Université Claude Bernard – Lyon 1
INSTITUT DE SCIENCE FINANCIERE ET D'ASSURANCES

Mémoire présenté
devant l'Institut de Science Financière et d'Assurances
pour l'obtention du diplôme d'Actuaire de l'Université de Lyon

le 7 janvier 2014

Par : Rémi BELLINA

Titre : Méthodes d'apprentissage appliquées à la tarification non-vie

Confidentialité : ☐ NON ☒ OUI (Durée : ☐ 1 an ☒ 2 ans)

Membres du jury de l'Institut des Actuaires :
M. Vladislav GRIGOROV
M. Pierre PETAUTON
M. Frédéric PLANCHET

Membres du jury I.S.F.A. :
Mme Ying JIAO
M. Nabil KAZI-TANI

Secrétariat :
Mme Marie-Claude MOUCHON

Bibliothèque :
Mme Patricia BARTOLO

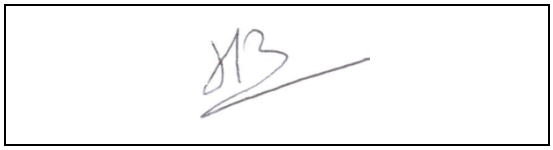
Entreprise :
Milliman

Directeur de mémoire en entreprise :
M. Fabrice TAILLIEU

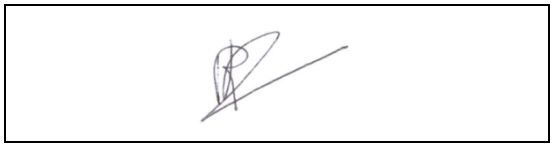
Invité :

Autorisation de mise en ligne sur un site de diffusion de documents actuariels *(après expiration de l'éventuel délai de confidentialité)*

Signature du responsable entreprise



Signature du candidat



50 Avenue Tony Garnier
69366 Lyon Cedex 07

Mémoire d'actuariat présenté devant
L'Institut de Science Financière et d'Assurances

Milliman

Méthodes d'apprentissage appliquées à la tarification non-vie

Mémoire Confidentiel (2 ans)

Auteur :
Rémi BELLINA

Directeur de mémoire :
Fabrice TAILLIEU

Entreprise :
Milliman

Date :
Janvier 2014

RÉSUMÉ

Mots-clefs
tarification, modèles linéaires généralisés, GLM, apprentissage statistique, arbres, CART, bootstrap, bagging, forêts aléatoires, boosting

La tarification constitue l'un des cœurs de métier de l'actuariat. Ce mémoire aborde les méthodes utilisées en tarification non-vie, et plus spécifiquement en tarification automobile. Tout l'enjeu est de comprendre les apports de l'apprentissage statistique par rapport aux méthodes plus classiques utilisées, dont les modèles linéaires généralisés (ou GLM) font figure de proue. Il est pour cela indispensable de se pencher sur les grands principes de la tarification afin de mettre en exergue un cadre général commun à toutes les méthodes, permettant de les comparer, et c'est là tout l'objet de l'introduction. Dans la première partie, nous présentons les données et les retraitements appliqués sur la base automobile à disposition comprenant plus de 500 000 individus. La deuxième partie met en avant les algorithmes dits linéaires, et principalement le GLM. Les troisième et quatrième parties abordent dans le détail certaines méthodes d'apprentissage statistique : les arbres de classification et de régression CART, ainsi que leurs algorithmes d'agrégation comme le bagging, les forêts aléatoires et le boosting. Enfin la dernière section propose une comparaison entre ces différentes procédures. On conclut que ces nouvelles approches non paramétriques fondées sur les arbres présentent de nombreux avantages, dont celui d'être facile à implémenter et de fournir une vision synthétique du portefeuille assuré. Nous tenons toutefois à nuancer, car leur qualité et leur performance ne sont pas qu'intrinsèques mais en partie liées à la base de données. Si celle-ci traduit l'exacte réalisation d'une loi paramétrique, alors une régression bien choisie sera sans doute plus adaptée. On retiendra néanmoins que les méthodes d'apprentissage offrent un cadre nouveau permettant d'avoir une compréhension plus poussée des risques sous-jacents au sein du portefeuille.

ABSTRACT

Keywords
pricing, generalized linear model, GLM, machine learning, trees, CART, bootstrap, bagging, random forest, boosting

Insurance pricing is a core business for insurance companies. Our aim here is to tackle the main methods used in non-life pricing and in particular within automobile insurance. The issue at stake is to have a better understanding of the ins and outs of machine learning systems applied to pricing. We compare them to more classical methods, like the very widespread generalized linear model (GLM). To begin with, we highlight a general mathematical framework for estimators in statistics. This enables us to compare the different methods. In the first part we present the automobile database used, which gathers half a million of insured. We also underline how to deal with outliers in the data. The second part focuses on the GLM. The aim of the third and fourth parts is to lay down the principles and the full computation of machine learning methods. We specifically go on about the classification and regression trees (CART) and the ensemble methods like bagging, random forests, and boosting. In the last part we eventually draw an analysis between all the results. We conclude with the advantages of the new non-parametric approaches based on trees. They are indeed easy to implement and they offer a synthetic vision of the insurance portfolio. Yet one needs to be cautious with these results. The high performance of the machine learning methods is linked to the database. If the data are the exact realization of a given parametric distribution then a regression model will fit almost perfectly. However, machine learning procedures give an opportunity to better understand the underlying risks of the portfolio, offering a new framework.

REMERCIEMENTS

Avant toute chose, je souhaite remercier mon directeur de mémoire Fabrice Taillieu, Principal et Directeur du département Non-Vie du cabinet Milliman Paris. Fabrice a su me faire confiance et accorder du crédit à mon travail tout en m'apportant son soutien pour réussir dans les meilleures conditions. Je remercie de plus le consultant Pierre-Aymeric Berthier pour sa remarquable contribution à ce mémoire. J'ai également beaucoup appris du manager Sébastien Delucinge qui a partagé son expérience sur la tarification non-vie.

De façon plus générale, je tiens à saluer l'ambiance dynamique dans laquelle il m'est actuellement permis d'évoluer au cours de cette première expérience professionnelle, avec une mention particulière pour les départements Non-Vie et R&D.

Je remercie, pour son implication, ma tutrice Diana Dorobantu, maître de conférences à l'Institut de Science Financière et d'Assurances, et plus généralement je n'oublie pas les personnes qui ont su m'accompagner durant ma scolarité.

SOMMAIRE

INTRODUCTION

Nous souhaitons avant toute chose présenter ici les grands principes de la tarification ainsi que de la théorie mathématique sous-jacente. Nous détaillons volontairement cette introduction afin de poser un cadre solide et cohérent, permettant de comparer les différentes méthodes que nous testerons.

GÉNÉRALITÉS SUR LA TARIFICATION

En général, on distingue la tarification a priori et a posteriori. Dans les deux cas, l'idée est de séparer les différents contrats et individus de la base de données en plusieurs classes ou catégories de sorte qu'en leur sein les risques puissent être considérés comme homogènes. En effet, en assurance l'hétérogénéité pose des problèmes, notamment liés au phénomène d'antisélection.

Supposons que nous ne segmentions pas suffisamment le portefeuille, et que nous appliquions la même prime pure à tous les assurés. Alors les mauvais risques seront plus enclins à se faire assurer tandis que les bons risques, jugeant la prime trop chère, iront voir la concurrence. Le résultat va donc se dégrader peu à peu. A contrario, une segmentation des risques trop fine, à l'échelle même de l'individu, n'est pas suffisamment robuste pour être effective. Dès lors on comprend l'importance d'une juste classification.

On parle de classes « a priori » lorsque la segmentation est réalisée à partir d'une information disponible portant sur l'assuré ou encore le bien assuré. On parle de classes « a posteriori » si la segmentation considère en outre l'historique de sinistralité de l'assuré. La tarification a posteriori est en lien direct avec la théorie de la crédibilité et les approches bayésiennes, elle ne sera pas développée ici (cf. DENUIT et CHARPENTIER, 2005). **Toutes les méthodes et théories que nous présenterons concernent la tarification a priori**.

Dans ce contexte, la tarification consiste à calculer la meilleure estimation de la prime à appliquer à un assuré : on parle de prime pure. Dans le modèle collectif, nous sommes amenés à écrire la sinistralité totale de la façon suivante, avec $(W_i)_{i\in\mathbb{N}^*}$ v.a. des montants de sinistres à valeurs dans $\mathbb{R}$ et N v.a. du nombre de sinistres à valeurs dans $\mathbb{N}$:

$$E\left[\sum_{i=1}^{N} W_i\right] = E[N]E[W]$$

pour des montants $(W_i)_{i\in\mathbb{N}^*}$ de variable parente W supposés indépendants et identiquement distribués (i.i.d.) et indépendants de N. Par suite, la prime pure se décompose classiquement en deux composantes :

- la fréquence moyenne,
- le coût moyen.

Nous y reviendrons dans la première partie plus longuement, mais notons d'emblée que cette séparation n'est pas que mathématique. En effet, les facteurs explicatifs ne sont souvent pas les mêmes. En assurance automobile, la fréquence est essentiellement liée au conducteur tandis que le coût moyen est plutôt expliqué par les caractéristiques du véhicule. Il y a de plus un décalage structurel entre les deux notions : le coût moyen n'est connu qu'au terme du développement final du sinistre alors que la fréquence est connue dès la déclaration.

En outre, la modélisation directe de la prime pure sans cette décomposition naturelle, est difficile. Nous entendons par là qu'il n'existe pas de lois usuelles (cf. annexe B) l'expliquant directement. À

l'opposé, on peut citer des lois usuelles pour la fréquence (Poisson, binomiale négative, etc.) et le coût moyen (log-normale, Pareto, etc.).

Enfin, il est important de noter que la prime pure n'est pas la prime que paiera effectivement l'assuré, on lui ajoute un chargement commercial (cf. partie 1.1.3). Avant de parler plus en profondeur de la modélisation, penchons-nous un instant sur les variables en jeu.

VARIABLES TARIFAIRES

La tarification repose avant tout sur une segmentation adaptée des individus assurés en classes de risques homogènes disposant de la même prime pure, cela présuppose donc un jeu de variables.

Les variables se distinguent selon leur type :

- variables qualitatives : ce sont des variables qui représentent des « qualités », ou « modalités ». Elles peuvent être de deux sous-types :
    - ordinale : il existe une relation d'ordre sous-jacente, comme on peut le retrouver dans un système de notation (AAA, AA, etc.),
    - nominale : il n'y a pas d'ordre précis, on peut citer le sexe par exemple (homme ou femme).

- variables quantitatives : ce sont des variables numériques, en particulier elles sont ordinales. Là encore elles sont classées en deux sous-catégories :
    - discrètes (âge de l'assuré),
    - continues (prix d'achat du véhicule).

Les variables tarifaires sont en général des variables à modalités finies, les variables continues sont volontiers regroupées en classe (on parle aussi de *bucket*). Une variable qualitative à k facteurs est souvent codée par $k-1$ variables binaires nulles pour le niveau de référence (notamment lorsque l'on applique un modèle linéaire généralisé).

Le phénomène d'antisélection se retrouve dans le choix des variables explicatives. L'assureur ne peut se permettre de choisir n'importe quelle variable pour mener à bien son étude. Ces informations doivent être vérifiables. Par exemple pendant longtemps, la distance totale parcourue n'a pas pu être utilisée pour une tarification car l'assuré aurait pu mentir sur le sujet (aujourd'hui il existe des offres *pay as you drive*).

Le choix des variables explicatives est une étape capitale dans la modélisation, mais qu'entendons-nous par modélisation ? Rappelons donc que notre objectif est d'avoir une vision claire et robuste de la prime pure, autrement dit de la fréquence moyenne et du coût moyen. Si l'on se penche sur la fréquence moyenne, que nous notons Y, il s'agit de l'expliquer en fonction des variables tarifaires notées X, de sorte que l'on puisse « prédire » la fréquence de sinistre d'un nouvel individu, ce qui déterminera le prix de la couverture.

THÉORIE MATHÉMATIQUE GÉNÉRALE

Nous souhaitons dans cette sous-partie prendre un recul nécessaire sur les méthodes d'estimation.

Contexte introductif

Il s'agit généralement d'étudier une variable aléatoire Y dépendant de plusieurs variables explicatives $X \in \mathbb{R}^p$. Nous disposons d'un échantillon $(X_i, Y_i)_{i\in\{1,\dots,n\}}$ et d'une réalisation $(x_i, y_i)_{i\in\{1,\dots,n\}}$. Nous souhaitons dans l'absolu tenter d'expliquer la variable Y par la variable X, eu égard à la réalisation de

l'échantillon. Classiquement, on peut donc chercher Y sous la forme d'une fonction $\phi: \mathbb{R}^p \to \mathbb{R}$ de X. Nous cherchons la « meilleure » (dans un sens à définir) des fonctions ϕ.

On instaure pour cela une fonction de perte Q permettant de quantifier la distance entre Y et $\phi(X)$. C'est donc une fonction de $\mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$, s'apparentant à la distance d'un espace métrique. Il y a une infinité de possibilités, nous pouvons citer (pour $u, v \in \mathbb{R}^n$) :

- $Q(u,v) = \|u-v\|_1 = \sum_{i=1}^n |u_i - v_i|$
- $Q(u,v) = \|u-v\|_2 = \sqrt{\sum_{i=1}^n (u_i - v_i)^2}$
- $Q(u,v) = \|u-v\|_2^2 = \sum_{i=1}^n (u_i - v_i)^2$
- $Q(u,v) = \|u-v\|_\infty = \sup_{i\in\{1,\dots,n\}} |u_i - v_i|$

Avant d'aller plus loin, il est nécessaire de faire quelques rappels sur l'espérance conditionnelle afin de choisir Q intelligemment.

Nous effectuons en annexe des calculs menés au sein de l'espace des variables aléatoires de carré intégrable, qui présente la particularité d'être complet pour la norme associée au produit scalaire (cf. annexe C). Dans ce contexte, les théorèmes de projection s'appliquent et nous parvenons à :

$$E[Y|X] = \arg\min_{W \in L^2(\mathcal{G})} \|Y - W\|_2 = \arg\min_{W \in L^2(\mathcal{G})} E[(Y-W)^2] \qquad (1)$$

Nous montrons ainsi que l'espérance conditionnelle est la solution d'un problème de minimisation de la norme associée au produit scalaire sur $L^2(\Omega)$.

Problématique de l'estimation

Revenons à présent sur notre problème initial d'estimation. Pour rappel, nous cherchions la meilleure des fonctions ϕ (sous réserve d'existence et d'unicité) au sens d'un critère de qualité Q permettant d'expliquer Y par X.

On note $\mathcal{F}$ l'ensemble des fonctions ϕ admissibles[1] de $\mathbb{R}^p \to \mathbb{R}$. On cherche ainsi à résoudre :

$$\phi^* = \arg\min_{\phi \in \mathcal{F}} Q\big(Y, \phi(X)\big)$$

Avec $Y = (y_i),\ X = (x_i) \in \mathbb{R}^n$. En se référant à la partie précédente[2], il apparait naturel de choisir une fonction de qualité quadratique $(u,v) = \|u-v\|_2$. Ainsi en utilisant l'équation (1) :

$$\begin{aligned} \phi^* &= \arg\min_{\phi \in \mathcal{F}} Q\big(Y, \phi(X)\big) \\ &= \arg\min_{\phi \in \mathcal{F}} \|Y - \phi(X)\|_2 \\ &= \arg\min_{W \in L^2(\mathcal{G})} \|Y - W\|_2 \\ &= E[Y|X] \end{aligned}$$

On peut donc conclure que :

Si Q est quadratique alors, $\phi^* = \arg\min_{\phi \in \mathcal{F}} Q\big(Y, \phi(X)\big) = \arg\min_{\phi \in \mathcal{F}} E\left[\big(Y - \phi(X)\big)^2\right] = E[Y|X]$ (2)

Dès lors, la meilleure représentation $\phi^*(X)$ que l'on puisse faire de Y sachant que l'on dispose de l'échantillon X est donnée par l'espérance conditionnelle $E[Y|X]$. C'est un résultat important car, nous

[1] Nous restons volontairement flous sur cette notion pour l'instant.
[2] Et le lemme : X est $\sigma(Y)$-mesurable ssi il existe une fonction g borélienne telle que $X = g(Y)$.

le verrons au cours de ce mémoire, il est le fondement de toute la théorie des modèles linéaires généralisés[1] mais aussi du machine learning.

