est 0.22 et il n'y a pas de preuve ici que le modèle est mal calibré. Cette technique est utile pour vérifier s'il n'y a pas de surajustement (auquel cas le modèle tend à retourner des probabilités très près de 0/1, mais qui ne correspondent pas à la réalité).

Ici, nous avons ajusté un seul modèle, celui contenant uniquement les 10 variables de base et nous nous sommes attardés au choix du point de coupure pour l'assignation aux classes. Il est possible qu'un autre modèle, contenant par exemple des termes d'interactions, des termes quadratiques ou d'autres transformations des variables, soit supérieur à celui-ci. Le choix du modèle de prévision se fait donc souvent en deux étapes :

1. choisir les variables explicatives
2. sélectionner un point de coupure.

Nous avons déjà vu des méthodes de sélections de variables au chapitre précédent. La section suivante reviendra sur ces méthodes dans le contexte de la régression logistique.

### 5.3.5 Sélection de variables en régression logistique

Les principes généraux, concernant la sélection de variables et de modèles, que nous avons vus au chapitre précédent sont toujours valides. Les critères AIC et BIC sont toujours disponibles puisqu'on estime le modèle par maximum de vraisemblance et les techniques générales de division de l'échantillon et de validation-croisée sont toujours valides. La principale différence est le coût d'estimation numérique des différents modèles : parce qu'il n'y a pas de solution explicite pour les

estimateurs du maximum de vraisemblance du modèle logistique, ajuster chacun de ces modèles est coûteux.

À la section précédente, nous avons inclus les 10 variables de base dans notre exemple d'envoi ciblé. Nous allons ici faire une recherche de type exhaustive parmi ces variables. La fonction `glmbb` du paquet éponyme fait une recherche à l'aide de l'algorithme de recherche arborescente dite par méthode de séparation et d'évaluation, qui ne nécessite pas de tester tous les modèles emboîtés. La sortie inclut les modèles qui sont à distance au plus cutoff du modèle optimal en ordre décroissant du critère d'information, avec une pondération associée qui peut servir comme succédané au mélange de modèle. La fonction permet de choisir entre les critères AIC et BIC et inclut toutes les modalités des variables explicatives pour les facteurs.

```
data(dbm, package = "hecmulti")
donnees <- dbm |> dplyr::filter(test == 0)
formule <- formula(yachat ~ x1 + x2 + x3 +
                     x4 + x5 + x6 + x7 +
                     x8 + x9 + x10)

select_BIC <-
  glmbb::glmbb(formule,
                data = donnees,
                criterion = "BIC",
                family = binomial(link = "logit"))
resultat_BIC <- summary(select_BIC)
# Formule du meilleur modèle
resultat_BIC$results$formula
# Valeurs de BIC des modèles
resultat_BIC$results$criterion
```

Ceci n'est qu'un exemple de stratégie de sélection de modèle parmi tant d'autre : le code qui suit explorera d'autres alternatives. Nous allons évaluer la performance de ces différentes stratégies avec comme critère de performance le revenu net de la stratégie si elle était appliquée aux 100 000 clients restants. Pour chacun des 100 000 clients à catégoriser, nous allons calculer la quantité suivante :

- Si le client n'est pas ciblé pour l'envoi d'un catalogue par le modèle, alors le revenu est nul.
- Si le client est ciblé pour l'envoi d'un catalogue par le modèle et qu'il n'achète rien, le revenu est de  $-10\$$  (le coût de l'envoi).
- Si le client est ciblé pour l'envoi d'un catalogue par le modèle et qu'il achète quelque chose, le revenu est de  $(ymontant - 10)\$$ , c'est-à-dire, le montant qu'il dépense moins le  $10\$\text{ du coût de l'envoi.}$

Pour une stratégie donnée, chaque individu n'appartient qu'à une seule des catégories. Le revenu net de la stratégie est la somme des revenus pour les 100 000 clients. Parmi ces derniers, 23 179

auraient acheté si on leur avait envoyé le catalogue et ces clients auraient généré des revenus de 1 601 212\$. Si on enlève le coût des envois ( $100\ 000 \times 10\$ = 1\ 000\ 000\$$ ), on obtient que la stratégie de référence permet un revenu net de 601 212\$.

Dans ce cas, nous allons estimer la probabilité d'achat avec un modèle de régression logistique. Nous allons ensuite trouver le meilleur point de coupure, avec une matrice de gain adéquatement choisie, afin d'avoir une règle d'assignation optimale. Nous avons déterminé des modèles potentiels à la section précédente. De plus, nous avons déjà vu comment trouver le meilleur point de coupure en spécifiant une matrice de gain, afin de maximiser le gain moyen à partir de la matrice de gain du Tableau 5.9. Nous allons donc trouver le meilleur point de coupure pour quelques-uns des modèles choisis à la section précédente, pour ensuite évaluer le revenu net de ces modèles.

Il faut encore une fois bien comprendre qu'en pratique, on ne pourrait pas faire cette comparaison, car on ne sait pas d'avance si les clients futurs vont acheter ou non. Mais dans cet exemple, les variables *yachat* et *ymontant* sont fournies pour ces 100 000 clients afin qu'on puisse voir ce qui se serait passé avec les différentes stratégies.

La stratégie de référence est celle qui consiste à envoyer le catalogue aux 100 000 clients sans les sélectionner. Le tableau qui suit montre des statistiques pour les variables *ymontant* et *yachat* pour les 100 000 clients à scorer. Le Tableau 5.11 résume la performance des différentes stratégies basées exclusivement sur le modèle logistique.

En résumé, la procédure numérique à réaliser est la suivante :

- Choisir les variables à essayer (interactions, etc.)
- Choisir l'algorithme ou la méthode de sélection
- Obtenir un modèle final et calculer le point de coupure optimal selon notre matrice de coût.
- Pour obtenir la performance finale, on obtient les prédictions pour les 100 000 clients de l'échantillon de validation et on classe pour prédire la classe de *yachat* pour les données de validation à l'aide du point de coupure optimal choisi.
- On calcule ensuite le revenu en soustrayant 10\$ pour chaque envoi et en additionnant les montants d'achats des personnes qui ont reçu le catalogue.

Quelques commentaires sur des raccourcis syntaxiques propres à R : dans une formule, spécifier `~.` indique que l'on ajoute au modèle de régression toutes les variables explicatives de la base de données, moins la variable réponse. On peut aussi utiliser `.^2` ou de manière équivalente `.*.` pour spécifier tous ces termes, ainsi que leurs interactions. Si on veut ajouter un terme quadratique pour une variable *x*, il faudra spécifier la transformation à l'intérieur de `I()`, par exemple `I(x^2)`.

```
# Diviser les bases de données
# en échantillons d'apprentissage
# et de validation
data(dbm, package = "hecmulti")
valid <- dbm[dbm$test == 1, ] |>
```

```

dplyr::select(! c(ymontant, test))
appr <- dbm[dbm$test == 0, ] |>
  dplyr::select(! c(ymontant, test))
# Formule du modèle avec toutes les interactions
# d'ordre 2 (.^2) et les termes quadratiques I(x^2)
formule <- formula(yachat ~ .^2 +
  I(x2^2) + I(x6^2) +
  I(x7^2) + I(x8^2) +
  I(x9^2) + I(x10^2))

# Nouvelles bases de données avec toutes ces variables
# On retire la première colonne (1, ordonnée à l'origine)
appr_c <- data.frame(
  cbind(model.matrix(formule, data = appr)[,-1]),
  y = as.integer(appr$yachat))
valid_c <- data.frame(
  cbind(model.matrix(formule, data = valid)[,-1]),
  y = as.integer(valid$yachat))
valid_ymontant <- with(dbm, ymontant[test == 1L])

# Ajustement des différents modèles

# Modèle avec toutes les variables principales
base <- glm(yachat ~ .,
  data = appr,
  family = binomial)
# Calcul du point de coupe optimal
# (par validation croisée)
base_coupe <- hecmulti::select_pcoupe(
  modele = base,
  c00 = 0,
  c01 = 0,
  c10 = -10,
  c11 = 57)
# Performance sur données de validation
base_pred <-
  predict(object = base,
    newdata = valid,
    type = "response") > base_coupe$optim
base_perfo <-

```

```

-10*sum(base_pred) +
  sum(valid_ymontant[base_pred], na.rm = TRUE)

# Modèle avec toutes les variables + interactions
# Ajustement
complet <- glm(formula = formule,
                 data = appr,
                 family = binomial)
# Sélection du point de coupure
complet_coupe <- hecmulti::select_pcoupe(
  modele = complet, c00 = 0,
  c01 = 0, c10 = -10, c11 = 57)
# Performance sur données de validation
complet_pred <-
  predict(object = complet,
          newdata = valid,
          type = "response") > complet_coupe$optim
# Revenu
complet_perfo <-
  -10*sum(complet_pred) +
    sum(valid_ymontant[complet_pred], na.rm = TRUE)

# Sélection de modèle avec algorithme glouton
# Recherche séquentielle (AIC)
seqAIC <- step(object = complet,
                  direction = "both", # séquentielle
                  k = 2, # AIC
                  trace = 0)
seqAIC_coupe <-
  hecmulti::select_pcoupe(
  modele = seqAIC, c00 = 0,
  c01 = 0, c10 = -10, c11 = 57)
seqAIC_pred <-
  predict.glm(object = seqAIC,
              newdata = valid,
              type = "response") >
  seqAIC_coupe$optim
seqAIC_perfo <-
  -10*sum(seqAIC_pred) +

```

```

    sum(valid_ymontant[seqAIC_pred],
        na.rm = TRUE)
# Recherche séquentielle (BIC)
seqBIC <- step(object = complet,
                 direction = "both", # séquentielle
                 k = log(nobs(complet)), #BIC
                 trace = 0)
seqBIC_coupe <- hecmulti::select_pcoupe(
  modele = seqBIC, c00 = 0,
  c01 = 0, c10 = -10, c11 = 57)
seqBIC_pred <-
  predict.glm(object = seqBIC,
               newdata = valid,
               type = "response") >
  seqBIC_coupe$optim
seqBIC_perfo <-
  -10*sum(seqBIC_pred) +
  sum(valid_ymontant[seqBIC_pred],
      na.rm = TRUE)

# Recherche exhaustive par algorithme génétique
# avec moins de variables
appr_r <- data.frame(
  cbind(model.matrix(seqAIC)[,-1],
        y = appr$yachat))
valid_r <- data.frame(
  model.matrix(formula(seqAIC),
               data = valid)[,-1])
library(glmulti)
exgen <- glmulti::glmulti(
  y = y ~ .,
  #nombre de variables limitées
  data = appr_r,
  level = 1,           # sans interaction
  method = "g",         # recherche génétique
  crit = "bic",         # critère (AIC, BIC, ...)
  confsetsize = 1,       # meilleur modèle uniquement
  plotty = FALSE,
  report = FALSE, # sans graphique ou rapport

```

```

fitfunction = "glm")

TASK: Genetic algorithm in the candidate set.
Initialization...
Algorithm started...
Improvements in best and average IC have been below the specified goals.
Algorithm is declared to have converged.
Completed.

# Redéfinir le modèle via "glm"
exgen_modele <-
  glm(exgen@objects[[1]]$formula,
      data = appr_r,
      family = binomial)
exgen_couple <-
  hecmulti::select_pcoupe(
    modele = exgen_modele,
    c00 = 0, c01 = 0, c10 = -10, c11 = 57)
exgen_pred <-
  predict(exgen_modele,
          newdata = valid_r,
          type = "response") > exgen_couple$optim
exgen_perfo <-
  -10*sum(exgen_pred) +
  sum(valid_ymontant[exgen_pred],
      na.rm = TRUE)

# LASSO
# Trouver le paramètre de pénalisation par
# validation croisée (10 groupes)
cvfit <- glmnet::cv.glmnet(
  x = as.matrix(appr_c[, -ncol(appr_c)]),
  y = appr_c$y,
  family = "binomial",
  type.measure = "auc") # aire sous courbe
# Le critère par défaut est la déviance (-211)
# Ajuster modèle avec pénalisation
lasso <- glmnet::glmnet(
  x = as.matrix(appr_c[, -ncol(appr_c)]),

```

TABLEAU 5.11 – Résumé des caractéristiques des modèles logistiques avec (a) référence, soit l'envoi sans sélection à tous les clients; (b) 10 variables de base sans sélection; (c) toutes les variables, incluant les termes quadratiques et les interactions d'ordre 2; (d) sélection séquentielle avec AIC (e) sélection séquentielle avec BIC (f) recherche exhaustive avec variables de la procédure séquentielle AIC (sélection selon BIC) (g) LASSO avec pénalité optimale selon le critère de l'aire sous la courbe. Les points de coupure optimaux ont été déterminés par validation-croisée sur l'échantillon d'apprentissage (sauf LASSO), tandis que la performance du modèle (sensibilité et taux de bonne classification) ont été calculés à partir de l'échantillon test de 100 000 individus.

modèle	no. variables	pt. coupure	sensibilité	taux bonne classif.	profit
(a)			1	0.232	601212
(b)	14	0.01	0.2674	0.366	732102
(c)	104	0.01	0.3605	0.595	918055
(d)	28	0.01	0.3245	0.52	878598
(e)	8	0.01	0.2924	0.44	804737
(f)	10	0.01	0.2978	0.455	818236
(g)	13	0.01	0.237	0.254	623345

```

y = appr_c$y,
family = "binomial",
lambda = cvfit$lambda.1se)
# Calculer performance selon les points de coupure
probs_lasso <-
  predict(lasso,
    newx = as.matrix(appr_c[,-ncol(appr_c)]),
    type = "resp")
lasso_coupe <- with(
  hecmulti::perfo_logistique(
    prob = probs_lasso,
    resp = appr_c$y),
  coupe[which.max(VP*57 - FN*10)])
lasso_pred <- c(predict(lasso,
  newx = as.matrix(valid_c[,-ncol(valid_c)]),
  type = "resp")) > lasso_coupe
lasso_perfo <- -10*sum(lasso_pred) +
  sum(valid_ymontant[lasso_pred], na.rm = TRUE)

```

Table :

Nous avons vu plus tôt, qu'avec les 10 variables de base, le meilleur point de coupure est de 0.11. En utilisant cette stratégie sur les 100 000 clients, le revenu net aurait été de 732102 dollars. C'est une énorme amélioration, de plus de 56%, par rapport à la stratégie de référence qui consiste à envoyer le catalogue à tout le monde (revenu net de 601 212\$). Si on inclut tous les termes quadratiques et les termes les interactions d'ordre deux (104 variables en tout), le revenu net est inférieur avec une valeur de 918055\$. Ici, le modèle est trop complexe et surajusté. Si on fait une sélection de variables (quasi méthodes sont présentées), suivie de la détermination du meilleur point de coupure, on fait alors toujours mieux qu'avec le modèle incluant les 10 variables de base seulement. L'approche la plus rentable parmi celles essayées aurait généré un profit de 918055 avec 104 variables explicatives : il s'agit d'un gain de 25.4% par rapport au modèle avec les 10 variables de base.

### 5.3.6 Modèle Heckit

Nous venons tout juste d'étudier des stratégies qui consistent essentiellement, à estimer  $\Pr(yachat = 1)$  et un point de coupure afin de décider à qui envoyer le catalogue en partant du postulat que tous les clients dépensent le même montant; le tout est basé uniquement sur la régression logistique. Le revenu moyen peut être estimé à partir de l'équation

$$E(ymontant) = E(ymontant | yachat = 1) \Pr(yachat = 1),$$

c'est-à-dire, la moyenne du montant dépensé est égale à la moyenne du montant dépensé étant donné qu'il y a eu achat, fois la probabilité qu'il ait eu achat. Une autre stratégie possible consiste donc à développer deux modèles : un pour  $E(ymontant | yachat = 1)$  et un autre pour  $\Pr(yachat = 1)$  et à les combiner afin d'obtenir des prévisions du montant dépensé.

#### Description du modèle Heckit

Le paragraphe qui suit est plus technique et peut être omis. Il ne serait pas justifié d'ajuster séparément les deux modèles pour  $E(ymontant | yachat = 1)$  et  $\Pr(yachat = 1)$  et de calculer les prévisions en prenant le produit :  $E(ymontant | yachat = 1) \Pr(yachat = 1)$ . Cela provient du fait que le modèle pour  $E(ymontant | yachat = 1)$  aurait été estimé seulement avec les clients qui ont acheté quelque chose et qu'ensuite on l'appliquerait (au moment de calculer les prévisions) à la fois aux clients qui vont acheter et à ceux qui ne vont pas acheter. Il y a donc un biais de sélection dans l'échantillon qui a servi à ajuster le modèle au départ. Une manière de contourner ce problème est d'ajuster conjointement les deux modèles avec un modèle de Tobit de type 2. Ce dernier est basé sur l'hypothèse que les deux variables observées ( $Y_1$  et  $Y_2$ ) proviennent de deux variables latentes non observées ( $Y_1^*$  et  $Y_2^*$ ), où

$$Y_1 = \begin{cases} 1 & \text{si } Y_1^* \geq 0, \\ 0 & \text{si } Y_1^* < 0, \end{cases} \quad Y_2 = \begin{cases} Y_2^* & \text{si } Y_1^* \geq 0, \\ 0 & \text{si } Y_1^* < 0. \end{cases}$$

Dans notre exemple,  $Y_1$  correspond à *yachat* et  $Y_2$  à *ymontant*. Ce qui lie les deux équations est le fait qu'on suppose que les variables sont binormales : les deux termes d'erreur sont de loi normale et sont corrélés,  $\boldsymbol{\varepsilon} \sim N_{\mathbf{O}_2}(\mathbf{0}_2, \boldsymbol{\Sigma})$ . Les variables dépendantes observées sont :

$$\begin{aligned} Y_1^* &= \beta_{01} + \beta_{11}X_{11} + \cdots + \beta_{1p}X_{p1} + \varepsilon_1 \\ Y_2^* &= \beta_{02} + \beta_{12}X_{12} + \cdots + \beta_{1p}X_{q2} + \varepsilon_2 \end{aligned}$$

Notez que les variables explicatives ne sont pas nécessairement les mêmes dans les deux équations. En estimant conjointement les deux équations, on élimine le biais de sélection mentionné plus haut. Le choix des variables doit être fait avant avec les méthodes qu'on a vues. Le modèle Tobit ajuste un modèle probit et non logistique à la variable binaire (la fonction de liaison).

Nous avons déjà développé des modèles de régression linéaire pour  $E(ymontant | yachat = 1)$  au chapitre précédent et nous venons de développer des modèles de régression logistique pour  $Pr(yachat = 1)$  dans ce chapitre. Nous avons donc tous les ingrédients pour implanter cette stratégie.

Nous allons cibler les clients dont la prévision du montant dépensé est plus grande que 10\$ (le coût de l'envoi du catalogue).

On pourrait faire une sélection de variables pour chaque modèle : pour faire simple, nous allons sélectionner les variables de la procédure séquentielle et choisir les variables qui donnent le modèle avec le plus petit BIC pour la partie de régression linéaire et la régression logistique.

Pour obtenir les prévisions, nous allons estimer conjointement les modèles pour  $E(ymontant | yachat = 1)$  et pour  $Pr(yachat = 1)$  avec un modèle Tobit de type 2 (aussi appelé modèle Heckit), dont une brève description est donnée à la fin de la section.

L'avantage de l'estimation simultanée est que l'on a pas à sélectionner le point de coupure, puisque l'on enverra le catalogue uniquement si le montant prédit pour  $E(ymontant)$  (non-conditionnel) est supérieur à 10\$.

```
library(sampleSelection)
formule_complet <- formula(ymontant ~
  (x1 + x2 + x3 + x4 + x5 +
   x6 + x7 + x8 + x9 + x10)^2 +
   I(x2^2) + I(x6^2) + I(x7^2) +
   I(x8^2) + I(x9^2) + I(x10^2))
select_modlin <-
  MASS::stepAIC(
  object = lm(formule_complet,
              data = dbm[dbm$test == 0,]),
  scope = formula(ymontant ~ 1),
  k = log(sum(dbm$test == 0)),
```

```

trace = FALSE)
fachat <- formula(seqBIC)
fmontant <- formula(select_modlin)
heckit.ml <- sampleSelection::heckit(
  selection = fachat,
  outcome = fmontant,
  method = "ml",
  data = dbm[dbm$test == 0,])
sortie_heckit <- summary(heckit.ml)
pred_achat <-
  predict(heckit.ml,
    part = "selection",
    newdata = dbm[dbm$test == 1,],
    type = "response") *
  predict(object = heckit.ml,
    part = "outcome",
    newdata = dbm[dbm$test == 1,])
#Remplacer valeurs manquantes par zéros
valid_ymontant[is.na(valid_ymontant)] <- 0
# On envoie le catalogue seulement si le montant d'achat prédit est supérieur à 10$

# Revenu total avec cette stratégie
heckit_perfo <-
  sum(valid_ymontant[which(pred_achat > 10)] - 10)

```

Le modèle Heckit aurait produit un revenu net de 10007980\$, un montant supérieur au revenu net de 918055\$, qui était le meilleur trouvé à la sous-section précédente.

Pour conclure cet exemple, il s'avère donc que la régression logistique permet d'effectuer un bon ciblage des clients potentiels afin de maximiser les revenus. L'approche générale consistant à obtenir des prévisions pour  $\text{Pr}(y_{achat} = 1)$  et ensuite trouver le meilleur point de coupure est très générale. D'autres types de modèles (arbre de classification, forêt aléatoire, réseau de neurones) pourraient être utilisés à la place de la régression logistique.

Nous reviendrons une dernière fois sur cet exemple dans le chapitre traitant des données manquantes. Nous verrons alors comment procéder si des valeurs manquantes sont présentes dans les variables explicatives.

## 5.4 Modèles pour données multinomiales

Supposons que la variable  $Y$  que vous cherchez à modéliser est une variable catégorielle pouvant prendre trois valeurs ou plus. Voici quelques exemples :

- Destination de vacances l'année dernière (Québec, États-Unis, ailleurs).
- Si les élections avaient lieu aujourd'hui au Québec, pour quel parti voteriez-vous (PLQ, PQ, CAQ, QS).
- Combien de fois êtes-vous allé au cinéma l'année dernière : moins de cinq fois (1), entre cinq et 10 fois (2), ou plus de 10 fois (3).
- Quelle importance accordez-vous au service après-vente ? Un parmi « pas important » (1), « peu important » (2), « moyennement important » (3), « assez important » (4), « très important » (5).

Dans les deux premiers exemples, la variable réponse  $Y$  est nominale (elle n'a pas d'ordre) alors qu'elle est ordinaire dans les deux derniers. Pour une variable ordinaire, le modèle logit multinomial peut être utilisé mais il existe d'autres possibilités comme le modèle logit cumulé. Nous couvrirons ces deux modèles.

### 5.4.1 Régression logistique multinomiale

En régression logistique,  $Y$  est une variable binaire qui vaut soit 0, soit 1 et la probabilité de succès est

$$\ln\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 X_{i1} + \cdots + \beta_p X_{ip},$$

$$p_i = \Pr(Y_i = 1 | X_i) = \text{expit}(\eta_i).$$

Dans ce modèle logistique,

$$\ln\left(\frac{p_i}{1-p_i}\right) = \ln\{\Pr(Y_i = 1 | X_i)\} - \ln\{\Pr(Y_i = 0 | X_i)\}$$

peut être vu comme étant le logit de la catégorie 1 en utilisant 0 comme catégorie de référence. Le modèle logistique multinomial procède de même en fixant une catégorie de référence et en modélisant le logit de chacune des autres catégories par rapport à la catégorie de référence. Avec  $K$  catégories ( $k = 1, \dots, K$ ) et en choisissant la catégorie 1 comme référence, le modèle devient

$$\ln\left(\frac{p_{ij}}{p_{i1}}\right) = \eta_{ij} = \beta_{0j} + \beta_{1j} X_{i1} + \cdots + \beta_{pj} X_{ip}, \quad (j = 2, \dots, K)$$

où  $p_{ik} = \Pr(Y_i = k | X_i)$  ( $k = 1, \dots, K$ ). Comme en régression logistique, on peut facilement exprimer ce modèle en termes des différentes probabilités,

$$p_{i1} = \Pr(Y_i = 1 | X_i) = \frac{1}{1 + \sum_{j=2}^K \exp(\eta_{ij})}$$

$$p_{ik} = \Pr(Y_i = k | X_i) = \frac{\exp(\eta_{ik})}{1 + \sum_{j=2}^K \exp(\eta_{ij})}, \quad k = 2, \dots, K.$$

On voit facilement que la somme des probabilités égale 1, c'est-à-dire  $p_{i1} + \dots + p_{iK} = 1$ , ce qui fait que connaître la probabilité de  $K - 1$  des catégories nous permet de déduire la dernière. En fait, le modèle logit multinomial ne fait que combiner plusieurs logit dans un seul modèle. L'interprétation des paramètres se fait comme en régression logistique sauf qu'il faut y aller équation par équation.

L'exemple qui suit traite du taux de participation lors des élections américaines et des facteurs expliquant qu'un électeur ou une électrice se prévaut de son droit de vote, ainsi que la fréquence de participation. Les données sont tirées d'un sondage Ipsos réalisé pour le site de nouvelles *FiveThirtyEight*. Les données sont accompagnées de pondérations provenant du recensement permettant de corriger la représentativité du sondage et de refléter l'électorat américain dans sa globalité.

La base de données `vote` contient 5837 observations obtenues par voie de sondage. Nous allons modéliser l'intention de vote, `catvote` à l'aide d'une régression logistique multinomiale.

On voit que les personnes plus fortunées, plus éduquées, plus âgées et celles qui s'associent à un parti politique principal (Républicains et Démocrates), votent davantage. L'écart selon l'âge est particulièrement édifiant, avec près de 50% des jeunes qui n'ont pas participé. Il faut garder en tête que le revenu, l'âge et le niveau d'éducation sont fortement associés et que les personnes plus jeunes ont eu moins d'occasions de voter (ce qui pourrait expliquer la plus grande propension pour les catégories de vote).

Une étude plus attentive révèle que la distribution conditionnelle de ceux qui votent toujours est bimodale. La Figure 5.9 montre clairement que les très jeunes et les personnes âgées en font partie. Ainsi, le modèle est potentiellement mal spécifié car le vrai effet de l'âge n'est visiblement pas linéaire au niveau du log de la cote. Cela dit, les primovotant(e)s n'ont souvent eu qu'une seule occasion de voter, ce qui peut expliquer le comportement sur le graphique et l'absence de réponses pour occasionnellement. Pour éviter cet artefact, on considère les personnes de plus de 30 ans uniquement.

Pour le modèle logit multinomial, nous allons prendre `rarement/jamais` comme catégorie de référence pour la variable réponse `catvote`. Notez qu'il est d'usage et préférable, pour réduire le risque de problèmes numériques et accélérer l'optimisation, de centrer et réduire les variables explicatives.

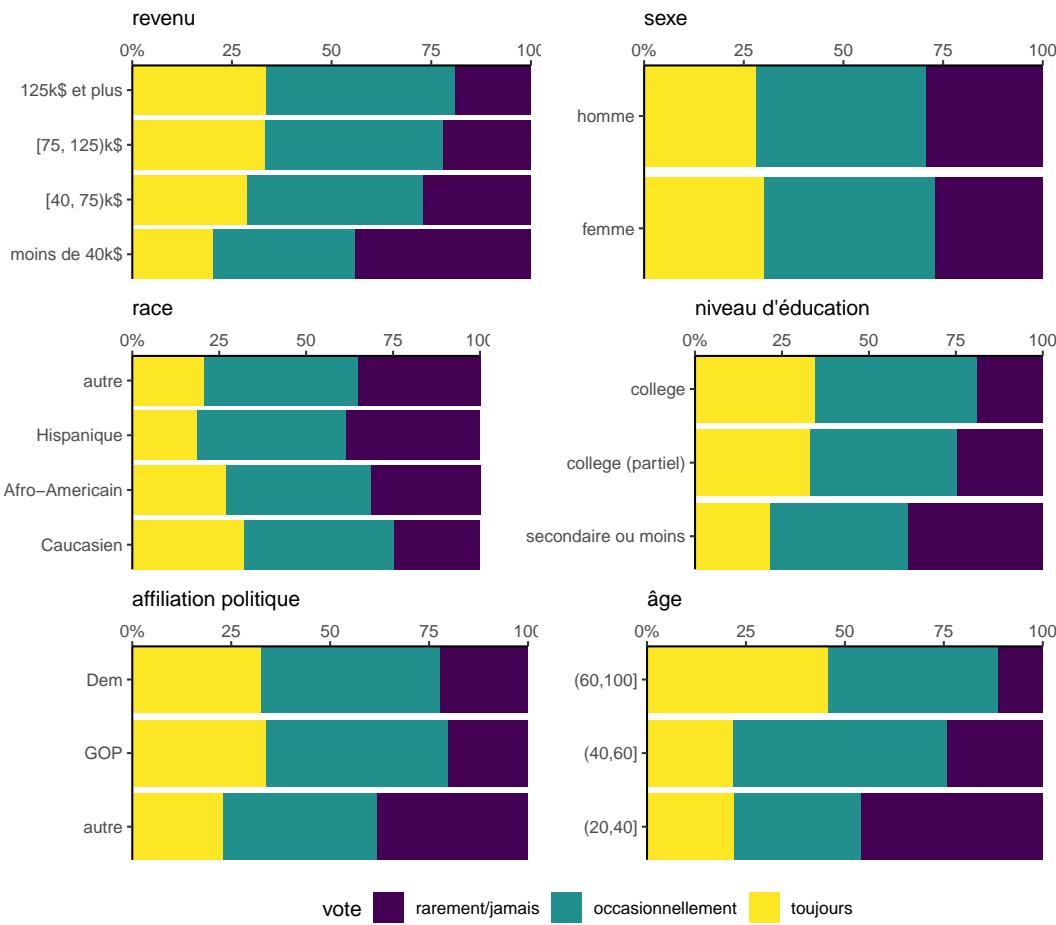


FIGURE 5.8 – Proportion des modalités des variables sociodémographiques des données de participation électorale.

```

data(vote, package = "hecmulti")
# Modèle multinomial
multi1 <- nnet::multinom(
  catvote ~ scale(age, scale = FALSE), # centrer
  data = vote,
  subset = age > 30,
  weights = poids,
  Hess = TRUE,
  trace = FALSE)
# Tableau résumé de l'ajustement

```

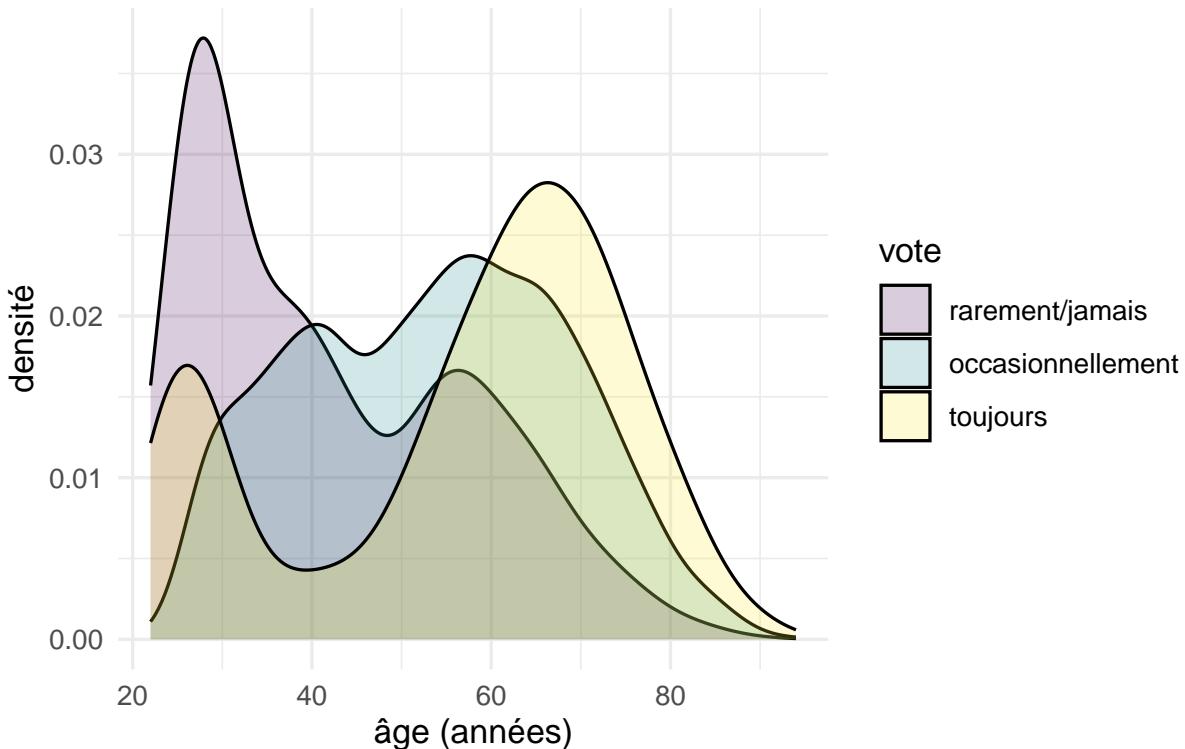


FIGURE 5.9 – Fréquence de vote selon l'âge.

```

summary(multi1)
# Estimations des coefficients
coef(multi1)
# Intervalles de confiance (Wald)
confint(multi1)
# Critères d'information
AIC(multi1)
BIC(multi1)
# Prédictions: probabilité de chaque scénario
predict(multi1, type = "probs")
# Prédictions: classe la plus susceptible
predict(multi1, type = "class")

```

Comme il y a trois catégories pour la variable dépendante, il y a deux équations pour le modèle

TABLEAU 5.12 – Estimation des coefficients et intervalles de confiance à 95 pourcent pour le modèle multinomial logistique avec les données de vote.