Il est à préciser que dans le cadre d'une estimation, on dispose d'une réalisation x de l'échantillon X. Le problème s'écrit alors :

$$\phi_n^* = \arg\min_{\phi \in \mathcal{F}} E\left[\left(Y - \phi(X)\right)^2\right] = \arg\min_{\phi \in \mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \phi(x_i)\right)^2 \tag{3}$$

On soulignera l'importance d'utiliser des critères quadratiques, notamment dans l'analyse de la performance d'un modèle. Lors d'une estimation nous cherchons ainsi à minimiser $\sum_{i=1}^{n}\left(y_i - \phi(x_i)\right)^2$, autrement dit la somme du carré des écarts entre les valeurs réelles et les prévisions. Cette approche fournit un cadre tout indiqué pour tester si un modèle est bon. Concrètement, si l'on dispose d'une estimation $\hat{\phi}_1$ obtenue à l'aide d'un premier modèle et d'une estimation $\hat{\phi}_2$ obtenue avec un second modèle, alors on dira que :

$$\begin{aligned} \text{Modèle 1} > \text{ Modèle 2} \quad &\Leftrightarrow \quad \sum_{i=1}^{n}\left(y_i - \hat{\phi}_1(x_i)\right)^2 < \sum_{i=1}^{n}\left(y_i - \hat{\phi}_2(x_i)\right)^2 \\ &\Leftrightarrow \quad \left\|y - \hat{\phi}_1(x)\right\|_2 < \left\|y - \hat{\phi}_2(x)\right\|_2 \end{aligned} \tag{4}$$

Toutefois, il n'est pas rare de voir d'autres critères de sélection de modèles, utilisant une autre norme que celle associée au produit scalaire sur l'espace L^2. On citera ainsi la norme 1 et la norme infinie. Ce sont des critères que nous utiliserons également mais seulement à titre indicatif, puisque nous venons de souligner toute la légitimité de la norme 2. Néanmoins en ouverture, nous reviendrons sur la norme 1, via la théorie de la robustesse (cf. partie 5.3).

À ce stade, notre estimation est solution de :

$$\phi_n^* = \arg\min_{\phi \in \mathcal{F}} \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \phi(x_i)\right)^2 = \arg\min_{\phi \in \mathcal{F}} \sum_{i=1}^{n}\left(y_i - \phi(x_i)\right)^2$$

Faisons un instant l'hypothèse que tous les profils x_i sont différents. Si nous créons une fonction ϕ interpolant exactement les y_i en les x_i, alors la somme est bien minimisée puisqu'elle est nulle. On comprend donc qu'il existe une infinité de fonctions $\phi: \mathbb{R}^p \to \mathbb{R}$ solutions du problème de minimisation, du moment que $\phi(x_i) = y_i$ pour tout $i \in \{1, \dots, n\}$. En dehors de ces points, la fonction peut prendre n'importe quelle valeur. Un exemple de solution est donc :

$$\phi_n: \mathbb{R}^p \to \mathbb{R}, x \longmapsto \sum_{i=1}^{n} y_i \mathbb{1}_{\{x = x_i\}} \tag{5}$$

Cette solution est-elle satisfaisante pour autant ? La réponse est bien évidemment non car il ne faut pas oublier le contexte. En effet, notre but est de trouver une fonction ϕ qui apprenne certes la réalisation de notre échantillon, mais afin d'en tirer une information générale permettant la prévision **sur une nouvelle base de données**. Dans notre exemple, n'importe quel autre vecteur de variables explicatives entré (typiquement un nouvel individu) se verra attribuer la valeur zéro, ce n'est clairement pas ce que nous souhaitions. Pour être parfaitement rigoureux, la fonction ϕ que nous cherchons satisfait un critère de moindres carrés sur l'échantillon à disposition mais devra également le faire sur un nouvel échantillon dont nous ignorons tout pour le moment. On parle de sur-apprentissage lorsque la fonction estimée n'a qu'un très faible pouvoir prédictif sur de nouvelles données.

[1] Les *GLM* supposent une certaine loi sur Y puis modélisent $g(E[Y|X]) = X\beta$.

SUR-APPRENTISSAGE

Le sur-apprentissage traduit une trop forte similitude entre la solution réelle connue et la prédiction (cf. DREYFUS, MARTINEZ, SAMUELIDES, GORDON, BADRAN et THIRIA, 2008). Lorsqu'il y a un sur-apprentissage total, alors la fonction de prédiction $\phi(.)$ est parfaitement égale à Y.

D'une certaine façon, en cas de sur-apprentissage la fonction ϕ se comporte comme une table contenant tous les échantillons utilisés lors de l'apprentissage et perd ses pouvoirs de prédiction sur de nouveaux échantillons, c'est le cas de la fonction donnée en exemple plus haut.

Afin de mieux comprendre ce phénomène, regardons de plus près la somme des écarts au carré :

$$\begin{aligned} E\left[\left(\hat{\phi}(X)-\phi(X)\right)^2\right] &= E\left[\hat{\phi}(X)^2\right]-2\phi(X)E\left[\hat{\phi}(X)\right]+\phi(X)^2 \\ &= \underbrace{E\left[\hat{\phi}(X)^2\right]-E\left[\hat{\phi}(X)\right]^2}_{V\left[\hat{\phi}(X)\right]}+\underbrace{E\left[\hat{\phi}(X)\right]^2-2\phi(X)E\left[\hat{\phi}(X)\right]+\phi(X)^2}_{\left(E\left[\hat{\phi}(X)-\phi(X)\right]\right)^2} \\ &= V\left[\hat{\phi}(X)\right]+\left(E\left[\hat{\phi}(X)-\phi(X)\right]\right)^2 \\ &= \text{Variance}+\text{Biais}^2 \end{aligned}$$

On rappelle que l'on veut minimiser cette erreur. Si le modèle est très simple, alors la variance sera petite mais le biais sera grand, c'est le cas contraire si le modèle est trop paramétré. Il y a donc un compromis à trouver entre biais et variance. Il y a sur-apprentissage lorsque la variance est très élevée, cela traduit l'*overfitting*.

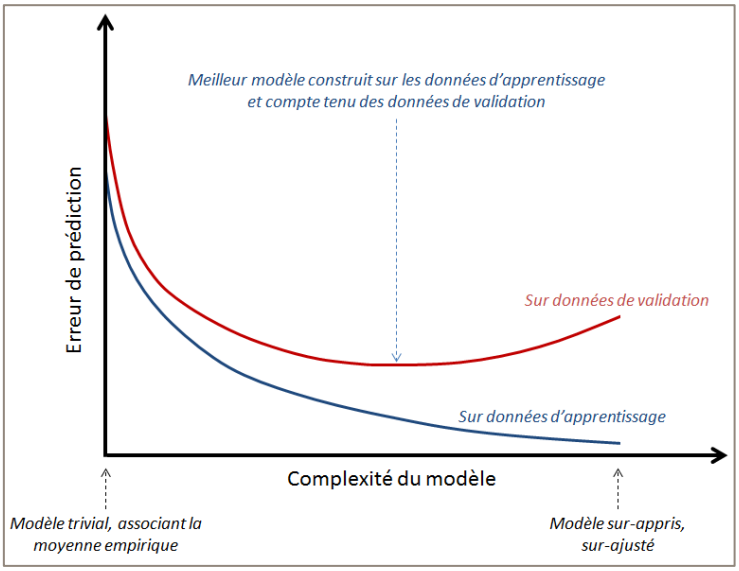


Figure 1 : Illustration du sur-apprentissage

Globalement, les méthodes d'apprentissage nous aident à quantifier cet aspect-là. L'idée est « d'apprendre » notre fonction ϕ sur notre réalisation de l'échantillon $(x_i, y_i)_{i\in\{1,\dots,n\}}$ mais pas complètement, pas de façon saturée, afin de conserver une forme de généralité. On cherchera ainsi à regrouper les variables qui se ressemblent pour extraire de l'information robuste et éviter le phénomène d'*overfitting*. Nous allons détailler certaines méthodes d'apprentissage dans ce mémoire, nous essayerons toujours de garder à l'esprit les considérations évoquées dans cette partie.

APPROCHES PARAMÉTRIQUES

Si on s'intéresse à une variable d'intérêt Y et que l'on dispose de variables explicatives $X \in \mathbb{R}^p$, on cherche alors à estimer $\phi(X) = E[Y|X]$ (où X est un vecteur de dimension p et Y est un scalaire).

Notre objectif est donc d'obtenir un estimateur $\hat{\phi}(X)$, tel que $\hat{\phi}(x) = E[Y|X = x]$, où $(x_i)_{i\in\{1,\dots,n\}}$ est la réalisation de l'échantillon. Toutefois estimer $\phi(X)$ n'est pas simple car cela revient à un problème d'estimation non paramétrique et on ne sait rien de la fonction ϕ. Ainsi pour rendre cette estimation plus simple, des hypothèses sont faites sur la loi de Y, pour transformer ce problème en un problème d'estimation paramétrique.

Pour cela, créons l'application $f: \mathcal{F} \to \mathbb{R}, \phi \mapsto \sum_{i=1}^{n}\big(y_i - \phi(x_i)\big)^2$, de sorte que $\phi_n^* = \arg\min_{\phi\in\mathcal{F}} f(\phi)$, on cherche donc le minimum global (sous réserve d'existence et d'unicité) de la fonction f en évitant le sur-apprentissage.

Le problème qui survient est que $\mathcal{F}$ n'est pas de dimension finie, or la théorie du calcul différentiel se complexifie en dimension infinie : on parle alors de calcul des variations. Une solution est de chercher ϕ parmi une famille de fonctions définies par des paramètres $\xi \in \mathbb{R}^r$. Le problème se transforme alors en $\phi_n^* = \phi_{\xi^*}$ avec $\xi^* = \arg\min_{\xi\in\mathbb{R}^r} f\big(\phi_\xi\big)$. Nous sommes donc ramenés à un cas en dimension finie plus favorable. Les méthodes *GLM* sont justement des méthodes paramétriques. Dans l'estimation paramétrique, on évite le sur-apprentissage en imposant une stricte inclusion $\mathcal{F} \subset \mathcal{F}(\mathbb{R}^p, \mathbb{R})$, la fonction ϕ solution a une forme imposée (en pratique on ne pourra donc pas obtenir l'équation (5)). Comme c'est la quantité $E[Y|X]$ qui est modélisée dans le modèle *GLM* et non directement Y, on peut en outre modéliser aussi bien une variable d'intérêt Y continue ou discrète.

La modélisation du *GLM* suppose que $E[Y|X]$, perturbée par une fonction de lien, est une relation linéaire en les variables explicatives : $g(E[Y|X]) = X\beta$, où $\beta \in \mathbb{R}^{p+1}$ est inconnu et à estimer. Dans le cas où Y est supposée gaussienne, et que la fonction de lien choisie est la fonction de lien dite « canonique », alors on se trouve dans le modèle linéaire classique.

Une fois l'hypothèse de loi (et donc le choix de la fonction g) réalisée, il suffit d'estimer les paramètres β ($\in \mathbb{R}^{p+1}$) car sous l'hypothèse de linéarité, estimer $E[Y|X]$ revient seulement à estimer β. L'estimation devient beaucoup plus facile à mener.

La statistique paramétrique traduit l'estimation d'un nombre fini de paramètres $\xi \in \mathbb{R}^r$. Toutefois les modèles sont souvent simples (approche linéaire), n'étant que des approximations, de plus les résultats sont très souvent asymptotiques. La réalité est bien plus complexe et le nombre d'observations est limité. La statistique non paramétrique s'intéresse à l'estimation d'une fonction inconnue appartenant à un espace fonctionnel de dimension infinie.

APPROCHES NON PARAMÉTRIQUES ET APPRENTISSAGE

Lorsque l'on parle d'apprentissage ou plus généralement de *data mining*, l'une des grandes familles de techniques employées repose sur le précepte algorithmique : « diviser pour régner ». Concrètement, il s'agit de procéder à des subdivisions récursives d'un problème initial de grande taille, afin de parvenir à de multiples problèmes de taille réduite. Les méthodes d'apprentissage par arbres de décision s'en inspirent. L'apprentissage permet d'identifier les sous-ensembles où les solutions sont identiques. Dès lors, lorsqu'un cas nouveau est soumis au système, l'attribution de la solution revient à l'attribution des données d'entrée dans un sous-ensemble préétabli.

Afin de gagner en efficacité et en précision, la tâche est parfois distribuée ou répétée plusieurs fois, comme confiée à différents experts. La solution générale s'écrit alors comme la combinaison des réponses de chacun. On parle de combinaison statique si celle-ci ne dépend pas de l'entrée (typiquement la moyenne des différentes réponses), on parlera de combinaison dynamique le cas contraire. Ces approches seront abordées sous l'appellation de « méthode ensemblistes » ou « méthode d'agrégation ».

On peut citer deux principaux problèmes en *data mining* :

- La classification supervisée consiste à attribuer une classe à des entrées,

- La classification non supervisée (ou *clustering*) permet de former des groupes homogènes au sein d'une population.

Dans le cadre de l'apprentissage supervisé, on cherche à construire un modèle du type $Y = \phi(X)$. La procédure se divise alors de la façon suivante :

1. Choix de la technique de modélisation : arbres de décision, méthodes d'ensemble, réseaux de neurones, *support vector machines*, etc.,

2. Spécification d'un protocole d'évaluation : séparation de la base en une partie d'apprentissage, une partie de validation et une partie de test, minimisation des écarts au carré, maximisation de l'indice de Gini sur les courbes de *lift*, etc.,

3. Construction du modèle sur la base d'apprentissage, optimisation des paramètres sur la base de validation,

4. Évaluation de la performance du modèle selon le critère choisi sur l'échantillon de test.

Dans le cadre d'une tarification automobile, nous pouvons illustrer le processus de prédiction par la donnée d'un questionnaire structuré, par exemple par une séquence de questions :

- Le conducteur est-il un homme ou une femme ?
- Quelle est la puissance de la voiture ?
- Quel est le type de véhicule ?
- Etc.

En général, les méthodes d'apprentissage n'imposent pas de restriction sur $\mathcal{F}$. Mener en l'état le problème de minimisation de l'équation (3) revient à obtenir une fonction sur-apprise de la forme (5). L'équation (3) est donc perturbée pour empêcher la saturation de la fonction ϕ_n^*. Dans la méthode *CART*, on ajoutera ainsi un critère d'arrêt matérialisé par un coefficient α (cf. partie 3.3.3). Si l'apprentissage est complet et bien réalisé, alors le tarif à appliquer (la réponse) est sans ambiguïté.

Il existe de nombreuses méthodes d'apprentissage statistique : les arbres de décision, les réseaux de neurones, les *Support Vector Machines*, etc. Si leurs fondements mathématiques, hypothèses et algorithmes de calculs sont tous très différents, elles ont ceci en commun que leur objectif est de prédire les valeurs d'une variable ciblée sur un ensemble d'observations à partir de variables dites explicatives, ou à défaut de classer ces dernières.

Dans le cas d'apprentissage supervisé où nous souhaitons expliquer une variable de sortie, l'échantillon est classiquement subdivisé en 3 parties :

- **Échantillon d'apprentissage** ($\sim 70\,\%$) : c'est l'échantillon principal où sont appliquées les méthodes, sur lequel les algorithmes apprennent. Il sert à ajuster le modèle.

- **Échantillon de validation** ($\sim 20\,\%$) : il permet d'optimiser les paramètres de la méthode, l'idée étant de tester le résultat de l'apprentissage avec plusieurs jeux de paramètres donnés sur cet échantillon afin d'en conserver le meilleur. Il permet d'ajuster la taille du modèle également.

- **Échantillon de test** ($\sim 10\ \%$) : ce dernier est utilisé pour tester l'adéquation du modèle optimal (au sens de la base de validation). Il n'a donc pas été utilisé pour l'apprentissage, la solution trouvée est indépendante de cet échantillon. L'idée source est de simuler la réception de nouvelles données afin de mettre en œuvre les méthodes, à la différence que sur cette base nous disposons des « vraies » valeurs de la variable à expliquer. Par suite, cet échantillon permet d'évaluer objectivement l'erreur réelle.

Arbres de décision

La principale qualité des arbres de décision est qu'ils sont très facilement lisibles et interprétables en un ensemble de règles simples. Leur utilisation est donc judicieuse lorsque les résultats doivent être exploités par un utilisateur lambda. En revanche, leur construction les rend peu pertinents lorsque les effectifs d'apprentissage sont faibles. Il existe trois principales méthodes d'induction d'arbres :

- *CHAID* est particulièrement utile sur de grandes bases de données mais nécessite un paramétrage compliqué,

- *C4.5* a l'avantage d'être efficace même sur de petits effectifs, et ne requiert pas non plus de paramétrage complexe. Néanmoins, elle construit des arbres de très grande taille, la rendant plus difficilement interprétable lorsque le nombre de données augmente,

- *CART* présente de meilleurs performances en classement – excepté sur les petites bases de données – et ne nécessite pas de paramétrage, à noter que la segmentation est toujours binaire, c'est la méthode que nous choisissons de développer.

Méthodes ensemblistes ou méthodes d'agrégation

Les méthodes ensemblistes, c'est-à-dire d'agrégation d'arbres, telles que le *bagging*, les forêts aléatoires et le *boosting*, fournissent de meilleures performances que les modèles simples, en introduisant de l'aléatoire (via des échantillons *bootstrap* par exemple) ou à défaut en mettant en œuvre des méthodes de descente. Le gain sur l'erreur est substantiel et les estimations sont plus robustes. Cependant en contrepartie, la sortie que ces modèles fournissent ne possède pas la même structure simple et intuitive des arbres de décision.

Autres méthodes

Les réseaux neuronaux constituent une méthode plus complexe à mettre en œuvre en raison des calculs qu'elle nécessite. Ils ne permettent pas de mesurer l'influence d'une variable sur la variable de sortie dès lors qu'une couche cachée est présente (aspect « boîte noire »), et la fonction de prédiction qu'ils produisent n'est pas interprétable en termes de règles simples. Néanmoins, si les données sont trop complexes pour les méthodes d'apprentissage traditionnelles, les réseaux neuronaux sont à privilégier, en ce sens qu'ils utilisent non seulement une pondération des variables mais également une pondération des liaisons entre celles-ci. Ils permettent ainsi de mettre en évidence des relations de causalité que les autres méthodes ne voient pas, en particulier dans les situations non-linéaires.

Les *Support Vector Machines* (*SVM*) sont une alternative non négligeable aux méthodes d'apprentissage les plus performantes, y compris les modèles agrégés. Le principal reproche qui leur est adressé est de n'être directement applicable que pour des variables à prédire à deux modalités, sans compter que les paramètres du modèle solution sont difficilement interprétables. Néanmoins, la construction et le choix des fonctions noyaux étant laissés libres à l'utilisateur, cette méthode s'avère être très flexible et adaptable à de nombreuses problématiques.

OBJECTIFS

L'enjeu de ce mémoire est de présenter les apports de ces nouvelles méthodes d'apprentissage supervisé, par rapport à des méthodes paramétriques plus classiques utilisées en tarification non-vie. La tarification constitue un domaine clé et stratégique pour une compagnie d'assurance, et pourtant les méthodes le plus souvent utilisées ne sont pas nécessairement les plus faciles à mettre en œuvre, ou pire, ne fournissent pas toujours de bons résultats. Nous utilisons comme support de l'étude des données automobile d'environ 500 000 individus.

Tout d'abord, nous présentons les bases de la tarification de contrats automobile, en définissant la notion de prime pure. Une fois cet objectif de modélisation mis en place, nous nous penchons en détail sur la base de données en dressant quelques analyses d'abord sommaires, puis plus poussées via des tests statistiques et des analyses graphiques. Cette étape de redressement et retraitement des données est primordiale, commune à toutes les méthodes. Nous faisons par ailleurs le choix, justifié, de nous concentrer exclusivement dans la suite sur le modèle de fréquence.

Dans la deuxième partie, le modèle linéaire généralisé est appliqué à la base de données. Nous mettons en œuvre une régression de Poisson, en insistant sur cette nécessité absolue et exigeante de préciser des hypothèses de loi lorsque l'on utilise cette méthode paramétrique.

Dans les troisième et quatrième parties, nous présentons dans le détail l'algorithme *CART* et ses méthodes d'agrégation. Nous nous efforçons de détailler les concepts mathématiques sous-jacents et leur logique de construction, leur application en actuariat étant récente. Ils sont décrits comme une alternative au *GLM* apportant une vision nouvelle du portefeuille.

Enfin dans l'ultime partie nous présentons une analyse détaillée et comparative entre les différentes méthodes mises en œuvre, avec des critères tant graphiques que quantitatifs. Une partie non négligeable sur la robustesse est aussi traitée, afin d'approfondir et tester les limites de l'algorithme *CART*.

En annexes A,B,C, après avoir fait quelques rappels de probabilités et présenté les lois usuelles utilisées en actuariat, le cadre mathématique de l'espérance conditionnelle est expliqué en lien avec l'introduction de ce mémoire.

Le modèle collectif permet d'obtenir la prime pure à l'aide du moment d'ordre 1 de la sinistralité. Cependant, si l'on veut appliquer un chargement commercial, une mesure de risque doit être choisie. De fait, on montre en annexe D comment trouver des moments d'ordre supérieurs.

Si le modèle *GLM* est classique, on ne peut en dire autant de la méthode *CART* et c'est pourquoi nous détaillons un exemple simple en annexe E, dans le but de parfaire la compréhension de l'algorithme. Au même titre, nous revenons plus largement sur les algorithmes de descente à l'origine du *gradient boosting* en annexe F. Le *stochastic gradient boosting*, nous le verrons, est mis en œuvre à l'aide d'un résultat sur les échantillons *bootstrap* (cf. partie 4.3.5), résultat que nous prouvons en annexe G.

Enfin, en annexe H sont construites les courbes de *Lift* (ou de Lorenz selon le contexte). Ces courbes sont souvent utilisées dans le contexte de tarification afin de tester le pouvoir explicatif d'un modèle, caractéristique à différencier de sa performance de prédiction.

1. TARIFICATION AUTOMOBILE

1.1. CONTEXTE

1.1.1. Enjeux

C'est dans la fin des années 1950 que l'assurance automobile devient obligatoire. Peu à peu, de nombreux assureurs ont été attirés par le marché. Aujourd'hui ce dernier est considéré comme étant saturé, et le parc automobile n'évolue plus tellement. C'est la raison pour laquelle la tarification automobile est primordiale, car la compétition y est très importante. C'est un enjeu majeur pour de nombreux assureurs.

En pratique, les assureurs sont intéressés par des analyses poussées sur l'ensemble des variables explicatives potentielles à disposition. Une fois ces analyses préliminaires réalisées, l'étude et la modélisation de la prime pure peut débuter, cela passe souvent par la définition d'un zonier (classification géographique adaptée) et d'un véhiculier.

Dans ce mémoire, nous considérons un portefeuille d'assurance automobile, que nous analysons à l'aide du logiciel R. On notera d'emblée que ce choix est motivé par le fait que ce logiciel est particulièrement à jour en ce qui concerne les méthodes d'apprentissage. En général, pour des bases de données volumineuses, les actuaires se tourneront plus volontiers vers SAS. Les méthodes que nous présenterons peuvent s'appliquer à d'autres domaines que l'assurance automobile.

Nous ne détaillerons pas de façon exhaustive les données pour des raisons de confidentialité, un facteur d'échelle a par ailleurs été appliqué. **La base utilisée est une base automobile comprenant plus de $500\,000$ individus**.

1.1.2. Définition de la prime pure

Revenons sur la notion de prime pure succinctement abordée dans l'introduction. La charge totale de l'assureur notée S est le risque transmis. Dans l'absolu, on souhaite remplacer cette variable aléatoire par une constante. Si l'on choisit un critère quadratique de quantification de l'écart entre S et une constante, alors celle-ci est égale à l'espérance $E[S]$: c'est la prime pure (cf. DENUIT et CHARPENTIER, 2004).

Dans l'approche fréquence-sévérité, nous avons recours à un modèle qui considère le nombre et le montant individuel de chaque sinistre pour un contrat donné. Cette approche peut être appliquée dans différents domaines en assurance IARD ainsi qu'en assurance maladie (habitation, automobile, responsabilité professionnelle, soins de santé, etc.).

Avec des notations usuelles, on définit une variable aléatoire discrète N représentant le nombre de sinistres survenus durant une période (annuelle par exemple). Le montant du k-ième sinistre est noté W_k. On a :

$$S = \sum_{k=1}^{N} W_k$$

On dit alors que S obéit à une loi composée. La v.a. N correspond à la fréquence, on la nomme aussi v.a. de comptage. Le montant d'un sinistre est appelé sévérité ou gravité. On suppose que les (W_k) sont i.i.d. et indépendants[1] de N, on notera W la variable parente de l'échantillon.

[1] En dehors de ce cadre favorable, cela relève de la recherche théorique.

On peut montrer (cf. annexe D) que :

$$\begin{aligned} E[S] &= E[N]E[W] \\ V[S] &= \underbrace{E[N]V[W]}_{aléa\ coûts} + \underbrace{E[W]^2V[N]}_{aléa\ nombres} \end{aligned}$$

La prime pure est définie comme étant la perte moyenne attendue, en l'occurrence $E[S]$. On remarque qu'elle est égale au produit entre la fréquence moyenne $E[N]$ et le coût moyen $E[W]$. Nous présentons plus en détail le modèle collectif en annexe D.

Il existe plusieurs façons de trouver la loi de S, les plus connues sont l'algorithme de Panjer et la *Fast Fourier Transform*. L'algorithme de Panjer est le plus couramment utilisé, il présuppose notamment que les probabilités de la loi de comptage satisfassent au critère de récurrence suivant : $p_{k+1} = \left(a + \frac{b}{k}\right) p_{k-1}$, où $p_k = \mathbb{P}[N = k]$ et $k \in \mathbb{N}^*$et $p_0 > 0$. On peut montrer que ce choix se restreint aux lois suivantes :

- Binomiale : $V[N] < E[N]$, on parle de sous-dispersion,
- Poisson : $V[N] = E[N]$, il y a équi-dispersion,
- Binomiale négative : $V[N] > E[N]$, c'est le cas le plus fréquent de la sur-dispersion.

Même si l'algorithme *Fast Fourier Transform* n'exige aucune forme particulière pour N, l'expérience montre que les actuaires cherchent souvent à calibrer le nombre de sinistres sur l'une de ces trois lois. Il en résulte que les lois de X sont appelées les lois binomiale composée, Poisson composée et binomiale négative composée.

C'est aussi la raison pour laquelle la régression de Poisson et la régression binomiale négative sont les plus souvent utilisées lors d'un modèle de fréquence via un *GLM*. Nous justifions par la même la propension naturelle à nous tourner vers une régression de Poisson dans la seconde partie du mémoire (cf. partie 2).

1.1.3. Chargement de sécurité

Pour comprendre pourquoi la prime pure n'est pas la prime chargée à l'assuré, il peut être intéressant de se pencher sommairement sur le modèle de Cramer-Lundberg en théorie de la ruine. Ce modèle décrit par un processus aléatoire l'état des réserves de la compagnie d'assurance. On considère le modèle en temps continu. Voici les notations et hypothèses :

- $(N_t)_{t\in\mathbb{R}^+}$ correspond au nombre de sinistres, il est modélisé par un processus de Poisson d'intensité λ constante,
- $(X_i)_{i\in\mathbb{N}^*}$ traduit les montants de sinistres, ils sont supposés i.i.d de variable parente X et de moyenne μ,
- Les processus $(N_t)_{t\in\mathbb{R}^+}$ et $(X_i)_{i\in\mathbb{N}^*}$ sont indépendants,
- $c > 0$ correspond au taux de prime, déterministe et constant,
- $u \geq 0$ correspond à la ressource initiale, déterministe et constante.

Les temps inter-sinistres notés $T_i = t_i - t_{i-1}$ sont des variables aléatoires i.i.d. de loi exponentielle de paramètre λ. On peut finalement écrire le processus de la charge cumulée de sinistres :

$$S_t = \sum_{i=1}^{N_t} X_i$$

Le processus de surplus, traduisant l'état des richesses de la compagnie à l'instant t, est défini de la façon suivante pour $t \in \mathbb{R}^+$:

$$\begin{cases} U_0 &= u \\ U_t &= u + c\,t - S_t \end{cases}$$

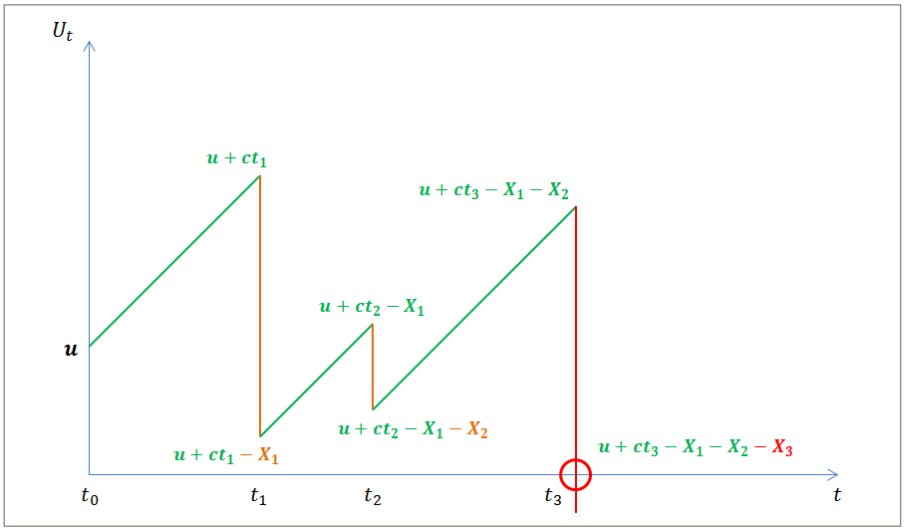


Figure 2 : Évolution de la richesse dans le modèle de Cramer-Lundberg

Sur le schéma ci-dessus, la ruine se produit en t_3. On comprend que la solvabilité de la compagnie sera d'autant plus garantie que u est grand. À horizon un an, les primes reçues sont égales à c et la perte attendue est $E[S_1]$. Il semble légitime de créer un coefficient de sécurité défini par la relation $c = (1+\eta)E[S_1]$, où le coefficient $\eta > 0$ permet de garantir qu'en moyenne la prime reçue sera supérieure à la perte attendue. Or :

$$\begin{aligned} E[S_1] &= E\left[\sum_{i=1}^{N_1} X_i\right] \\ &= E\left[E\left[\sum_{i=1}^{N_1} X_i \middle| N_1\right]\right] \\ &= E\big[N_1 E[X]\big] \\ &= E[N_1]E[X] \\ &= \lambda\,\mu \end{aligned}$$

On en déduit que le chargement de sécurité est défini par :

$$\eta = \frac{c - \lambda\,\mu}{\lambda\,\mu}$$

Il est en fait nécessaire d'avoir $\eta > 0$, autrement dit $c > \lambda\,\mu$. En effet, nous avons le lemme suivant :

si $c \leq \lambda\,\mu$ alors la ruine est presque sûre

Pour définir correctement la ruine, il est nécessaire de créer la variable concrétisant le temps de ruine :

$$\tau = \inf\{t \geq 0 : U_t < 0\}$$

avec la convention $\inf\{\emptyset\} = +\infty$. La probabilité de ruine en temps fini est donnée par :

$$\psi(u, \eta; t) = \mathbb{P}[\tau \leq t] = \mathbb{P}[\exists s \in [0, t] : U_s < 0]$$

De même, la probabilité de ruine en temps infini est donnée par :

$$\psi(u, n) = \lim_{t \to \infty} \psi(u, \eta; t) = \mathbb{P}[\tau < \infty] = \mathbb{P}[\exists s \geq 0 : U_s < 0]$$

Avec ces notations, le lemme que nous avons cité se réécrit :

$$\eta \leq 0 \quad \Longrightarrow \quad c \leq \lambda\, \mu \quad \Longrightarrow \quad \psi(u, n) = 1$$

En effet, on peut écrire que $E[S_t] = E[N_t]E[X] = t\, \lambda\, \mu$, de sorte que $E[U_t] = u + (c - \lambda\, \mu)t$.