(a) catégorie rarement/jamais vs toujours			
	coefficient	IC (2.5%)	IC (97.5%)
cst	0.783	0.709	0.858
age	0.031	0.025	0.036
(b) catégorie occasionnellement vs toujours			
	coefficient	IC (2.5%)	IC (97.5%)
cst	-0.128	-0.224	-0.032
age	0.082	0.075	0.089

ajusté. En regardant les coefficients dans le Tableau 5.12, on obtient :

$$\ln \left\{ \frac{\Pr(\text{catvote}_{1i} = \text{occasionnellement} | \text{age}_i)}{\Pr(\text{catvote}_{1i} = \text{rarement/jamais} | \text{age}_i)} \right\} = 0.783 + 0.031\text{age}_i,$$

$$\ln \left\{ \frac{\Pr(\text{catvote}_{1i} = \text{toujours} | \text{age}_i)}{\Pr(\text{catvote}_{1i} = \text{rarement/jamais} | \text{age}_i)} \right\} = -0.128 + 0.082\text{age}_i.$$

Plus l'âge du répondant augmente, plus la probabilité que la personne vote toujours augmente. Ainsi, la cote moyenne pour toujours versus la référence rarement/jamais est multipliée par  $1.031 = \exp(0.031)$  pour chaque année de plus. Pour faire simple, on a employé une seule variable explicative, mais il est clair au vu de l'analyse exploratoire que d'autres variables sont utiles pour comprendre le comportement des électeurs et électrices. Qui est plus, la taille de la base de données nous permettrait de mesurer d'autres effets.

On peut comparer les modèles emboîtés à l'aide de tests de rapport de vraisemblance.

```
# Ajuster modèle sous H0: les
# prédictions correspondent à la
# proportion empirique de chaque catégorie
multi0 <- nnet::multinom(catvote ~ 1,
                           weights = poids,
                           data = vote,
                           subset = age > 30,
                           trace = FALSE)

# Test de rapport de vraisemblance
anova(multi0, multi1)
```

TABLEAU 5.13 – Analyse de déviance : test du rapport de vraisemblance pour la variable explicative dans le modèle multinomial logistique.

modèle	DL	déviance	DL	rapport vrais.	valeur-p
cst	9692	9781.07			
age	9690	9077.63	2	703.44	< 0.0001

Cette valeur est donnée dans la dernière colonne du tableau. De plus, cet effet est significatif car la valeur- $p$  est inférieure à  $10^{-4}$ .

Pour une comparaison directe entre les deux autres catégories, rarement/jamais et occasionnellement, il suffit de changer la catégorie de référence.

#### 5.4.2 Régression logistique cumulative à cotes proportionnelles

Si les modalités de la réponse sont ordinaires, la régression logistique multinomiale est toujours appropriée. Il peut néanmoins être préférable d'utiliser un modèle qui utilise l'ordre des modalités pour obtenir un modèle plus facile à interpréter et plus parcimonieux. Le modèle de régression logistique cumulative à cotes proportionnelles [@McCullagh:1980] est une simplification du modèle multinomial sous l'hypothèse que les rapports de cotes sont les mêmes pour toute la catégorie.

Supposons que les  $K$  modalités de la variable **ordinale**  $Y$  sont en ordre croissant ( $1 \leq \dots \leq K$ ) et que l'on dispose de  $p$  variables explicatives  $X_1, \dots, X_p$  pour chaque observation.

Soit  $p_{ik} = \Pr(Y_i = k | \mathbf{X}_i)$  ( $k = 1, \dots, K$ ) la probabilité que  $Y_i$  prenne la valeur  $k$ . On dénote

$$S_{ij} = \sum_{k=j}^K p_{ik} = \Pr(Y_i > j-1 | \mathbf{X}_i), \quad j = 2, \dots, K.$$

La quantité  $S_{ij}$  est la probabilité que  $Y_i$  soit plus grand ou égal à  $j$ ;  $S_{i1}$  est égal à 1 et  $S_{iK} = \Pr(Y_i = K | \mathbf{X}_i) = p_{iK}$ .

Le modèle logistique cumulé spécifie que

$$\ln\left(\frac{S_{ij}}{1 - S_{ij}}\right) = \beta_{0j} + \beta_1 X_{i1} + \dots + \beta_p X_{ip}, \quad j = 2, \dots, K.$$

Comme pour le modèle multinomial logistique, il y a  $K - 1$  équations pour le rapport de cote. En revanche, si l'ordonnée à l'origine change d'une équation à l'autre, les paramètres quantifiant les effets des variables explicatives,  $\beta_1, \dots, \beta_p$  sont les mêmes pour chacune des cotes. Par conséquent, pour modéliser une variable ordinale  $Y$  ayant  $K$  valeurs possibles et avec  $p$  variables explicatives, le modèle cumulatif logistique utilise  $p + K - 1$  paramètres. Le modèle logit multinomial, qui peut également être utilisé pour les données ordinaires, utilise plutôt  $(K - 1) \cdot (p + 1)$  paramètres. Le

modèle logistique cumulatif à cotes proportionnelles est donc plus parcimonieux et, pour autant qu'il soit approprié, mènera à des estimations des paramètres plus précises qu'avec le modèle de régression logistique multinomiale. Les deux modèles sont identiques au modèle de régression logistique si la variable ordinaire a uniquement deux modalités (variable binaire).

La cote  $S_{ij}/(1 - S_{ij})$  mesure à quel point il est plus probable que  $Y_i$  prenne une valeur plus grande ou égale à  $j$  par rapport à une valeur plus petite que  $j$ , viz.

$$\frac{S_{ij}}{1 - S_{ij}} = \exp(\beta_{0j} + \beta_1 X_{i1} + \cdots + \beta_p X_{ip}).$$

Dans cet exemple, nous n'avons aucune transformation des variables explicatives, ni aucune interaction dans le modèle; l'interprétation des paramètres est donc simplifiée. Si le paramètre  $\beta_j$  est positif, cela indique que plus  $X_j$  prend une valeur élevée, plus la variable  $Y$  a tendance à prendre aussi une valeur élevée. Inversement, si le paramètre  $\beta_j$  est négatif, cela indique que plus  $X_j$  prend une valeur élevée, plus la variable  $Y$  a tendance à prendre une valeur basse. Plus précisément, pour chaque augmentation d'une unité de  $X_j$ , la cote  $S_k$  versus  $(1 - S_k)$  est multipliée par  $\exp(\beta_j)$ , peu importe la valeur de  $Y$ . En terme de probabilités cumulées d'excéder  $k$ ,

$$\begin{aligned} S_{ik} &= \Pr(Y_i \geq k | \mathbf{X}_i) \\ &= \text{expit}(\beta_{0k} + \beta_1 X_{i1} + \cdots + \beta_p X_{ip}), \quad j = 2, \dots, K. \end{aligned}$$

En utilisant ces expressions, on peut obtenir la probabilité de chaque catégorie,

$$\begin{aligned} \Pr(Y_i = k | \mathbf{X}_i) &= \Pr(Y_i \geq k | \mathbf{X}_i) - \Pr(Y_i \geq k + 1 | \mathbf{X}_i) \\ &= S_k - S_{k+1}. \end{aligned}$$

On peut répéter le même modèle que précédemment, même s'il est peu susceptible que l'hypothèse de cotes proportionnelles soit valide. La variable réponse doit être de classe `ordered`, une forme particulière de facteur dont les niveaux sont ordonnés en ordre croissant. On ajuste un modèle, cette fois avec sexe pour nous permettre de pratiquer l'interprétation d'une variable catégorielle.

```
# Modèle de régression logistique
# multinomiale ordinale à cote proportionnelle
with(vote, is.ordered(catvote))
multi2a <- MASS::polr(
  catvote ~ sexe,
  data = vote,
  subset = age > 30,
  weights = poids,
  method = "logistic",
```

```
Hess = TRUE)

multi2b <- nnet::multinom(
  catvote ~ sexe,
  data = vote,
  subset = age > 30,
  weights = poids,
  Hess = TRUE,
  trace = FALSE)
# Le modèle est paramétré en terme
# du rapport de cote, ascendant
summary(multi2a)
# Test du rapport de vraisemblance pour
# modèle à cote proportionnelle
# deviance = -2*ll
pchisq(deviance(multi2a) - deviance(multi2b),
       df = length(coef(multi2a)),
       lower.tail = FALSE)
# Intervalles de confiance pour beta_x
# - vraisemblance profilée
confint(multi2a)
# Critères d'information
AIC(multi2a)
BIC(multi2a)

# Tableau des coefficients
# Ordonnées à l'origine:
multi2a$zeta
# Uniquement pour variables explicatives
# exp(beta) avec l'IC de vraisemblance profilée
exp(c(coef(multi2a), confint(multi2a)))
# On peut obtenir les intervalles de Wald
# avec confint.default

# Test d'adéquation
# (rapport de vraisemblance, comparaison avec modèle saturé)
pchisq(q = deviance(multi2a),
       df = df.residual(multi2a),
       lower.tail = FALSE)
```

TABLEAU 5.14 – Tableau des estimations des coefficients du modèle pour réponses ordinaires pour la régression logistique à cotes proportionnelles avec sexe.

effet	coefficient	erreur-type
sexe [homme]	-0.166	0.055
cst [rarement/jamais occasionnellement]	-1.297	0.044
cst [occasionnellement toujours]	0.865	0.041

```
# Petite valeur-p = modèle inadéquat
```

Ici, l'effet estimé d'être un homme plutôt qu'une femme (*sexe*) est  $-0.166$  et ce paramètre est significativement différent de zéro (valeur-*p* de 0.003 obtenue en faisant un test de rapport de vraisemblance).

Ainsi, les hommes sont moins susceptibles de voter fréquemment que les femmes. Plus précisément, la cote d'être dans une catégorie plus élevée de *catvote*, par rapport à une catégorie plus basse, est multipliée par  $\exp(r'sprintf(round(as.numeric(coef(multi2a)), 3), fmt = "$ , ce qui correspond à une diminution de la cote des femmes 15% (et donc la probabilité d'être dans une catégorie plus élevée est plus faible).

Avant toute chose, il faut s'assurer que le modèle est approprié. Rappelez-vous que l'une des hypothèses de ce modèle est que les effets des variables explicatives sont les mêmes pour chaque équation.

- $\mathcal{H}_0$  : l'effet de chaque variable est le même pour les  $K$  logit du modèle multinomial logistique, soit  $\beta_{11} = \dots = \beta_{1K}, \dots, \beta_{p1} = \dots = \beta_{pK}$ .

Une très petite valeur-*p* (rejet de  $\mathcal{H}_0$ ) pour ce test serait une indication que le modèle de régression multinomiale ordinaire n'est pas approprié et que le modèle multinomial logistique serait préférable. Comme la valeur-*p* est négligeable, on rejette pas  $\mathcal{H}_0$  et l'hypothèse de cote proportionnelle ne tient pas la route.

On a précédemment utilisé un test du rapport de vraisemblance pour valider l'hypothèse des cotes proportionnelles : il n'y avait aucune indication que la simplification n'était pas adéquate. Ce n'est pas le cas pour le modèle qui ne contient que la variable *age* ou plusieurs variables explicatives : la Figure 5.10 montre les différences de probabilités ajustées pour les deux modèles qui incluent uniquement l'âge comme variable explicative. Règle générale, on n'ajustera jamais un modèle avec une seule des variables : un test du rapport de vraisemblance indique que **toutes** les variables explicatives sont utiles pour expliquer le comportement.

On peut également vérifier si le modèle est adéquat pour décrire les données en comparant le modèle pour les données ordinaires avec un modèle saturé (qui contient autant de paramètres que

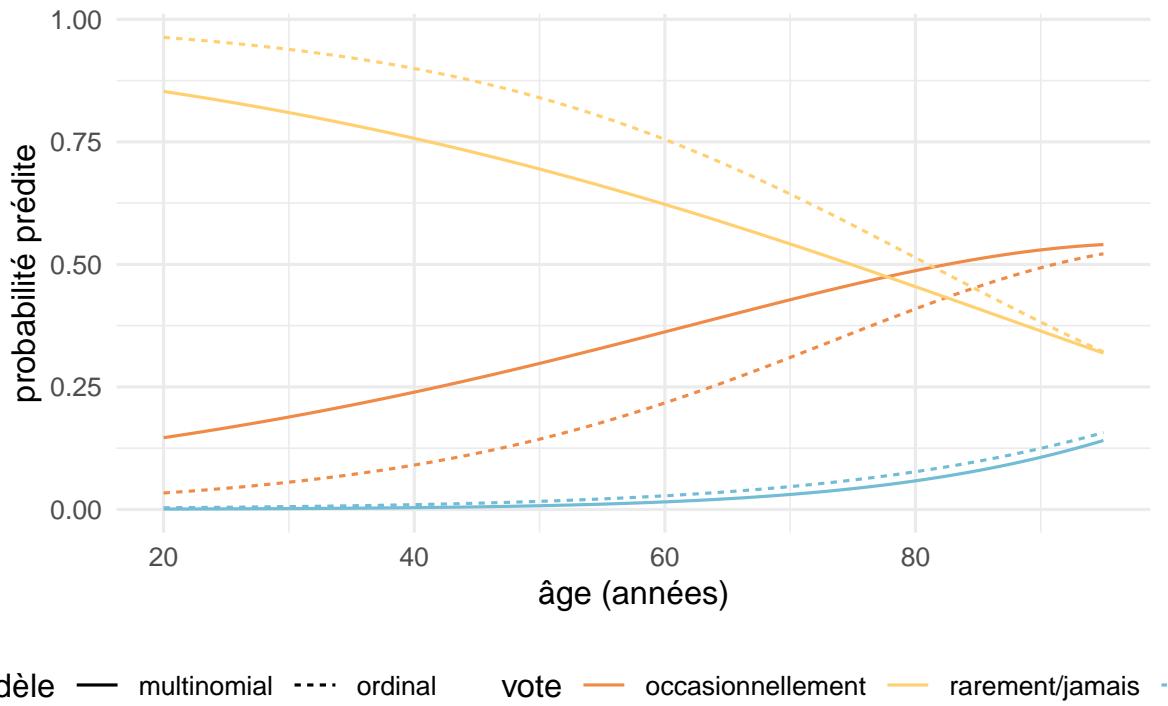


FIGURE 5.10 – Probabilités prédictives pour chaque modalité selon le modèle de régression multinomiale logistique et le modèle de régression ordinale à cotes proportionnelles selon l'âge.

d'observations/niveaux) : la valeur-*p*, infime, indique que le modèle plus complexe, soit le modèle saturé, est préférable. Cela nous indique que le modèle a un piètre pouvoir explicatif.



# **Chapitre 6**

## **Analyse de survie**

### **6.1 Introduction**

Cette section traite de temps avant qu'un événement survienne. Le traitement de ce type de données, dites données de survie en référence au domaine médical, est particulier parce que l'information disponible est incomplète. Plusieurs mécanismes peuvent impacter la survie.

On considère comme exemple une étude sur le chômage dû à la crise du coronavirus. On s'intéresse à tous ceux qui étaient en recherche active d'emploi entre mars et juin ; seuls ceux qui étaient au chômage durant cette période seront considérés. Certaines personnes seront déjà au chômage en avril et donc leur durée de chômage en mars est déjà longue (troncature à gauche). Lors de notre suivi, d'autre mentionneront avoir trouvé un emploi lors de notre appel, mais ne pourront nous renseigner sur la date exacte de leur embauche : cette dernière précèdera notre prise de contact, mais nous est inconnue (censure à gauche). D'autres personnes seront toujours au chômage en juin à la fin de l'étude et on ignorera le nombre réel de mois passés au chômage (censure à droite). Enfin, certaines personnes cesseront de chercher activement un emploi et donc quitteront l'étude. Tous ces méchanismes (complexes) peuvent être dictés par certaines covariables (employabilité, découragement) et être aléatoires ou pas. Pour estimer le taux de chômage, il faudra prendre en compte les méchanismes de survie dans notre modèle. On se concentrera sur le cas simple des données censurées à droite de façon aléatoire.

#### **6.1.1 Exemple du temps d'abonnement**

Une entreprise oeuvrant dans le secteur des télécommunications s'intéresse aux facteurs influençant le temps qu'un client reste abonné à son service de téléphone cellulaire. Des données provenant d'un échantillon de clients se trouvent dans le fichier `survie1`, qui contient les variables suivantes :

- **temps** : temps (en semaines) que le client est resté abonné au service de téléphone cellulaire. Il s'agit du vrai temps si le client n'est plus abonné et d'un temps censuré à droite si le client est toujours abonné.
- **censure** : variable binaire qui indique si la variable  $t$  est censurée (0 si le client est toujours abonné) ou non (1, la variable  $t$  est la durée finale de l'abonnement).
- **age** : âge du client au début de l'abonnement.
- **sexe** : sexe du client, soit femme (1), soit homme (0).
- **service** : nombre de services en plus du cellulaire auquel le client est abonné parmi internet, téléphone fixe, télévision (câble ou antenne parabolique).
- **region** : région où habite le client en ce moment (valeurs entre 1 et 5).

### 6.1.2 Contexte

On s'intéresse au temps avant qu'un événement survienne. On observe chaque sujet jusqu'à ce que l'une des deux choses suivantes se produise : l'événement survient avant la fin de la période d'observation ou bien l'étude se termine et l'événement n'est toujours pas survenu. Dans l'exemple, l'événement correspond au fait d'interrompre son abonnement. On dispose donc d'une variable « temps », que l'on nomme  $T$ , pour chaque individu qui est soit censurée soit non censurée. Si l'individu a expérimenté l'événement avant la fin de la période d'observation, la valeur de  $T$  n'est pas censurée. Si l'événement n'est toujours pas survenu à la fin de la période d'observation, la valeur de  $T$  est censurée. Pour chaque individu, on dispose également d'un ensemble de variables explicatives  $X_1, \dots, X_p$ . Pour l'instant, supposons que les valeurs de ces variables sont fixes dans le temps mais on reviendra plus loin au cas où leurs valeurs peuvent varier dans le temps. Bien que le terme analyse de survie semble implicitement référer à la santé, de nombreux autres exemples sont envisageables

- temps qu'un client demeure abonné à un service offert par notre compagnie.
- temps de survie d'un individu après avoir été diagnostiqué avec un certain type de cancer.
- ancienneté d'un travailleur au service d'une compagnie.
- durée de vie d'une franchise.
- temps avant la faillite d'une entreprise (ou d'un particulier).
- temps avant le prochain achat d'un client.
- temps durant lequel un(e) employé(e) est au chômage.

Si aucune observation n'est censurée, c'est-à-dire, si on a observé le « vrai » temps pour chaque sujet, on pourrait alors simplement modéliser  $T$  en incluant des covariables dans les paramètres de vraisemblance d'une loi positive (par exemple, une régression log-linéaire). En revanche, si des observations sont censurées dans l'échantillon, leur omission biaiserait l'analyse.

Ce chapitre se veut une introduction à l'analyse de données de survie. Comme le développement de la théorie de l'analyse de survie est assez complexe (plus encore que celle de la régression linéaire ou logistique), on s'intéressera ici uniquement aux principes de base afin d'être en mesure d'appliquer

les méthodes et de bien interpréter les résultats. Plusieurs extensions sont également possibles. Un survol de ces dernières sera effectué dans des sections plus loin.

Il existe deux grandes approches pour analyser des données de survies :

- i) nonparamétrique ou semi-paramétrique : estimateur de Kaplan–Meier, modèle de Cox (à risques proportionnels).
- ii) paramétrique : modèle paramétrique avec loi continue (Weibull, log-normal, log-logistique, gamma).

Nous discuterons seulement de la première approche dans ce chapitre. Le tableau suivant fait une analogie entre ce que nous ferons dans ce chapitre et des méthodes que vous connaissez.

réponse $Y$	résumé descriptif	comparaison de deux groupes	modèle général
continue	moyenne	test- $t$ pour deux échantillons	régression linéaire
binaire	proportion	test d'indépendance du khi-deux	régression logistique
temps de survie (censure à droite)	fonction de survie temps de survie médian	test log-rang test de Wilcoxon généralisé (Gehan)	modèle de Cox

La structure de données de base que l'on doit avoir pour travailler est la suivante :

- 1) une variable temps,  $T$ .
- 2) une variable binaire  $C$  (censure).
- 3) d'autres variables explicatives  $X_1, \dots, X_p$ , une par colonne.

## 6.2 Fonctions de survie et de risque

Un des éléments de base d'une analyse de survie (*survival analysis*) est la **fonction (ou courbe) de survie**. Soit  $F(t) = \Pr(T \leq t)$  la fonction de répartition du temps de survie  $t$  et  $f(t) = d/dt F(t)$ . La fonction de survie est

$$S(t) = \Pr(T > t) = 1 - F(t)$$

et donne la probabilité que le temps de survie soit supérieur à  $t$ . On verra plus loin comment estimer cette fonction avec un échantillon et comment tester l'égalité de deux (ou plusieurs) fonctions de survie.

La **fonction de risque** (en anglais, *hazard*) est

$$h(t) = \frac{f(t)}{S(t)}$$

où  $f(t)$  est la fonction de densité (pour  $T$  continu) ou de masse pour  $T$  discret. Dans le cas discret où le temps peut seulement prendre les valeurs  $0, 1, 2, \dots$ , la fonction de risque est donc simplement la probabilité que l'événement survienne au temps  $t$ , étant donné qu'il n'était pas survenu avant,

$$\Pr(T = t | T > t) = \Pr(T = t) / \Pr(T > t) = f(t) / S(t);$$

c'est une probabilité conditionnelle. Dans le cas général, la fonction de risque est nécessairement positive mais peut prendre des valeurs supérieures à un. On ne peut donc pas, à strictement parler, la voir comme une probabilité et c'est pourquoi on parle plutôt de risque. En fait, cette fonction mesure le risque instantané que l'événement survienne au temps  $t$ , étant donné qu'il n'était pas survenu avant.

Cette fonction est importante car il s'agit de celle que nous allons modéliser avec le modèle de régression de Cox. Si, en régression logistique, on modélise le logarithme des cotes, on modélise plutôt la fonction de risque en analyse de survie. Les fonctions de survie et de risque sont intimement reliées et

$$h(t) = -\frac{d \ln\{S(t)\}}{dt}, \quad S(t) = \exp\left\{-\int_0^t h(u)du\right\}.$$

Ainsi, si on connaît la fonction de survie, on peut retrouver la fonction de risque et vice-versa. Par conséquent, un modèle pour la fonction de survie spécifie une fonction de risque (et vice-versa).

### 6.3 Estimation d'une courbe de survie et de risque

L'estimateur nonparamétrique le plus couramment utilisé pour l'estimation de la fonction de survie en présence de censure à droite est l'estimateur de Kaplan-Meier. De plus, cette méthode est nonparamétrique en ce sens qu'on ne suppose aucun modèle et qu'on suppose uniquement que la censure est non-informative.

Si l'échantillon ne contient aucune observation censurée (on a des temps exacts pour tous les sujets), l'estimateur de Kaplan-Meier de la fonction de survie à un temps  $t$  donné est alors simplement la proportion des observations dans l'échantillon qui possède un temps de survie supérieur à  $t$ . Par convention, on considère qu'une observation censurée à droite faisait partie de l'ensemble d'observations à risque au temps de censure observé.

On considère l'exemple des temps d'abonnement pour illustrer le concept. L'estimation de la fonction de survie selon la méthode Kaplan-Meier est obtenue grâce aux commandes suivantes :

TABLEAU 6.2 – Estimation de la fonction de survie (Kaplan–Meier) pour les données de survie d'abonnement.

temps	nb à risque	nb échecs	nb cumul.	survie	erreur-type
2	500	1	1	0.998	0.002
11	499	1	2	0.996	0.003
14	498	1	3	0.994	0.003
18	497	1	4	0.992	0.004
27	496	1	5	0.990	0.004
29	495	1	6	0.988	0.005
30	494	1	7	0.986	0.005
34	493	4	11	0.978	0.007
189	13	1	331	0.204	0.028
202	6	1	332	0.170	0.039
216	2	2	334	0.000	

```

library(survival)
data(survie1, package = "hecmulti")
# Estimateur de Kaplan–Meier
# La réponse "temps" est le temps de survie
# et l'indicateur de censure "censure" est
# "0" pour censuré à droite, "1" pour événement
kapm <-
  survfit(Surv(temps, censure) ~ 1,
          conf.type = "log",
          data = survie1)
summary(kapm)
quantile(kapm)
plot(kapm,
     ylab = "fonction de survie",
     xlab = "temps")

```

La fonction résumé (`summary`) renvoie l'estimation de la fonction de survie pour chaque temps d'échec (événement) : l'estimateur est indéfini à ces valeurs. Le Tableau 6.2 offre une sortie (tronquée) du résumé, modifié pour mieux illustrer les changements. L'estimation de la probabilité que le temps d'abonnement soit supérieur à 30 semaines est  $\hat{S}(30) = 0.986$ .

```

ggsurv <- survminer::ggsurvplot(kapm, palette = 1)
ggsurv$plot + # objet de class 'ggplot'
  theme(legend.position = "none") +
  labs(x = "temps",
       subtitle = "Fonction de survie",
       y = "")

```

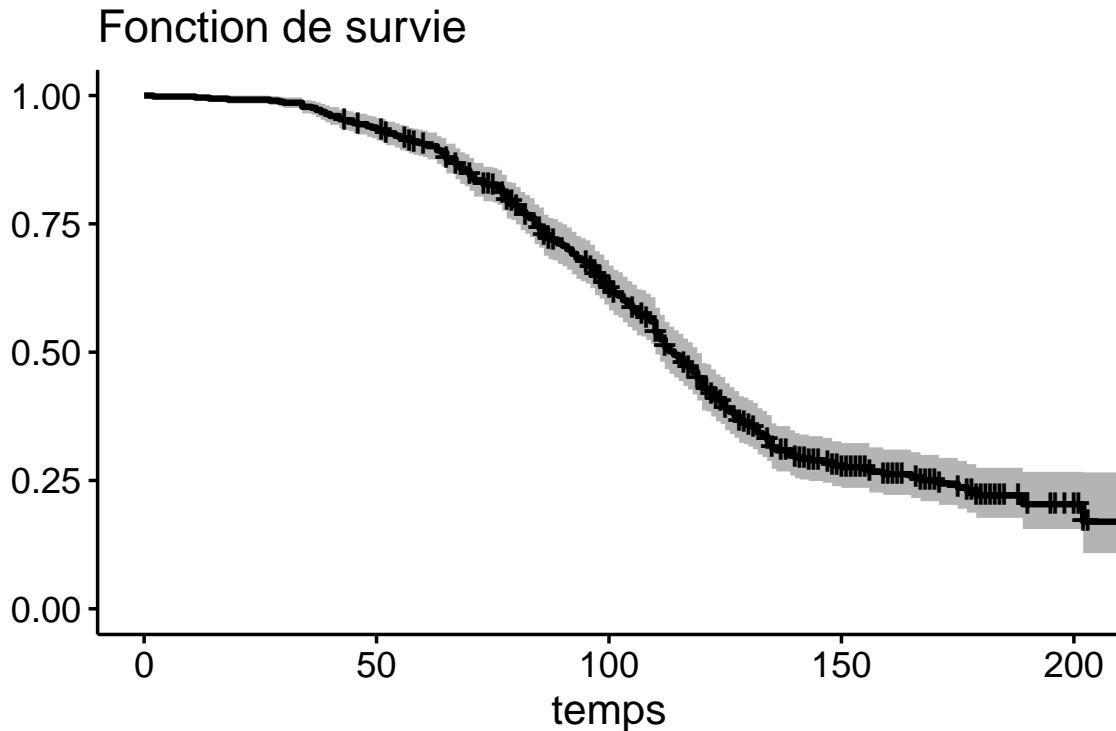


FIGURE 6.1 – Estimation de Kaplan–Meier de la fonction de survie pour les données d’abonnement avec intervalles de confiance ponctuels à 95%.

On peut également utiliser la fonction `quantile` pour obtenir une estimation des quartiles et un intervalle de confiance. On utilise généralement le temps de survie médian (au lieu de la moyenne) dans ce type d’étude. Ici, l’estimé du temps de survie médian est de 114 semaines : on estime que la moitié des clients vont avoir une durée d’abonnement supérieure à 114 semaines. De même, la moitié des clients vont avoir une durée d’abonnement inférieure à 114 semaines. Un intervalle de confiance de niveau 95% pour ce temps médian est [110; 119].

Un estimé de la moyenne et de l’écart-type est donné, mais ce dernier est biaisé (trop bas) puisque les données censurées ne donnent qu’une borne inférieure pour la vraie valeur. Avec un modèle paramétrique pour la survie (par ex., une loi exponentielle), les paramètres estimés du modèle

dicteraient ces deux valeurs. Le modèle de Kaplan–Meier estime la survie, mais si la plus grande observation est censurée, la courbe n'atteindra pas zéro.

Le graphique de la fonction de survie permet de lire le temps de survie pour une probabilité donnée. Les bandes donnent un intervalle de confiance ponctuel de niveau 95% pour chaque temps donné. Une information pertinente de la sortie est le nombre de données censurées : parmi les 500 observations, il y a 334 clients qui ont terminé leur abonnement et 166 qui sont censurées (le client est toujours abonné et le temps est donc une borne inférieure de la durée d'abonnement). Les données censurées contiennent moins d'information que les temps observés de défaillance puisqu'on sait uniquement la borne inférieure de la plage possible des valeurs pour le vrai temps de défaillance.

### 6.3.1 Calcul de l'estimateur de Kaplan–Meier

Cette partie peut être omise ; elle est incluse seulement par souci de complétude. Soit  $T_1, \dots, T_n$  les  $n$  réalisations aléatoires de la variable temps (certaines censurées, d'autres pas). Supposons qu'il y a  $m$  temps distincts où au moins un individu expérimente l'événement. Soient  $t_{(1)} < \dots < t_{(m)}$ , ces temps ordonnés en ordre croissant et  $r_i$  le nombre d'individus à risque au temps  $t_{(i)}$  (les individus qui n'ont pas encore expérimenté l'événement, et qui ne sont pas encore censurés avant  $t_{(i)}$ ). On note  $d_i$  le nombre d'individus qui expérimentent l'événement au temps  $t_{(i)}$ . L'estimateur de Kaplan–Meier de la fonction de survie à un temps  $t$  est

$$\hat{S}(t) = \left(1 - \frac{d_1}{r_1}\right) \times \dots \times \left(1 - \frac{d_{i(t)}}{r_{i(t)}}\right), \quad t_{(1)} \leq t \leq t_{(m)},$$

où  $i(t) = \max(j \in \{1, \dots, m\} : t \geq t_j)$ , soit le plus grand indice parmi  $1, \dots, m$  tel que  $t \geq t_{i(t)}$ . Par convention, si  $t < t_{(1)}$ , on fixe  $\hat{S}(t) = 1$ . La fonction de survie n'est pas définie aux temps de défaillance observée, mais la convention veut qu'elle soit continue à gauche.

Avec la troncation à gauche, la même logique s'applique mais les personnes ne sont à risque qu'à partir du temps minimum observé.

## 6.4 Comparaison de deux courbes de survie

Supposons que les individus ont été divisés en deux groupes et que  $S_1(t)$  et  $S_2(t)$  dénotent respectivement la fonction de survie du premier groupe et du deuxième groupe. On est souvent intéressé à tester l'égalité des fonctions de survie, c'est-à-dire, les hypothèses  $\mathcal{H}_0 : S_1(t) = S_2(t)$  pour tout  $t$  et  $\mathcal{H}_1 : S_1(t) \neq S_2(t)$  pour au moins une valeur de  $t$ .

Par exemple, dans une étude sur le temps de survie après avoir été diagnostiquée avec un certain type de cancer, on pourrait vouloir comparer le temps de survie des individus ayant reçu le traitement standard (groupe 1) au temps de survie des individus ayant reçu un nouveau traitement (groupe 2).

Les deux tests utilisés habituellement sont le test du log-rang (*log-rank test*) et le test de Wilcoxon généralisé (ou test de Gehan).

Testons l'hypothèse que la courbe de survie des clients masculins est la même que celle des clients féminins dans l'exemple des données d'abonnement :

```
## valeur-p du test asymptotique
survdiff(formula = Surv(temp, censure) ~ sexe,
          data = survie1)

Call:
survdiff(formula = Surv(temp, censure) ~ sexe, data = survie1)

      N Observed Expected (0-E)^2/E (0-E)^2/V
sexe=0 309      217      181      7.33     16.4
sexe=1 191      117      153      8.63     16.4

Chisq= 16.4 on 1 degrees of freedom, p= 5e-05

# valeur-p exact par simulation
# (pour petits échantillons, utiliser distribution="exact")
coin::logrank_test(
  Surv(temp, censure) ~ factor(sexe),
  data = survie1,
  distribution = coin::approximate(nresample = 1e5))

Approximative Two-Sample Logrank Test

data: Surv(temp, censure) by factor(sexe) (0, 1)
Z = -4.1415, p-value = 4e-05
alternative hypothesis: true theta is not equal to 1

# Le test du log-rang calcule si les
# deux courbes sont identiques
# Idée de base: sous H0, les étiquettes
# 'sexe' ne sont pas pertinentes
# On peut les permute et comparer les courbes
```

On retrouve dans la sortie les estimés de la fonction de survie, de même que les quartiles par strate : la première strate correspond aux hommes (`sexe=0`) et la deuxième aux femmes (`sexe=1`).

Si on utilise les méthodes (`print`, `summary`, `plot`, `quantile`), on obtiendra les résultats pour chaque strate (ou niveaux de la variable catégorielle). Par exemple, il y a 309 hommes et 191 femmes et l'estimation du temps de survie médian est de 110 semaines pour les hommes et de 123 semaines pour les femmes.

La fonction `survdiff` avec la formule retourne le résultat du test asymptotique du log-rang pour l'hypothèse d'égalité des fonctions de survie. La valeur-*p* du test est inférieure à  $10^{-4}$  : on rejette donc  $\mathcal{H}_0$  pour conclure qu'il y a donc une différence significative entre les deux courbes de survie.

Les courbes de survie selon le sexe sont représentées dans la Figure 6.2. On voit que la courbe des femmes est systématiquement au-dessus de celle des hommes. Les femmes ont donc tendance à rester abonnées plus longtemps que les hommes, et cette différence est significative.

```
plot(survfit(Surv(temps, censure) ~ sexe,
              data = survie1),
      conf.int = FALSE,
      col = c("red", "blue"),
      xlab = "temps",
      ylab = "fonction de survie")
```

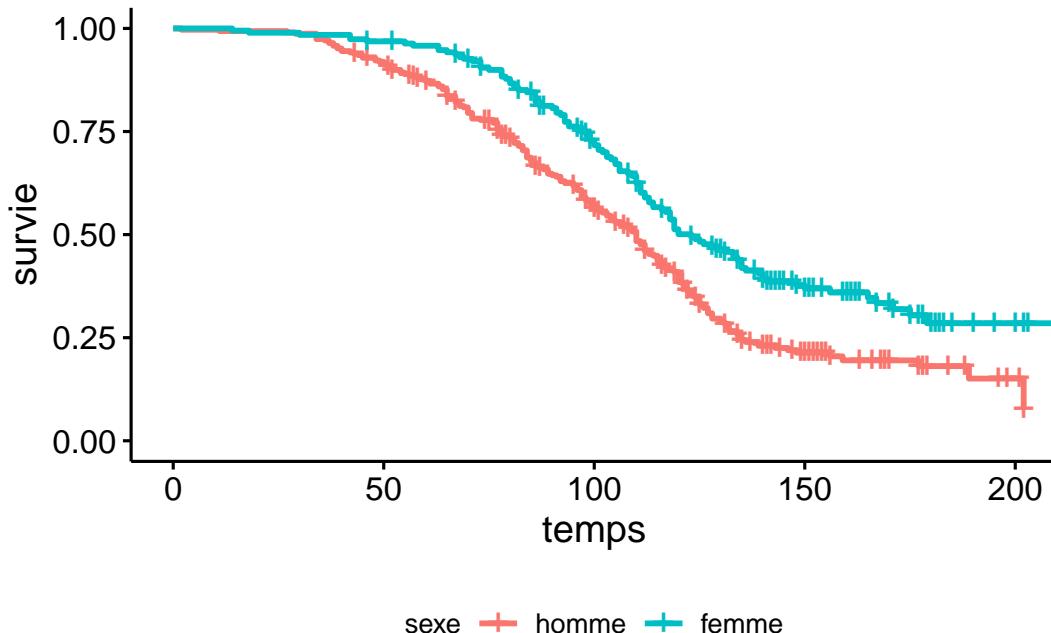


FIGURE 6.2 – Courbes de survie pour les durées d'abonnement selon le sexe de l'individu.

Il est également possible de tester l'égalité des courbes de survie avec plus de deux groupes. Par exemple, s'il y a  $k$  groupes, l'hypothèse nulle est alors  $\mathcal{H}_0 : S_1(t) = S_2(t) = \dots = S_k(t)$  pour tout  $t$ ,

versus l'alternative qu'au moins deux des fonctions ont une valeur différente pour au moins une valeur de  $t$ ; la loi nulle asymptotique du test est alors  $\chi^2_{k-1}$ .

L'estimateur de Kaplan–Meier ne permet pas l'inclusion de variables explicatives à proprement parler : si on peut voir les différences au niveau de la survie selon les modalités d'une variable explicative catégorielle, on divise pour ce faire l'échantillon en sous-groupes et on utilise l'estimateur de Kaplan–Meier pour chacune des modalités en gardant en tête que cela réduit la taille de l'échantillon disponible et que l'estimation résultante est possiblement trop incertaine pour être utile.

### En résumé

- L'analyse de survie est l'étude de temps d'attente (variable positive) avant que survienne un événement.
- L'étude des temps de défaillance nécessite l'utilisation d'outils statistiques spécifiques en raison des mécanismes de censure et de troncation.
- La fonction de survie  $S(t)$  encode la probabilité que le temps de défaillance excède le temps  $t$ .
- La fonction de risque encode la probabilité de mourir au temps  $t$  sachant qu'on a survécu jusque là.
- La connaissance de la fonction de survie permet d'obtenir la fonction de risque et vice-versa.
- Le mécanisme d'information partielle le plus courant en analyse de survie est la censure à droite (on ne connaît qu'une borne inférieure pour le temps de défaillance, l'événement n'étant toujours pas survenu au temps donné).
- Si on traitait les temps de censure comme des temps de défaillance observée, on *sous-estimerait* la durée de survie.
- L'estimateur de Kaplan–Meier est l'estimateur du maximum de vraisemblance nonparamétrique si on a de la censure à droite aléatoire ou non-informative. Il ne fait aucun postulat sur la distribution de la survie.
- Pour que l'estimation soit de qualité, il faut un nombre *conséquent* d'observations (disons 1000). La quantité d'observations censurées impacte la précision de l'estimation.
- L'estimateur est déficient si le plus grand temps observé est censuré à droite (l'estimation de la fonction de survie ne décroît pas à zéro)
- Le test du log-rang permet de valider si deux fonctions de survie sont égales (en tout temps).
- On peut estimer la fonction de survie indépendamment pour chaque modalité d'une variable explicative catégorielle en stratifiant : cela réduit la taille de l'échantillon pour chaque strate.

## 6.5 Modèle à risques proportionnels de Cox

Le modèle à risques proportionnels de Cox (*proportional hazard model*) est l'un des modèles les plus utilisés pour l'analyse des données de survie.

### 6.5.1 Description du modèle de Cox

Soit  $h(t; \mathbf{x})$  la valeur de la fonction de risque au temps pour un individu dont les valeurs des variables explicatives sont  $X_1 = x_1, \dots, X_p = x_p$ . Le modèle à risques proportionnels est

$$h(t; \mathbf{x}) = h_0(t) \exp(\beta_1 x_1 + \dots + \beta_p x_p)$$

où  $h_0(t)$  est la fonction de risque de base; il n'est pas nécessaire de spécifier cette dernière, d'où la nature semiparamétrique du modèle de Cox. Le postulat de risques proportionnels implique que le terme de droite  $\exp(\mathbf{X}\boldsymbol{\beta})$  ne dépend pas du temps, et plus particulièrement  $\beta_1, \dots, \beta_p$  ne dépend pas du temps. Nous verrons subséquemment une extension qui permet de prendre en compte les variables explicatives dont la valeur change dans le temps en scindant ces observations.

Lorsque toutes les variables explicatives prennent la valeur zéro,  $\mathbf{X} = \mathbf{0}$ , on recouvre  $h(t; \mathbf{0}) = h_0(t)$ . Par conséquent, la fonction  $h_0(t)$  peut être interprétée comme la fonction de risque lorsque toutes les variables explicatives valent zéro. Toutefois, tout comme la valeur de l'ordonnée à l'origine dans un modèle de régression linéaire, cette interprétation n'est pas nécessairement valide si la situation où toutes les variables explicatives valent zéro n'est pas possible ou si elle ne survient pas dans notre échantillon.

La deuxième partie du modèle,  $\exp(\beta_1 x_1 + \dots + \beta_p x_p)$ , vient modéliser l'effet d'un changement des valeurs des variables explicatives sur la fonction de risque de base. Tout comme dans le cas de la régression logistique (l'effet des variables sur la cote), c'est un effet multiplicatif, d'où le terme **risques proportionnels**.

Pour l'interprétation des paramètres, il sera plus simple de penser en termes de rapport de risque (*hazard ratio*), qui est défini comme étant le rapport des fonctions de risque pour deux ensembles de valeurs des variables explicatives. Pour simplifier l'illustration, supposons que nous avons seulement une variable explicative  $X$  et que  $h(t; x) = h_0(t) \exp(\beta x)$ . Le rapport de risque lorsque  $X = x_1$  par rapport à  $X = x_0$  est

$$\frac{h(t; x_1)}{h(t; x_0)} = \exp\{\beta(x_1 - x_0)\}.$$

Par conséquent, l'impact d'une augmentation de  $X$  d'une unité (quand  $x_1 - x_0 = 1$ ) est  $\exp(\beta)$ . Ainsi, pour chaque augmentation d'une unité pour  $X$ , le risque que l'événement survienne est multiplié par  $\exp(\beta)$ .

TABLEAU 6.3 – Rapport de risques et intervalles de confiance de Wald à niveau 95%.

terme	exp(coef)	borne inf.	borne sup.
sexé	0.63	0.5	0.79

Le terme **risques proportionnels** fait référence à la situation où le rapport de risque dépend seulement de la différence  $x_1 - x_0$  et non pas du temps lui-même. Le rapport de risque est constant par rapport au temps  $t$ . Cela implique que l'effet d'une variable est stable dans le temps. Nous verrons plus loin comment faire en sorte que l'effet d'une variable puisse varier dans le temps.

Débutons avec un exemple simple en utilisant les données d'abonnement : on ajuste un modèle de Cox en utilisant seulement la variable binaire sexe.

```
cox1 <- coxph(Surv(temps, censure) ~ sexe,
                 data = survie1,
                 ties = "exact")
# Coefficients, tests et IC 95% de Wald
summary(cox1)
# Test du rapport de vraisemblance
car::Anova(cox1, type = 3)
```

La sortie inclut notamment des tests de significativité globale basés sur la vraisemblance comparant le modèle sans variable explicative avec celui ajusté (rapport de vraisemblance, score et Wald). Ces trois statistiques ciblent la même hypothèse, aussi on peut ne s'attarder qu'au test du rapport de vraisemblance, qui est plus fiable et généralement plus puissant. Le tableau des coefficients donne les estimations  $\hat{\beta}$ . Le rapport de risque pour sexe=1 versus sexe=0 est  $\exp(\hat{\beta})$ , tandis qu'on obtiendrait le rapport de risque pour sexe=0 versus sexe=1, soit  $\exp(-\hat{\beta})$ . Les statistiques de test et la valeur- $p$  associée sont basées sur la statistique de Wald, soit  $\hat{\beta}/\text{se}(\hat{\beta})$ , tandis que les intervalles de confiance à 95% pour le rapport de risque sont des intervalles de Wald de la forme  $\exp\{\hat{\beta} \pm 1.96 \cdot \text{se}(\hat{\beta})\}$  pour le rapport de risque. Un rapport de risque de 1 signifie que le risque n'est pas affecté par la variable explicative. Ici, le test correspondant à l'hypothèse nulle  $\beta = 0$  ne mène pas au rejet de l'hypothèse qui veut que la variable n'impacte pas le risque. On pourrait obtenir les intervalle

Il y a une seule variable explicative, le sexe de l'individu. L'estimation du paramètre de l'effet de sexe est -0.47. Ce paramètre est significativement différent de 0 (valeur- $p$  du test du rapport de vraisemblance inférieure à  $10^{-10}$ ). Pour l'interprétation, on utilise la colonne exp(coef) qui contient la valeur  $\exp(\hat{\beta}_{\text{sexé}}) = \exp(-0.466) = 0.628$ . Ainsi, le rapport du risque d'une femme par rapport à un homme est

$$\frac{\hat{h}(t; \text{sexé} = 1)}{\hat{h}(t; \text{sexé} = 0)} = 0.628.$$

Par conséquent, le risque qu'une femme interrompe son abonnement est 0.628 fois celui d'un homme. Une femme est donc moins à risque de quitter qu'un homme. Nous avions déjà vu cela à la section précédente lorsque nous avions comparé les courbes de survie des hommes et des femmes. Il est important de se rappeler qu'avec ce modèle, l'effet d'une variable est le même dans le temps (peu importe la valeur de  $t$ ). Donc, une femme est moins à risque de quitter qu'un homme à tout moment, d'après ce modèle. Inversement, le ratio du risque d'un homme par rapport à une femme est  $\exp(-\hat{\beta}_{\text{sexe}}) = 1/0.628 = 1.59$ . Ainsi, à tout moment, un homme a un risque d'interrompre son abonnement qui est 59% plus élevé que celui d'une femme.

Comme il y a un seul paramètre ici, les tests basés sur la vraisemblance pour  $\mathcal{H}_0 : \boldsymbol{\beta} = \mathbf{0}$  reviennent à tester l'effet de la variable sexe. Le test de Wald est le même que celui du tableau des coefficients. Dans le cas particulier où il y a une seule variable explicative catégorielle comme ici, le test du score est équivalent au test du log-rang que nous avons vu à la section précédente à une petite différence près lorsqu'il y a des doublons (ex aequo) dans les temps de survie.

On pourrait également utiliser une variable explicative continue plutôt qu'une variable binaire; le principe est le même.

```
cox2 <- coxph(Surv(temp, censure) ~ age,
                 data = survie1,
                 ties = "exact")
summary(cox2)
```

Le rapport de risque pour âge est 0.959 et donc le risque diminue de 4.1% chaque fois que l'âge augmente d'un an — le risque d'interrompre l'abonnement diminue lorsque l'âge augmente et cet effet est significatif (valeur- $p$  du test de signification globale inférieures à  $10^{-4}$ ).

Généralement, on considérera le modèle de Cox avec toutes les variables explicatives simultanément. La variable `region` est nominale tandis que la variable `service` est ordinaire (avec quatre modalités). Nous allons les incorporer, comme d'habitude, en utilisant des variables indicatrices avec `region=1` et `service=0` (le client n'est abonné à aucun autre service) comme catégories de référence.

```
with(survie1, table(service))
cox3 <- coxph(Surv(temp, censure) ~
                age + sexe + region + service,
                data = survie1,
                ties = "exact")
summary(cox3)
# Effets de type 3
# (modèle avec toutes les variables sauf une)
car::Anova(cox3,
```

TABLEAU 6.4 – Rapport de risque et intervalles de confiance à niveau 95% de Wald pour le modèle de Cox de base avec toutes les variables explicatives.

terme	exp(coef)	borne inf.	borne sup.
age	0.95	0.94	0.96
sexe	0.51	0.40	0.65
region2	0.67	0.47	0.98
region3	1.03	0.73	1.46
region4	0.80	0.57	1.13
region5	0.97	0.68	1.37
service1	0.35	0.27	0.45
service2	0.17	0.12	0.25
service3	0.12	0.07	0.19

TABLEAU 6.5 – Tests du rapport de vraisemblance pour les effets de type III pour le modèle de Cox avec toutes les variables explicatives.

terme	statistique	ddl	valeur-p
age	68.07	1	<1e-04
sexe	32.82	1	<1e-04
region	7.67	4	0.1
service	159.31	3	<1e-04

`type = 3)`

Les effets des variables sont maintenant des effets marginaux. Ainsi, lorsque les autres variables demeurent fixes, le risque de quitter d'une femme est 0.511 fois plus petit que celui d'un homme. L'effet marginal (une fois que les autres variables sont incluses) de la variable sexe est significatif (valeur-*p* inférieure à  $10^{-4}$ ).

Toutes autres choses étant égales, chaque augmentation de l'âge d'un an fait diminuer le risque d'interrompre l'abonnement. Plus précisément, le risque est multiplié par 0.95 lorsque l'âge augmente d'un an et cet effet est significatif.

Pour la variable service, l'interprétation se fait par rapport à la catégorie de référence, qui est la catégorie 0 (abonné à aucun autre service). Ainsi, si le client est abonné à un autre service, son risque de quitter est 0.353 fois celui d'un client qui n'est pas abonné à un autre service (toutes autres choses étant égales). Si le client est abonné à deux autres services, son risque de quitter est encore plus petit comparativement à un client qui n'est pas abonné à un autre service (rapport de risque

de 0.174) Finalement, si le client est abonné à trois autres services, son risque de quitter est encore plus petit (rapport de risque de 0.115). Les paramètres de ces trois variables sont tous significatifs. Ainsi, les clients qui sont abonnés à un, deux ou trois services ont un risque de quitter qui est significativement plus faible que celui d'un client qui n'est pas abonné à un autre service. Le tableau d'analyse de déviance (effets de type 3) donne les statistiques du rapport de vraisemblance pour tester globalement la significativité d'une variable explicative modélisée avec plusieurs indicatrices. Pour la variable `service`, le test présenté teste l'hypothèse nulle  $\mathcal{H}_0 : \beta_{\text{service}_1} = \beta_{\text{service}_2} = \beta_{\text{service}_3} = 0$  contre l'alternative qu'un moins un de ces trois paramètres est non-nul. Le test est largement significatif (statistique du rapport de vraisemblance valant 159.31 avec une valeur- $p$  inférieure à  $10^{-4}$ ). L'effet de la variable `service` est donc globalement significatif. Afin de comparer les autres modalités entre elles, par exemple afin de voir si le risque de quitter est différent entre un client qui a deux services et un autre qui a trois services, il suffit de changer la catégorie de référence à la commande `class` et de réajuster le modèle.

Finalement, la variable `region` n'est pas globalement significative (statistique du rapport de vraisemblance de 7.67 avec une valeur- $p$  de 0.1 basée sur la loi asymptotique  $\chi^2_4$ ).

### 6.5.2 Estimation de la fonction de survie pour des valeurs particulières des variables explicatives

Il est possible d'obtenir l'estimation de la fonction de survie pour des valeurs particulières des variables explicatives. Pour ce faire, il faut avoir un autre fichier de données qui contient les valeurs des variables explicatives pour lesquelles on veut une estimation de la fonction de survie. Si on ajuste le modèle avec aucune variable explicative, on retrouvera alors l'estimation de Kaplan–Meier de la fonction de survie.

Supposons qu'on ajuste le modèle avec les variables sexe et âge seulement dans l'exemple du temps d'abonnement, et que l'on désire la fonction de survie pour les hommes de 25 et 60 ans et pour les femmes de 25 et 60 ans. Le fichier `survie2` contient les données qui seront utilisées à cette fin.

```
cox4 <- survival::coxph(
  Surv(time = temps,
        event = censure,
        type = "right") ~
    sexe + age,
  data = survie1,
  ties = "exact"
)
data(survie2, package = "hecmulti")
# Calculer la fonction de survie
fsurv <- survfit(cox4,
```

```

    newdata = survie2,
    type = "kaplan-meier")
# Tracer la fonction de survie selon le profil
plot(fsurv,
      col = 1:4,
      ylab = "survie",
      xlab = "temps d'abonnement (semaines)",
      bty = "n")
legend(x = "bottomleft",
       legend = c("homme, 25 ans",
                 "femme, 25 ans",
                 "homme, 60 ans",
                 "femme, 60 ans"),
       col = 1:4,
       lty = 1,
       bty = "n")

```

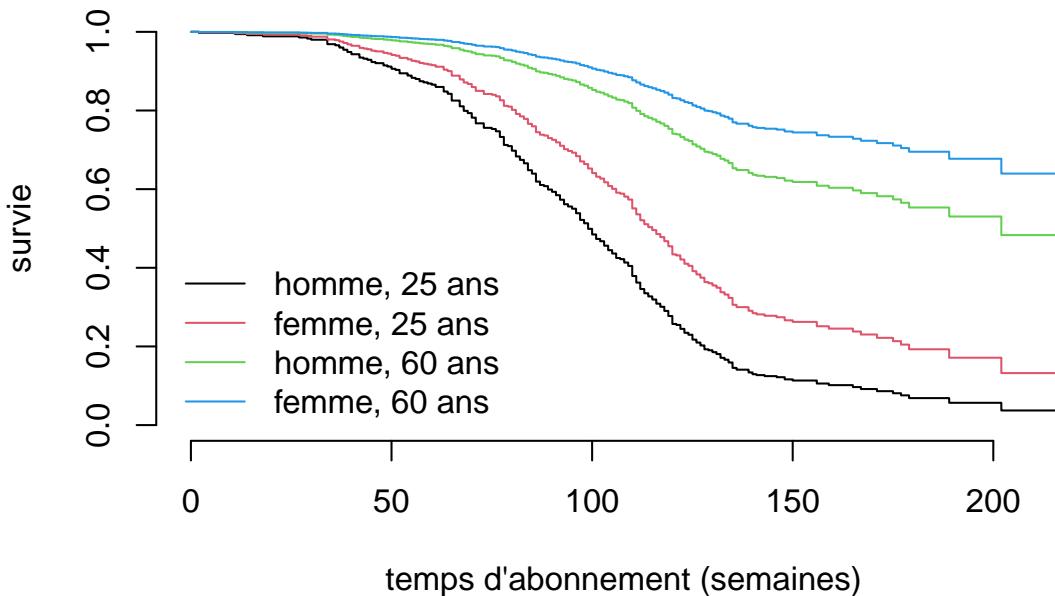


FIGURE 6.3 – Courbes de survies du modèle de Cox pour les quatre profils client.

Les quatre fonctions de la Figure 6.3 correspondent aux profils pour lesquels nous désirons une estimation de la courbe de survie. La courbe 1 est pour les hommes de 25 ans, la courbe 2 pour les femmes de 25 ans, la courbe 3 pour les hommes de 60 ans et la courbe 4 pour les femmes de 60 ans. On voit donc que, parmi ces quatre profils, les hommes de 25 ans sont le plus à risque de quitter

tandis que les femmes de 60 ans sont le moins à risque de quitter.

### 6.5.3 Postulat de risques proportionnels

Le modèle de Cox fait l'hypothèse que la fonction de risque de base est identique pour toute valeur des variables catégorielles. La Figure 6.4 illustre à quoi ce postulat correspond dans un cas simple où il y a uniquement une variable binaire explicative.

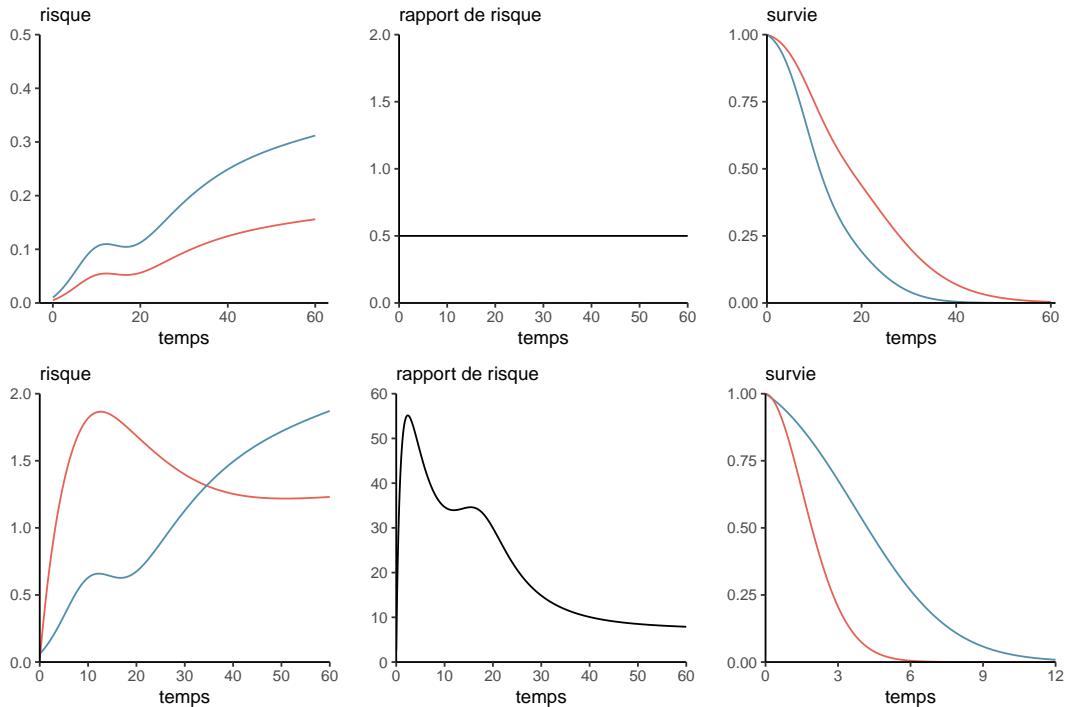


FIGURE 6.4 – Courbes de risques proportionnelles (panneau supérieur) et non proportionnelles (panneau inférieur) avec rapport de risque et fonctions de survie correspondantes.

Si ce n'est pas le cas, alors les résultats du modèle ne sont pas nécessairement fiables.

Comme pour un modèle de régression, il est possible de créer des résidus du modèles et de faire des graphiques diagnostics pour potentiellement infirmer le postulat de risques proportionnels [Grambsch.Therneau :1994]. Si l'hypothèse tient la route, alors il ne devrait pas y avoir de tendance temporelle dans les résidus.

La commande `cox.zph` permet de tester le postulat de risques proportionnels à l'aide d'un test du score pour voir si la pente  $\beta(t)$  associée à une covariable est nulle en fonction du temps  $t$ ; si la valeur- $p$  est grande, cela indique une absence de preuve. La fonction `plot` permet également d'obtenir un graphique des résidus en fonction du temps.

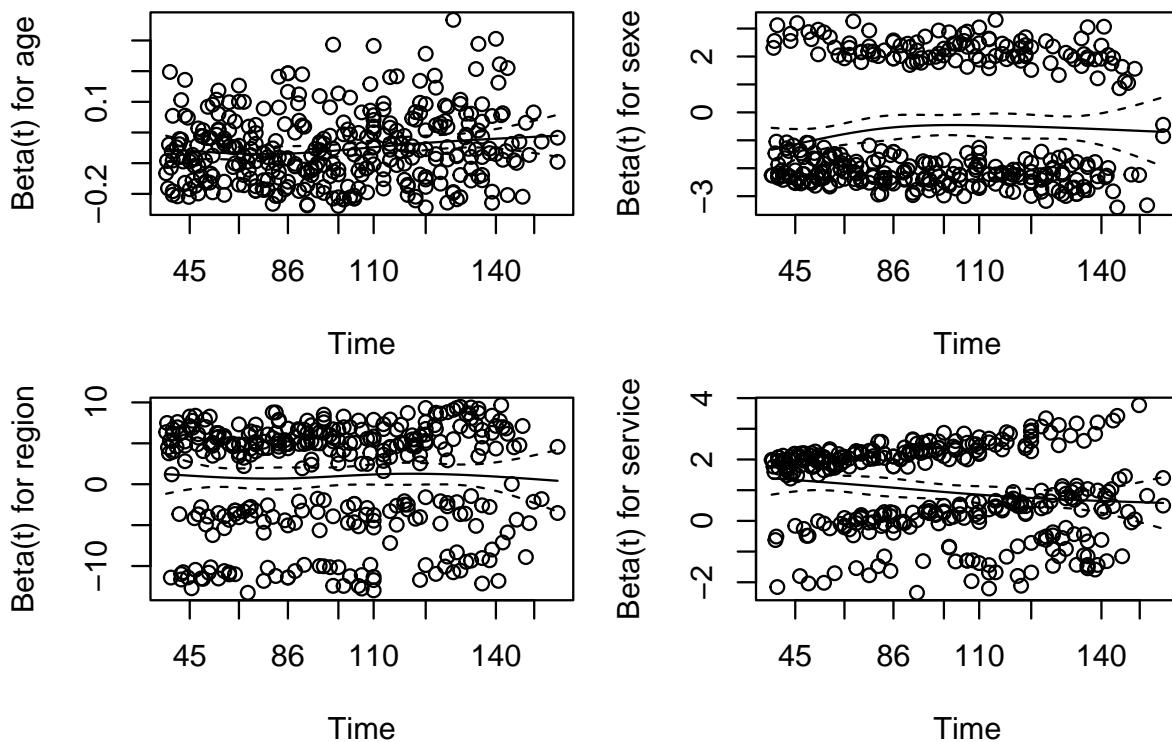


FIGURE 6.5 – Estimations des coefficients en fonction du temps basés sur les moindres carrés pondérés (diagnostic graphique de Grambsch et Therneau).