On notera que si $\eta > 0$ alors la ruine reste bien sûr possible. De cette étude on comprend que la prime pure (matérialisée dans notre exemple par $\lambda\, \mu$) ne suffit pas et qu'il est indispensable d'ajouter un chargement commercial. Ce chargement est également censé couvrir les frais de gestion.

De plus, il faut bien comprendre que la prime dérivant d'un modèle, qu'il soit paramétrique ou non, n'est pas lissée. Imaginons par exemple que la prime réclamée pour les individus de moins de 25 ans soit deux fois plus grande que celle des plus de 25 ans. Pour des raisons commerciales, on préférera interpoler des fonctions continues satisfaisant certaines contraintes (un écart maximal de tarif entre deux profils différents, tarif maximal à ne pas dépasser, etc.). Des méthodes d'optimisation sous contraintes sont parfois mises en œuvre. Il faudra toutefois veiller à ne pas imposer des contraintes trop fortes sous peine de perdre la justesse et l'essence de nos estimations de primes pures.

Le lecteur intéressé pourra se reporter au chapitre 4 du livre de DENUIT et CHARPENTIER (2004), où sont notamment traitées les questions de choix de chargement (en lien avec les mesures de risque).

Dans la suite nous ne parlerons que de la prime pure.

1.2. PRÉSENTATION DE LA BASE DE DONNÉES

En assurance automobile, la tarification peut être liée aux caractéristiques propres du véhicule (puissance, vitesse maximale, ancienneté, etc.), à son usage, à la zone de circulation, au profil de l'assuré etc. Classiquement, voici listées ci-dessous de façon non exhaustive des variables que l'on peut retrouver en assurance auto.

VARIABLES	
Intérêts/Sorties	**Explicatives**
- Nombre de sinistres, - Montant des sinistres, - Etc.	- Police souscrite, - Mesure d'exposition, - Date de souscription, - Date d'expiration, - Adresse de l'assuré, - Age de l'assuré, - Type de véhicule, - Age moyen du véhicule, - Coût moyen du véhicule, - Lieu de parking, - Etc.

Les variables concernent différents domaines : la police souscrite, la résidence de l'assuré, la sinistralité, le véhicule, le conducteur etc. Il est important de rappeler que les variables relatives à la sinistralité passée de l'assuré ne peuvent être incorporées dans le cadre d'une tarification a priori. L'inclusion du passé de sinistres conduirait ici à une double pénalisation des assurés ayant déclaré un sinistre. En revanche, l'assureur pourra utiliser ces informations lors de la mise en place d'un système bonus-malus par exemple.

Il est aussi intéressant de noter que les caractéristiques personnelles se rapportent souvent au souscripteur de la police à défaut du conducteur effectif. Les conclusions des modèles seront dès lors toujours à nuancer.

Pour notre étude, voici la liste des variables explicatives à disposition :

1. COUVERTURE : type de couverture,
2. COUPE : le véhicule est un coupé (Oui/Non),
3. UTILITAIRE : le véhicule est un utilitaire (Oui/Non),
4. CAMION : le véhicule est un camion (Oui/Non),
5. 4X4 : le véhicule est 4 roues motrices (Oui/Non),
6. LUXE : le véhicule est une berline de luxe (Oui/Non),
7. CITADINE : le véhicule est une petite citadine (Oui/Non),
8. PUISSANTE : le véhicule est puissant (Oui/Non),
9. EPAVE : le véhicule est en mauvais état (Oui/Non),
10. NEUVE : le véhicule est neuf (Oui/Non),
11. AGE_VEHICULE : l'âge du véhicule,
12. TAILLE_MOTEUR : la taille du moteur,
13. CYLINDRE : le nombre de cylindres,
14. TRANSMISSION : le type de transmission (Automatique/Manuelle),
15. USAGE : l'usage du véhicule (Commercial/Privé/Business),
16. SUM_INS : la somme assurée,
17. FINANCE : si la voiture est autofinancée (Oui/Non),
18. SEXE : le sexe du conducteur (Homme/Femme/Inconnu),
19. JEUNE_COND : si le conducteur est jeune (Oui/Non),
20. NOTE_COND : une note attribuée au conducteur (de 0 à 9),
21. AGE_COND : l'âge du conducteur,
22. NCB : *no claims bonus*,
23. NOMBRE_COND : le nombre de conducteurs.

On souligne l'absence remarquée :

- D'une variable renseignant sur le lieu de vie et l'environnement dans lequel vit l'assuré. Est-ce à la campagne ou plutôt en ville que le véhicule va se déplacer ? Être garé ? Si la commune est très urbanisée, la fréquence de sinistre devrait être plus élevée car de nombreux véhicules sont en circulation mais les vitesses réduites devraient diminuer les montants. En conséquence on s'attend à une sinistralité plutôt attritionnelle en ville. La classification en fonction de ce type de variable relève du zonier, et ne sera pas développée ici faute de données exploitables.

- D'une variable d'exposition, correspondant en général au nombre de jours où la police a été en vigueur, elle permet de mesurer l'exposition au risque afin de pouvoir redresser les données. En effet, elle permet de tenir compte du fait qu'un sinistre déclaré pour une police exposée sur un mois est plus mauvais qu'un sinistre déclaré sur police annuelle. On pourrait

aussi utiliser le kilométrage parcouru par l'automobiliste. On fera ici l'hypothèse que l'exposition est de 1, et que le portefeuille est homogène et stable eu égard à ce critère.

- D'une variable relative à l'année, on supposera que les données sont relatives à une seule et unique année.

Voici la liste des variables d'intérêt :

1. CHARGE_SIN : le montant de la charge de sinistres,
2. NOMBRE_SIN : le nombre de sinistres obtenus.

Le coût total de sinistres correspond à la charge des sinistres pour l'année de la base. Il s'agit d'un coût total en euros que l'assuré a chargé à la compagnie d'assurance, intégrant les paiements, les réserves et les frais de gestion.

Précisons que les variables continues ou discrètes à nombreuses modalités sont bien souvent rassemblées en « paquets » ou *buckets* avant la modélisation. Nous ne détaillerons pas les méthodes permettant de procéder à cette segmentation, en effet elle n'est pas rigoureusement nécessaire dans les méthodes d'apprentissage. On pourra par exemple utiliser un modèle additif généralisé (GAM) pour catégoriser des variables continues ou à trop grand nombre d'issues (cf. DENUIT et CHARPENTIER, 2005). Ici, les *buckets* ont déjà été réalisés[1].

Bien sûr, et nous n'insisterons jamais assez, il faudra toujours nuancer les données à disposition. En effet le nombre de sinistres est bien celui « déclaré » et non pas le nombre réellement survenu. Il arrive parfois que l'assuré décide de ne pas déclarer un sinistre, pour ne pas subir un malus par exemple. L'interprétation des données se fait toujours via le prisme de l'aléa moral et l'antisélection.

1.3. ANALYSES PRÉLIMINAIRES

1.3.1. Données aberrantes

Avant de procéder à la modélisation de la prime pure, via le nombre et le coût des sinistres, des analyses préliminaires sont primordiales afin de bien connaitre la base de données et effectuer des retraitements.

En général, la phase d'extraction et de mise en forme des données constitue une étape capitale de l'étude et peut représenter une part considérable du temps consacré à l'étude. À titre d'exemple, les données manquantes ne sont pas à passer sous silence. Elles peuvent parfois révéler des indications précieuses sur l'assuré. On préfèrera donc ajouter des variables indiquant si des données sont manquantes ou non, puis de vérifier leur caractère aléatoire, et c'est seulement dans ce cas que l'on pourra les négliger. Dans notre base, un très grand nombre d'assurés n'a pas renseigné le sexe, et nous observons une sinistralité différente sur ces individus, cette information produit donc du sens.

Les assureurs sont de plus en plus exigeants quant à la constitution, la maintenance et la mise à jour de bases de données (en lien avec la notion de *big data*). La généralisation de la saisie par voie électronique abonde d'ailleurs en ce sens, limitant de fait les erreurs. Les données dont nous disposons ont été retraitées et nous ne détaillerons pas plus ici les méthodes.

[1] Nous aurons l'occasion de parler plus longuement des impacts du regroupement.

1.3.2. Sinistres graves

En assurance automobile, une tempête de grêle sur un parc auto, ou bien encore des dégâts corporels majeurs en responsabilité civile, peuvent générer de graves sinistres. En tarification, certains sinistres à faible fréquence et forte sévérité peuvent donc perturber l'estimation. En pratique, on préfère traiter à part les sinistres graves. Mais comment décider si un sinistre est grave ou non ? C'est la théorie des valeurs extrêmes qui apporte ces réponses.

On considère pour cela un échantillon i.i.d. de variables aléatoires $X_1, \dots, X_n$ de distribution parente F. Il s'agit d'estimer la queue de distribution de F. En pratique, voilà le genre de graphique que l'on peut obtenir, avec en bleu la densité théorique et en rouge les réalisations observées :

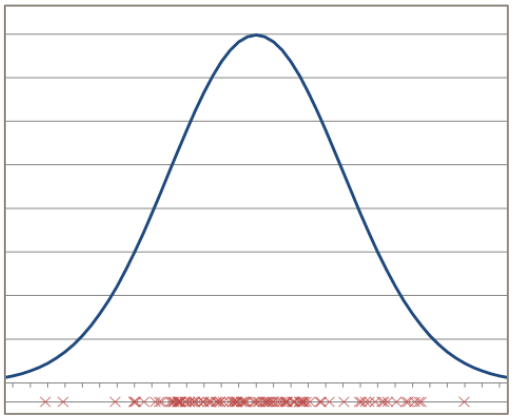


Figure 3 : Illustration de la théorie des valeurs extrêmes

On comprend à quel point il peut être difficile d'estimer cette queue de distribution quand si peu de points sont disponibles dans ce domaine. La théorie des valeurs extrêmes fournit des éléments mathématiques pour l'évaluation de lois, conditionnellement au fait que la variable dépasse un certain seuil μ. On s'intéresse, pour $x > 0$, à la fonction de survie de l'excès au-dessus de μ :

$$\begin{aligned} \bar{F}_\mu(x) &= \mathbb{P}[X - \mu > x | X > \mu] \\ &= \frac{\mathbb{P}[X > x + \mu, X > \mu]}{\mathbb{P}[X > \mu]} \\ &= \frac{\mathbb{P}[X > x + \mu]}{\mathbb{P}[X > \mu]} \\ &= \frac{\bar{F}(x + \mu)}{\bar{F}(\mu)} \end{aligned}$$

Le théorème de Pickands – Balkema – de Haan donne la forme de la loi limite pour cette fonction de survie de l'excès moyen. Dans le cas où $x^F = \sup\{x \in \mathbb{R} : F(x) < 1\} = +\infty$ on peut dire que si F satisfait aux critères de la théorie des valeurs extrêmes et appartient au domaine d'attraction déterminé par ξ, alors il existe une fonction positive $\beta(\mu)$ telle que :

$$\lim_{\mu \to \infty} \left(\sup_{x \geq 0} \left| \bar{F}_\mu(x) - \bar{G}_{\xi, \beta(\mu)}(x) \right| \right) = 0 \qquad (6)$$

Où $\bar{G}_{\xi,\beta(\mu)}$ est la fonction de survie d'une loi de Pareto généralisée (*GPD*) définie par :

$$G_{\xi,\beta(\mu)}(x) = \begin{cases} 1 - \left(1 + \xi \dfrac{x}{\beta(\mu)}\right)^{-\frac{1}{\xi}} & si\ \xi \neq 0 \\ 1 - \exp\left(-\dfrac{x}{\beta(\mu)}\right) & si\ \xi = 0 \end{cases}$$

$x \geq 0$ si $\xi \geq 0$ et $0 \leq x \leq -\frac{\beta(\mu)}{\xi}$ si $\xi < 0$

$\beta(\mu) > 0$ est un paramètre d'échelle

$\xi \in \mathbb{R}$ est un paramètre de forme

La propriété (6) laisse entendre que des distributions à queue lourde à droite sont proches de la distribution *GPD*. Nous avons donc pour μ grand :

$$\frac{\bar{F}(x+\mu)}{\bar{F}(\mu)} \sim \bar{G}_{\xi,\beta(\mu)}(x)$$

En faisant le changement de variable $x \longleftarrow x+\mu$:

$$\bar{F}(x) \sim \bar{F}(\mu)\bar{G}_{\xi,\beta(\mu)}\left(x-\bar{F}(\mu)\right)$$

Tout l'enjeu est donc de choisir un seuil μ à partir duquel l'hypothèse d'une loi *GPD* est satisfaisante. Généralement, μ est déterminé graphiquement en exploitant la linéarité de la fonction d'excès moyen pour la GPD. En effet nous savons (cf. EMBRECHTS, KLUPPELBERG et MIKOSCH, 1997) que pour une loi de Pareto généralisée $GPD(\xi,\beta)$ alors $E[X-\mu|X>\mu] = \frac{\beta}{1-\xi} + \frac{\xi}{1-\xi}\mu$.

En pratique, on observe donc le moment où la courbe du *mean excess plot* (représentation empirique de $E[X-\mu|X>\mu]$) devient affine pour pouvoir supposer une ressemblance avec une loi *GPD* et donc entrer dans le cadre de la théorie des extrêmes. Le seuil μ est choisi de cette façon, au-delà on peut considérer le sinistre comme étant grave.