Dans la Figure 6.5, on voit que le coefficient pour service augmente au fil du temps pour tous les groupes. On pourrait capturer cette interaction ou stratifier pour calculer le risque selon le nombre de services, au risque d'avoir trop peu d'observations pour estimer de manière fiable le risque de base. C'est explicable par le fait que les personnes avec plus de services tendent à avoir une survie plus longue.

```
diag_risqueprop <- cox.zph(cox3)
print(diag_risqueprop)
plot(diag_risqueprop)
```

Si le postulat n'est pas validé, on peut interpréter l'effet comme un rapport de risque moyen pondéré sur la période de suivi, mais ce dernier change selon le moment [@Stensrud.Hernan :2020]. Cela implique également que les erreurs-types associées aux estimations sont trompeuses. La section suivante permettra de généraliser le modèle de Cox et traiter ce cas de figure.

TABLEAU 6.6 – Postulat de risques proportionnels : test du score de Grampsch et Therneau (1994) pour les coefficients constants dans le temps et valeur-p asymptotique basée sur la loi nulle khi-deux.

effet	score	ddl	valeur-p
age	4.22	1	0.040
sexe	1.11	1	0.291
region	3.81	4	0.432
service	10.97	3	0.012
global	21.23	9	0.012

## 6.6 Extensions du modèle de Cox

Dans cette section, nous allons voir deux extensions du modèle à risques proportionnels de base.

- i) Inclusion de variables explicatives dont la valeur change dans le temps.
- ii) Modèle à risques compétitifs (*competing risks*) pour étudier la situation où il y a plusieurs manières de quitter l'état.

La vignette *Using Time Dependent Covariates and Time Dependent Coefficients in the Cox Model* de Therneau, Crowson et Atkinson du paquet survival offre des détails sur l'implémentation des méthodes qui suivent.

### 6.6.1 Variables explicatives dont la valeur change dans le temps

Il est clair que certaines caractéristiques d'un individu évoluent dans le temps (*time-varying covariates*). Si le sexe d'un individu est stable dans le temps, son revenu, son statut matrimonial, l'endroit où il habite, sont par contre des caractéristiques qui peuvent changer dans le temps. Il peut alors être intéressant d'en tenir compte dans l'analyse. Rappelez-vous que le modèle à risques proportionnels est

$$h(t; \mathbf{x}) = h_0(t) \exp(\beta_1 x_1 + \cdots + \beta_p x_p).$$

Supposons que la variable  $X_1$  change au fil du temps et que les autres demeurent fixes. On peut alors réécrire le modèle

$$h(t; \mathbf{x}) = h_0(t) \exp\{\beta_1 x_1(t) + \cdots + \beta_p x_p\},$$

où  $x_1(t)$  indique que la valeur de  $X_1$  dépend du temps  $t$ .

TABLEAU 6.7 – Aperçu des cinq premières observations de la base de données survie3.

id	debut	fin	evenement	age	sexe	region	service
1	0	130	0	48	1	3	2
1	130	178	0	48	1	3	1
2	0	159	0	31	1	3	2
3	0	110	1	36	1	4	0
4	0	109	1	30	0	2	0
5	0	78	0	22	0	5	1
5	78	108	1	22	0	5	0

Supposons que la variable `service`, qui représente le nombre d'autres services souscrits, est la seule que nous voulons modéliser comme une variable qui varie dans le temps. Pour l'âge, nous prenons simplement l'âge au début de l'abonnement, idem pour la région.

Le plus difficile est de créer correctement le fichier de données pour effectuer ce genre d'analyse. Pour chaque personne, on peut identifier plusieurs moments où l'une ou l'autre des valeurs des variables explicatives change. Supposons qu'on a observé un événement au temps  $t_1$ , et que la valeur de la covariable change à  $t_0$ , où  $0 < t_0 < t_1$ . On peut envisager cette observation donne deux contributions : une pour la trajectoire sur l'intervalle  $(0, t_0]$  (censure à droite) et l'autre, après la modification, sur l'intervalle  $(t_0, t_1]$  (troncature à gauche). Puisque la valeur est observée passée le premier intervalle, on a besoin de l'information sur toute la fenêtre (les deux bornes) et le type d'événement pour la première fenêtre sera de la censure à droite.

La base de données `survie3` contient le format adéquat : une colonne `evenement` qui indique la censure (tout intervalle intermédiaire pour un individu est traité comme de la censure à droite) et les bornes de la fenêtre, `debut` et `fin`. Dans cet exemple, il y a eu au plus un changement dans la variable `service`, comme présenté dans le fichier `survie3`. Les variables explicatives `age`, `sexe` et `region` sont comme précédemment.

On regarde plus en détail le profil des cinq premiers clients présenté dans le Tableau 6.7, dont seuls deux ont changé le nombre d'abonnements ; les individus 3–5 se sont désabonnés du service cellulaire à un moment donné. Le premier client était abonné à deux autres services au début de son abonnement au téléphone cellulaire mais, après 130 semaines d'abonnement, a effectué un changement à son forfait pour ne conserver qu'un autre service en plus du cellulaire. Pour le deuxième client, comme `temps_ch` est manquante, il est toujours abonné à deux autres services et ce, jusqu'à la fin de l'étude.

```
data(survie3, package = "hecmulti")
cox4 <- coxph(Surv(time = debut,
```

```

    time2 = fin,
    event = evenement) ~
    age + sexe + region + service,
    data = survie3)

```

L'interprétation se fait comme précédemment; on a omis l'option `ties = "exact"` parce que cette option est trop gourmande en calcul. Puisque c'est la valeur d'une variable qui varie dans le temps et non pas son effet, on a l'interprétation usuelle. Par exemple, le risque de quitter pour un client qui a un autre service est 0.601 fois celui d'un client qui n'a aucun autre service (référence). Le fait d'avoir deux ou trois services diminue encore plus le risque de quitter (rapports de risque de 0.265 et 0.209, respectivement).

### 6.6.2 Modèle à risques compétitifs

Parfois, la raison pour laquelle un individu quitte l'état étudié peut avoir un intérêt en soi. Par exemple si on s'intéresse au temps qu'un employé demeure au service de la compagnie, la distinction entre le fait qu'il ait démissionné ou bien qu'il ait été renvoyé peut avoir un impact sur l'effet des variables explicatives. Comme autre exemple, si on s'intéresse au temps de survie d'un individu après qu'il ait été diagnostiqué avec un certain type de cancer, il pourrait être important de distinguer selon la cause exacte de la mort.

De manière générale, supposons qu'il y a  $K$  manières possibles que l'événement survienne. On peut alors spécifier  $K$  fonctions de risques (une pour chaque manière) et obtenir le modèle de Cox à risques compétitifs (*competing risks*),

$$\begin{aligned}
 h_1(t; \mathbf{x}) &= h_{01}(t) \exp(\beta_{11}x_1 + \cdots + \beta_{p1}x_p) \\
 &\vdots \\
 h_K(t; \mathbf{x}) &= h_{0K}(t) \exp(\beta_{1K}x_1 + \cdots + \beta_{pK}x_p)
 \end{aligned}$$

Notez que les coefficients sont différents d'une équation à l'autre. En estimant ce modèle, on obtient donc une estimation de l'effet des variables selon la raison du départ de l'état. De plus, on peut aussi inclure des variables dont la valeur change dans le temps, comme vu précédemment. Ce qui simplifie énormément la situation est qu'il est prouvé qu'on peut estimer les paramètres de chaque équation séparément sans perte de précision. Par conséquent, en pratique, il suffira d'ajuster  $K$  modèles de Cox séparément.

Dans notre exemple d'abonnement cellulaire, supposons que nous avons trois causes possibles pour la perte d'un client : soit il a interrompu son abonnement pour aller chez le compétiteur A, soit pour aller chez le compétiteur B, soit il n'a plus de cellulaire du tout.

TABLEAU 6.8 – Rapport de risque et intervalles de confiance à niveau 95% pour le modèle à risques compétitifs (probabilité de quitter pour compétiteur A).

terme	exp(coef)	borne inf.	borne sup.
age	0.95	0.94	0.97
sexe	0.44	0.32	0.62
region2	0.54	0.32	0.92
region3	0.82	0.50	1.35
region4	0.72	0.45	1.16
region5	1.05	0.66	1.66
service1	0.38	0.27	0.54
service2	0.18	0.11	0.30
service3	0.11	0.05	0.22

Les données pour cet exemple se trouvent dans le fichier `survie4`. La seule nouveauté par rapport au fichier original est la variable `censure` qui est maintenant codée ainsi

- 1, si le temps est censuré (l’individu est toujours abonné à notre service)
- 2, si l’individu a quitté pour aller chez le compétiteur A
- 3, si l’individu a quitté pour aller chez le compétiteur B
- 4, si l’individu a quitté parce qu’il n’a plus besoin de cellulaire.

On peut calculer la fréquence de chaque modalité avec `table`. Ainsi, il y a donc 166 clients toujours abonnés, 170 qui nous ont quitté pour aller chez A, 121 pour aller chez B, et 43 qui n’ont plus de cellulaires.

Pour ajuster le modèle lorsque la cause du départ est le compétiteur A, le code est

```
# Rappel pour `event`:
#   1 pour observation,
#   0 pour censure à droite
# On utilise la convention TRUE = 1, FALSE = 0
data(survie4, package = "hecmulti")
cox5 <- coxph(Surv(time = temps,
                     event = censure == 2,
                     type = "right") ~
                age + sexe + region + service,
                data = survie4,
                ties = "exact")
summary(cox5)
```

Notez qu'on précise que les valeurs 1, 3 et 4 sont des observations censurées. Ici, l'événement d'intérêt est que le client est parti chez le compétiteur A. S'il est toujours abonné (`censure=1`), s'il est parti chez le compétiteur B (`censure=3`) ou s'il nous a quitté car il n'a plus de cellulaire (`censure=4`), alors l'événement « quitter pour aller chez A » n'est pas survenu. C'est pourquoi on doit traiter ces situations comme des censures.

Ainsi, on voit que l'événement est survenu 170 fois et qu'il y a 330 censures. L'interprétation des paramètres se fait comme précédemment. Sauf qu'il faut préciser qu'il s'agit du risque de quitter pour aller chez le compétiteur A. Par exemple, le risque de quitter pour aller chez le compétiteur A d'une femme est 0.444 fois le risque de quitter pour aller chez le compétiteur A d'un homme. Ainsi, les femmes sont moins à risque de quitter pour aller chez le compétiteur A que les hommes.

Pour ajuster le modèle lorsque la cause du départ est le compétiteur B, procède de la même manière. Notez que cette fois-ci, ce sont les valeurs 1, 2 et 4 de la variable censure qui correspondent au fait que l'événement n'est pas survenu. Il y a 121 clients qui ont quitté pour aller chez B et 379 cas autre (censure). L'interprétation des paramètres se fait en termes de risque de quitter pour aller chez le compétiteur B.

Si on ajuste le modèle pour le cas de figure où la cause de départ est que le client n'a plus besoin de cellulaire, on obtient une sortie similaire. Comme il y a seulement 43 événements (quitter car on n'a plus besoin de cellulaire), les estimations des paramètres sont moins précises, ce qu'on peut voir avec les erreurs-types qui sont plus élevés. Les observations censurées contiennent moins d'information que les événements, d'où cette perte de précision.

Dans **R**, on peut aussi ajuster simultanément tous les modèles en spécifiant que l'événement d'intérêt est un facteur : c'est alors la catégorie de référence qui fait fois de l'état de départ. Il faut également une variable qui identifie l'observation — dans le cas qu'on considère, c'est simplement une colonne avec des valeurs de 1 à  $n$ .

```
rc_cox <- coxph(
  Surv(time = temps,
        event = factor(censure)) ~ sexe + age + service,
  data = survie4 |>
    dplyr::mutate(id = 1:nrow(survie4)),
  id = id)
```

On obtient l'ensemble des coefficients dans le tableau résumé, un pour chaque événement compétitif autre que la référence.

Si on voulait ajuster le modèle de Kaplan–Meier, pour une transition d'un état de base (abonné) vers un état dit absorbant (désabonnement, soit chez compétiteur A ou B ou abandon du cellulaire), il faut obligatoirement ajuster toutes les courbes simultanément. On ajustera le modèle multi-état en spécifiant un facteur pour l'événement, où encore une fois la catégorie de référence est abonnement (`censure=1`).

```
rc_km <- survfit(Surv(time = temps,
                        event = factor(censure)) ~ 1,
                        data = survie4)
```

Les représentations graphiques pour le modèle à risque compétitif sont légèrement différente. Si on dichotomise l'événement d'intérêt (survie ou échec), il y a deux options possibles et la probabilité d'échec est complémentaire à la survie. Avec plus d'un choix, on obtiendra un graphique avec une estimation de la probabilité pour chaque modalité de l'événement : la Figure 6.6 montre ceci pour la sortie du modèle de Cox et le modèle de Kaplan-Meier en ajoutant la courbe correspondant à la probabilité de demeurer abonné.

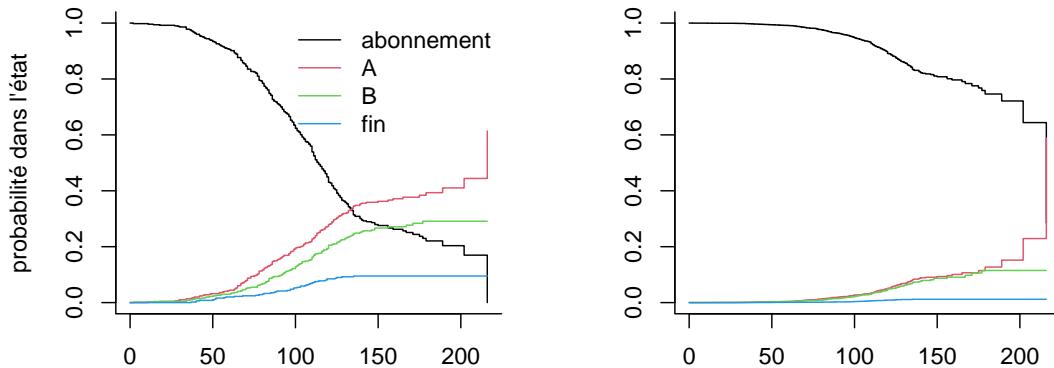


FIGURE 6.6 – Probabilité d'événement sans variable explicative (Kaplan-Meier, gauche) et avec âge, service et sexe (modèle de Cox, droite).

## 6.7 Risques non proportionnels

Comme son nom l'indique, le modèle à risques proportionnels suppose que les risques sont proportionnels. Cela implique que l'effet d'une variable est stable dans le temps. Nous verrons dans cette section deux façons de modéliser le cas de risques non proportionnels.

### 6.7.1 Non-proportionnalité avec un terme d'interaction avec le temps

Pour simplifier l'exposition, supposons que nous avons une seule variable explicative  $X$ . L'équation du modèle à risques proportionnels est  $h(t; x) = h_0(t) \exp(\beta x)$  et suppose que la fonction de risque de base  $h_0(t)$  est indépendante de la variable explicative  $X$ . Une manière de modéliser la non-proportionnalité est d'inclure un terme d'interaction entre la variable et le temps. Il existe plusieurs façons de le faire. Par exemple, on pourrait inclure une nouvelle variable qui est le produit entre le

temps et la variable  $X$ . Le modèle est alors

$$h(t; x) = h_0(t) \exp(\beta_1 x + \beta_2 x t).$$

Pour ce modèle, le rapport de risque, pour une augmentation d'une unité de  $X$  est  $\exp(\beta_1 + \beta_2 t)$  et dépend du temps  $t$  : c'est un modèle avec risques non proportionnels. On retombe sur le modèle à risques proportionnels lorsque  $\beta_2 = 0$ .

Supposons que l'effet du nombre de service augmente avec le temps. On peut inclure à la fois `service`, qui capture l'effet au temps zéro, et le surenchérissement ou la diminution à mesure que le temps d'abonnement progresse.

```
# Créer variables binaires par service
service_bin <- model.matrix(
  ~service,
  data = survie1)[-1]
# enlever ordonnée à l'origine (1e colonne)
survie1m <- cbind(survie1, service_bin)
cox_np <- survival::coxph(
  Surv(temps, censure) ~
    age + sexe + service +
    tt(service1) + tt(service2) + tt(service3),
  data = survie1m,
  tt = function(x, t, ...){t * x})
```

Le bloc code précédent illustre comment créer le terme non proportionnel : on spécifie avec l'option `tt()` la variable qui change dans le temps et on spécifie par la suite la nature de l'interaction temporelle en définissant une fonction `tt`.

Les variables catégorielles n'étant pas supportées en l'état, elles doivent préalablement être transformées en variable indicatrices binaires. Une fois les nouvelles variables dans la base de données, on procède à la spécification du terme de risque non proportionnel.

Le Tableau 6.9 contient les résultats. Les coefficients pour l'interaction avec  $t$  sont petits, mais c'est parce que la variable temps n'est pas standardisée et qu'elle s'étend de 0 à 200 semaines : de petites variations sont possiblement importantes quand le temps augmente. Les estimations des coefficients pour l'interaction sont tous positifs, ce qui suppose que le risque augmente avec le temps. Si on considère comme source plausible d'effet du nombre de services un quelconque rabais, il semble que cet effet protecteur s'amenuise. Deux des termes d'interaction sont significatifs à niveau 5% (statistiques de Wald  $Z$  de 2.173, 1.584 et 2.476 et valeurs- $p$  correspondantes de 0.03, 0.113 et 0.013).

Ajustons maintenant un autre terme d'interaction, cette fois de manière à répliquer les changements qui interviennent au sein d'une variable explicative de manière plus simple.

TABLEAU 6.9 – Rapport de risque et intervalles de confiance à niveau 95% pour le modèle à risques non proportionnels (interaction linéaire entre temps et service).

terme	exp(coef)	test de Wald	valeur-p
age	0.953	-7.368	<0.001
sexe	0.543	-5.241	<0.001
service1	0.144	-4.293	<0.001
service2	0.072	-3.762	<0.001
service3	0.010	-4.139	<0.001
tt(service1)	1.010	2.173	0.0298
tt(service2)	1.010	1.584	0.1132
tt(service3)	1.023	2.476	0.0133

```
# interaction entre service et temps
# objet avec 'tt' varie dynamiquement
cox6 <- coxph(
  Surv(temp, censure) ~
    tt(age) + sexe + service,
  data = survie1,
  tt = function(x, t, ...){x + t/52})
summary(cox6)
```

L'âge (en années) augmente à mesure que le temps d'abonnement (en semaines) passe, d'où  $\text{age}(t) = \text{age} + t/52$ . Ici, on inclut uniquement la variable transformée  $\text{tt}(\text{age})$  et son effet est toujours fortement significatif pour expliquer le désabonnement potentiel : les personnes plus âgées sont moins susceptibles de résilier leur abonnement.

```
cox_np <- survival::coxph(
  Surv(temp, censure) ~
    tt(age) + sexe + service,
  data = survie1,
  tt = function(x, t, ...){x + t/52})
summary(cox_np)
```

On spécifie avec l'option `tt()` dans la formule la variable qui change dans le temps et par la suite la nature de l'interaction temporelle avec l'argument `tt`.

L'interaction employée ici n'est pas la seule fonction du temps qu'on pourrait spécifier : on pourrait par exemple inclure une fonction de type escalier  $I(T < t_0)$  qui indique que l'effet de la variable dis-

TABLEAU 6.10 – Rapport de risque et intervalles de confiance à niveau 95% pour le modèle à risques non proportionnels (interaction linéaire entre temps et âge).

terme	exp(coef)	borne inf.	borne sup.
age	0.91	0.87	0.95
tt(age)	1.00	1.00	1.00
sexe	0.52	0.41	0.65
region2	0.70	0.48	1.02
region3	1.03	0.73	1.46
region4	0.80	0.56	1.12
region5	0.97	0.69	1.37
service1	0.36	0.28	0.46
service2	0.18	0.12	0.26
service3	0.12	0.07	0.20

paraît après le temps  $t_0$ , si par exemple un rabais disparaît après une certaine durée d'abonnement, avec `rabais * I(t<t0)` pour une valeur numérique  $t_0$  fixée.

### 6.7.2 Stratification

Une manière de modéliser la non-proportionnalité est par la stratification. Il est important de comprendre qu'on ne pourra pas estimer l'effet de la variable de stratification comme d'ordinaire en étudiant le coefficient  $\beta$  associé.

Supposons que nous avons une variable explicative catégorielle  $Z = 1, \dots, K$  pour lequel le postulat de risque proportionnels n'est pas valide. On s'intéresse à l'effet des variables  $\mathbf{X}$ . Typiquement, on ne s'intéresse pas directement à l'effet de la variable  $Z$ . Le modèle de Cox avec stratification (pour la variable  $Z$ ) est

$$h(t; \mathbf{x}, z = k) = h_k(t) \exp(\boldsymbol{\beta} \mathbf{x}),$$

où  $h_k(t)$  est la fonction de risque de base quand  $Z = k$  ( $k = 1, \dots, K$ ). L'effet des autres variables explicatives  $\mathbf{X}$  est supposé être le même peut importe la valeur de  $Z$ , mais la fonction de risque de base peut différer. Le rapport de risque pour  $Z = k$  versus  $Z = j$  est  $h_k(t)/h_j(t)$ ; cette quantité dépend du temps  $t$ , mais pas des autres caractéristiques mesurées par  $\mathbf{X}$ . L'effet de la variable est donc variable dans le temps (et non pas constant). Ce modèle permet donc de modéliser la non-proportionnalité pour la variable  $Z$ . Si on stratifie par rapport à une variable, il ne faut pas l'inclure dans le modèle en plus car elle est déjà modélisée via la stratification. Notez que les paramètres  $\boldsymbol{\beta}$  seront estimés à l'aide des données de toutes les strates, mais les fonctions de risque  $h_k(t)$  seront obtenues à l'aide des sous-échantillons correspondant aux valeurs de  $Z$ .

TABLEAU 6.11 – Rapport de risques et intervalles de confiances à niveau 95% pour le modèle de Cox stratifié par service.

terme	exp(coef)	borne inf.	borne sup.
age	0.96	0.94	0.97
sexe	0.61	0.44	0.85

L'avantage de la stratification est que cette méthode permet de modéliser n'importe quel changement dans l'effet d'une variable dans le temps sans devoir spécifier un type de changement particulier, comme lorsqu'on doit choisir la forme de l'interaction. Par contre, on perd la possibilité de tester l'effet de la variable de stratification ou réduit la taille de l'échantillon pour l'estimation de la fonction de risque de base. On devrait principalement utiliser la stratification seulement avec des variables pour lesquelles nous n'avons pas besoin d'estimer l'effet (variables secondaires ou de contrôles).

```
# Stratification par service
cox7 <- coxph(Surv(temp, 1-censure) ~
                 age + sexe + strata(service),
                 data = survie1)

# Décompte par région
with(survie1, table(service))

# Coefficients
summary(cox7)
```

On voit à la lecture de la sortie dans le Tableau 6.11 qu'il n'y a plus de paramètres pour la variable `service`. Les paramètres des autres variables s'interprètent comme d'habitude. On peut néanmoins résumer l'information pour `service` en calculant une statistique descriptive, par exemple les différences de survie à des temps donnés.

La Figure 6.7 illustre l'effet de la stratification sur l'estimation du risque de base et des courbes de survie. On voit que la résolution des courbes est moindre, puisque chaque fonction est estimée à partir d'un sous-ensemble des données.

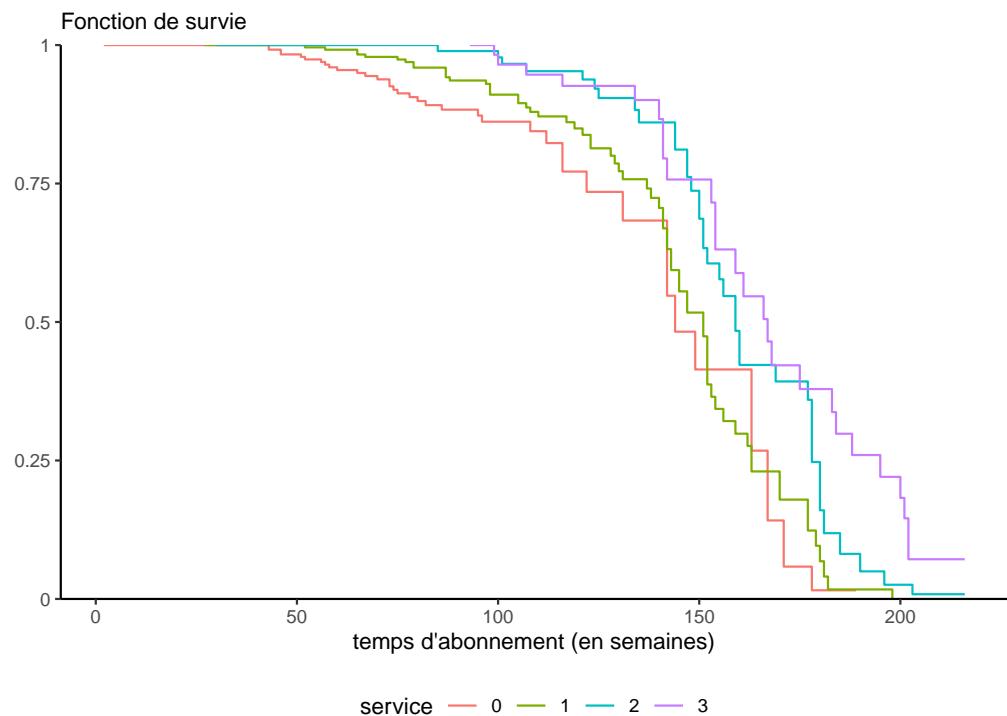


FIGURE 6.7 – Courbes de survie estimées par nombre de service pour le modèle de Cox avec stratification pour un homme de 40 ans.



## Chapitre 7

# Analyse de regroupements

### 7.1 Introduction

Si la publicité ciblée personnalisée a pris de l'essort ces derniers années en commercialisation, la segmentation de consommateurs reste une partie prenante essentielle de toute campagne de publicité ou de développement de produits.

L'analyse de regroupement est une technique d'**analyse descriptive** qui sert à combiner des sujets en groupes de telle sorte que les individus d'un même groupe soient le plus semblables possible et que les groupes soient le plus différent possible les uns des autres, avec des valeurs aberrantes clairement identifiées. Cette similarité est définie selon des caractéristiques provenant de variables explicatives. Le résultat de l'analyse de regroupement sera une étiquette associée à chaque observation l'assignant à un regroupement ou l'identifiant comme aberrance, nous permettant ainsi de caractériser par le biais de statistiques descriptives les différents **segments** obtenus.

Il y a une certaine analogie avec l'analyse factorielle. En analyse factorielle, on cherche à déterminer s'il y a des groupes de **variables** corrélées entre elles et à les regrouper pour réduire le nombre de variables. En analyse de regroupements, on cherche plutôt à créer des groupes d'**observations** similaires. Les deux méthodes servent pour l'analyse exploratoire ou descriptive.

Pour créer les regroupements, on utilisera  $p$  variables explicatives  $X_1, \dots, X_p$  pour chacune des  $n$  observations, où  $X_{ij}$  dénotera la valeur de la  $j$ e variable explicative pour le  $i$ e sujet.

#### 💡 Étapes d'une analyse de regroupements

1. Choisir les variables pertinentes à l'analyse. Cette étape peut nécessiter de créer, transformer de nouvelles variables ou d'agréger les données.
2. Décider quel méthode sera utilisée pour la segmentation.

3. Choisir les hyperparamètres de l'algorithme (nombre de regroupements, rayon, etc.) et la mesure de dissemblance.
4. Valider la qualité de la segmentation (interprétabilité, taille des groupes, homogénéité des regroupements).
5. Avec les étiquettes, calculer un prototype de groupe.
6. Interpréter les regroupements obtenus à partir des prototypes

## 7.2 Données

Voici en vrac quelques exemples de bases de données sur lesquelles on pourrait effectuer une analyse de regroupements.

Les programmes de fidélisation font partie de la stratégie de commercialisation de plusieurs grandes chaînes (pharmacies, épiceries) : en échange de rabais et d'offres promotionnelles, la clientèle fournit des informations sociodémographique (nom, adresse, date de naissance, etc.) et utilise un identifiant numérique, une carte ou une application pour inscrire chaque achat : ce faisant, le système peut traquer les habitudes de consommation.<sup>1</sup> Créer des regroupements permet de mieux cerner les besoins et habitudes de segments de consommateurs et ainsi d'adapter l'offre promotionnelle. Les algorithmes utilisés pour l'analyse de regroupements peuvent également servir à la résolution d'entité, qui consiste à fusionner les profils de bases de données sans identifiant unique client.

Un autre exemple d'application de l'analyse de regroupements est la segmentation de la clientèle de transport en commun. Dans la région métropolitaine de Montréal, l'Agence régionale de transport métropolitain recueille des informations sur les passages et transactions par le biais des cartes à puce Opus (achat de passes mensuelles ou de billets unitaires, lieu de l'achat, etc.) ainsi que les passages (heure, type de véhicule, emplacement approximatif pour les services d'autobus ou station de métro). En créant des regroupements, une agence de transport peut ainsi ajuster son offre et proposer des abonnements ou des produits qui reflètent les besoins de sa clientèle. Un exemple extrême de traquage de compagnie de transport est *Nederlandse Spoorwegen* (NS) : toute personne qui veut voyager en train sur les chemins de fers néerlandais doit acheter une carte à puce et la charger, en plus de composter son billet au départ et à l'arrivée de son voyage. Cette approche, qui peut sembler intrusive, permet néanmoins de mesurer précisément la demande sur les lignes en fonction du moment de la journée et de l'associer à chaque client.

Souvent, les bases de données marketing sont souvent de nature longitudinale : chaque ligne correspond à une transaction, mais plusieurs d'entre elles peuvent être le fait d'une même personne/compte. Une fois l'analyse exploratoire des données complétée, on procédera à l'aggrégation

1. Il existe bien sûr d'autres méthodes de traque pour les personnes qui n'ont pas de compte client, notamment par le biais de numéros de carte de débit ou de crédit qui permettent de regrouper les transactions.

des observations par compte client, puisque la segmentation doit être effectuée à cette échelle. C'est également à ce stade qu'on pourra créer de nouvelles variables explicatives à partir de l'information présente dans la base de données : par exemple, on pourrait considérer la fréquence moyenne d'achat, le montant moyen par transaction, le mode du moment de la journée, la variabilité de cette fréquentation, le pourcentage des ventes provenant d'articles en solde, la variabilité du montant du panier, etc. Cette liste, non exhaustive, illustre l'étape cruciale de l'extraction de l'information utilisée dans l'analyse statistique : il faut être conscients que la qualité de la segmentation dépend du choix de variables employées.

### 7.3 Choix des variables

L'analyste est libre de choisir quelles variables seront incluses dans le modèle. Le choix des variables est important : en général on veut créer des groupes d'individus qui sont homogènes par rapport à certains aspects de leur comportement ou de leur situation. On ne doit alors inclure que les variables pertinentes à cet aspect. Inclure de nombreuses variables pour lesquelles il y a une forte similitude entre individus contribue à diluer les différences.

Par exemple, si le but de l'analyse est de segmenter nos clients selon leurs habitudes de consommation (genre de boutiques fréquenté, fréquence, etc.), on n'inclura pas des variables démographiques qui feraient ressortir les différences de genre, d'âge, de revenu, etc. En fait, souvent l'analyse de regroupements servira justement à créer des groupes qui seront comparés par rapport à d'autres variables qui n'ont pas été utilisées pour créer les groupes.

La compréhension de la base de données est cruciale pour comprendre le comportement. Si on essaie de faire une segmentation du comportement d'utilisateurs et utilisatrices de transports en commun à partir d'informations auxiliaires comme le temps de passage, le nombre de correspondance et la fréquence d'utilisation, il peut être utile de créer de nouvelles variables (par exemple, une variable indicatrice qui indique si la personne voyage durant les heures de traffic entre 7h30 et 9h et 16h à 18h, le nombre hebdomadaire moyen de jours ouvrables pendant lesquels elle se déplace, etc). L'inclusion des ces variables auxiliaires peut augmenter la qualité de la segmentation.

Pour voir si certaines variables sont inutiles, il peut être utile de comparer les représentants des groupes (par exemple, le barycentre ou une observation lambda du groupe) pour voir si les moyennes ou caractéristiques diffèrent. Si ce n'est pas le cas, on pourrait envisager de recommencer la procédure en enlevant cette variable.

Si on a un nombre important de variables explicatives à disposition, il est parfois utile de réduire préalablement la dimension (par exemple, en effectuant une analyse en composantes principales) et à ne retenir que les premières composantes pour faciliter la tâche. Cette approche n'est pas la panacée : quelquefois, cette réduction de la dimension masque les différences entre groupes et mène à une segmentation inférieure à l'utilisation des variables originales.

Malheureusement, il n'est pas évident de prime abord de déterminer quelles variables inclure dans la base de données pas plus qu'il n'est facile de juger de la qualité d'une segmentation ou du nombre de regroupements à effectuer. Les choix individuels auront un impact certain sur les regroupements obtenus : on recommande d'essayer plusieurs alternatives et de vérifier graphiquement ou à l'aide de critères d'ajustement si les regroupements obtenus sont homogènes et compacts.

Si certaines variables définissent naturellement des groupes, par exemple l'âge des personnes, et fait qu'ils et elles ont des caractéristiques intrinsèquement différentes, il peut être utile de faire une segmentation indépendamment pour chacun de ces sous-groupes.

Dans ce chapitre, nous utiliserons des données simulées inspirées de campagnes de financement d'organismes de charité. Ces dernières font souvent du démarchage publicitaire auprès de donateurs ou envoient par publipostage des demandes de dons à toutes les adresses postales. Ces efforts ont un coût important : nous essaierons de créer des catégories de donateurs afin de mieux cibler les donateurs et donatrices et le moment adéquat pour ce démarchage. Plusieurs grandes compagnies sont associées à ces organismes et les parrainent : notre base de données contiendra le profil de toutes ces personnes, qu'elles fassent un don ou pas.

La base de données dons contient 49730 observations pour 16 variables : le Tableau 7.1 fournit les statistiques descriptives. Elle a été créée en regroupement les identifiants : de nombreuses variables explicatives sont dérivées des données brutes, notamment le temps entre dons, les statistiques descriptives (montant moyen, minimum maximum) pour les dons monétaires. Ces choix de variables sont loin d'être anodins et peuvent influencer la segmentation décrite dans ce chapitre. Une rapide exploration des données révèle que près de 61% des employé(e)s n'ont pas donné à l'organisme. Une poignée de dons sont très élevés, mais la plupart des montants tourne autour de 5\$, 10\$, 20\$, etc.

La grande proportion de données manquantes pose un problème immédiat pour la segmentation, puisque la plupart des procédures ne permettent pas de traiter ces dernières et éliminent d'office les observations correspondantes de la base des données. Ici, plusieurs valeurs manquantes (NA) peuvent être logiquement remplacées par des valeurs numériques : par exemple, la valeur cumulative des dons (vdons) d'une personne qui n'a jamais donné est nulle.<sup>2</sup> En revanche, le temps d'attente entre deux dons pour une personne qui a fait un don ou moins n'est pas bien défini.

Si on essaie de créer manuellement des groupes, il apparaît logique de séparer en trois segments initiaux la base de données : les personnes qui n'ont jamais donné à l'organisme de charité mais dont les caractéristiques sont connues, les personnes qui ont fait un seul don et celles qui ont fait des dons multiples. Un algorithme ferait de toute façon vraisemblablement ressortir cette information, mais nous empêcherait d'exploiter pleinement l'ensemble des variables explicatives et de ses dérivées. On pourra effectuer la segmentation séparément sur chaque groupe avec en intrant des variables explicatives différentes.

Les intrants de l'analyse de regroupement (soit le choix des variables) est laissé à la discrédition

---

2. Imputer par la moyenne ou utiliser une méthode plus sophistiquée serait illogique (et incorrect).

TABLEAU 7.1 – Statistiques descriptives des variables du jeu de données dons.

variable	moyenne	écart-type	min	max	manquant
ndons	5.13	5.21	1	27.0	30377
recence	86.41	81.80	2	299.0	30377
anciennete	168.72	95.77	2	302.0	30377
vdons	117.66	431.08	1	19260.0	30377
vdonsmax	30.72	88.13	1	2460.0	30377
vdonsmin	10.98	28.79	1	1570.0	30377
npromesse	1.72	2.37	0	15.0	30377
vpromesse	61.32	282.07	0	12680.0	30377
nradiations	0.52	0.89	0	10.0	30377
vradiations	28.09	141.47	0	11815.0	37519
ddons	2.13	1.94	0	23.9	36113
ddonsmax	3.85	3.24	0	23.9	36113
ddonsmin	1.32	1.83	0	23.9	36113
nrefus	1.67	1.98	0	11.0	0
nrefusconsec	1.43	1.86	0	11.0	0
nindecis	0.31	0.76	0	8.0	0

de l'analyste. Dans notre exemple, on pourrait aisément créer de nouvelles variables pour faire ressortir des informations jugées pertinentes. Est-ce qu'on s'intéresse au montant moyen des dons, soit `vdons/ndons`? Est-ce que la valeur des radiations nous intéresse, ou bien devrait-on plutôt considérer le pourcentage de la valeur promise réalisée?

On considère ci-dessous l'ensemble des personnes qui ont fait plusieurs dons. On modifie certaines variables explicatives pour réduire la corrélation entre variables et obtenir des variables plus évocatrices : le montant moyen de dons, le nombre de refus relatif à l'ancienneté du donneur ou de la donatrice et finalement la valeur de la promesse moyenne, si applicable (zéro sinon). Plusieurs variables (délais minimum et maximum entre dons, valeurs minimum, radiations, etc.) sont également abandonnées pour simplifier l'exposition et pour éviter qu'elles ne ressortent indûment. On voit également que plusieurs valeurs de radiations sont manquantes : cette variable est éliminée d'office.

```
donsmult <- dons |>
  filter(ndons > 1L) |>
  mutate(mtdons = vdons/ndons,
         snrefus = nrefus/anciennete*mean(anciennete),
         mpromesse = case_when(
           npromesse > 0 ~ vpromesse/npromesse,
           TRUE ~ 0)) |>
  select(!c(
    vradiations, # valeurs manquantes
    nindecis, vdons, ddonsmax,
    ddonsmin, vdonsmin, npromesse,
    vpromesse, nrefus, nradiations)) |>
  relocate(mtdons)
```

Le champ des applications de l'analyse de regroupements est parfois surprenant. Par exemple, cet article de FiveThirtyEight propose une segmentation des électeurs démocrates new-yorkais ou des quartiers de Los Angeles. Un autre exemple incongru est la compression d'images : la Figure 7.1 montre une image du bâtiment Decelles (coin supérieur gauche) et la reconstruction avec trois, quatre et 10 couleurs obtenues en appliquant l'algorithme des  $K$ -moyennes sur la matrice formée par les valeurs des canaux (rouge, vert, bleu) de l'image.

## 7.4 Mesures de dissemblance

Comment mesurer si deux observations appartiennent à un même regroupement et sont similaires ? Idéalement, on aimerait avoir une situation comme dans la Figure 7.2 où les regroupements sont clairement visibles. On aimerait que la similarité entre observations d'un même groupe, ou intra-



FIGURE 7.1 – Compression d'image avec l'algorithme des  $K$ -moyennes : image originale (en haut à gauche), compression avec trois (en haut à droite), quatre (en bas à gauche) et 10 (en bas à droite) couleurs.

groupe, soit élevée et que la similarité entre groupe soit faible. Les regroupements devraient être éloignés les uns des autres, tandis que les observations au sein de ces regroupements devraient être proches. Dans la plupart des cas, il y aura des observations isolées qui n'appartiennent pas nécessairement logiquement à l'un ou l'autre des groupes : on appelle parfois ces observations aberrances.

#### 7.4.1 Mesures de dissemblance

Les algorithmes de segmentation comparent les observations entre elles : souvent, la matrice de données est réduite à une mesure de distance entre observations (soit les lignes de la base de

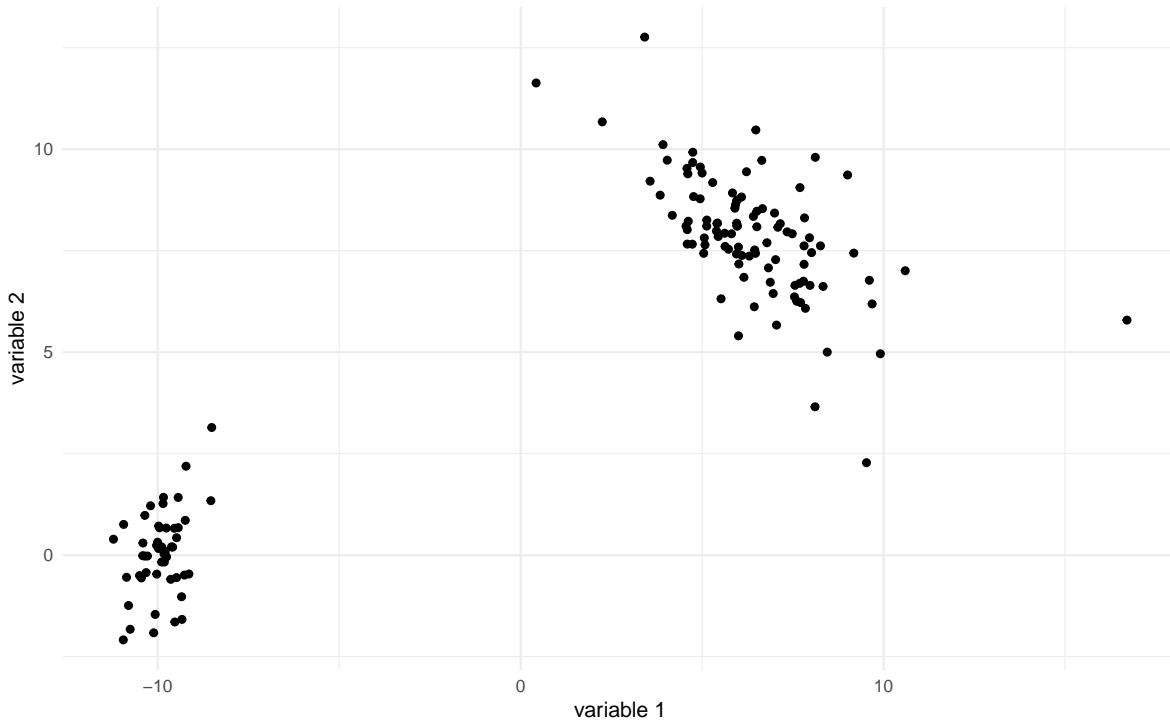


FIGURE 7.2 – Données simulées avec deux regroupements hypothétiques.

données). Une **mesure de dissemblance** sert à quantifier la proximité de deux objets à partir de leurs coordonnées. Elle mesure la distance entre deux vecteurs lignes d’observations  $\mathbf{X}_i$  et  $\mathbf{X}_j$  en se basant sur les  $p$  variables explicatives. Plus la dissemblance est petite, plus les sujets  $\mathbf{X}_i$  et  $\mathbf{X}_j$  sont similaires. La plupart des mesures de dissemblances  $d$  ont les propriétés mathématiques suivantes :

- 1)  $d(\mathbf{X}_i, \mathbf{X}_j) \geq 0$  (positivité), avec égalité (distance nulle) si et seulement si  $\mathbf{X}_i = \mathbf{X}_j$  (mêmes caractéristiques pour toutes les variables explicatives);
- 2)  $d(\mathbf{X}_i, \mathbf{X}_j) = d(\mathbf{X}_j, \mathbf{X}_i)$  (symmétrie);

Toute mesure de distance<sup>3</sup> est une mesure de dissemblance. La mesure de dissemblance la plus utilisée en pratique est la distance euclidienne entre sujets, soit

$$d(\mathbf{X}_i, \mathbf{X}_j; l_2) = \{(X_{i1} - X_{j1})^2 + \cdots + (X_{ip} - X_{jp})^2\}^{1/2}.$$

C'est tout simplement la longueur du segment qui relie deux points dans l'espace  $p$  dimensionnel.

---

3. Une fonction de distance respecte en plus l'inégalité du triangle.

Plus généralement, la distance de Minkowski ou distance  $l_q$  entre les vecteurs ligne  $\mathbf{X}_i$  et  $\mathbf{X}_j$  est

$$d(\mathbf{X}_i, \mathbf{X}_j; l_q) = \left( \sum_{k=1}^p |X_{ik} - X_{jk}|^q \right)^{1/q}, \quad q > 0;$$

la distance Euclidienne correspondant à  $q = 2$ , et la distance de Manhattan à  $q = 1$ .<sup>4</sup> Finalement, si  $q = \infty$ , la distance se réduit à  $\max_{k=1}^p |X_{ik} - X_{jk}|$ , soit le maximum des différences entre coordonnées des vecteurs d'observations.

Il existe un très grand nombre d'autres mesures de dissemblance pour variables quantitatives, ordinaires, nominales et binaires. Si les variables sont toutes binaires, la mesure d'appariement simple (*simple matching*), qui mesure la proportion des variables pour lesquelles les deux sujets ont des valeurs différentes, est une mesure de dissemblance adéquate.

Dans le cas de jeux de données avec des variables mixtes, une option populaire est la distance de Gower [@Gower :1971]. Cette dernière compare deux individus selon leurs caractéristiques et est construite à partir de similarité, avec  $\mathbf{D} = (\mathbf{I}_n - \mathbf{S})^{1/2}$  comme matrice de dissimilarité des  $n$  observations. La similarité entre deux individus est définie comme

$$S_{ij} = \frac{\sum_{k=1}^p s_{ijk} \delta_{ijk}}{\sum_{k=1}^p \delta_{ijk}}$$

où  $\delta_{ijk}$  est un poids qui vaut zéro si la variable  $X_k$  est manquante pour l'un ou l'autre des individus.

On distingue trois types de variables dans la distance de Gowers :

- les variables binaires asymétrique de type absence/présence donnent une valeur de  $\delta = 1, s = 1$  si les deux sont présentes  $X_{ik} = X_{jk} = 1$ ,  $\delta_{ijk} = 1$  et  $s_{ijk} = 0$  si  $X_{ik} \neq X_{jk}$  et  $\delta_{ijk} = 0$  si  $X_{ik} = X_{jk} = 0$ .
- $s_{ijk} = 1$  les variables qualitatives ont la même modalité et  $s_{ijk} = 0$  sinon
- $s_{ijk} = 1 - |X_{ik} - X_{jk}|/R_k$  pour une variable continue, où  $R_k$  est l'étendue de la variable  $R_k = \max_i X_{ik} - \min_i X_{ik}$  dans l'échantillon.

La dissemblance résultante pour les types mixtes vaut zéro quand toutes les variables sont similaires/égales et un si elles sont complètement différentes/maximalemen distantes.

On peut traiter les variables ordinaires soit comme des variables continues, soit comme des variables nominales avec la mesure d'appariement simple; ce faisant, on n'utilise pas l'ordre entre les modalités.

---

4. La distance de Manhattan est la somme des valeurs absolues entre chaque composante. En deux dimensions, si on considère une ville comme New York dont les rues sont quadrillées, cela revient à marcher le long des rues alors que la distance Euclidienne traverse les édifices.

### 7.4.2 Dissemblance et valeurs manquantes

Dans plusieurs cas, on se trouvera en présence de valeurs manquantes dans le jeu de données. Cela peut arriver pour plusieurs raisons valables (aucune candidature ne représente un parti dans une circonscription donnée pour un parti lors d'une élection, l'information est manquante, une femme ne peut avoir de cancer de la prostate, etc.) Il faut bien penser à vérifier si l'algorithme de votre choix peut gérer ces valeurs manquantes. Sinon, ces dernières devront être imputées préalablement à l'analyse de regroupements ou vous devrez faire sans les variables explicatives correspondantes.

Les définitions des distances révèlent que chaque variable explicative a le même poids. En revanche, plus une variable a une grande variance, plus elle aura de l'influence sur le calcul de la distance, ce qui peut être bon ou mauvais selon la structure des groupes. Règle générale, il est préférable d'éviter qu'une variable domine dans la segmentation. La standardisation des variables et les transformations préalables effectuées sur les variables (log, arcsin, etc.) impacteront le résultat.

On peut standardiser au préalable les variables avant de faire l'analyse. Par défaut, les variables continues seront centrées et réduites, ou standardisées, afin d'avoir une moyenne de zéro et une variance de un (`scale`). On peut ensuite faire les analyses comme précédemment. Si on a des valeurs aberrantes, cela peut impacter le calcul des moyennes et variances ; d'autres estimateurs de localisation et d'échelles plus robustes, par exemple la médiane et la déviation absolue par rapport à la médiane (`mad`) peuvent alors être plus adéquats pour diminuer l'impact des valeurs aberrantes même si le coût de calcul associé est plus conséquent. Notez qu'il est illogique de standardiser les variables binaires et catégorielles.

```
# Standardisation usuelle
# (soustraire moyenne, diviser par écart-type)
donsmult_std <- scale(donsmult)
# Standardisation robuste
donsmult_std_rob <- apply(
  donsmult,
  MARGIN = 2,
  FUN = function(x){(x - median(x))/mad(x)})
# apply permet d'appliquer une fonction
# par ligne, colonne ou cellule
# MARGIN = 2 indique colonne
# (on centre chaque colonne tour à tour)
# Déviation absolue par rapport à la médiane
# mad = moyenne de |obs - mediane|
```

## 7.5 Algorithmes pour la segmentation

L'analyse de regroupements est une branche de l'apprentissage non-supervisé : contrairement à la classification, il n'existe pas de vraies étiquettes sur lesquelles se baser pour déterminer la qualité d'une segmentation. Des critères graphiques et des mesures d'homogénéité peuvent néanmoins déterminer à quel points les segments créés sont distincts les uns des autres.

L'analyse de regroupements cherche à créer une division de  $n$  observations de  $p$  variables en  $k$  regroupements. Il existe un grand nombre d'algorithmes qui permettent de partitionner les données en regroupements à partir d'un jeu de données ou d'une matrice de dissemblance. Les sections suivantes survoleront différents algorithmes en s'attardant à l'heuristique de l'implémentation, aux différentes étapes de la procédure, aux hyperparamètres qui influencent le résultat (par ex., le nombre de groupes, la distance minimale entre regroupements, la forme des regroupements, les éléments représentatifs) qui détermine la sortie ainsi que les forces et faiblesses des algorithmes. À l'ère des mégadonnées, la complexité d'un algorithme de regroupements, une mesure du nombre d'opérations nécessaires pour effectuer le calcul, impactera le choix possible : l'algorithme de regroupements hiérarchiques (agglomératif ou divisif), de même que l'algorithme de partition autour des médoïdes (PAM) sont à proscrire dans ces scénarios. Outre l'algorithme, il y a des coûts associés au calcul de la matrice de dissemblance entre chacune des paires des  $n$  observations : cette opération nécessite  $O(n^2 p)$  flops pour le calcul et  $O(n^2)$  entrées de stockage.<sup>5</sup> Dans le cas de matrice creuses avec beaucoup de zéros, le coût de stockage et le coût pour réaliser des opérations matricielles (décomposition en valeurs propres et vecteurs propres) peut être réduit à l'aide d'algorithmes dédiés.

Les méthodes de regroupement peuvent être regroupées grossièrement dans les catégories suivantes :

1. méthodes basées sur les centroïdes et les médoïdes ( $k$ -moyennes,  $k$ -médoides PAM, CLARA)
2. mélanges de modèles (mélanges Gaussiens, etc.)
3. méthodes basées sur la connectivité (regroupements hiérarchiques, AGNES et DIANA)
4. méthodes basées sur la densité (DBScan)

Dans certaines méthodes paramétriques (catégories 1 à 3), le nombre de groupes est fixé a priori et est un hyperparamètre du modèle. Les méthodes nonparamétriques déterminent plutôt ce nombre automatiquement, mais spécifient un paramètre qui contrôle le degré de lissage.

Nous survolerons uniquement les caractéristiques des principales méthodes.

---

5. Soit 740MB d'espace dans la mémoire vive pour stocker la moitié de la matrice de dissemblance 13617 par 13617 (la matrice étant symétrique).

### 7.5.1 *K*-moyennes

L'algorithme des *K*-moyennes est un des plus couramment employé en raison de son faible coût. L'idée est la suivante : on assigne chaque observation à un de *K* regroupements et on calcule la distance entre cette dernière et un prototype  $\mu_k$  pour le regroupement *k*. La fonction objective que l'on cherche à minimiser est

$$\min_{\mu_1, \dots, \mu_K} \min_{\substack{r_{ik} \in \{0,1\} \\ r_{i1} + \dots + r_{iK} = 1}} \sum_{i=1}^n \sum_{k=1}^K r_{ik} d(\mathbf{X}_i, \mu_k) \quad (7.1)$$

distance entre obs. *i* et le prototype le plus près

où  $r_{ik} = 1$  si l'observation  $\mathbf{X}_i$  (soit la *i*e ligne de la base de données) est assignée au groupe *k*. Si on utilise la distance Euclidienne carrée, alors la fonction objective correspond à la somme du carré des erreurs au sein de chaque regroupement et on cherche à minimiser l'erreur quadratique moyenne. Les coordonnées optimales  $\hat{\mu}_k$  pour le prototype si on connaît les étiquettes de groupes sont celles du barycentre des  $n_k$  observations du groupe *k*, soit

$$\hat{\mu}_k = \frac{\sum_i r_{ik} \mathbf{X}_i}{n_k}, \quad k = 1, \dots, K;$$

d'où l'appellation *K*-moyennes. Si on utilise plutôt la distance de Manhattan ( $l_1$ ), alors la solution est la médiane coordonnée par coordonnées des observations du groupe. Il n'est pas possible de déterminer l'allocation optimale de *n* observations en *K* groupes (problème NP complet), mais il est en revanche possible de trouver rapidement une solution approximative au problème.

Pour ce faire, on sélectionne préalablement un nombre *K* de regroupements et les coordonnées de départ pour les prototypes. L'algorithme itère entre deux étapes :

1. **Assignation** (étape E) : calculer la distance entre chaque observation et les prototypes ; assigner chaque observation au prototype le plus près.
2. **Mise à jour** (étape M) : estimer les coordonnées des nouveaux prototypes ; si on utilise la distance Euclidienne, cela revient à calculer le barycentre (la moyenne variable par variable) des observations assignées aux regroupements.

En pratique, l'algorithme convergera rapidement vers une solution locale. Cette dernière est simplement une assignation pour laquelle, d'une itération à l'autre, aucune observation ne change de groupe.

L'algorithme des *K*-moyennes présenté offre une forme de partitionnement dite rigide : chaque observation est assignée à un seul regroupement. Si cette appartenance unique peut être logique pour les points à proximité du barycentre, ceux situés à l'intersection des frontières qui définissent les différents regroupements pourraient parfois légitimement faire partie d'un ou l'autre de ces derniers. On pourrait plutôt assigner un poids représentant la probabilité d'être dans un des *K* regroupements, appelé responsabilité et dénotée  $r_{ik}$ . Avec une assignation rigide,  $r_{ik} = 1$  si l'observation *i* est dans le regroupement *k* et  $r_{ik} = 0$  sinon.

Quelquefois, on peut vouloir prédire les étiquettes de groupes de nouvelles observations. Sans réentraîner l'algorithme, on pourrait ainsi assigner de nouvelles observations au barycentre le plus près.

Voici quelques forces et faiblesses de la méthode des  $K$ -moyennes

- L'algorithme des  $K$ -moyennes a une complexité **linéaire** dans la dimension et dans le nombre de variables, soit  $O(np)$ . Ce faible coût de calcul est un avantage avec des mégadonnées ( $n$  grand) et en haute dimension  $p$  grand).
- L'algorithme converge rapidement vers une solution et on a des garanties que la solution est un maximum local, puisque l'algorithme minimise les répartitions et les prototypes tour à tour.
- Les  $K$ -moyennes créent des regroupements globulaires d'apparence sphérique si on utilise la distance Euclidienne : cela revient à faire une séparation linéaire de l'espace (voir Figure 7.6).
- Chaque observation est assignée à un seul des  $K$  regroupements (assignation rigide).
- Comme toutes les observations font partie des  $K$  groupes, les valeurs aberrantes ne sont pas traitées à part. Or, la présence de valeurs aberrantes impacte le barycentre des observations du groupe. Comme ce dernier donne le prototype du groupe, l'algorithme manque de robustesse.
- L'algorithme est sensible aux valeurs initiales des prototypes et retourne des solutions différentes selon ces dernières.

### Choix des hyperparamètres

L'algorithme des  $K$ -moyennes comporte plusieurs paramètres, dit hyperparamètres, qui sont fixés par l'utilisateur préalablement à la segmentation. Ces derniers incluent

1. les valeurs initiales des prototypes
2. le nombre de groupes  $K$
3. le choix de la mesure de distance.

### Valeurs initiales des prototypes

Comme mentionné précédemment, les regroupements obtenus peuvent varier fortement en fonction des valeurs de départ : la Figure 7.4 montre trois regroupements visibles avec une segmentation qui fusionne deux groupes apparents (gauche), et une solution plus sensée à droite. Une segmentation sera supérieure à une autre si elle a une plus petite valeur de la fonction objective de l'Équation 7.1 : les points seront moins dispersés autour de leurs prototypes.

La solution la plus simple est de choisir aléatoirement des coordonnées initiales pour les prototypes et de répéter la segmentation plusieurs fois, en choisissant à la fin celle qui a la plus petite valeur du critère objectif.

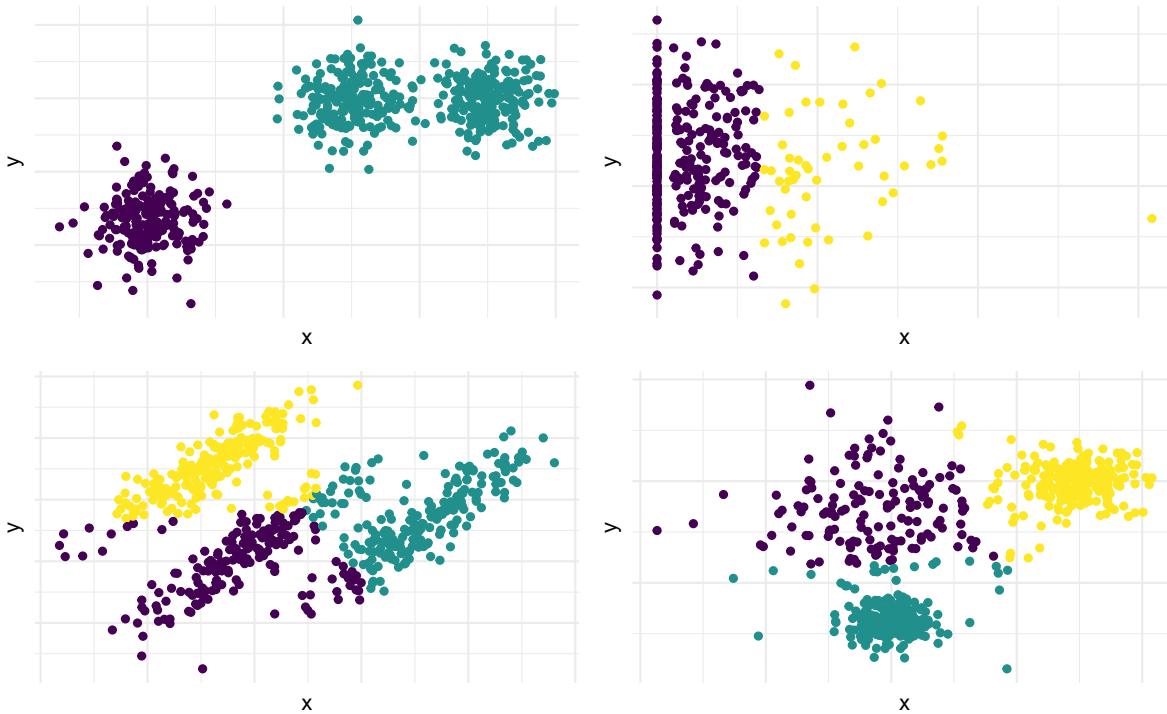


FIGURE 7.3 – Performance de l'algorithme des  $K$ -moyennes en fonction de différents scénarios : (haut, à gauche) nombre incorrect de classe et données normales de même variance, bien séparées, (haut, à droite) données avec excès de zéro, un cas où les  $K$ -moyennes ignorent la topologie des regroupements, et ne segmente pas adéquatement les regroupements connectés (bas, à gauche) données elliptiques de même variance, mais fortement corrélées. Comme le critère minimise la distance intra-groupe sans pondération, les points regroupés appartiennent à différentes classes et (bas, à droite) données sphériques de variances différentes. L'algorithme des  $K$ -moyennes réussit une bonne segmentation si les groupements sont compacts et bien séparés.

On peut également choisir des valeurs suffisamment éloignées : l'algorithme des  $K$ -moyennes<sup>++</sup> est une variante algorithmique qui propose de choisir des barycentres éloignés les uns des autres (ce qui réduit typiquement le nombre d'itérations). Cette méthode d'initialisation sélectionne une observation au hasard et on l'assigne comme premier prototype, disons  $\mu_1$ . Par la suite, on procède avec  $k = 2, \dots, K$  aux étapes suivantes :

1. calcul de la distance carrée minimale entre l'observation  $\mathbf{X}_i$  et les prototypes précédemment choisis,

$$p_i = \min\{d(\mathbf{X}_i, \mu_1; l_2)^2, \dots, d(\mathbf{X}_i, \mu_{k-1}; l_2)^2\}$$

2. Choisir la valeur initiale du  $k^{\text{e}}$  prototype au hasard parmi les observations avec une probabilité de  $p_i / \sum_j p_j$  pour l'observation  $\mathbf{X}_i$ .

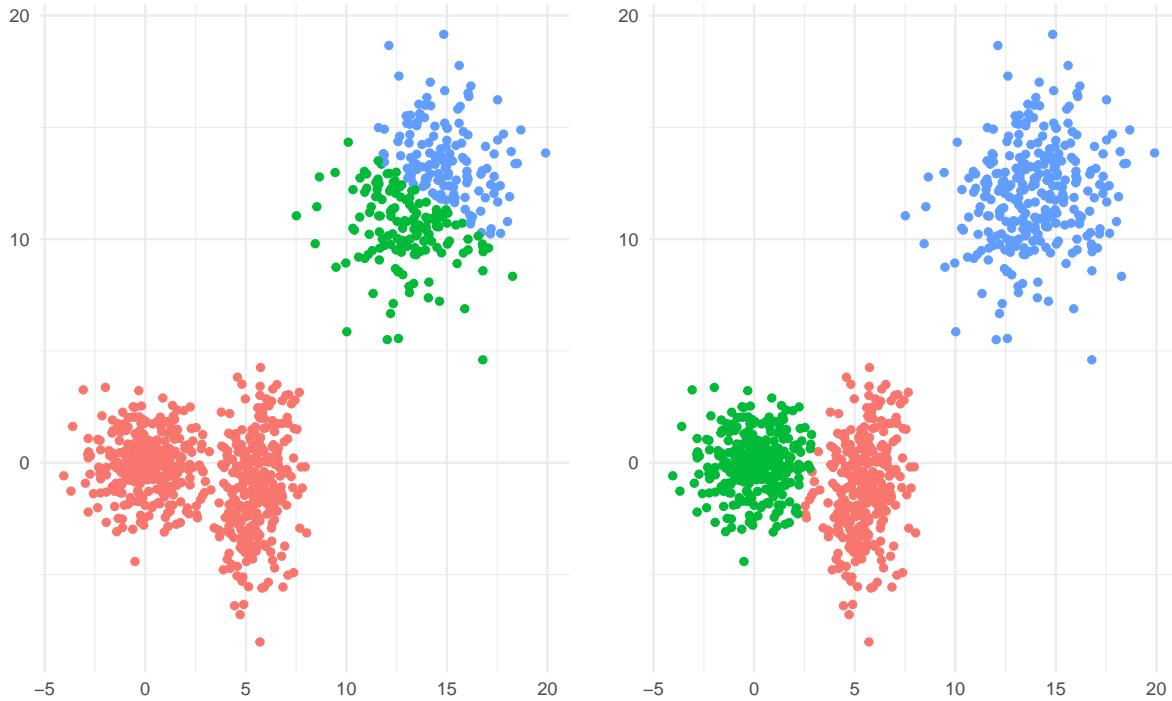


FIGURE 7.4 – Résultat d'une analyse de regroupement avec  $K = 3$  groupes avec une mauvaise initialisation principale (gauche) et une bonne initialisation (droite).