Voici représenté le *mean excess plot* avec notre base de données :

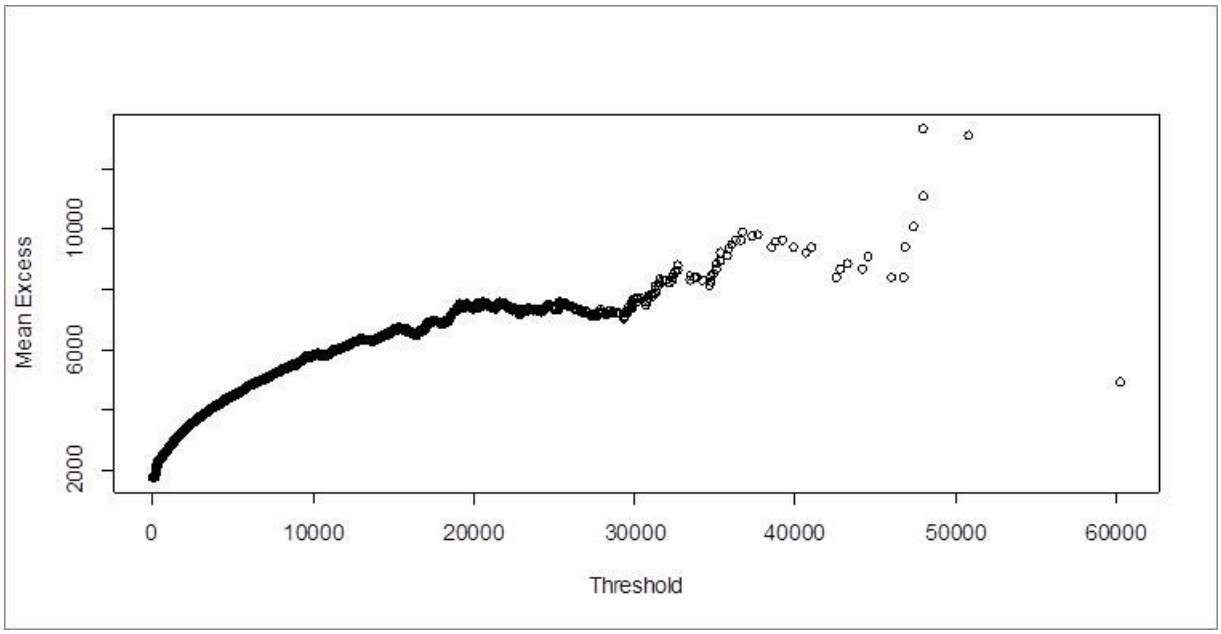


Figure 4 : Mean Excess Plot du montant de sinistres

On retiendra ici un seuil $\mu = 20\,000€$. On en décompte moins de $0{,}1\,\%$ dans la base, ils sont retirés (écrêtement). Fondamentalement, il faudrait leur appliquer un traitement à part mais au regard de leur faible nombre nous n'y reviendrons plus dans la suite.

1.3.3. Test d'indépendance du khi-deux

Un premier tri au sein des variables explicatives peut être opéré au moyen d'un test du khi-deux. Une variable intéressante à considérer est la déclaration ou non d'un sinistre. Pour toutes les variables explicatives, on évalue la statistique du khi-deux (H0 : la variable observée est indépendante de la variable « déclaration de sinistre Oui/Non ») et on donne la *p-value* du test associé. Si les *p-values* sont élevées, alors on peut retirer en amont ces variables de l'étude.

Voici les résultats obtenus :

RÉSULTATS DU TEST D'INDÉPENDANCE			
Statistique	**Degré de liberté**	**p-value**	**Variable**
X-squared = 1905,925	df = 7	p-value < 2,2e-16	COUVERTURE
X-squared = 11,0781	df = 1	p-value = 0,0008735	COUPE
X-squared = 0,1884	df = 1	p-value = 0,6643	UTILITAIRE
X-squared = 1,7123	df = 1	p-value = 0,1907	CAMION
X-squared = 381,7343	df = 1	p-value < 2,2e-16	4X4
X-squared = 23,5079	df = 1	p-value = 1,244e-06	LUXE
X-squared = 50,6456	df = 1	p-value = 1,106e-12	CITADINE
X-squared = 9,2987	df = 1	p-value = 0,002293	PUISSANTE
X-squared = 669,5913	df = 1	p-value < 2,2e-16	EPAVE
X-squared = 2,79	df = 1	p-value = 0,09485	NEUVE
X-squared = 1824,927	df = 18	p-value < 2,2e-16	AGE_VEHICULE
X-squared = 238,2736	df = 4	p-value < 2,2e-16	TAILLE_MOTEUR
X-squared = 151,2717	df = 3	p-value < 2,2e-16	CYLINDRE
X-squared = 1,2053	df = 1	p-value = 0,2723	TRANSMISSION
X-squared = 433,2142	df = 2	p-value < 2,2e-16	USAGE
X-squared = 1498,871	df = 17	p-value < 2,2e-16	SUM_INS
X-squared = 730,3749	df = 1	p-value < 2,2e-16	FINANCE
X-squared = 300,1863	df = 2	p-value < 2,2e-16	SEXE
X-squared = 255,9238	df = 1	p-value < 2,2e-16	JEUNE_COND
X-squared = 126,6329	df = 8	p-value < 2,2e-16	NOTE_COND
X-squared = 359,4087	df = 7	p-value < 2,2e-16	AGE_COND
X-squared = 1422,459	df = 9	p-value < 2,2e-16	NCB
X-squared = 448,9822	df = 4	p-value < 2,2e-16	NOMBRE_COND

On rejette l'hypothèse d'indépendance au seuil $5\,\%$ dans la plupart des cas, on comprend que les variables sont « liées ». En revanche nous identifions en bleu 4 variables qui sont jugées indépendantes de la survenance ou non d'au moins un sinistre : nous les retirons de notre étude. Voici donc le jeu de variables explicatives final que nous utilisons :

1. COUVERTURE : type de couverture,
2. COUPE : le véhicule est un coupé (Oui/Non),
3. 4X4 : le véhicule est 4 roues motrices (Oui/Non),
4. LUXE : le véhicule est une berline de luxe (Oui/Non),
5. CITADINE : le véhicule est une petite citadine (Oui/Non),
6. PUISSANTE : le véhicule est puissant (Oui/Non),
7. EPAVE : le véhicule est délabré (Oui/Non),
8. AGE_VEHICULE : l'âge du véhicule,
9. TAILLE_MOTEUR : la taille du moteur,
10. CYLINDRE : le nombre de cylindres,
11. USAGE : l'usage du véhicule (Commercial/Privé/Business),

12. SUM_INS : la somme assurée,
13. FINANCE : si la voiture est autofinancée (Oui/Non),
14. SEXE : le sexe du conducteur (Homme/Femme/Inconnu),
15. JEUNE_COND : si le conducteur est jeune (Oui/Non),
16. NOTE_COND : une note attribuée au conducteur (de 0 à 9),
17. AGE_COND : l'âge du conducteur,
18. NCB : *no claims bonus*,
19. NOMBRE_COND : le nombre de conducteurs.

Soit un total de **19 variables explicatives**. On peut alors évaluer la densité empirique du nombre et du montant de sinistres (sachant que l'on a un sinistre). Nous présentons ci-dessous les résultats :

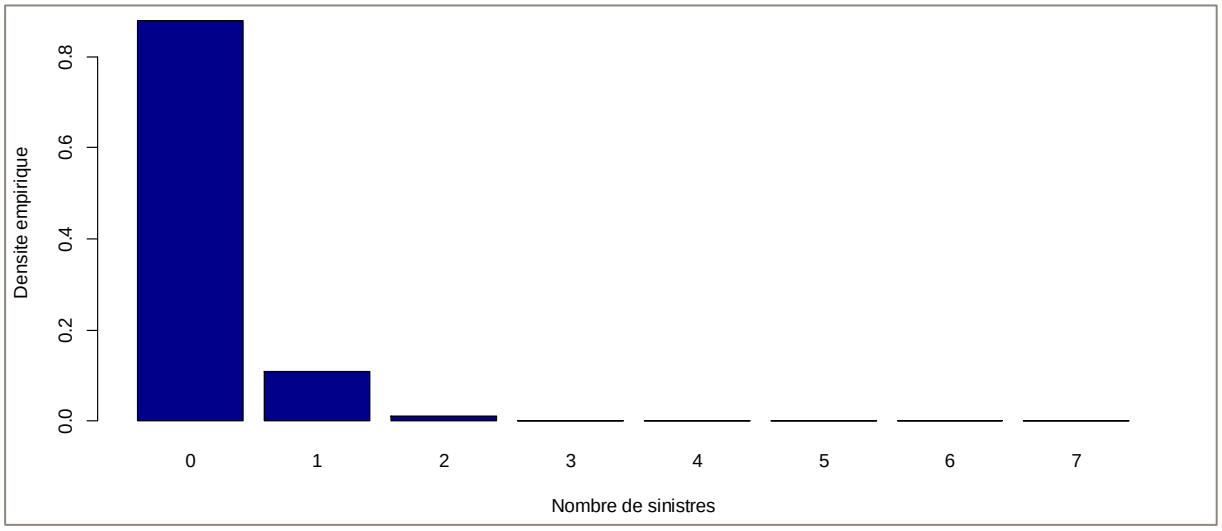


Figure 5 : Densité empirique du nombre de sinistres sur la base totale retenue

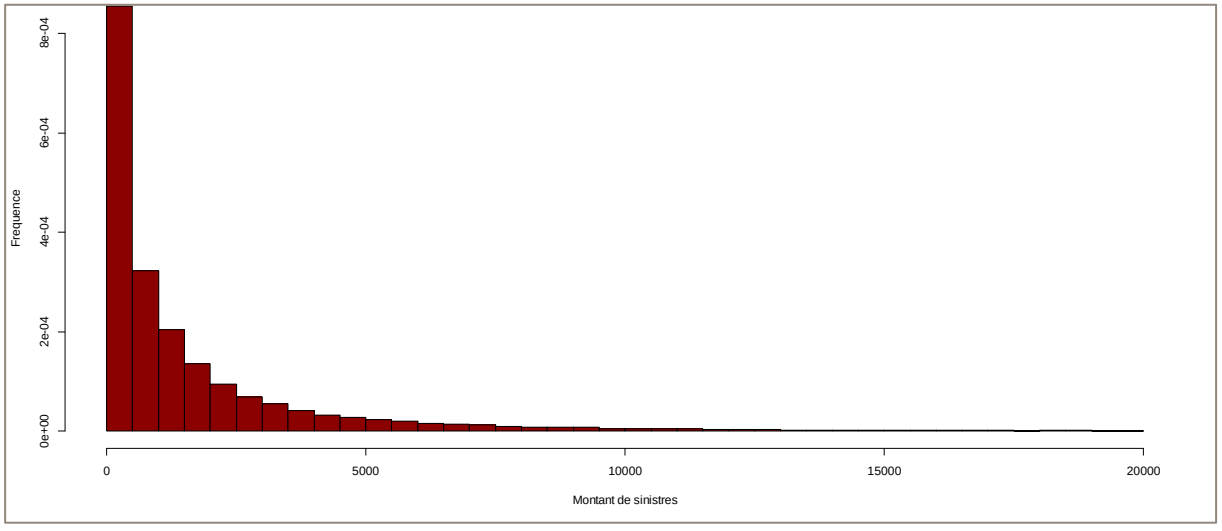


Figure 6 : Densité empirique du montant de sinistres sur la base totale retenue

On peut également estimer les moyennes empiriques :

- fréquence empirique moyenne : 13,41 %,
- coût moyen empirique : 1653€,
- **prime pure empirique : 222€.**

À noter que cette valeur de prime pure est obtenue sans aucune modélisation, c'est la prime la plus simple que l'on réclamerait à n'importe quel individu.

1.3.4. Analyses graphiques

On présente ci-dessous quelques histogrammes : répartition au sein du portefeuille, fréquences moyennes de sinistres observés et coûts moyens observés entre les différentes modalités de la variable explicative choisie.

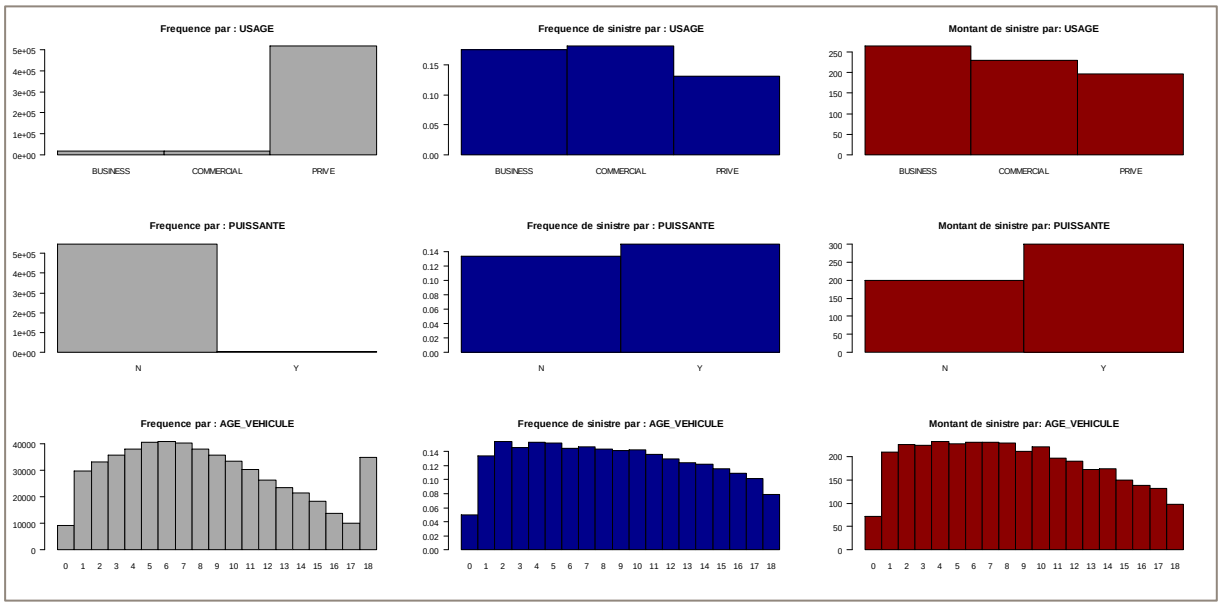


Figure 7 : Histogrammes sur des variables liées au véhicule

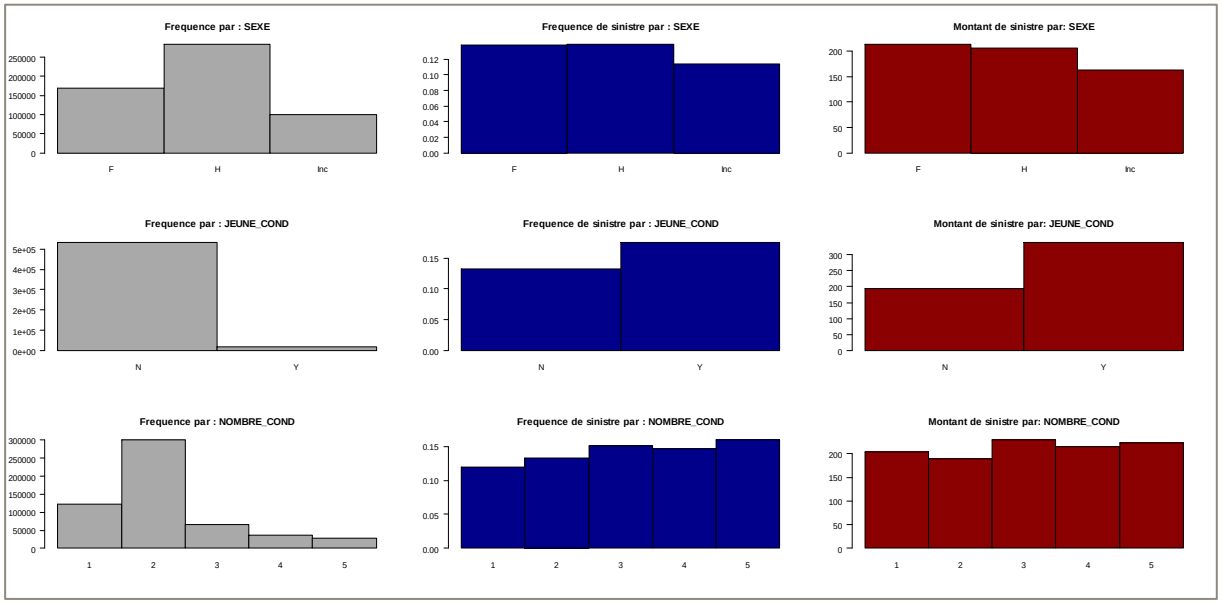


Figure 8 : Histogrammes sur des variables liées à l'assuré

On peut par exemple déduire de ces graphiques que, relativement à la variable JEUNE_COND, le nombre de sinistres sera plus élevé si l'individu est jeune. Pour la variable AGE_VEHICULE, on peut reconnaitre le regroupement de modalités que nous avons réalisé au-delà de 18 ans.