À la fin, on obtiendra  $K$  valeurs initiales qui serviront à l'initialisation. Ce faisant, on peut espérer ne pas avoir à faire plusieurs allocations aléatoires, puisque les valeurs de départ choisies sont raisonnablement éloignées les unes des autres.

### **Nombre de regroupements**

L'autre paramètre crucial des  $K$ -moyennes est le nombre de regroupements,  $K$ . Il est difficile de savoir combien de regroupements sélectionner a priori, puisque la visualisation en haute dimension est difficile et on est souvent loin de la situation présentée dans la Figure 7.4. On pourrait envisager de rouler l'algorithme avec plusieurs valeurs de  $K$  et de comparer les résultats, mais sur quelle base?

La fonction objective de l'Équation 7.1 avec la distance Euclidienne représente la somme du carré des distances (SCD) entre les observations d'un groupe et leur barycentre, soit la variabilité totale des observations des  $K$  différents groupes autour de leur barycentre,

$$\text{SCD}_K = \text{SCD}_{1,K} + \cdots + \text{SCD}_{K,K}$$

où la somme du carré des distances des observations du groupe  $k$  (pour lesquelles  $r_{ik} = 1$ )

$$\text{SCD}_{k,K} = \sum_{i=1}^n r_{ik} \|\mathbf{X}_i - \boldsymbol{\mu}_k\|_2 .$$

distance  $l_2$  entre obs. du groupe  $k$  et barycentre  $k$

La somme des carrés totales correspond à la somme du carré des distances au barycentre avec un seul regroupement,  $\text{SCT} = \text{SCD}_1$ .

La valeur optimale de la somme du carré des distances mesure va mécaniquement diminuer à mesure que le nombre de regroupements augmente parce que le modèle aura plus d'opportunités pour réduire la variabilité intra-groupe, donc  $\text{SCD}_1 > \text{SCD}_2 \dots$ . En pratique, cela peut ne pas être le cas si le minimum local est sous-optimal. Si la réduction de la somme du carré des distances est négligeable, on pourrait penser que la valeur ajoutée d'un groupe supplémentaire (qui implique plus de paramètres à estimer et plus de segments à interpréter) est faible.

On peut calculer un coefficient de détermination, qui mesure pourcentage de variance expliquée, soit  $R_K^2 = 1 - \text{SCD}_K/\text{SCT}$ . De la même manière, on s'attend à une diminution du critère et on pourrait calculer le  $R^2$  semi-partiel  $(\text{SCD}_k - \text{SCD}_{k-1})/\text{SCT}$  pour  $k \geq 2$ .<sup>6</sup>

On pourrait aussi tracer un diagramme de la somme du carré des distances en fonction de  $K$  en ajoutant une pénalité à notre fonction objective. En effet, avec la distance Euclidienne carrée, il y a une analogie à faire avec un modèle de régression et on peut légitimement utiliser un critère d'information pour guider notre choix de  $K$  : le nombre de paramètres est  $Kp$ , soit les valeurs des  $p$  coordonnées des  $K$  barycentres. On utilisera donc un critère d'information de type BIC.

Il est possible que ces critères donne beaucoup plus de regroupements que ce que l'analyste est prêt(e) à envisager. Il faut garder en tête que, davantage qu'un critère mathématique, l'interprétabilité des regroupements est notre principale critère. Les critères d'information peuvent retourner trop ou pas assez de groupe : à titre d'exemple, le panneau de gauche de la Figure 7.5 montre la somme du carré des distances pour la Figure 7.4; on voit un coude à  $K = 2$ , mais il y avait visiblement trois regroupements, dont deux rapprochés.

### Mesure de distance

Toutes les distances  $l_q$  peuvent être utilisées, mais le choix de la distance Euclidienne carrée est particulièrement commode et populaire<sup>7</sup> entraîne une partition linéaire de l'espace, comme l'illustre la Figure 7.6. La solution du problème d'optimisation est explicite, ce qui accélère les calculs (les prototypes correspondent aux barycentres). Sauf indication contraire, on supposera dans ce

---

6. Ces critères servent également pour les regroupement hiérarchiques avec le critère de Ward.

7. La fonction objective s'apparente alors à la somme du carré des erreurs, et donc il y a une analogie à faire avec la vraisemblance d'un modèle Gaussien en dimension  $p$  de covariance sphérique. Cela légitime l'emploi de critères d'information pour le choix du nombre de regroupements.

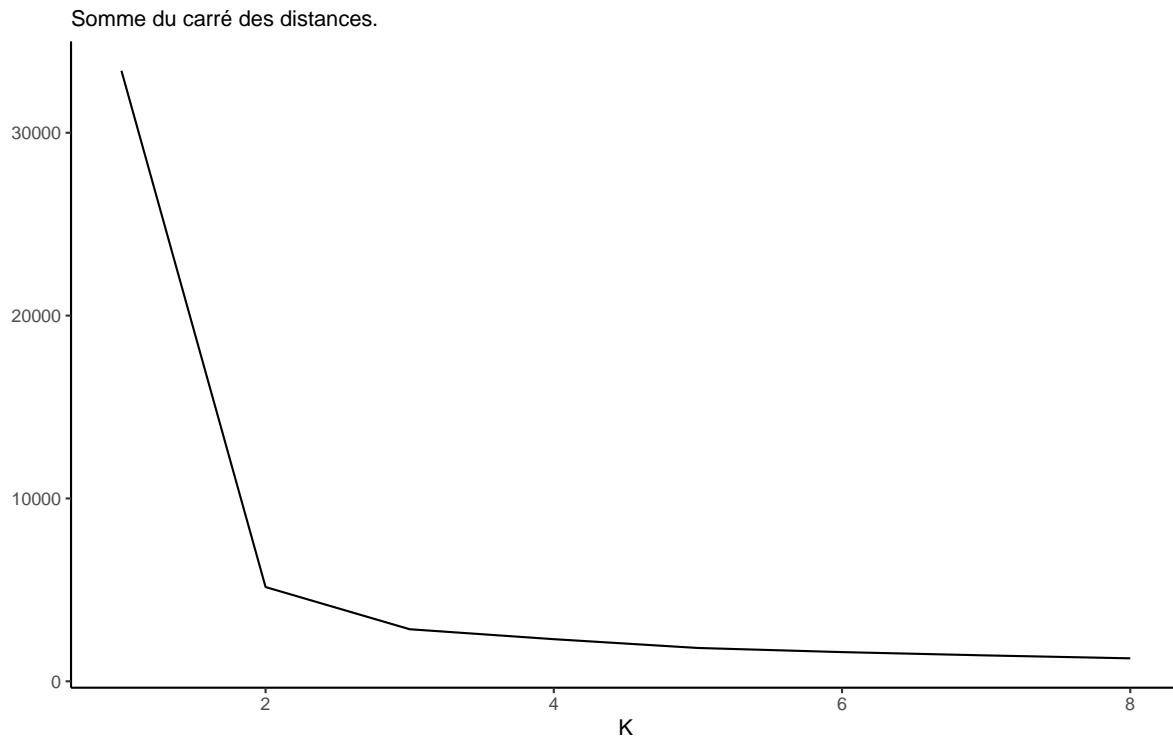


FIGURE 7.5 – Valeur de la fonction objective (somme du carré des distances) en fonction du nombre de regroupements  $K$ .

qui suit que la distance entre un point et un prototype est calculée avec la distance Euclidienne au carré.

### 7.5.1.1 Application en R

Dans **R**, la fonction `kmeans` dans le paquet de base `stat` permet de faire l'analyse de regroupement. Elle ne prend pas en charge les valeurs manquantes. La fonction a plusieurs arguments, dont les coordonnées initiales des prototypes (`center`; cet argument peut également être un entier qui dicte le nombre de groupes), le nombre maximum d'itération de l'algorithme EM (`iter.max`) et le nombre de fois qu'on redémarre l'algorithme avec des valeurs aléatoires (`nstart`).

On va estimer le modèle en faisant varier le nombre de regroupements avec pour chaque valeur de  $K$  10 ensembles de valeurs de départ aléatoires.

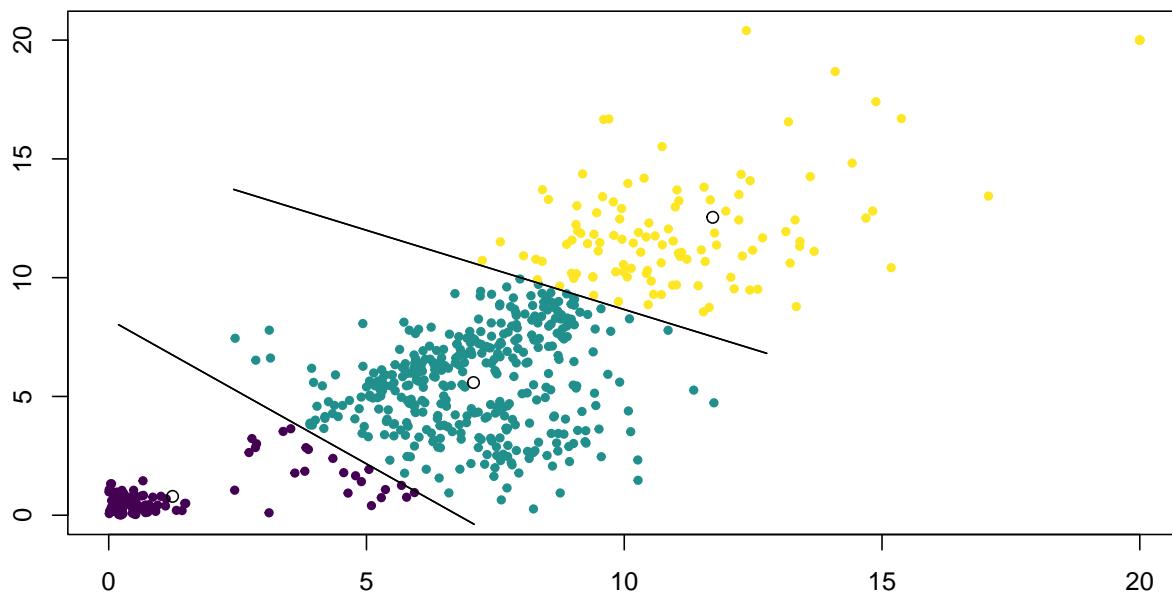


FIGURE 7.6 – Partitions de Voronoï pour les barycentres (cercles) obtenus dans la solution des  $K$ -moyennes. La ligne de démarcation qui sépare les groupes est linéaire.

```
set.seed(60602)
kmoy <- list()
ngmax <- 10L
for(i in seq_len(ngmax)){
  kmoy[[i]] <- kmeans(donsmult_std,
    centers = i,
    nstart = 10)
}
```

Il suffit ensuite de choisir le nombre de regroupements voulus. Rappelez-vous que le résultat des  $k$ -moyennes est aléatoire (parce que les valeurs initiales des prototypes le sont) et les étiquettes peuvent être permutées d'une fois à l'autre même si les regroupements sont les identiques.

À des fins d'illustration, regardons la solution avec  $K = 5$  regroupements. On pourrait également utiliser l'algorithme  $K$ -moyennes<sup>++</sup> avec `kcca` du paquet `flexclust`. Le code ci-dessous montre le résultat avec la distance de Manhattan ( $K$ -médianes)

```
set.seed(60602)
kmed5 <- flexclust::kcca(
  x = donsmult_std,
```

```

k = 5,
family = flexclust::kccaFamily("kmedians"),
control = list(initcent = "kmeanspp"))
# Vérifier répartition
kmed5@clusinfo
# Coordonnées des K-médianes (standardisées)
t(t(kmed5@centers)*dm_std + dm_moy)
# Étiquettes
kmed5@cluster

```

Il est toujours utile de regarder la taille des regroupements pour voir si on ne se trouve pas avec des regroupements fortement débalancés.

```

kmoy5 <- kmoy[[5]]
# Regarder la répartition
kmoy5$size
## [1] 993   64 3812 4496 4252

```

On peut étudier les coordonnées des prototypes (par exemple, avec `kmoy5$centers`), mais ici les données standardisées ne sont pas directement interprétables. On procède plutôt au calcul des statistiques descriptives des profils rapportées dans le Tableau 7.2.

```

donsmult |>
  group_by(groupe = kmoy5$cluster) |>
  summarise_all(mean)

```

Les regroupements obtenus sont interprétables :

- Groupe 1 : Petits donateurs, faible nombre de dons. N'ont pas donné depuis longtemps. Refus fréquents et délai entre dons élevés
- Groupe 2 : Grands donateurs fidèles : plus petit groupe. Ces personnes ont fait plusieurs dons, leur valeur maximale est élevée. N'ont pas donné récemment.
- Groupe 3 : Petits donateurs récidivistes. Dons plus élevés que la moyenne mais beaucoup de dons de faible valeur et peu fréquents.
- Groupe 4 : Petits nouveaux. Moins d'ancienneté, dons fréquents et refus fréquents relativement à l'ancienneté.
- Groupe 5 : Petits donateurs inactifs. Plutôt anciens, plusieurs refus.

On peut représenter graphiquement les regroupements obtenus sur les premières composantes principales avec les deux mesures de dissemblance.

Avec les  $K$ -médianes, les personnes qui ont fait des dons plus élevés sont fusionnés avec d'autres

TABLEAU 7.2 – Moyenne des variables explicatives par segment (segmentation avec K-moyennes et cinq regroupements).

	1	2	3	4	5
décompte	993	64	3812	4496	4252
mtdons	13.92	445.49	24.98	15.32	12.11
ndons	2.98	11.38	13.71	4.00	4.63
recence	64.56	67.14	27.34	29.00	172.06
anciennete	219.46	255.45	252.77	83.59	247.85
vdonsmax	22.39	1069.30	61.19	22.53	19.23
ddons	7.49	1.92	1.65	1.60	1.87
nrefusconsec	1.82	0.52	0.47	0.62	3.15
snrefus	3.17	0.88	1.05	4.23	2.71
mpromesse	15.32	620.32	45.13	17.70	7.67

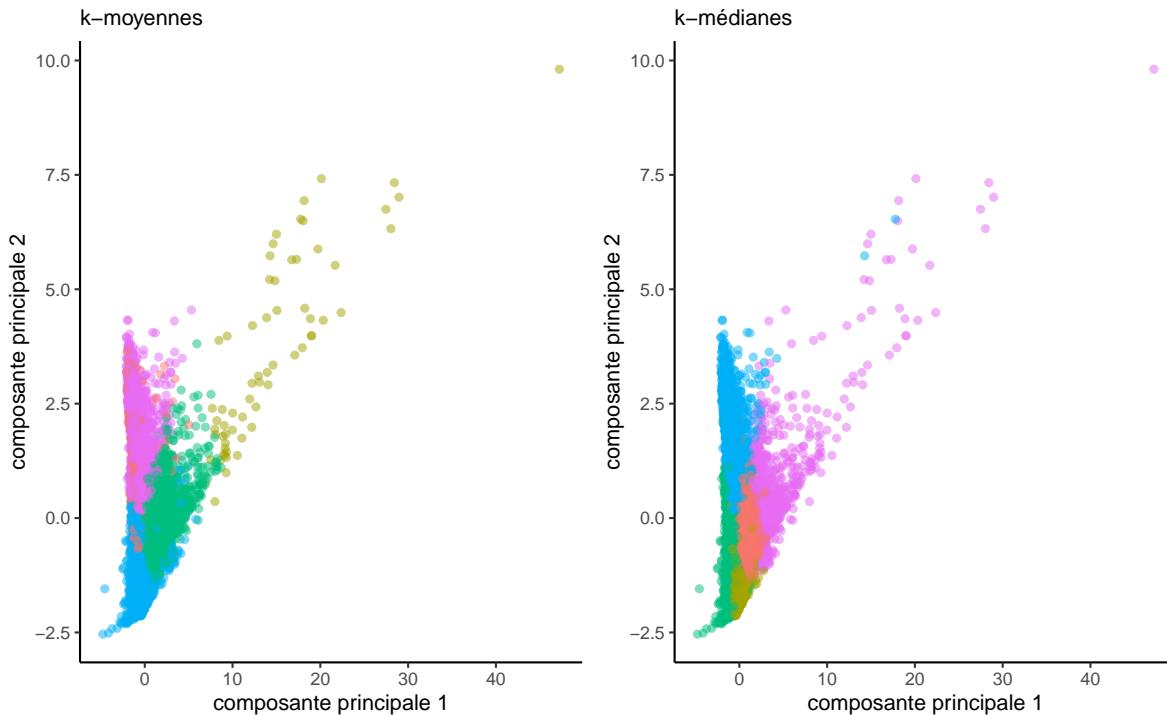


FIGURE 7.7 – Nuage de points des deux premières composantes principales des observations de dons multiples avec les étiquettes des regroupements obtenus selon la méthode des  $K$ -moyennes et  $K$ -médianes avec  $K = 5$  regroupements.

personnes qui ont fait des dons moins élevés et les groupes sont plus de taille comparable. Selon l'objectif des regroupements, cela peut être avantageux, mais cibler les donateurs les plus généreux semble plus logique dans le contexte.

On peut étudier l'impact de l'augmentation du nombre de groupes à l'aide de différents critères. Le premier est la somme des carrés des distances intra-groupes.

```
scd <- sapply(kmoy, function(x){x$tot.withinss})
# Graphiques
homogene <- homogeneite(scd)
bic_kmoy <- sapply(kmoy, BIC)
```

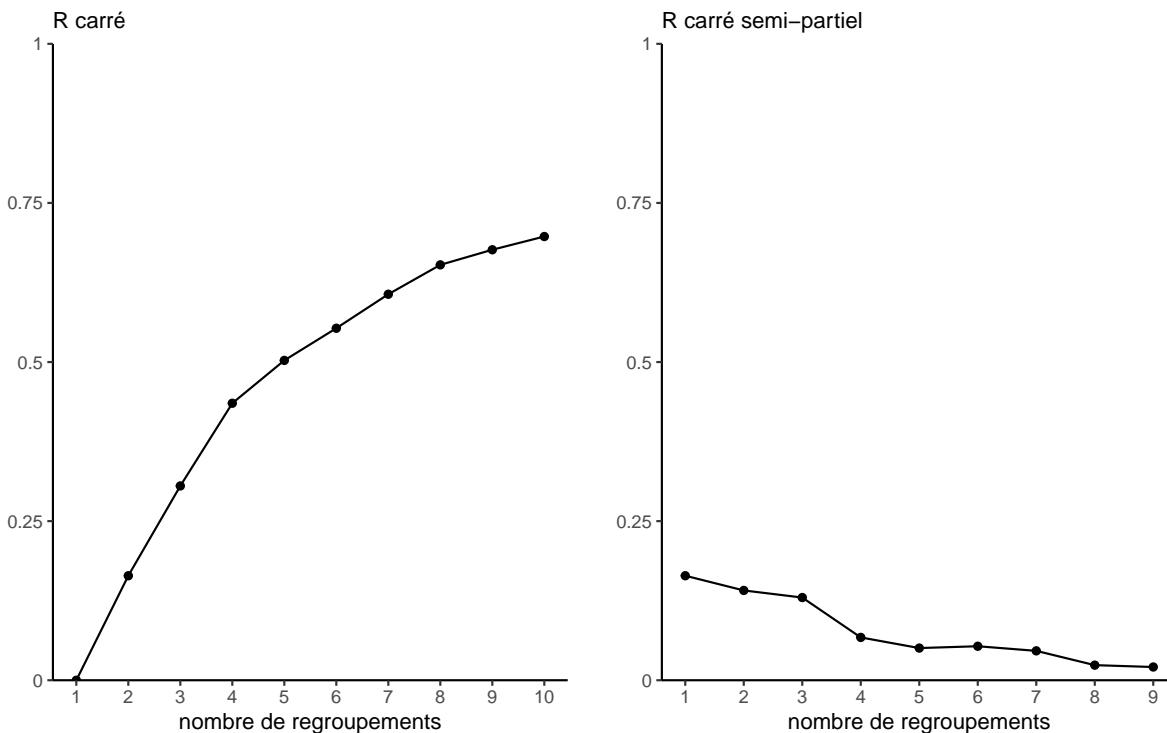


FIGURE 7.8 – Graphiques de l'homogénéité ( $R$  carré et  $R$  carré semi-partiel).

On peut aussi observer directement la diminution de la somme du carré des erreurs en incluant une pénalité. Ici, tous les critères pointent vers un nombre de regroupements plus élevé que 10, mais ce peut être trop.

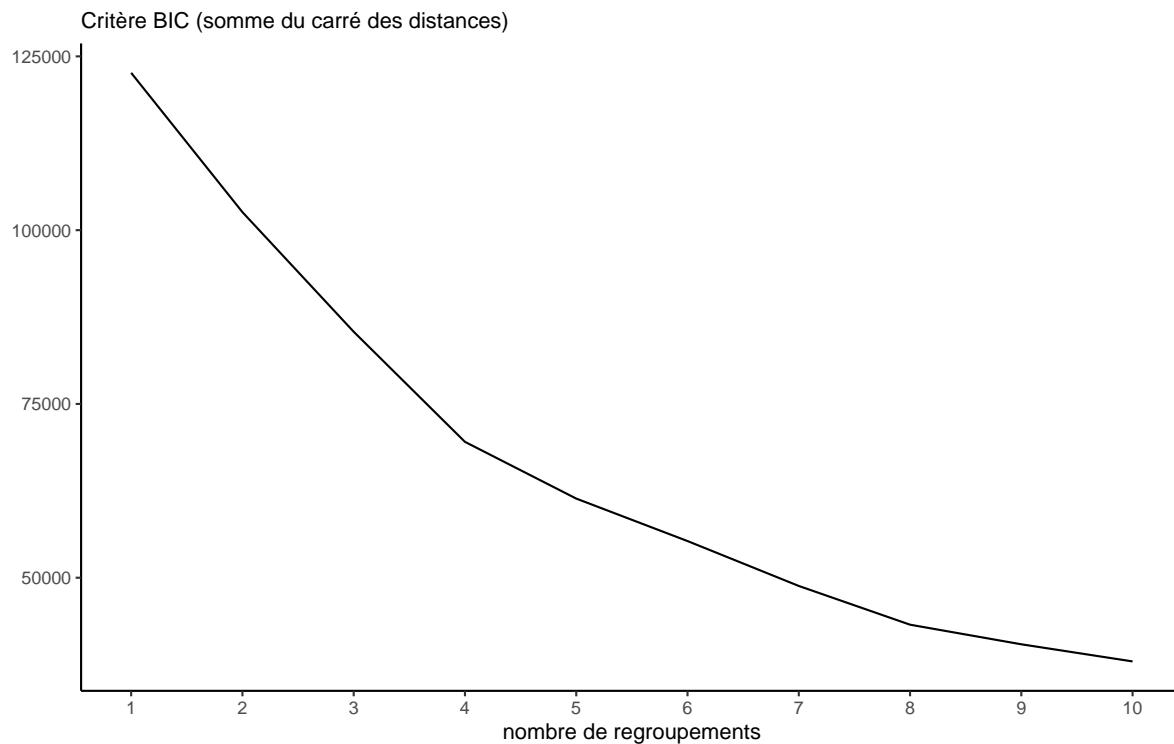


FIGURE 7.9 – Coefficient BIC pour les  $K$ -moyennes en fonction du nombre de regroupements.

### 7.5.2 $K$ -médoïdes

L'algorithme des  $K$ -moyennes spécifie que le barycentre des regroupements est le prototype. On pourrait également choisir pour ce dernier une des observations du groupe. Cette approche dite des médoïdes est plus coûteuse en calcul, mais permet d'avoir une observation réellement observée et est un peu moins sensible aux extrêmes et aux aberrances, bien que ce fait soit disputé.

L'algorithme de partition autour des médoïdes (PAM) procède comme suit :

1. Initialisation : sélectionner  $K$  des  $n$  observations comme médoïdes initiaux.
2. Assigner chaque observation au médoïde le plus près.
3. Calculer la dissimilarité totale entre chaque médoïde et les observations de son groupe.
4. Pour chaque médoïde ( $k = 1, \dots, K$ ) : considérer tous les  $n - K$  observations à tour de rôle et permutez le médoïde avec l'observation. Calculer la distance totale et sélectionner l'observation qui diminue le plus la distance totale.
5. Répéter les étapes 2 à 4 jusqu'à ce que les médoïdes ne changent plus.

Puisque qu'on considère chaque observation comme candidat à devenir un médoïde à chaque étape, le coût de calcul est prohibitif.

L'algorithme CLARA, décrit dans @Kaufman.Rousseeuw :1990, réduit le coût de calcul et de stockage en divisant l'échantillon en  $S$  sous-échantillons de taille approximativement égale (par défaut 5) et en utilisant l'algorithme PAM sur chacun. Une fois les médoïdes obtenus, le reste de toutes les observations de l'échantillon sont assignées au regroupement du médoïde le plus près. La qualité de la segmentation est pour chacune des  $S$  segmentations est calculée en obtenant la distance moyenne entre les médoïdes et les observations; on retourne la solution qui a la plus petite distance moyenne.

La qualité des regroupements est obtenue en utilisant la moyenne des distances entre les regroupements et leurs médoïdes. On peut également tracer un graphique des silhouettes : pour chaque observation, on calcule la moyenne des dissimilarités entre l'observation  $X_i$  et celles de chaque regroupement, disons  $a_i$ . On calcule de la même manière la distance moyenne entre  $X_i$  et chaque autre regroupement et on retient le minimum de ces distances,  $b_i$ .

La valeur de la silhouette est simplement  $s_i = (b_i - a_i) / \max\{a_i, b_i\}$ . Il est possible que la silhouette  $s_i$  soit négative : cela indique généralement des observations mal regroupées. De bons regroupements seront obtenus si la silhouette est élevée : on s'attend, si les groupes sont très éloignées les uns des autres, à avoir des profils plus uniformes et une silhouette moyenne plus élevée.

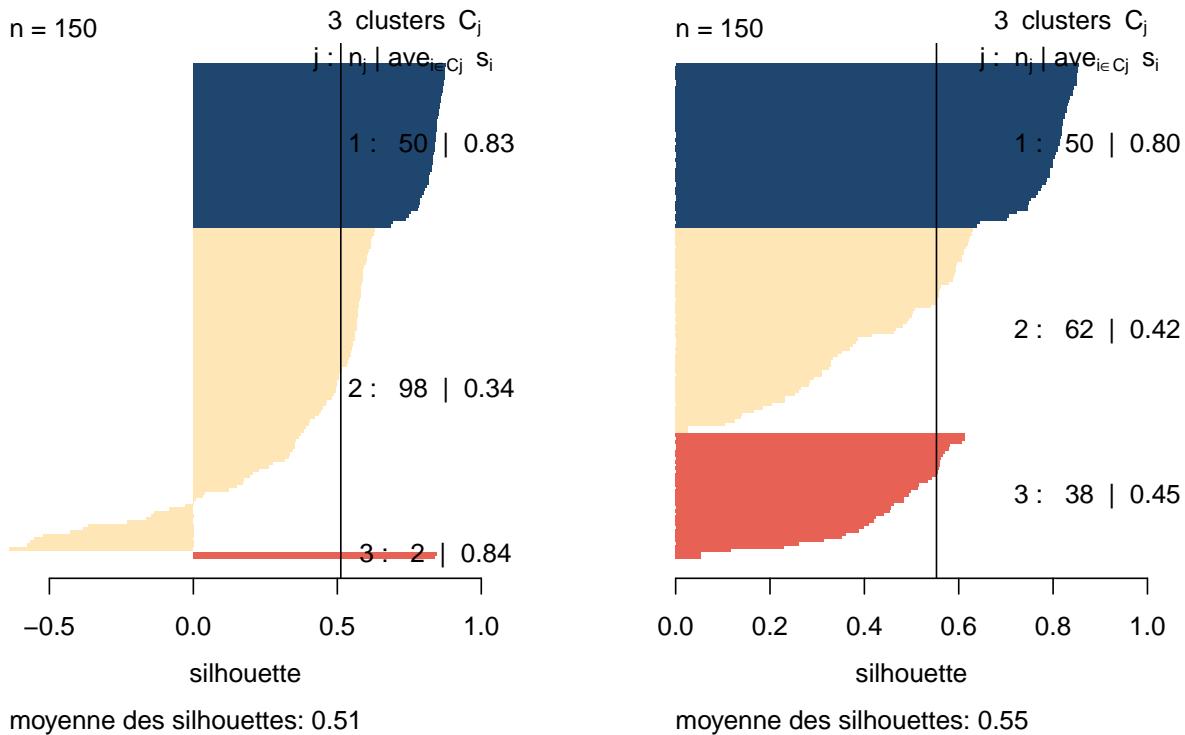


FIGURE 7.10 – Profil des silhouettes pour deux regroupements d'un jeu de données : la segmentation de droite est supérieure parce que les regroupements sont plus homogènes et mieux équilibrés.

On estime avec nos données de dons multiples les regroupements. Étant donné la taille conséquente de la base de données, il est préférable d'utiliser l'algorithme CLARA (*Clustering large applications*).

```
kmedoide <- list()
set.seed(60602)
for(k in seq_len(ngmax)){
  # Algorithme quadratique en sampszie
  kmedoide[[k]] <- cluster::clara(x = donsmult_std,
    k = k,
    sampszie = 500,
    metric = "euclidean", # distance,
    #cluster.only = TRUE, # ne conserver que étiquettes
    rngR = TRUE, # germe aléatoire depuis R
    pamLike = TRUE, # même algorithme que PAM
    samples = 10) #nombre de répétitions
}
```

Comme les  $K$ -moyennes, on fera plusieurs essais pour trouver de bonnes valeurs de départ. On peut tracer le profil des silhouettes (Figure 7.11)

```
plot(factoextra::fviz_silhouette(kmedoide[[4]]))
##   cluster size ave.sil.width
## 1       1   146      0.29
## 2       2   190      0.25
## 3       3    90      0.33
## 4       4    74      0.26
```

Puisque les prototypes (médoides) sont des observations, on peut simplement extraire leur identifiant. La sortie inclut plusieurs éléments dont la taille des regroupements, la valeur du critère PAM, etc.

```
medoides_orig <- donsmult[kmedoide[[4]]$i.med,]
medoides_orig
# Taille des regroupements
kmedoide[[4]]$clusinfo
```

Voici quelques avantages et inconvénients des  $K$ -médoides.

- les prototypes sont des observations de l'échantillon.
- la fonction objective est moins impactée par les extrêmes.
- le coût de calcul est prohibitif avec des mégadonnées.

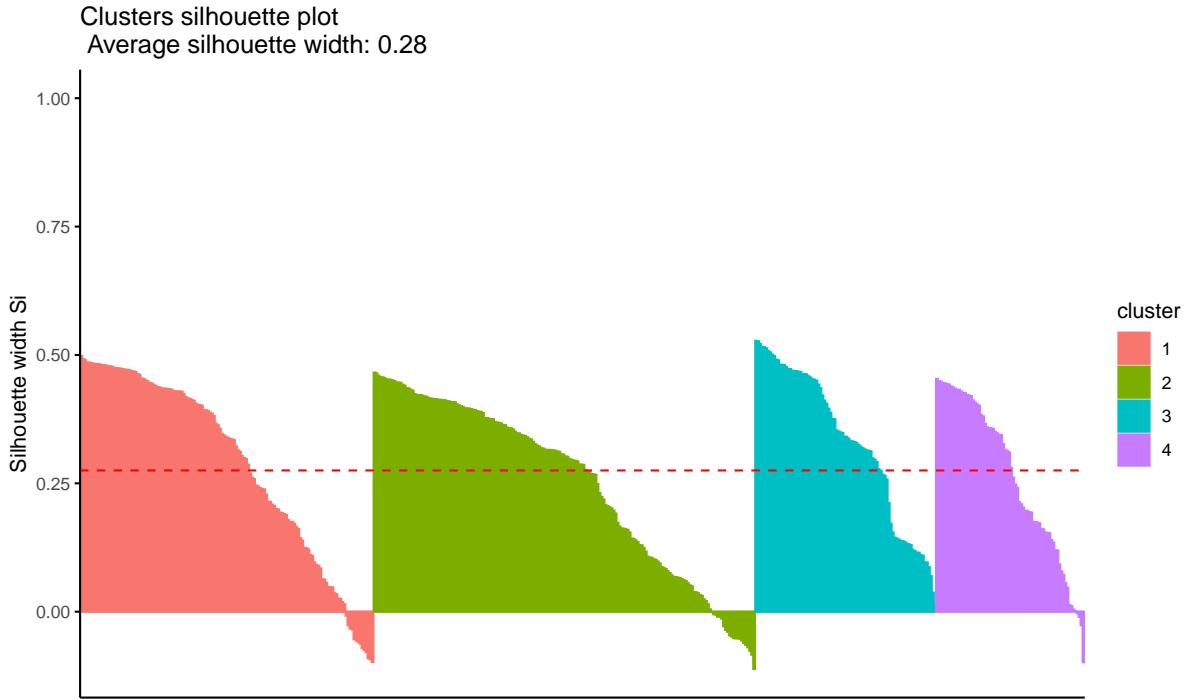


FIGURE 7.11 – Silhouettes pour les données de dons multiples avec l'algorithme CLARA pour  $K = 4$  regroupements.

### 7.5.3 Mélange de modèles

L'algorithme des  $K$ -moyennes fait une allocation rigide : chaque observation est assignée à un seul regroupement, ignorant de ce fait l'incertitude rattachée à l'étiquetage des observations. Les frontières de la région, obtenue en calculant l'intersection des courbes sphériques de regroupement, sont linéaires.

Peut-être plus problématique, la distance Euclidienne non pondérée impose des regroupements convexes et sphériques de taille semblable : la qualité des regroupements des  $K$  moyennes est donc mauvaise si les regroupements ne sont pas sphériques ou globulaires, ou sont de concentrations inégales.

Une approche plus générale considère que  $X_1, \dots, X_p$  sont tirées d'un mélange à  $K$  composantes de lois spécifiées. Généralement, on choisit une loi normale multidimensionnelle pour le  $k$ e groupe  $G$ ,

$$X | G = k \sim \text{No}_p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

On suppose qu'on a  $K$  groupes, chacun caractérisé par une densité de dimension  $p$ , soit  $f_k(\mathbf{X}_i; \boldsymbol{\theta}_k)$  si  $\mathbf{X}_i$  provient du groupe  $k$  pour  $k = 1, \dots, K$ .

On réécrit la vraisemblance en fonction de  $\pi_k$ , la probabilité qu'une observation  $\mathbf{X}_i$  tombe dans le groupe  $k$ ,

$$L_i(\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K; \pi_1, \dots, \pi_K, \mathbf{X}_i) = \sum_{k=1}^K \pi_k f_k(\mathbf{X}_i, \boldsymbol{\theta}_k).$$

Si on savait de quelle composante l'observation originait, on pourrait simplement obtenir les estimation du maximum de vraisemblance pour les paramètres de moyenne et de variance. Inversement, si on avait les valeurs des paramètres, on pourrait déterminer de quel composante l'observation est la plus susceptible de parvenir à l'aide des poids. Le modèle est estimé à l'aide de l'algorithme d'espérance-maximisation, qui itère entre l'estimation des probabilités, et celles des autres composantes. Le paramètres retournés correspondent à un maximum local, et on peut également obtenir un estimé de la variabilité de ces paramètres. Ainsi, le mélange de modèle nous donne accès à la fois à l'incertitude des paramètres et à la probabilité  $\pi_k$  qu'une observation appartiennent au groupe  $G_k$ .

La loi multinormale est caractérisée par une moyenne (qui peut servir de prototype) et par une matrice de covariance  $\Sigma_k$ . Si on paramétrise cette dernière, on peut obtenir plus de flexibilité selon que les variances soient différentes d'une variable à l'autre, ou que les variables soient corrélées. On peut également spécifier que certains éléments (structure de corrélation, variances) de  $\Sigma_k$  sont communes à tous les regroupements. En laissant les paramètres varier, on peut capturer l'effet de regroupements de tailles et de densité différente au prix de plus de paramètres et d'un plus petit nombre d'observations pour estimer chacun d'entre eux.

Si  $p$  est élevé, la structure de covariance non structurée possède trop de paramètres pour être utile. On limitera ce nombre en choisissant plutôt une paramétrisation plus parsimonieuse qui impose des contraintes sur la forme des ellipsoïdes, propres ou communes à tous les groupes.

La matrice de covariance dans `mclust` est paramétrisée en fonction de  $\lambda$ , qui contrôle le volume, une matrice diagonale  $\mathbf{A}$  qui contrôle les variances de chaque observation et  $\mathbf{D}$  une matrice orthogonale qui permet de créer de la corrélation entre observations. Un index  $k$  spécifie que cette composante varie d'un regroupement à l'autre.

Les trois lettres de l'identifiant pour volume/forme/orientation déterminent si cette composante est égale (E), si elle varie d'un regroupement à l'autre (V) ou si elle est indéterminée (I). Par exemple, EII spécifie une matrice de covariance où chaque composante a variance  $\lambda$  et où les composantes sont indépendantes. Voir `mclust.options("emModelNames")` et la documentation dans le Tableau 3 de `@mclust5`.

Voici quelques avantages et inconvénients des mélanges de modèles Gaussiens

- cette approche est plus flexible que les  $K$ -moyennes.
- l'ajout d'une composante uniforme permet de gérer les aberrances (supporté par `mclust`).
- l'algorithme EM garantie la convergence à un minimum local (comme pour les  $K$ -moyennes)

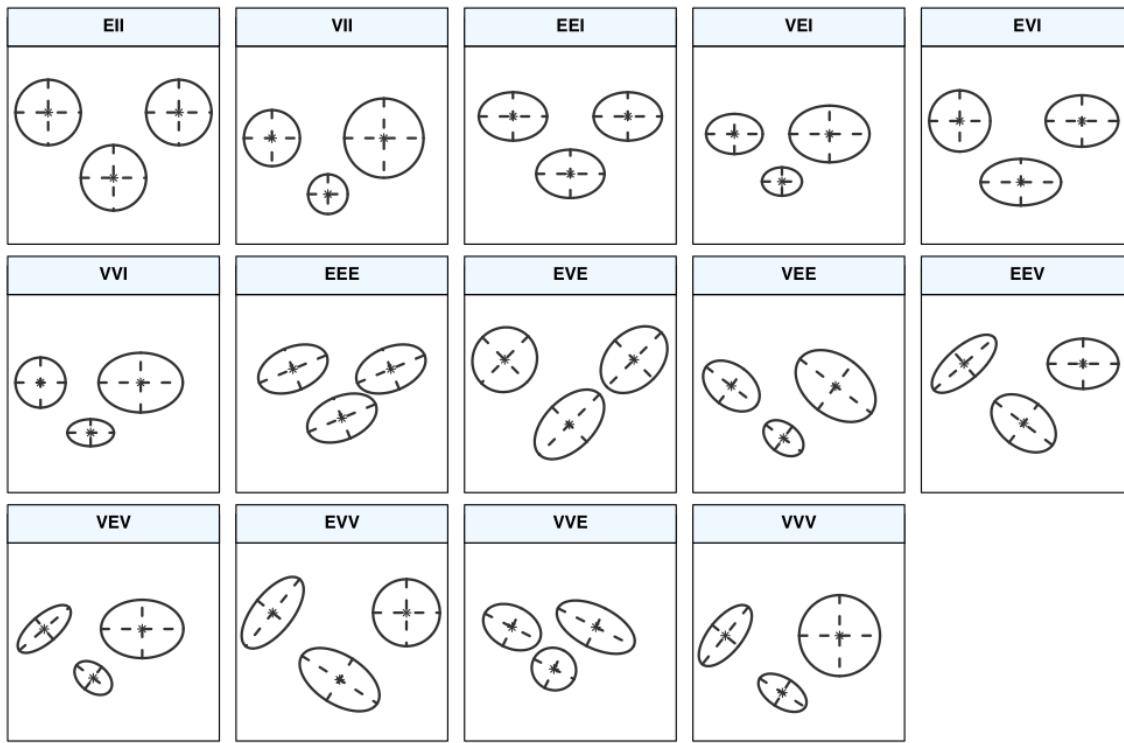


FIGURE 7.12 – Forme des ellipsoïdes pour le mélange de modèle selon la forme de la structure de covariance. Image extraite de @mclust5 (Figure 2) partagée sous licence CC BY 4.0.

- on obtient une assignation probabiliste plutôt que rigide, également pour la classification
- le coût de calcul est plus élevé que les  $K$ -moyennes
- le nombre de paramètre des matrices de covariance augmente rapidement avec la dimension  $p$

### 7.5.3.1 Hyperparamètres

Pour le mélange de modèle, on doit fixer a priori le nombre de groupes  $K$ , la forme des ellipsoïdes et les valeurs pour l'initialisation. Les mêmes considérations pratiques qu'avec les  $K$ -moyennes s'appliquent, bien qu'ici l'utilisation des critères d'information permette plus légitimement de choisir le nombre de regroupements.

La forme des ellipsoïdes est un compromis entre simplicité (d'estimation) et nombre de paramètres : un modèle plus flexible sera plus difficile à estimer et nécessitera plus de temps de calcul et un plus grand nombre d'échantillon. En petite dimension, il peut être utile d'effectuer une visualisa-

tion préalable pour déterminer quel type de modèle serait suffisant. Règle générale, il faut aussi considérer le nombre de paramètres à estimer (qui dépend de  $p$ ) et le nombre d'observations par regroupement. Comme tous les modèles sont estimés avec la méthode du maximum de vraisemblance, on peut toujours ajuster tous les types de structures de covariance pour un nombre de regroupements  $K$  donné et retourner les critères d'information (BIC) pour sélectionner le meilleur mélange de modèles. La fonction `mclustBIC` du paquet `mclust` permet de calculer ces modèles et la méthode `summary` retourne les trois meilleurs modèles selon le critère d'information.

### 7.5.3.2 Paquet `mclust`

La stratégie de base du paquet `mclust` [@mclust5] est d'ajuster des mélanges de modèles gaussiens avec plusieurs structures de covariance en faisant varier le nombre de regroupements. Le modèle sélectionné parmi tous les candidats est celui qui a la plus petite valeur du critère BIC : ce dernier dépend de la qualité de l'ajustement et la pénalité prend en compte le nombre de paramètres de covariance, en plus des moyennes. Il est possible d'ajouter une composante pour le bruit, de manière à éviter que les valeurs aberrantes impactent négativement la segmentation.

Une fois le modèle obtenu, plusieurs fonctionnalités sont disponibles pour représenter graphiquement les ellipses des modèles pour chaque paire de variable, les nuages de points des paires de variables avec différents symboles et couleurs pour les regroupements, etc.

```
## Mélanges de modèles gaussiens
set.seed(60602)
library(mclust)
## Package 'mclust' version 6.0.0
## Type 'citation("mclust")' for citing this R package in publications.
mmg <- Mclust(data = donsmult_std,
  G = 1:10,
  # Ajouter composante uniforme
  # pour bruit (aberrances)
  initialization = list(noise = TRUE))
# Résumé de la segmentation
summary(mmg)
```

On peut obtenir les étiquettes (avec 0 pour le bruit) avec `mmg$classification`. Le graphique du critère d'information Bayésien (BIC) montre le négatif : on cherche donc la structure de covariance et le nombre qui maximise  $-\text{BIC}$ .

```
plot(mmg, what = "BIC")
```

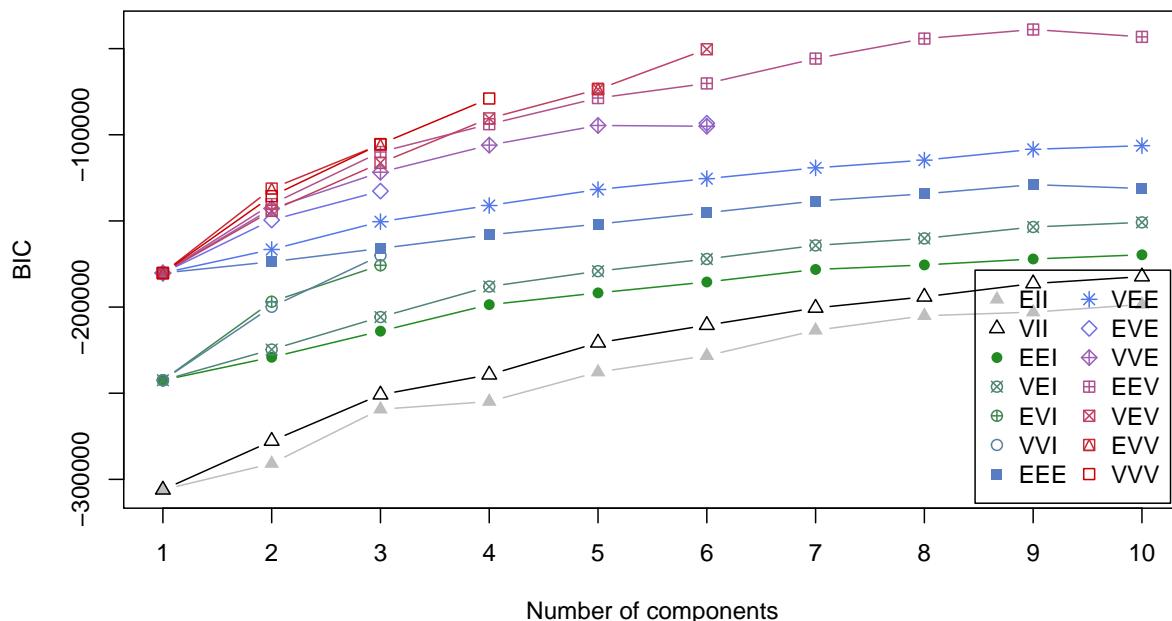


FIGURE 7.13 – Valeur du négatif du critère d’information Bayésien pour les mélanges de modèles gaussiens selon le nombre de regroupements et la structure de covariance.

Avec notre grande base de données, le modèle identifie neuf regroupements et un volume variable. On peut utiliser des techniques de réduction de la dimension pour obtenir une représentation graphique.

```
# Matrice des nuage de points (paires de variables)
# plot(mmg, what = "classification")
# Réduction de la dimension
reduc_dim_mmg <- mclust:::MclustDR(mmg)
par(mfrow = c(1,2)) # graphiques côte-à-côte
plot(reduc_dim_mmg, what = "contour")
## Error in parameters$variance$sigma[, , k]: indice hors limites
plot(reduc_dim_mmg, what = "scatterplot")
```

#### 7.5.4 Regroupements hiérarchiques

Historiquement très utilisés dans les années 70, les méthodes de regroupement hiérarchique offrent une méthode déterministe de regroupement à partir d’une matrice de dissimilarité.

L’algorithme pour la procédure agglomérative procède comme suit :

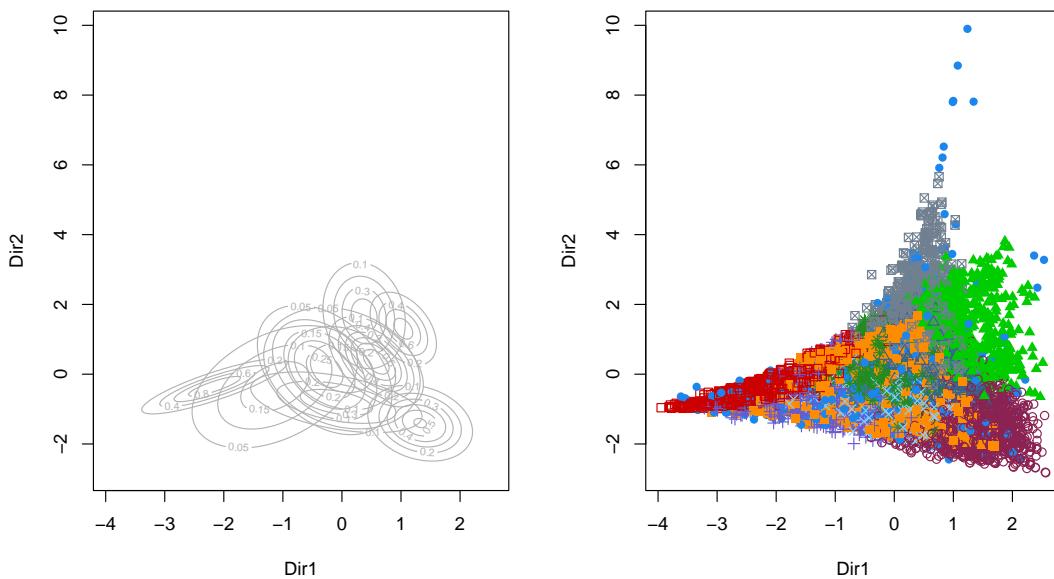


FIGURE 7.14 – Projection des observations, colorées par regroupement (gauche) et structure des regroupements avec ellipsoïdes de confiance (droite).

1. Initialisation : chaque observation forme son propre groupe.
2. les deux groupes les plus rapprochés sont fusionnés; la distance entre le nouveau groupe et les autres regroupements est recalculée.
3. on répète l'étape 2 jusqu'à obtenir un seul regroupement.

La procédure divisive procède de la même façon, mais en partant d'un seul ensemble et en subdivisant ce dernier jusqu'à ce qu'il y ait autant d'observations que de groupes. Cette dernière est préférable si on veut isoler de grands regroupements, mais est rarement employée.

Il y a plusieurs façons de calculer la distance entre deux groupes d'observations. Selon notre définition, nous obtiendrons des regroupements différents. Les méthodes les plus populaires incluent

- liaison simple (plus proches voisins)
- liaison complète (voisins les plus éloignés)
- liaison moyenne : utilise la moyenne des distances entre toutes les paires de sujets (un pour chaque groupe) provenant des deux groupes.
- méthode de Ward : calcul de l'homogénéité globale

```
## [1] 33
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "color" n'est pas un
## paramètre graphique
```

```
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "color" n'est pas un
## paramètre graphique
```

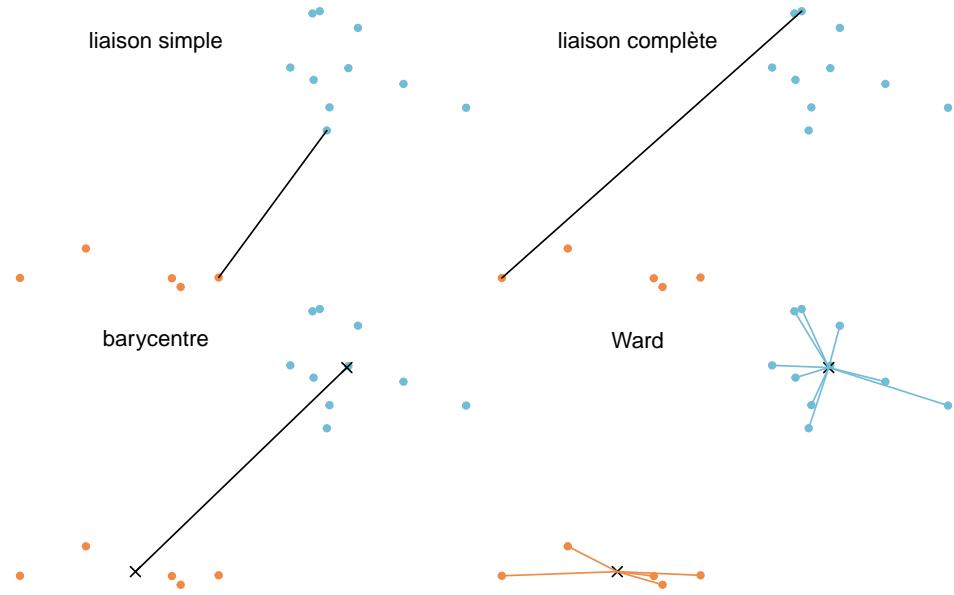


FIGURE 7.15 – Distances entre regroupements selon la liaison (simple, complète, barycentre, homogenéité de Ward).

La méthode de Ward n'est pas définie en terme de distance entre représentants de groupes, mais plutôt en terme de mesure d'homogénéité au sein des groupes. Supposons qu'à une étape du processus hiérarchique, nous avons  $M$  groupes et que nous voulons passer à  $M - 1$ . Pour chaque groupe  $k$ , nous pouvons calculer la somme des carrés des distances par rapport à la moyenne du groupe, disons  $SCD_k$  : plus cette distance est petite, plus le groupe est compact et homogène. On calcule ensuite l'homogénéité globale en faisant la somme de l'homogénéité de tous les groupes, soit  $H^{(M)} = SCD_1 + \dots + SCD_M$ . La méthode de Ward va regrouper les deux groupes qui feront augmenter le moins possible l'homogénéité.

En général, les algorithmes de regroupement hiérarchiques stockent une matrice de dissemblance  $n \times n$ , et donc un coût de stockage quadratique et un coût de calcul  $\Omega(n^2)$  avec  $O(n^3)$ . Il faut réaliser que ce coût de calcul est **prohibitif** en haute dimension. Certains algorithmes efficaces pour la méthode de liaison simple permettent un temps de calcul quadratique sans calcul de toutes les distances, à coût  $O(n)$ . Si la méthode de liaison simple est la moins coûteuse du lot, elle n'est pas aussi populaire car elle fonctionne bien si l'écart entre deux regroupements est suffisamment grand. S'il y a du bruit entre deux regroupements, la qualité des regroupements en sera affectée. La méthode de liaison complète est moins sensible au bruit et aux faibles écarts entre regroupements,

mais a tendance à casser les regroupements globulaires. Puisque le critère d'homogénéité de Ward ressemble à celui des  $K$ -moyennes, la sortie aura tendance à bien regrouper les amas globulaires.

Généralement, le résultat de la procédure agglomérative avec la méthode de liaison simple inclura quelques valeurs isolées et un seul grand regroupement. Une alternative récente [@Gagolewski :2016], appelée Genie, modifie la fonction objective de la méthode de liaison simple en retenant son efficacité de calcul. Plutôt que de simplement trouver la paire de regroupements à distance minimale, cette fusion n'est appliquée que si une mesure d'inéquité est inférieur à un seuil spécifié par l'utilisateur. Si les regroupements sont fortement inéquitables, la fusion survient entre les regroupements dont un de la taille minimale courante. L'implémentation R [@Gagolewski :2021] dans le paquet `genieclust` est nettement plus rapide que les autres alternatives et ne nécessite pas de calculer la matrice de dissimilarité.

On peut comparer les performances des regroupements hiérarchiques selon la méthode de regroupement. La page web de scikit-learn developers montre la performance sur des exemples jouets très artificiels, qui montre que selon la structure des données, l'impact de la fonction de liaison. Ici, aucun approche hiérarchique ne performe mieux que les autres dans tous les exemples.

#### 7.5.4.1 Sélection des hyperparamètres

Outre le choix de la fonction de liaison qui déterminera la distance entre les regroupements à chaque étape, on devra choisir le nombre de regroupements.

On peut représenter le modèle à l'aide d'un **dendrogramme**, un arbre dont les feuilles indiquent les regroupements à chaque étape jusqu'à la racine à la dernière étape. La distance entre chaque embranchement est déterminée par notre critère : cela nous permet de sélectionner un nombre de regroupements  $K$  après inspection du dendrogramme et d'extraire la solution en élaguer l'arbre à cette profondeur.

La hauteur du dendrogramme donne la valeur du critère associé à la mesure de regroupement : on peut sélectionner le nombre de regroupements  $K$  en sélectionnant une étape où la qualité de l'ajustement diminue drastiquement. Pour le critère de Ward qui utilise l'homogénéité, on peut créer le pourcentage de variance expliquée,  $R^2$  en calculant  $R_{(M)}^2 = 1 - H_{(M)}/H_{(1)}$ , où  $H_{(1)}$  est simplement la somme du carré des distances par rapport à la moyenne lorsque toutes les observations sont dans un même groupe. Le R-carré semi-partiel, qui mesure la perte d'homogénéité d'une étape à l'autre, renormalisée par  $H_{(1)}$ , permet également de mesurer la perte d'homogénéité (relative) en combinant ces deux groupes. On peut faire un graphique de ces deux critères en fonction du nombre de regroupements et chercher un point d'inflection (un coude) à partir duquel la perte d'homogénéité est moindre ou encore le  $R^2$  augmente plus lentement.

La fonction `stat::hclust` permet de faire des regroupements agglomératifs (`agnes`), mais `fastcluster` propose une version avec une empreinte mémoire inférieure. Le paquet `cluster` offre de son côté l'algorithme divisif (`diana`).

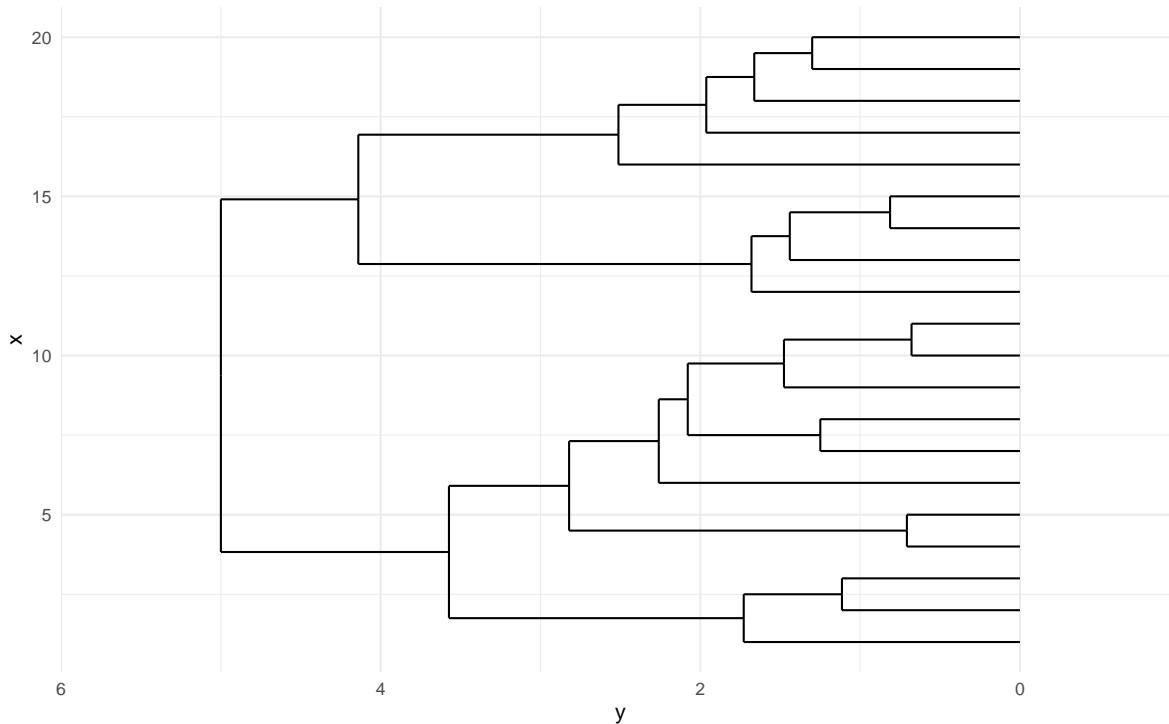


FIGURE 7.16 – Dendrogramme pour l'exemple de regroupement hiérarchique avec la méthode de Ward et 20 observations.

Voici quelques particularités des méthodes de regroupement hiérarchique.

- la solution du regroupement hiérarchique est toujours la même (déterministe)
- l'assignation d'une observation à un regroupement est finale
- les aberrances ne sont pas traitées et sont souvent assignées dans des regroupements à part
- les méthodes d'arborescence sont faciles à expliquer
- le nombre de groupes n'a pas à être spécifié a priori (une seule estimation)
- le coût de calcul est prohibitif, avec une complexité quadratique de  $O(n^2)$  pour la méthode de liaison simple et autrement  $O(n^3)$  pour la plupart des autres fonctions de liaison.

### 7.5.5 Méthodes basées sur la densité

L'algorithme DBSCAN (*density-based spatial clustering of applications with noise*) est une méthode de partitionnement basée sur la densité des points. L'idée de base de l'algorithme est de tracer une boule de rayon  $\epsilon$  autour de chaque observation et de voir si elle inclut d'autres observations. L'algorithme contient deux hyperparamètres : le rayon  $\epsilon$  et  $M$ , le nombre minimal de points pour

former un regroupement. L'algorithme classe les observations en trois catégories : aberrance, point central et point frontière.

- Un point central est une observation qui possède  $M - 1$  voisins à distance  $\epsilon$ .
- Un point frontière est un point qui est distant de moins de  $\epsilon$  d'un point central, sans en être un.
- Un point isolé est une observation qui n'est pas rattachée à aucun regroupement.

L'algorithme répète les étapes suivantes jusqu'à ce que chaque observation ait été visitée.

1. Choisir un point aléatoirement parmi ceux qui n'ont pas été visités.
2. Si le point n'est pas étiqueté, calculer le nombre de points voisins qui se trouvent dans un rayon  $\epsilon$  : s'il y a moins de  $M$  observations, provisoirement étiqueter l'observation comme point isolé, sinon comme point central.
3. Si l'observation est un point central avec  $M - 1$  voisins ou plus, créer un regroupement.
4. Étiqueter chaque point à distance  $\epsilon$  créé et l'ajouter au regroupement, ainsi que tout point à distance  $\epsilon$  de ces voisins.

[Ce site web] offre une visualisation interactive des différentes étapes de l'algorithme et de comparer la performance de DBSCAN selon le type de regroupements.

Puisque chaque point est visité à tour de rôle et comparé aux autres pour trouver les plus proches voisins, la complexité brute est  $O(n^2)$  mais une implémentation efficace permet de réduire ce coût à  $O(n \ln n)$  avec un coût pour l'allocation de la mémoire linéaire de  $O(n)$ .

Voici quelques caractéristiques de DBSCAN :

- le traitement des aberrances est automatique et l'algorithme est robuste.
- le nombre de regroupements n'a pas à être spécifié a priori.
- la forme des regroupements est arbitraire, peut être non convexe et de taille différente.
- la complexité de l'algorithme est d'au mieux  $O(n \ln n)$ .
- les hyperparamètres ont une interprétation physique, mais leur choix n'est pas aisés.
- DBSCAN ne permet pas de traiter le cas où la densité des regroupements change et risque de fusionner des regroupements s'il y a une série d'observations qui permet de relier deux regroupements.
- comme la plupart des algorithmes, le voisinage des points devient épars quand  $p$  augmente en raison du fléau de la dimension.