Avant d'adopter un modèle paramétrique ou de mettre en œuvre des méthodes d'apprentissage, il est souvent utile voire indispensable d'analyser les liens entre les données. Il existe plusieurs méthodes.

On peut citer l'analyse en composantes principales (ACP), l'analyse en composantes binaires (ACOBI), l'analyse factorielle de correspondances multiples (AFCM). Présentons la plus classique, à savoir l'ACP.

L'ACP permet d'obtenir des représentations et des réductions de l'information contenue dans des bases de données volumineuses. On note par M la matrice de la base de donnée, ainsi le coefficient m_{ij} correspond à la valeur de la j ième variable explicative ($j \in \{1, \dots, p\}$) pour le i ième individu ($i \in \{1, \dots, n\}$). Dès lors l'espace naturel pour représenter les données est $\mathbb{R}^p$, sous forme du nuage de points suivant :

$$\text{Pour } i = 1 \dots n, (x_{i1} \quad \dots \quad x_{ip})'$$

Dans l'analyse de données, on considère un second espace, en l'occurrence $\mathbb{R}^n$, dans lequel on trace ce nuage de points :

$$\text{Pour } j = 1 \dots p, (x_{1j} \quad \dots \quad x_{nj})'$$

Dans l'espace $\mathbb{R}^p$, les points représentent les p caractéristiques de l'individu i. Dans l'espace $\mathbb{R}^n$, ils représentent les valeurs prises par la variable j sur l'ensemble des n individus.

Il est difficile de se rendre compte de la forme des nuages de points dans les grandes dimensions, l'on préfère alors les projeter sur des droites ou sur des plans. On peut montrer que les sous-espaces de projections maximisant la distance, ou « l'inertie », sont engendrés par les vecteurs propres de la matrice de corrélation entre les variables explicatives. Ces vecteurs forment les axes factoriels. En pratique, on représente le cercle des corrélations qui à chaque point-variable associe un point dont la coordonnée sur un axe factoriel est une mesure de la corrélation entre cette variable et le facteur.

Voici le cercle obtenu sur les variables quantitatives :

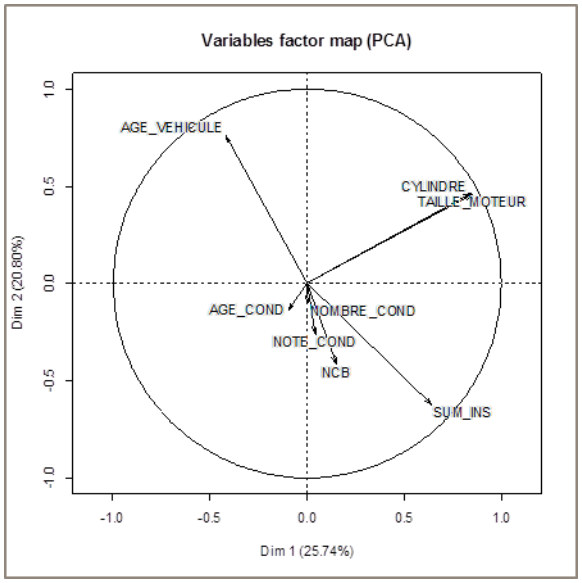


Figure 9 : ACP sur les variables explicatives quantitatives

On ne peut interpréter que les variables proches du cercle, seuls ces points sont bien projetés. On retiendra donc surtout ici que la taille du moteur est – sans surprise – étroitement liée au nombre de cylindres. Dans la suite (cf. partie 2.2.2), lors de l'utilisation de la régression de Poisson via un *GLM*, on s'apercevra que la variable cylindre est justement redondante avec la taille du moteur.

1.4. DÉCOUPAGE DE LA BASE

En assurance automobile, le nombre de sinistres mérite une attention toute particulière là où le coût des sinistres n'est habituellement pas sujet à une classification aboutie. La fréquence de sinistres concentre tout l'enjeu de la personnalisation a posteriori de la prime. En effet, les systèmes bonus-malus n'intègrent souvent que cette variable pour l'ajustement sur l'expérience de la prime.

L'analyse des coûts de sinistres est sensiblement plus complexe que celle de la fréquence. Là où tous les individus sont utilisés pour la modélisation du nombre de sinistres, l'on comprend que seules les polices sinistrées doivent être considérées lors de l'estimation du coût moyen, ce qui limite le nombre d'observations. En outre, l'influence de la duration est majeure pour les coûts.

En effet bien souvent les sinistres sont longs à être clôturés, introduisant une forme de latence dans le coût inscrit dans la base. Ce phénomène est très marqué dans le cas d'un accident corporel, où la victime peut nécessiter des soins sur une durée très longue. La duration des sinistres en RC auto peut ainsi être importante. Certains actuaires suggèrent d'ailleurs de considérer à part les sinistralités à développement long. Toujours en RC auto, et selon que l'assuré renverse une personne âgée ou un jeune homme, le coût ne sera pas le même pour l'assureur. Néanmoins l'on perçoit mal comment les variables explicatives peuvent capter ce phénomène. L'idée à retenir est que le coût des sinistres est nettement plus complexe. Il est expliqué par des données exogènes bien au-delà de la base de données.

C'est pour l'ensemble de ces raisons que **nous nous focaliserons dans la suite sur l'estimation du modèle de fréquence**, qui nécessite une analyse de toute la base là où le modèle de sévérité se limite aux lignes ayant eu un sinistre.

Pour rappel, nous souhaitons mettre en œuvre un *GLM* ainsi qu'un arbre *CART* complété par des méthodes ensemblistes. Si pour le *GLM* le découpage en une base d'apprentissage et une base de test est suffisant, ce n'est pas le cas des méthodes d'apprentissage. Nous en avions parlé en introduction :

- La base d'apprentissage permet de trouver les paramètres du *GLM* et d'apprendre les arbres de régression,

- La base de validation permet d'optimiser les paramètres des arbres,

- La base de test permet d'estimer les erreurs entre prédictions et les réalisations, tant pour le *GLM* et les méthodes d'apprentissage, ce sont ces erreurs que nous comparerons (cf. partie 5.2).

En proportion, voici la segmentation de la base totale de $n \sim 500\,000$ individus que nous avons choisie[1] :

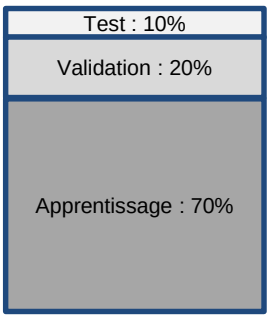


Figure 10 : Découpage de la base totale

Nous insistons sur le fait que, pour le moment, nous avons effectué des regroupements (*buckets*) au sein de certaines variables explicatives. Il est à noter que dans la suite nous présentons aussi des résultats sur une base de données qui n'a pas subi ces regroupements. En effet les arbres autorisent ce genre de liberté, là où les *GLM* présupposent ces *buckets*. Le découpage de la base est le même que ce soit sur les variables ayant subi des regroupements que sur les variables initiales.

Nous pouvons désormais nous pencher sur les méthodes d'estimation.

[1] Dans la suite et par abus, on nommera indifféremment n la taille de ces échantillons.

2. MODÈLE LINÉAIRE GÉNÉRALISÉ

Pendant longtemps, les actuaires se sont limités au modèle linéaire gaussien classique. Toutefois la complexité des problèmes statistiques les ont poussés vers des méthodes plus complètes, les modèles linéaires généralisés. À titre d'exemple, la régression de Poisson est devenue la méthode la plus répandue en tarification automobile.

On rappelle la segmentation classique que l'on retrouve lorsque l'on veut effectuer des régressions :

- **Variables à expliquer/réponses/endogènes** : Nombre de sinistres, montant de sinistres, probabilité d'avoir au moins un sinistre, etc.

- **Variables explicatives/prédictives/exogènes** : âge de l'assuré, sexe de l'assuré, Cadre Socio Professionnel de l'assuré, type de véhicule, zone géographique, etc.

Dans le cas de *GLM*, on peut présenter ce tableau récapitulatif **non exhaustif** :

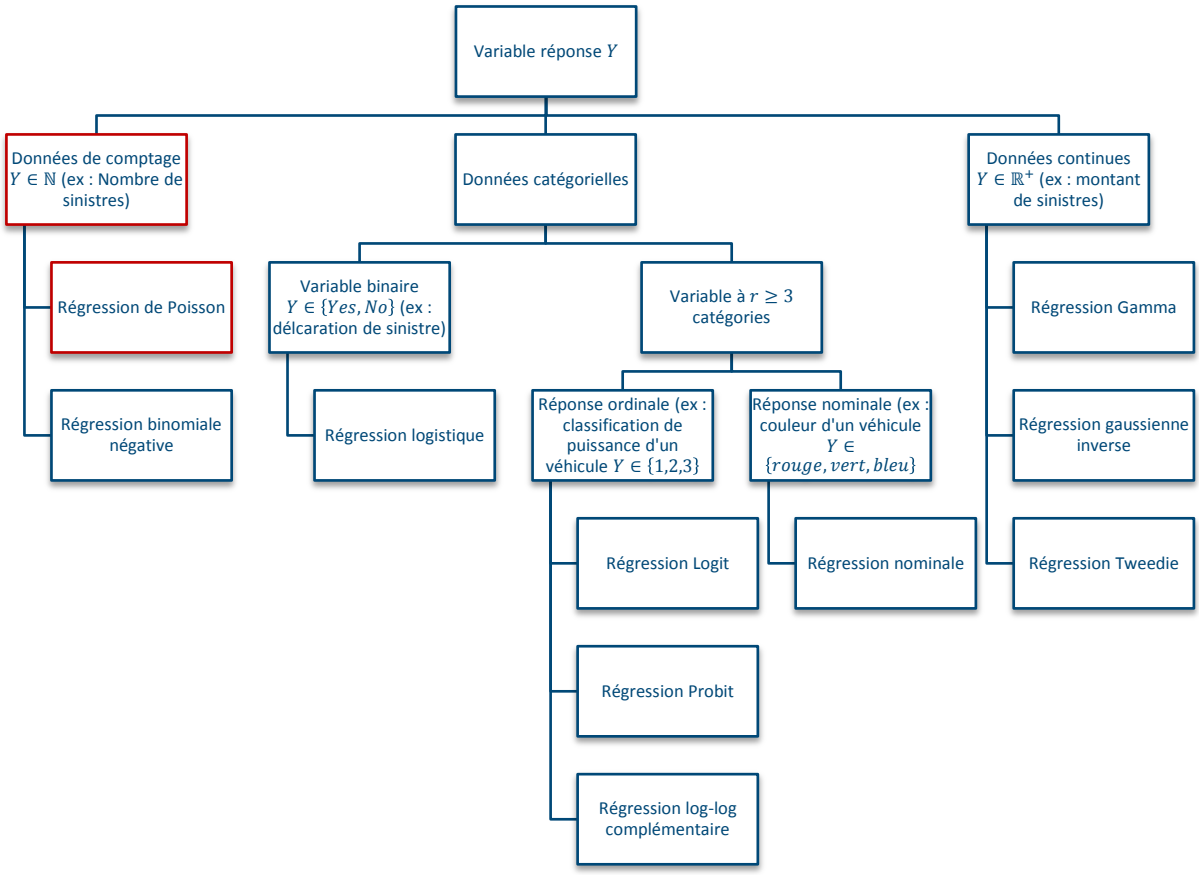


Figure 11 : Schéma classique de choix de régression pour un GLM

Nous avons mis en évidence en rouge la régression qui nous intéresse dans le cadre de l'estimation du modèle de fréquence.

2.1. DÉFINITION DU GLM

Le modèle linéaire généralisé se distingue en trois composantes (cf. DENUIT et CHARPENTIER, 2005).

2.1.1. Composante aléatoire

La variable à expliquer est $Y = (Y_1 \quad \dots \quad Y_n)'$ dont les densités appartiennent à la loi famille exponentielle. On dit que f_{Y_i} appartient à la loi famille exponentielle si et seulement si on peut trouver $\theta \in \mathbb{R}$ (paramètre canonique, ou de la moyenne), $\phi \in \mathbb{R}$ (paramètre de dispersion), a fonction définie sur $\mathbb{R}$ non nulle, b fonction définie sur $\mathbb{R}$ deux fois dérivable, c fonction définie sur $\mathbb{R}^2$, tels que :

$$f(y;\theta,\phi) = \exp\left\{\frac{y\theta - b(\theta)}{a(\phi)} + c(y;\phi)\right\}$$

La densité d'une loi de Poisson satisfait ces critères, elle est donnée par :

$$\begin{aligned} f(y) &= \frac{e^{-\mu}\mu^y}{y!} \\ &= \exp\{y\ln\mu - \mu - \ln(y!)\} \\ &= \exp\left\{\frac{y\theta - b(\theta)}{a(\phi)} + c(y;\phi)\right\} \end{aligned}$$

avec $\phi = 1$, $a(\phi) = 1$, $c(y;\phi) = c(y) = -\ln(y!)$, $\theta = \ln\mu$, $b(\theta) = \mu = e^\theta$

2.1.2. Composante déterministe

Soit $i = 1 \dots n$, alors pour chaque Y_i on dispose de la valeur d'un $p - uplet$ $(X_{1i} \quad \dots \quad X_{pi})'$, des p variables explicatives décrivant Y_i. Les vecteurs $X_j = (X_{j1} \quad \dots \quad X_{jn})'$ pour $j = 1 \dots p$ sont les vecteurs explicatifs.

2.1.3. Fonction de lien

C'est une fonction g déterministe strictement monotone définie sur $\mathbb{R}$ et telle que :

$$g_n\left(\underbrace{\mathbb{E}[Y]}_{\mu}\right) = \underbrace{\beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p}_{\eta:\, score} = X\beta$$

avec $g_n : \mathbb{R}^n \to \mathbb{R}^n, (x_1, \dots, x_n) \mapsto \big(g(x_1), \dots, g(x_n)\big)$.

On a avec nos notations :

$$g\left(\underbrace{\mathbb{E}[Y_i]}_{\mu_i}\right) = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} = x_i'\beta = \eta_i$$

Chacune des lois de probabilités de la famille exponentielle possède une fonction de lien spécifique, dite « canonique », et définie par $\theta = \eta$. Le lien canonique est tel que $g(\mu_i) = \theta_i$, or on sait que $\mu_i = b'(\theta_i)$ ainsi formellement $g^{-1} = b'$. La fonction de lien canonique d'une loi de Poisson est la fonction logarithmique.

Dans le contexte de notre base de données automobile, nous allons mettre en œuvre **la régression de Poisson et nous choisissons la fonction de lien canonique $g = \ln$**.