#### 7.5.5.1 Choix des hyperparamètres

Les deux paramètres,  $M$  et  $\epsilon$ , sont positivement corrélés : si on augmente le nombre minimal de point  $M$  par regroupement, il faudra également augmenter le rayon  $\epsilon$  pour éviter d'avoir un nombre trop élevé de points isolés étiquetés comme points isolés ou comme aberrances.

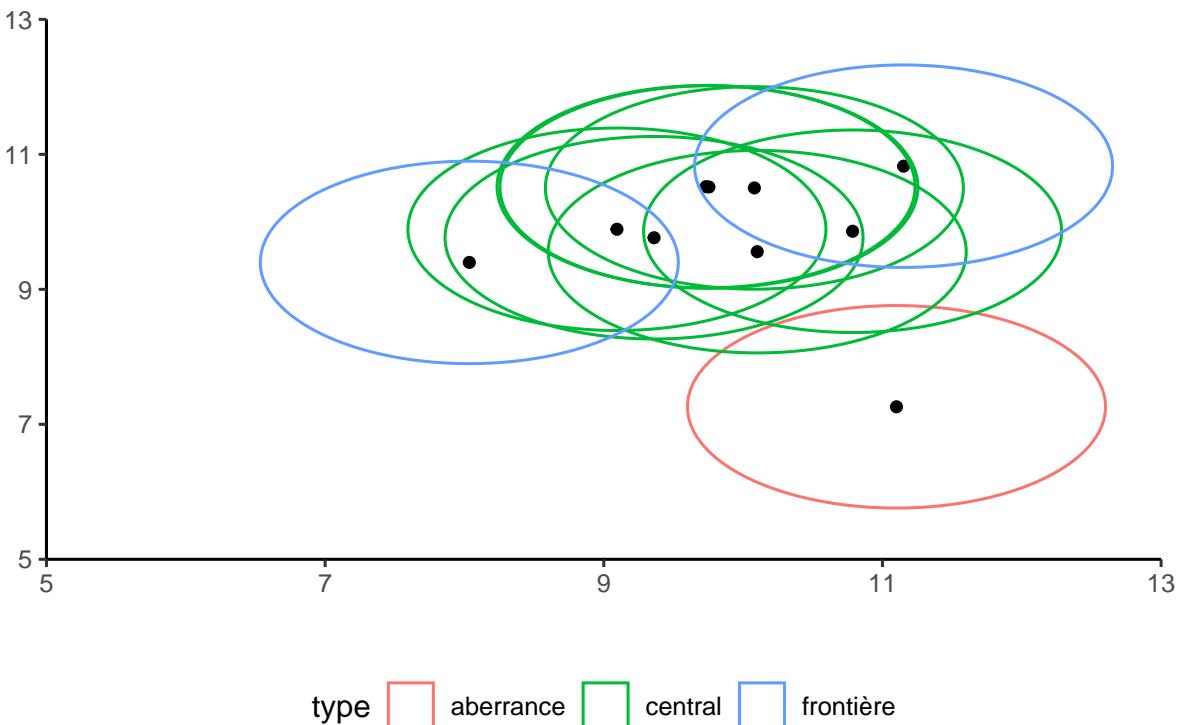


FIGURE 7.17 – Illustration de la classification des points avec DBSCAN : toutes les observations sont assignées à un regroupement, moins une aberrance.

Pour spécifier le nombre minimal d'observations voisines  $M$  pour créer un point central, il faut aussi considérer la dimension  $p$  des variables explicatives : la recommandation est de requérir au moins  $p + 1$  points dans le voisinage. Le choix du rayon peut être plus difficile à déterminer : . Une option est de fixer le nombre de plus proches voisins  $M$  et de considérer la distance entre chaque observation et ses plus proches voisins : au sein d'un regroupement, on s'attend à ce que cette distance soit petite. Cela permettra également de déterminer un seuil acceptable pour  $\epsilon$  pour éviter que trop d'observations soient isolées.

La fonction `kNNdistplot` du paquet `dbscan` permet de tracer un graphique de la distance moyenne des  $k$  plus proches voisins pour chaque observation : en prenant  $k = M - 1$ , on peut calculer la distance entre le  $k$  plus proche voisin de chaque observation et ordonner ces distances. La recommandation est de choisir  $\epsilon$  en prenant une distance où la plupart des observations ne sont pas voisines (critère du coude).

Une variante de l'algorithme DBSCAN, intitulée OPTICS, est plus coûteuse mais permet de gérer le cas de regroupements de densités variables en évitant la spécification de  $\epsilon$ .

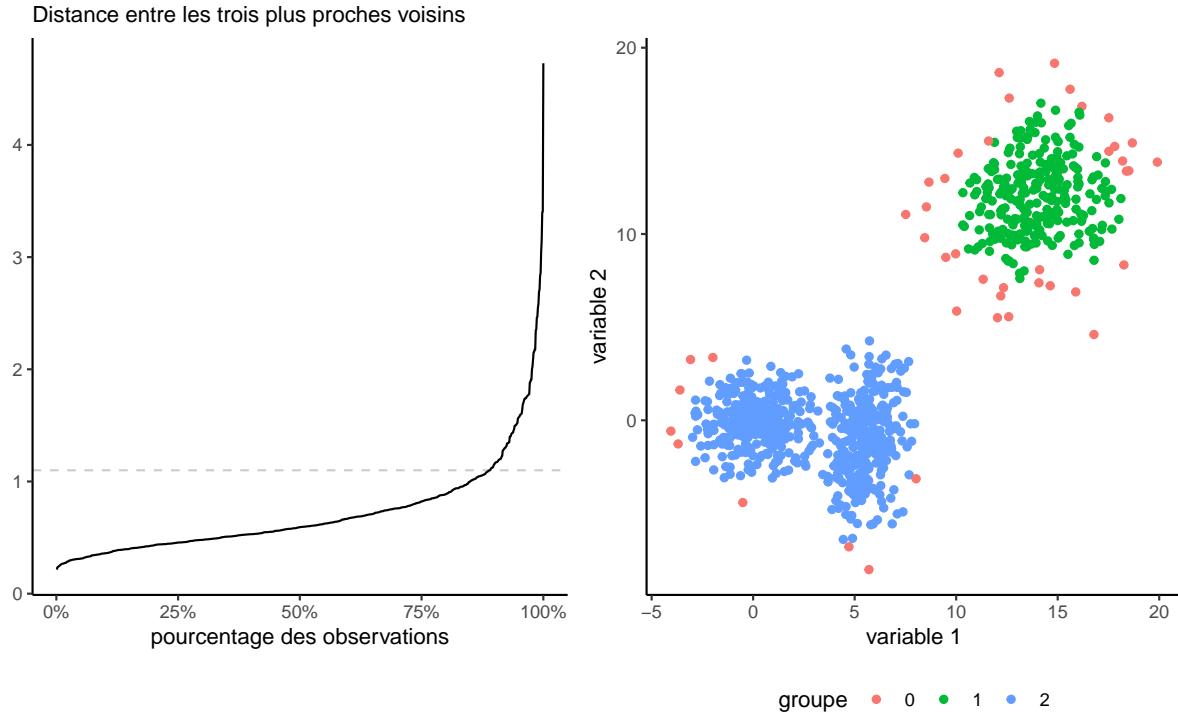


FIGURE 7.18 – Graphique des distances entre chaque observation et son troisième plus proche voisin (gauche), en fonction du pourcentage d’observations à moins de cette distance et regroupements obtenus avec DBSCAN avec  $M = 10$  et  $\epsilon = 1.1$  (droite).

## 7.6 Conclusion

Le résultat d’une analyse de regroupements est une étiquette pour chaque observation. Parfois, la méthode d’analyse de regroupement retourne également un prototype (le barycentre, une observation du groupe ou la médiane coordonnée par coordonnée) qui permet d’interpréter les regroupements.

L’analyse de regroupement est une méthode d’apprentissage non-supervisé : l’objectif est de déduire la structure présente dans un ensemble de points sans étiquette préalable (contrairement à la classification). Ainsi, une fois qu’on a obtenu les étiquettes, on peut comparer les regroupements entre eux avant d’effectuer le profilage. Est-ce que les regroupements sont homogènes et que les observations sont près de leur représentant de groupe ? On pourrait calculer les silhouettes et voir si les groupes sont bien équilibrés, etc. S’il n’existe pas de solution, il existe des segmentations de moins bonne qualité (parce que difficilement interprétables, avec des regroupements qui contiennent une poignée d’observations). Si la segmentation n’est pas satisfaisante, on retourne à la planche à dessin et on modifie les variables, la méthode ou la calibration des hyperparamètres jusqu’à ce

qu'on soit satisfaits du résultat.

#### **i** En résumé

- L'objectif d'une analyse de regroupement est de mettre en commun des observations de telle sorte que les observations d'un même groupe soient le plus semblables possible, et que les groupes soient le plus différent possible les uns des autres.
- Chaque observation se voit assigner une étiquette de groupe.
- On procède ensuite à une analyse descriptive, segment par segment, à l'aide de prototypes
- L'analyse de regroupement est une méthode d'apprentissage non-supervisée : il n'y a pas de véritable séparation.
- La segmentation n'est utile que si elle a une valeurs ajoutée.
- Plusieurs choix de l'analyste (mesure de dissemblance, algorithme ou méthode de regroupement, choix des hyperparamètres) peuvent donner une segmentation différente. L'analyste a une grande marge de manœuvre.
- Chaque algorithme de segmentation a des avantages et inconvénients.
- L'algorithme des  $K$ -moyennes est le plus employé et son faible coût permet son utilisation avec des mégadonnées.
- Aucun algorithme ne performe uniformément mieux, mais certains sont plus faciles à employer que d'autres.
  - avec des mégadonnées, la complexité est un facteur important à considérer pour le choix de la méthode.
  - la plupart du temps, le choix des hyperparamètres nécessite un peu d'essai-erreur.
  - la segmentation peut être médiocre parce que les hyperparamètres sont mal choisis.
- Le nombre de groupes peut être guidé par le contexte : les formules et indicateurs de qualité servent de balises.



# Chapitre 8

## Données manquantes

Il arrive fréquemment d'avoir des valeurs manquantes dans notre échantillon. Ces valeurs peuvent être manquantes pour diverses raisons. Si on prélève nous-mêmes nos données, un répondant peut refuser de répondre à certaines questions. Si on acquiert nos données d'une source externe, les valeurs de certaines variables peuvent être manquantes directement dans le fichier obtenu. Si on ne prend pas en compte le mécanisme générant les valeurs manquantes, ces dernières peuvent également biaiser nos analyses. Le but de ce chapitre est de faire un bref survol de ce sujet.

### 8.1 Principes de base

Soit  $X$  une variable pour laquelle des données sont manquantes. Voici la définition de trois processus de génération de données manquantes.

- 1) Les données manquantes de  $X$  sont dites **manquantes de façon complètement aléatoire** (MCAR, de l'anglais *missing completely at random*) si la probabilité que la valeur de  $X$  soit manquante ne dépend ni de la valeur de  $X$  (qui n'est pas observée), ni des valeurs des autres variables.

Le fait qu'une variable est manquante peut être relié au fait qu'une autre soit manquante. Des gens peuvent refuser systématiquement de répondre à deux questions dans un sondage. Dans ce cas, si la probabilité qu'une personne ne réponde pas ne dépend pas des valeurs de ces variables (et de toutes les autres), nous sommes encore dans le cas MCAR. Si par contre, la probabilité que les gens ne répondent pas à une question sur leur revenu augmente avec la valeur de ce revenu, alors nous ne sommes plus dans le cas MCAR.

Le cas MCAR peut se présenter par exemple si des questionnaires, ou des pages ont été égarés ou détruits par inadvertance (effacées du disque rigide, etc.) Si les questionnaires manquants

constituent un sous-ensemble choisi au hasard de tous les questionnaires, alors le processus est MCAR. L'hypothèse que les données manquantes sont complètement aléatoires est en général considérée comme trop restrictive.

- 2) Les données manquantes de  $X$  sont dites **données manquantes de façon aléatoire** (MAR, de l'anglais *missing at random*) si la probabilité que la valeur de  $X$  soit manquante ne dépend pas de la valeur de  $X$  (qui n'est pas observée) une fois qu'on a contrôlé pour les autres variables.

Il est possible par exemple que les femmes refusent plus souvent que les hommes de répondre à une question (et donc, le processus n'est pas MCAR). Si pour les femmes, la probabilité que  $X$  est manquante ne dépend pas de la valeur de  $X$  et que pour les hommes, la probabilité que  $X$  est manquante ne dépend pas de la valeur de  $X$ , alors le processus est MAR. Les probabilités d'avoir une valeur manquante sont différentes pour les hommes et les femmes mais cette probabilité ne dépend pas de la valeur de  $X$  elle-même. L'hypothèse MAR est donc plus faible que l'hypothèse MCAR.

- 3) Les données manquantes de  $X$  sont dites **manquantes de façon non-aléatoire** (MNAR, de l'anglais *missing not at random*) si la probabilité que la valeur de  $X$  soit manquante dépend de la valeur de  $X$  elle-même.

Par exemple, les gens qui ont un revenu élevé pourraient avoir plus de réticences à répondre à une question sur leur revenu. La méthode de traitement que nous allons voir dans ce chapitre, l'imputation multiple, est très générale et est valide dans le cas MAR (et donc aussi dans le cas MCAR). Le cas MNAR est beaucoup plus difficile à traiter et ne sera pas considéré ici.

## 8.2 Méthodes d'imputation

### 8.2.1 Cas complets

La première idée naïve pour une analyse est de retirer les observations avec données manquantes pour conserver les cas complets (*listwise deletion*, ou *complete case analysis*).

Cette méthode consiste à garder seulement les observations qui n'ont aucune valeur manquante pour les variables utilisées dans l'analyse demandée. Dès qu'une variable est manquante, on enlève le sujet au complet. C'est la méthode utilisée par défaut dans la plupart des logiciels, dont **R**.

- Si le processus est MCAR, cette méthode est valide car l'échantillon utilisé est vraiment un sous-échantillon aléatoire de l'échantillon original. Par contre, ce n'est pas nécessairement la meilleure solution car on perd de la précision en utilisant moins d'observations.
- Si le processus est seulement MAR ou MNAR, cette méthode produit généralement des estimations biaisées des paramètres.

En général, l'approche des cas complets est la première étape d'une analyse afin d'obtenir des estimés initiaux que nous corrigerons pas d'autre méthode. Elle n'est vraiment utile que si la proportion d'observations manquantes est très faible et le processus est MCAR. Évidemment, la présence de valeurs manquantes mène à une diminution de la précision des estimateurs (caractérisée par une augmentation des erreurs-types) et à une plus faible puissance pour les tests d'hypothèse et donc ignorer l'information partielle (si seulement certaines valeurs des variables explicatives sont manquantes) est sous-optimal.

### 8.2.2 Imputation simple

La deuxième approche est l'**imputation simple**. L'idée ici est de ne pas enlever les observations avec des valeurs manquantes mais de remplacer ces valeurs par des valeurs raisonnables. Par exemple, on peut remplacer les valeurs manquantes d'une variable par la moyenne de cette variable dans notre échantillon. On peut aussi ajuster un modèle de régression avec cette variable comme variable dépendante et d'autres variables explicatives comme variables indépendantes et utiliser les valeurs prédites comme remplacement. Une fois que les valeurs manquantes ont été remplacées, on fait l'analyse avec toutes les observations.

Il existe d'autres façons d'imputer les valeurs manquantes mais le problème de toutes ces approches est que l'on ne tient pas compte du fait que des valeurs ont été remplacées et on fait comme si c'était de vraies observations. Cela va en général sous-évaluer la variabilité dans les données. Par conséquent, les écarts-type des paramètres estimés seront en général sous-estimés et l'inférence (tests et intervalles de confiance) ne sera pas valide. Cette approche n'est donc **pas recommandée**.

Une manière de tenter de reproduire correctement la variabilité dans les données consiste à ajouter un terme aléatoire dans l'imputation. C'est ce que fait la méthode suivante, qui possédera l'avantage de corriger automatiquement les écarts-type des paramètres estimés.

### 8.2.3 Imputation multiple

Cette méthode peut être appliquée dans à peu près n'importe quelle situation et permet d'ajuster les écarts-type des paramètres estimés. Elle peut être appliquée lorsque le processus est MAR (et donc aussi MCAR).

L'idée consiste à procéder à une imputation aléatoire, selon une certaine technique, pour obtenir un échantillon complet et à ajuster le modèle d'intérêt avec cet échantillon. On répète ce processus plusieurs fois et on combine les résultats obtenus.

L'estimation finale des paramètres du modèle est alors simplement la moyenne des estimations pour les différentes répétitions et on peut également obtenir une estimation des écarts-type des paramètres qui tient compte du processus d'imputation.

Plus précisément, supposons qu'on s'intéresse à un seul paramètre  $\theta$  dans un modèle donné. Ce modèle pourrait être un modèle de régression linéaire, de régression logistique, etc. Le paramètre  $\theta$  serait alors un des  $\beta$  du modèle.

Supposons qu'on procède à  $K$  imputations, c'est-à-dire, qu'on construit  $K$  ensemble de données complets à partir de l'ensemble de données initial contenant des valeurs manquantes. On estime alors les paramètres du modèle séparément pour chacun des ensembles de données imputés. Soit  $\hat{\theta}_k$ , l'estimé du paramètre  $\theta$  pour l'échantillon  $k \in \{1, \dots, K\}$  et  $\hat{\sigma}_k^2 = \text{Va}(\hat{\theta}_k)$  l'estimé de la variance de  $\hat{\theta}_k$  produite par le modèle estimé.

L'estimation finale de  $\theta$ , dénotée  $\hat{\theta}$ , est obtenue tout simplement en faisant la moyenne des estimations de tous les modèles, c'est-à-dire,

$$\hat{\theta} = \frac{\hat{\theta}_1 + \dots + \hat{\theta}_K}{K}.$$

Une estimation ajustée de la variance de  $\hat{\theta}$  est

$$\begin{aligned}\text{Va}(\hat{\theta}) &= W + \frac{K+1}{K}B, \\ W &= \frac{1}{K} \sum_{k=1}^K \hat{\sigma}_k^2 = \frac{\hat{\sigma}_1^2 + \dots + \hat{\sigma}_K^2}{K}, \\ B &= \frac{1}{K-1} \sum_{k=1}^K (\hat{\theta}_k - \hat{\theta})^2.\end{aligned}$$

Ainsi, le terme  $W$  est la moyenne des variances et  $B$  est la variance entre les imputations. Le terme  $(1+1/K)B$  est celui qui vient corriger le fait qu'on travaille avec des données imputées et non pas des vraies données en augmentant la variance estimée du paramètre.

C'est ici qu'on voit l'intérêt à procéder à de l'imputation multiple. Si on procédait à une seule imputation (même en ajoutant une part d'aléatoire pour essayer de reproduire la variabilité des données), on ne serait pas en mesure d'estimer la variance inter-groupe de l'estimateur. Notez que la formule présentée n'est valide que pour le cas unidimensionnel; l'estimation de la variance dans le cas multidimensionnel est différente (voir Little et Rubin, 2002).

La méthode d'imputation multiple possède l'avantage d'être applicable avec n'importe quel modèle sous-jacent. Une fois qu'on a des échantillons complets (imputés), on ajuste le modèle comme d'habitude. Mais une observation imputée ne remplacera jamais une vraie observation. Il faut donc faire tout ce qu'on peut pour limiter le plus possible les données manquantes.

Il faut utiliser son jugement. Par exemple, si la proportion d'observations perdues est petite (moins de 5%), ça ne vaut peut-être pas la peine de prendre des mesures particulières et on peut faire une analyse avec les données complètes seulement. S'il y a un doute, on peut faire une analyse avec les données complètes seulement et une autre avec imputations multiples afin de valider la première.

TABLEAU 8.1 – Tableau des configurations des données manquantes.

$X_1$	$X_2$	$X_3$	$X_4$	$X_5$	$X_6$	$y$
1	4	0	1	35	2	0
.	1	0	.	33	3	0
2	3	1	.	46	3	0
5	2	1	.	32	1	1
3	2	1	.	38	3	1
.	4	0	0	36	3	0
.	3	0	.	35	3	0
.	5	1	0	26	2	0
.	3	1	1	39	2	1
5	2	1	.	38	3	0

Si, à l'inverse, une variable secondaire cause à elle seule une grande proportion de valeurs manquantes, on peut alors considérer l'éliminer afin de récupérer des observations. Par exemple, si vous avez une proportion de 30% de valeurs manquantes en utilisant toutes vos variables et que cette proportion baisse à 3% lorsque vous éliminez quelques variables peu importantes pour votre étude (ou qui peuvent être remplacées par d'autres jouant à peu près le même rôle qui elles sont disponibles), alors vous pourriez considérer la possibilité de les éliminer. Il est donc nécessaire d'examiner la configuration des valeurs manquantes avant de faire quoi que ce soit.

### 8.3 Example d'application de l'imputation

On examine l'exemple de recommandations de l'association professionnelle des vachers de la section Section 5.2.3.

Le but est d'examiner les effets des variables  $X_1$  à  $X_6$  sur les intentions d'achat; la base de données manquantes contient les observations. Il s'agit des mêmes données que celles du fichier `logit1` mais avec des valeurs manquantes.

Les points (.) indiquent des valeurs manquantes. Le premier sujet n'a pas de valeur manquante. Le deuxième a une valeur manquante pour  $X_1$  (emploi) et  $X_4$  (éducation), etc.

Une première façon de voir combien il y a de valeurs manquantes consiste à faire sortir les statistiques descriptives avec `summary`. Ainsi, il y a 192 valeurs manquantes pour  $X_1$ , 48 pour  $X_2$  et 184 pour  $X_4$ . Les autres variables n'ont pas de valeurs manquantes, incluant la variable dépendante  $Y$ . La procédure unidimensionnelle nous permet seulement de voir combien il y a de valeurs manquantes variable par variable.

TABLEAU 8.2 – Pourcentage de valeurs manquantes par variable.

	x1	x2	x3	x4	x5	x6	y
nombre	192	49	0	184	0	0	0
pourcentage	0.384	0.098	0	0.368	0	0	0

```

data(manquantes, package = 'hecmulti')
summary(manquantes)
# Pourcentage de valeurs manquantes
apply(manquantes, 2, function(x){mean(is.na(x))})
# Voir les configurations de valeurs manquantes
md.pattern(manquantes)

```

La Figure 8.1 donne une indication sur les différentes combinaisons de données complètes (cases bleues) et les observations manquantes (cases roses) avec leur fréquence. Les variables sont indiquées au dessus, les effectifs manquants en dessous, le nombre de cas de chaque combinaisons à gauche et le nombre de variables avec des valeurs manquantes à droite. Ainsi, il y a 180 sujets (36% de l'échantillon) avec aucune observation manquante. Il y en a 99 avec seulement  $X_4$  manquante et ainsi de suite. On voit donc, par exemple, que pour 14 sujets, à la fois  $X_1$  et  $X_2$  sont manquantes.

La recommandation d'usage est d'imputer au moins le pourcentage de cas incomplet, ici 64% donc 64 imputations. Si la procédure est trop coûteuse en calcul, on peut diminuer le nombre d'imputations, mais il faut au minimum 10 réplications pour avoir une bonne idée de la variabilité.

On peut comparer l'inférence avec toutes les variables explicatives pour les données sans valeurs manquantes ( $n = 500$  observations), avec les cas complets uniquement ( $n = 180$  observations). Le Tableau 8.3 présente les estimations des paramètres du modèle de régression logistique s'il n'y avait pas eu de valeurs manquantes, avec les cas complets et les résultats de l'imputation multiple.

Si on ajuste un modèle à une base de données qui contient des valeurs manquantes, le comportement par défaut est de retirer les observations qui ont au moins une valeur manquante pour une des variables nécessaires à l'analyse (voir la sortie de `glm(y ~ ., data = manquantes)`). Il ne serait pas raisonnable de faire l'analyse avec seulement 180 observations et de laisser tomber les 320 autres. De plus, comme nous l'avons vu plus haut, ce n'est pas valide à moins que le processus ne soit MCAR. La partie du milieu du Tableau 8.3 présente les estimations obtenues. Plusieurs variables significatives à niveau  $\alpha = 0.05$  ne le sont plus (puisque il y a moins d'information quand on réduit le nombre d'observations). Il y a même pire : non seulement la variable  $I(X_2 = 1)$  est passée de significative à non significative, mais en plus l'estimé de son paramètre a changé de signe.

Nous allons donc faire l'analyse avec l'imputation multiple, en prenant la méthode d'imputation par défaut

```

library(mice)
# Imputation multiple avec équations enchaînées
# Intensif en calcul, réduire `m` si nécessaire
impdata <- mice(data = manquantes,
                  m = 50,
                  seed = 2021,
                  method = "pmm",
                  printFlag = FALSE)
# Chaque copie est disponible (1, ..., 50)
complete(impdata, action = 1)
# ajuste les modèles avec les données imputées

adj_im <- with(data = impdata,
                 expr = glm(y ~ x1 + x2 + x3 + x4 + x5 + x6,
                             family = binomial(link = 'logit')))

# combinaison des résultats
fit <- pool(adj_im)
summary(fit)

```

La procédure `mice` du paquet éponyme crée les copies complètes du jeu de données. On peut ensuite appliquer une procédure quelconque et combiner les estimations avec `pool`.

On peut remarquer que la précision est systématiquement meilleure avec l'imputation multiple; les erreurs-type pour l'imputation multiple sont plus petits que celle du modèle qui retire les données incomplètes.

On voit que la variable  $X_3$  (sexe) est significative avec l'imputation multiple. Son paramètre estimé est 1.19, comparativement à 1.349 s'il n'y avait pas eu de valeurs manquantes. La précision dans l'estimation avec l'imputation multiple est seulement un peu moins bonne (erreur-type de 0.27) que celle s'il n'y avait pas eu de manquantes (erreur type de 0.26). Le paramètre de  $I(X_6 = 2)$  redevient aussi significatif, alors qu'il ne l'était plus si on retirait les manquantes. Il est peu probable que les données soit MCAR et donc les résultats de l'analyse des cas complets seraient biaisés.

TABLEAU 8.3 – Estimés, erreurs-type et valeurs-p des paramètres avec les 500 données complètes (gauche), avec les 180 cas complets (milieu) et avec l'imputation multiple (droite).

	Données complètes			Cas complets			Imputation multiple		
	$\hat{\beta}$	se( $\hat{\beta}$ )	valeur- $p$	$\hat{\beta}$	se( $\hat{\beta}$ )	valeur- $p$	$\hat{\beta}$	se( $\hat{\beta}$ )	valeur- $p$
cste	-6.89	1.02	0.00	-5.25	1.70	0.00	-6.57	1.04	0.00
$x_1 = 1$	0.36	0.48	0.46	-0.09	0.85	0.92	0.55	0.54	0.31
$x_1 = 2$	-0.47	0.37	0.21	-0.57	0.66	0.39	-0.13	0.45	0.78
$x_1 = 3$	-0.31	0.35	0.37	-0.47	0.66	0.47	0.07	0.44	0.87
$x_1 = 4$	-0.32	0.40	0.43	-0.93	0.74	0.21	-0.04	0.48	0.93
$x_2 = 1$	1.33	0.60	0.03	-0.74	1.14	0.52	1.10	0.65	0.09
$x_2 = 2$	1.15	0.50	0.02	0.46	0.91	0.61	1.03	0.55	0.06
$x_2 = 3$	0.77	0.48	0.11	-0.41	0.89	0.64	0.52	0.52	0.31
$x_2 = 4$	-1.11	0.54	0.04	-2.74	1.02	0.01	-1.04	0.57	0.07
$x_3$	1.35	0.26	0.00	0.80	0.44	0.07	1.19	0.27	0.00
$x_4$	1.83	0.30	0.00	2.25	0.58	0.00	1.52	0.37	0.00
$x_5$	0.11	0.02	0.00	0.11	0.03	0.00	0.10	0.02	0.00
$x_6 = 1$	2.41	0.38	0.00	2.23	0.66	0.00	2.26	0.38	0.00
$x_6 = 2$	1.04	0.25	0.00	0.83	0.44	0.06	1.00	0.25	0.00

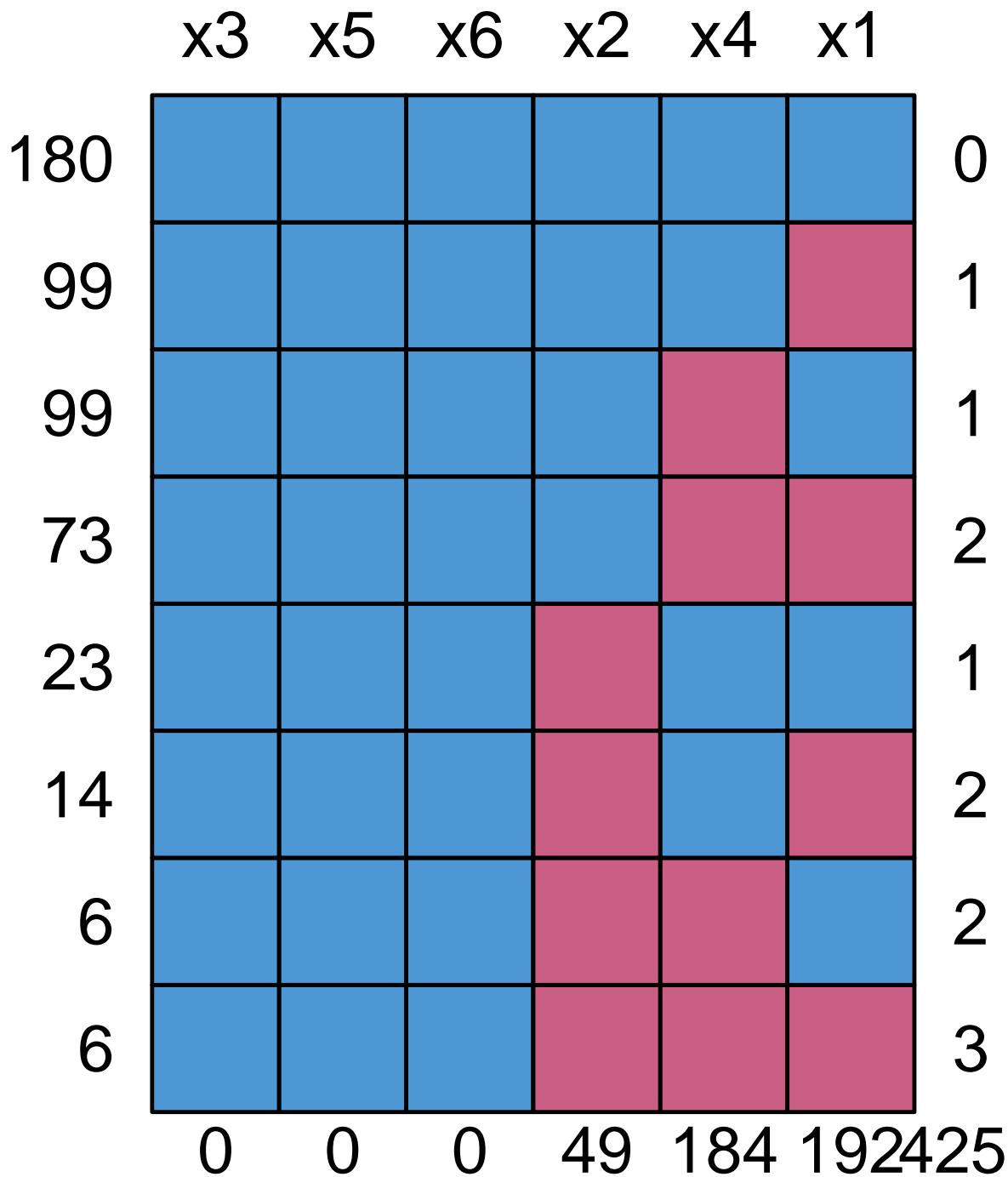


FIGURE 8.1 – Configurations des valeurs manquantes pour la base de données manquantes.