2.2. ESTIMATION DU MODÈLE DE FRÉQUENCE

2.2.1. Vérification de l'hypothèse de loi

On suppose que $Y \sim \mathcal{P}(\mu)$. Le modèle *GLM* associé s'écrit (avec la fonction de lien canonique $g = \ln$) :

$$g_n(\mu) = X\beta$$

Nous somme désireux de vérifier l'hypothèse de loi. Nous nous basons pour cela sur un *Q-Q plot*.
Le théorème de Glivenko – Cantelli (cf. annexe A) assure que :

$$\hat{F}_n^{-1} \to F_Y^{-1} \text{ presque sûrement quand } n \to \infty$$

Nous rappelons que nous disposons d'une réalisation $\{y_1, \dots, y_n\}$. Pour $\alpha \in]0,1[$, le tracé de la courbe de $\hat{F}_n^{-1}(\alpha)$ en fonction de $F_Y^{-1}(\alpha)$ devrait être une droite. En pratique, on trace donc :

$$\left\{\left(y_{(k)}, F_Y^{-1}\left(\frac{k}{n+1}\right)\right), k \in \{1, \dots, n\}\right\}$$

où la suite des $\left(y_{(k)}\right)_{k\in\{1,\dots,n\}}$ correspond à la suite des $(y_k)_{k\in\{1,\dots,n\}}$ rangés en ordre croissant[1]. On obtient le graphique suivant avec une loi de Poisson pour F_Y :

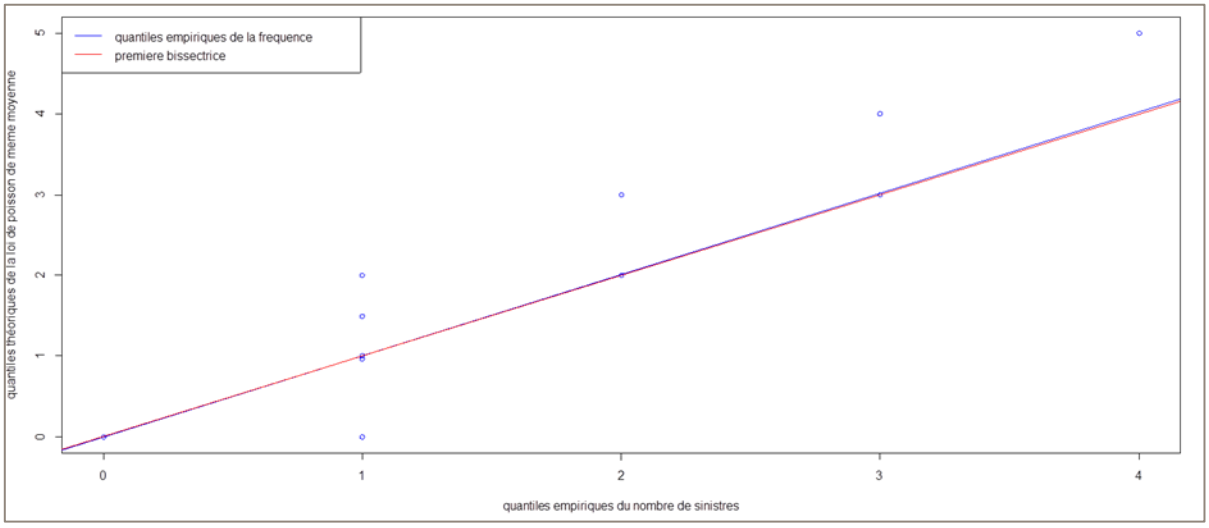


Figure 12 : Analyse graphique via un Q-Q plot du nombre de sinistres

On déduit que l'hypothèse d'une loi de Poisson est satisfaisante.

[1] Référence aux statistiques d'ordre.

2.2.2. Résultats

En déclarant les variables comme étant catégorielles, on fournit un aperçu de la sortie R :

```
factor(SUM_INS)35000       0.535768   0.051461  10.411  < 2e-16 ***
factor(SUM_INS)40000       0.573569   0.050484  11.361  < 2e-16 ***
FINANCEY                   0.126224   0.010202  12.372  < 2e-16 ***
SEXEH                     -0.040900   0.010284  -3.977 6.98e-05 ***
SEXEInc                   -0.208085   0.014301 -14.551  < 2e-16 ***
JEUNE_CONDY                0.247739   0.022965  10.788  < 2e-16 ***
factor(NOTE_COND)2        -0.045800   0.015216  -3.010 0.002612 **
factor(NOTE_COND)3        -0.017550   0.012435  -1.411 0.158134
factor(NOTE_COND)4         0.003281   0.015716   0.209 0.834619

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 223104  on 387081  degrees of freedom
Residual deviance: 218294  on 386992  degrees of freedom
AIC: 315129

Number of Fisher Scoring iterations: 6
```

Figure 13 : Aperçu de la sortie R d'une régression de Poisson avec toutes les variables

Les valeurs des paramètres sont obtenues par la méthode du maximum de vraisemblance présentée plus haut. On note que certaines variables ne sont pas significatives.

Toutefois le modèle de Poisson suppose l'équi-dispersion puisque pour une loi de Poisson la variance est égale à l'espérance. Une variable jugée pertinente dans un modèle de Poisson peut ne plus l'être si l'on prend en considération l'effet de sur-dispersion. La solution est de considérer le modèle de régression binomiale négative ou modèle de quasi-Poisson. Testons donc une régression de quasi-poisson, on observe la sortie suivante :

```
(Dispersion parameter for quasipoisson family taken to be 1.067269)

    Null deviance: 223104  on 387081  degrees of freedom
Residual deviance: 218294  on 386992  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 6
```

Figure 14 : Aperçu de la sortie R d'une régression de quasi-Poisson avec toutes les variables

Le paramètre de dispersion étant pratiquement égal à 1, on supposera que la régression de Poisson est suffisante, de même que l'on juge inutile de tester une régression binomiale négative. Toutefois, nous remarquons que de nombreuses variables ne sont pas significatives, nous proposons donc une procédure de sélection. Au passage, si l'on choisit les données non regroupées, alors on obtient le résultat suivant avec une régression de Poisson :

```
(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 223104  on 387081  degrees of freedom
Residual deviance: 219505  on 387054  degrees of freedom
AIC: 316216

Number of Fisher Scoring iterations: 6
```

Figure 15 : Aperçu de la sortie R d'une régression de Poisson sur des données sans bucket

On voit que le modèle est moins bon que lorsque que l'on a effectué quelques regroupements de modalités. Dans la suite, on considère comme acquis le fait que les données avec *buckets* soient plus adaptées au *GLM*. C'est donc sur la première régression de Poisson que nous effectuons une sélection de variables.

2.2.1. Calcul de la déviance

La log-vraisemblance de la loi est donnée page suivante.

$$\begin{aligned} l(Y,\mu) &= \ln L(Y,\mu) \\ &= \ln \prod_{i=1}^{n} \frac{e^{-\mu_i}\mu_i^{y_i}}{y_i!} \\ &= \sum_{i=1}^{n} -\mu_i + y_i \ln \mu_i - \ln(y_i!) \end{aligned}$$

Le modèle estimé est comparé au modèle dit « parfait » ou « saturé ». Dans notre modèle, la vraisemblance est fonction des observations Y et des prévisions $\hat{\mu}$: $L = L(Y, \hat{\mu})$. Dans un modèle saturé, nous avons plutôt $L_{sat} = L(Y, Y)$, il y a un complet *overfitting*.

Le modèle décrira bien les données lorsque la vraisemblance du modèle estimé est proche du modèle saturé, $L \cong L_{sat}$, a contrario l'estimation sera mauvaise si $L \ll L_{sat}$. La statistique du rapport de vraisemblance est donnée par :

$$\begin{aligned} \lambda &= \frac{L_{sat}}{L} \\ \ln \lambda &= \ln L_{sat} - \ln L \end{aligned}$$

La déviance réduite, ou normalisée, est définie par :

$$D = 2 \ln \lambda = 2(\ln L_{sat} - \ln L)$$

On jugera que le modèle est de mauvaise qualité si D est grand (statistique d'un test du khi-deux). La statistique permet également de comparer deux modèles entre eux.

La déviance d'une régression de Poisson est donnée par :

$$\begin{aligned} D &= 2(l_{sat} - l) \\ &= 2\big(l(Y,Y) - l(Y,\hat{\mu})\big) \\ &= 2\left(\sum_{i=1}^{n} -y_i + y_i \ln y_i - \ln(y_i!) - \sum_{i=1}^{n} -\hat{\mu}_i + y_i \ln \hat{\mu}_i - \ln(y_i!)\right) \\ &= 2\sum_{i=1}^{n} \left(\hat{\mu}_i - y_i + y_i \ln \frac{y_i}{\hat{\mu}_i}\right) \end{aligned} \qquad (7)$$

Nous utiliserons cette équation dans la partie 5, à titre indicatif.

2.2.2. Procédure de sélection de variables

Généralement, deux méthodes opposées sont citées pour sélectionner des variables :

- L'algorithme *forward* : méthode ascendante qui consiste à partir du modèle constant en ajoutant une à une des variables au modèle,
- L'algorithme *backward* : méthode descendante qui consiste à partir du modèle comprenant toutes les variables explicatives puis de les retirer une à une.

Pour comparer des modèles ayant des nombres de paramètres différents, et ainsi conserver le meilleur, on peut utiliser les critères suivants, où K est le nombre de paramètres à estimer :

- Critère d'Akaïke : $AIC = -2 \ln L + 2K$
- Critère de Schwartz : $BIC = -2 \ln l + K \ln n$

On préfèrera le modèle pour lequel ces critères ont la valeur la plus faible. Pour des bases de données de taille relativement faible, le BIC ne choisit pas toujours le « vrai » modèle car il a tendance à en choisir de trop simples en raison de sa plus forte pénalisation.

Dans notre contexte, on choisit le critère AIC avec une méthode *backward*. Voici les résultats obtenus :

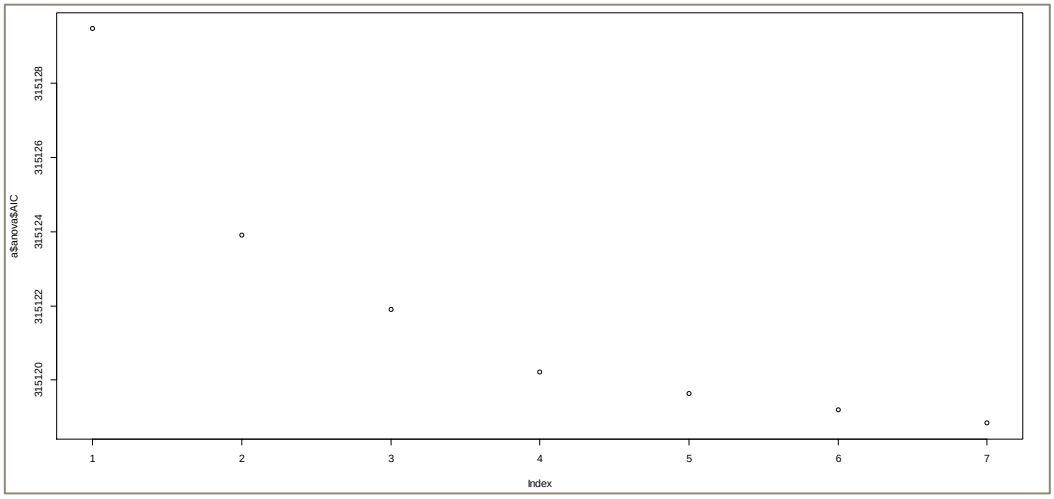


Figure 16 : Évolution du critère AIC en fonction du nombre de paramètres retirés

Bien sûr la courbe a été tronquée mais elle croit au-delà de 7. Il s'agit pour nous d'enlever donc 6 paramètres, qui sont les suivants :

	Step	Df	Deviance	Resid. Df	Resid. Dev	AIC
1				386992	218294.2	315129.5
2	- factor(CYLINDRE)	3	0.429587929	386995	218294.7	315123.9
3	- LUXE	1	0.000293617	386996	218294.7	315121.9
4	- PUISSANTE	1	0.320495961	386997	218295.0	315120.2
5	- EPAVE	1	1.415766726	386998	218296.4	315119.6
6	- COUPE	1	1.566676223	386999	218298.0	315119.2
7	- CITADINE	1	1.637283119	387000	218299.6	315118.8

Figure 17 : Variables à retirer du modèle de régression de Poisson

On peut alors effectuer une nouvelle régression en enlevant ces variables, le modèle est effectivement meilleur. On propose également de tester des interactions. Après quelques essais, c'est la triple interaction SEXE:JEUNE_COND:factor(CYLINDRE) qui est significative (on passe d'une erreur sur la base de test de 86,21766 à 86,21496). Dans la suite, on appellera « *GLM* sélectionné » ce modèle issu de l'algorithme backward avec la triple interaction.

2.3. LIMITATIONS

Le modèle *GLM* est dit paramétrique, en effet il nécessite de préciser une loi pour la variable d'intérêt $Y|X = x$. Ici nous avons choisi une loi de Poisson. Le modèle est de plus linéaire et donc l'impact des variables explicatives également. Pour relâcher cette hypothèse on pourra se tourner vers les modèles additifs généralisés (*GAM*).

Pour une même variable explicative donnée, les *GLM* sont incapables de modéliser des effets différents : le même coefficient est appliqué pour une variable continue. Il y a une forme de monotonie. On notera aussi que le traitement des valeurs atypiques ou manquantes est délicat.

Enfin, la modélisation des interactions entre les variables, bien que possible, relève souvent de l'avis d'expert, au même titre que leur sélection en amont. D'autant qu'une variable éliminée par une méthode basée sur l'AIC peut se révéler intéressante si couplée à une autre variable. Les modèles sont parfois longs à s'exécuter et l'on ne peut se permettre de tester toutes les interactions envisageables pour conserver le meilleur modèle.

Pour toutes ces raisons, nous nous tournons à présent vers de nouvelles méthodes non paramétriques, que sont les arbres de régression.

3. ARBRE DE CLASSIFICATION ET DE RÉGRESSION CART

3.1. PRÉSENTATION

3.1.1. Généralités

La technique des arbres de décision est fondée sur la classification d'un objet par une suite de tests sur les attributs qui le décrivent, ou variables explicatives. Ces tests sont organisés de façon hiérarchique, en ce sens que la réponse à un test influence les tests suivants, d'où la notion de récursivité à travers le terme d'arbres.

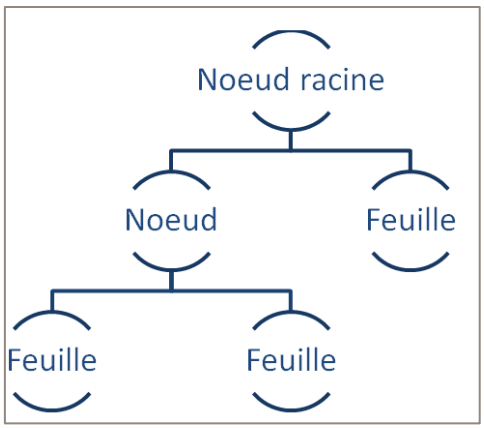


Figure 18 : Schéma simplifié d'un arbre binaire CART

Une feuille de l'arbre est associée à une classe d'individu et un nœud est associé à un test (une forme de sélection). Selon la réponse au test, on se déplacera vers tel ou tel fils du nœud. Le processus de classification est donc récursif, jusqu'à la rencontre d'une feuille (ou nœud terminal). Ce type de structure est qualifié d'arbre de décision.

Il est essentiel de rappeler d'emblée que la construction d'un arbre de décision passe par une phase d'apprentissage. En effet, la ramification d'un arbre est avant tout la traduction de l'expérience. La construction s'appuie sur un ensemble d'apprentissage et non pas sur une expertise.

Les arbres de décision constituent une famille majeure de méthodes de *data mining* – exploration de données – et se classent parmi les techniques d'apprentissage supervisé. Leur objectif est de répartir un ensemble d'individus en groupes les plus homogènes possibles du point de vue d'une variable ciblée, appelée variable à prédire, à partir de données portant sur d'autres variables, dites variables prédictives. Différentes techniques d'induction d'arbre existent, parmi lesquelles la méthode CHAID (Chi-Squared Automatic Interaction Detector) publiée par Gordon Kass en 1980, *CART* (Classification and Regression Trees) développée par Leo Breiman en 1984, ou encore la méthode C4.5, publiée par Ross Quinlan en 1993 en vue d'améliorer son premier algorithme ID3. Rappelons ici que **nous nous concentrons sur l'algorithme *CART***.

Les arbres de classification et de régression sont tous deux des méthodes d'estimation non paramétriques, et se distinguent par la nature de la variable d'intérêt.

- Si cette variable est qualitative, on veut prévoir la classe ou modalité à laquelle va appartenir la réponse, et on utilisera alors un arbre de classification.

- Si la variable d'intérêt est quantitative, on utilisera un arbre de régression. Mais le principe reste le même : déterminer des sous-groupes dans la population selon leurs variables explicatives et dont les valeurs de la variable d'intérêt sont aussi similaires que possible.

3.1.2. Une approche différente du GLM

Les arbres de classification et de régression (**C**lassification **A**nd **R**egression **T**ree) sont des méthodes non paramétriques, c'est-à-dire qu'elles ne nécessitent pas d'hypothèse sur la distribution des données. Cette approche non paramétrique est un gros avantage lorsque comparée à l'approche paramétrique *GLM*, car préciser une distribution pour les données est souvent difficile et très approximatif, ce qui conduit à des résultats faussés.

De nombreux modèles sont non linéaires et sont tout de même traités linéairement par les méthodes *GLM* pour plus de simplicité car il est difficile de préciser un modèle lorsque les relations ne sont pas linéaires et que le nombre de variables explicatives est important. Ainsi souvent, le « vrai » modèle est approximé par un modèle linéaire, or lorsque l'on veut prédire des données différentes de celles qui ont servi à construire le modèle, les prévisions risquent d'être très mauvaises. Les arbres sont des algorithmes non linéaires ce qui permet de relâcher cette hypothèse forte des *GLM*.

Les arbres ne sont pas monotones, c'est-à-dire qu'une même variable explicative continue peut avoir différents effets, par exemple s'il existe une valeur seuil en dessous de laquelle cette variable a un effet positif sur la variable d'intérêt et au-dessus un effet négatif, un nœud pourrait être construit en cette valeur seuil pour partager la population en deux groupes plus homogènes et partageant le même effet. A contrario le modèle *GLM* estime un unique β_j exprimant l'effet moyen de cette variable explicative X_j, ce qui conduit souvent à une mauvaise estimation si la variable a des effets différents selon ses valeurs.

De même les arbres, contrairement aux *GLM*, permettent de modéliser les multiples interactions entre les variables explicatives, et d'une certaine façon ce sont justement ces interactions qui permettent de créer les différents nœuds composant l'arbre pour créer des groupes d'observations plus homogènes.

La sélection des variables explicatives est un point important dans la construction des *GLM* pour obtenir un modèle à la fois performant et parcimonieux. Ce travail n'est pas nécessaire lorsque l'on utilise des arbres de classification ou de régression car les variables explicatives sont hiérarchisées selon l'importance de leur effet sur la variable d'intérêt dans la création des nœuds de haut en bas de l'arbre.

Les problèmes de données manquantes sont facilement résolus lorsque l'on travaille avec des arbres : s'il manque certaines variables explicatives, l'on peut tout de même obtenir une estimation de la variable d'intérêt à partir des variables explicatives renseignées en descendant l'arbre jusqu'au dernier nœud applicable.

Enfin, la facilité d'interprétation des arbres comparée aux autres méthodes de modélisation statistique est un autre avantage, qui explique la popularité de la méthode *CART*.

Ces solutions apportées par *CART* aux limites des modèles linéaires généralisés en font donc une bonne alternative dans de nombreuses situations. C'est la raison pour laquelle nous allons désormais présenter en détail cette méthode. Nous recommandons au lecteur de se référer à l'annexe E pour suivre le cheminement complet de construction d'un arbre sur des données simples.

3.2. CONSTRUCTION DE L'ARBRE SATURÉ

3.2.1. Principe

Un arbre de classification ou de régression (cf. THERNEAU et ATKINSON, 1997, cf. BERK, 2004 et cf. PAGLIA et PHELIPPE-GUINVARC'H, 2010) débute à partir d'une racine ou nœud initial puis se

divise en deux branches conduisant à deux nouveaux nœuds qui se divisent (ou non) en de nouvelles branches et donc à des nouveaux nœuds, etc.

Les nœuds terminaux, aussi appelés feuilles, sont situés en bas de l'arbre et regroupent des ensembles homogènes d'observations, c'est-à-dire des observations partageant des combinaisons de modalités de variables explicatives ayant un effet commun sur la variable d'intérêt et ainsi ayant des valeurs pour la variable d'intérêt aussi homogènes que possible. Ces feuilles donnent une estimation de la variable d'intérêt sachant certaines valeurs des variables explicatives.

La construction d'un arbre binaire consiste à déterminer une séquence de nœuds. En effet, on commence à la racine ou nœud initial contenant l'ensemble de l'échantillon que l'on divise selon une partition des modalités d'une variable explicative en deux classes pour obtenir deux sous-ensembles de l' échantillon les plus homogènes possibles.

Ces deux sous-ensembles sont des nouveaux nœuds que l'on va chercher à diviser en deux nouveaux sous-ensembles. La division se fait :

- selon une partition des modalités d'une variable explicative autre que celle qui a déjà servi pour la première division,
- ou alors selon une partition du sous-groupe de modalités créé lors de la première division.

Nous obtenons alors quatre nœuds terminaux que l'on peut encore diviser en deux sous-groupes, etc.

L'arbre est **élaboré sur la base d'apprentissage**.

3.2.2. Fonctions d'hétérogénéité

Les fonctions d'hétérogénéité sont des fonctions positives qui croient en fonction de l'hétérogénéité des valeurs prises par la variable d'intérêt au sein d'un même nœud. Ces fonctions diffèrent selon la nature de la variable d'intérêt.

Variable d'intérêt discrète

Définition : *Une fonction d'hétérogénéité pour une variable d'intérêt discrète à k modalités, est une fonction $i: I_k \to \mathbb{R}^+, (p_1, \dots, p_k) \longmapsto i(p_1, \dots, p_k)$, avec $I_k = \{(p_1, \dots, p_k) \in [0,1]^k : \sum_{i=1}^k p_i = 1\}$, vérifiant les propriétés suivantes :*

- *i admet un unique maximum en $\left(\frac{1}{k}, \dots, \frac{1}{k}\right)$,*
- *i admet k différents minima en chaque $(e_i)_{i=1\dots k}$ de la base canonique de $\mathbb{R}^k$,*
- *i est une fonction symétrique en les $p_1, \dots, p_k$, i.e. invariante par permutation des variables.*

Elle s'évalue pour un nœud N représentatif d'une séparation d'une variable à k modalités, on note donc volontiers $i(N)$ son évaluation.

Les trois fonctions d'hétérogénéité les plus utilisées sont l'indice de Gini, la fonction d'entropie, et l'erreur de Bayes. Pour un nœud N et une variable d'intérêt discrète avec k modalités notées $\{m_1, m_2, \dots, m_k\}$, ces 3 fonctions sont toutes de forme additive, autrement dit on peut écrire :

$$\begin{cases} g: [0,1] \to \mathbb{R}^+ \\ p_{j,N} = \mathbb{P}[Y = m_j | N] \\ i(N) = \sum_{j=1}^{k} g(p_{j,N}) \end{cases}$$

où $p_{j,N}$ représente la probabilité que la variable d'intérêt soit égale à m_j (la j-ième modalité) dans le nœud N, i.e. sachant que les variables explicatives prennent des valeurs qui respectent les conditions imposées par le nœud. On pourrait écrire $p_{j,N} = \mathbb{P}[Y = m_j | X \in N]$.

Voici les 3 fonctions couramment choisies pour g :

- Indice de Gini : $g(p) = p(1-p)$,
- Fonction d'Entropie : $g(p) = -p \ln p$,
- Erreur de Bayes : $g(p) = \min(p, (1-p))$.

En pratique, la probabilité $p_{j,N}$ est estimée par $\frac{n_{j,N}}{n_N}$ où $n_{j,N}$ est l'effectif dans le nœud de la modalité m_j et n_N l'effectif total du nœud N.

Ainsi pour mesurer la réduction hétérogénéité lors de la division d'un nœud N en deux nœuds fils gauche N_G et droit N_D, on applique la formule suivante :

$$\Delta = i(N) - \left(\mathbb{P}[N_G] i(N_G) + \mathbb{P}[N_D] i(N_D)\right)$$

où $\mathbb{P}[N_x]$ est la probabilité pour une observation du nœud N de tomber dans le nœud N_x, estimé par $\frac{n_{N_x}}{n_N}$, avec $x = G$ ou D.

On a ainsi l'estimateur de Δ :

$$\widehat{\Delta} = \sum_{j=1}^{k} g\left(\frac{n_{j,N}}{n_N}\right) - \left(\frac{n_{N_G}}{n_N} \sum_{j=1}^{k} g\left(\frac{n_{j,N_G}}{n_{N_G}}\right) + \frac{n_{N_D}}{n_N} \sum_{j=1}^{k} g\left(\frac{n_{j,N_D}}{n_{N_D}}\right)\right)$$

Variable d'intérêt continue

Définition : *La fonction d'hétérogénéité pour une variable continue est la variance intra-nœud, formellement nous avons :*

$$\Delta = n_N \{V[Y|N] - (\mathbb{P}[N_G] V[Y|N_G] + \mathbb{P}[N_D] V[Y|N_D])\}$$

Notons que nous avons ici n_N en facteur, cela simplifie la formule de $\widehat{\Delta}$ tout en ne modifiant pas la division le maximisant. On obtient en effet :

$$\widehat{\Delta} = \sum_{i=1, i \in N}^{n_N} \left(y_i - \bar{y}(N)\right)^2 - \left(\sum_{i=1, i \in N_G}^{n_{N_G}} \left(y_i - \bar{y}(N_G)\right)^2 + \sum_{i=1, i \in N_D}^{n_{N_D}} \left(y_i - \bar{y}(N_D)\right)^2\right)$$

où $\bar{y}(N)$ est l'espérance empirique de Y sachant le nœud N. Nous notons abusivement $i \in N$ pour signifier que l'on ne considère que les indices i tels que les variables explicatives X_i aient des

modalités appartenant au nœud. On parlera indifféremment de réduction de MSE ou de réduction déviance[1].

3.2.3. Algorithme récursif de création d'un nœud

La division d'un nœud revient à maximiser la réduction d'hétérogénéité, on cherche donc à tester tous les cas envisageables pour N_G et N_D afin de trouver la séparation s^* qui maximise $\widehat{\Delta}$.

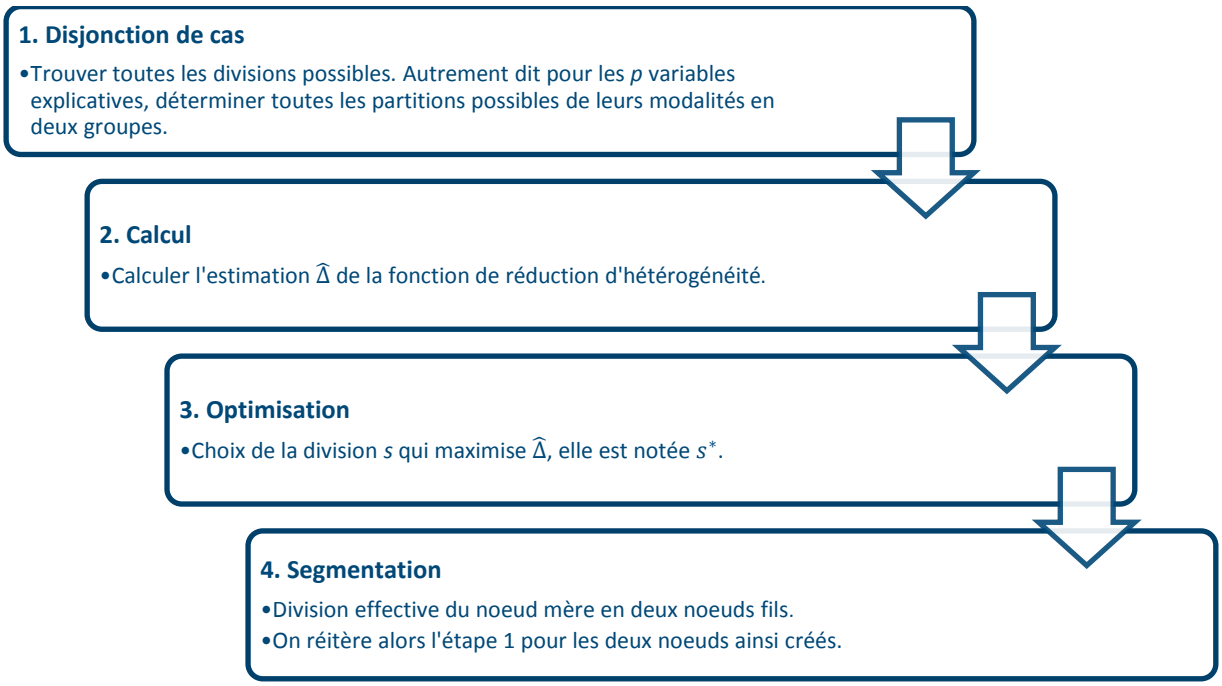


Figure 19 : Algorithme de création d'un arbre

Il s'agit donc de répéter ce schéma pour les deux nouveaux nœuds créés en utilisant seulement le sous ensemble, de variables explicatives et modalités, généré par la division. La division optimale s^*, qui correspond au choix d'une variable explicative et d'une partition de ses modalités, forme alors le premier nœud et conduit à de nouveaux nœuds, appelés nœuds fils.

On répète cette opération jusqu'à obtenir un profil par feuille (nœud terminal), et on obtient ainsi un arbre appelé **arbre saturé**, nous y revenons plus bas.

3.2.4. Séparation selon le type de variable

Variable explicative purement qualitative

Le nombre de divisions possibles s en deux sous-ensembles non vides de modalités peut être très grand, en effet pour une variable explicative discrète non ordinale à k modalités, il existe $2^{k-1}-1$ partitions différentes de ces modalités.

Si par exemple, notre variable explicative est une couleur prenant comme modalité {rouge, vert, bleu}, il y a donc 3 partitions possible pour cette variable :

- {rouge, vert} et {bleu}

[1] Ce terme de déviance est utilisé par les codes R mais est à différencier de la notion de vraisemblance.

- {vert, bleu} et {rouge}
- {rouge, bleu} et {vert}

Variable explicative ordinale

Pour une variable discrète ordinale à k modalités, l'ensemble des divisions possibles en deux sous-ensembles non vides est au nombre de $k-1$, car on peut ordonner les modalités. Ainsi pour l'âge qui est une variable discrète ordinale, si on a les observations {1, 2, 3, 5} par exemple, on va considérer comme divisions possibles :

- supérieur ou inférieur à 1,5, donc {1} et {2, 3, 5},
- supérieur ou inférieur à 2,5 donc {1, 2} et {3, 5},
- supérieur ou inférieur à 4 donc {1, 2, 3} et {5}.

Variable explicative continue

Pour une variable continue, le principe de division en sous-groupes de modalités est le même que pour une variable discrète ordinale sauf que le nombre de modalités est en général égal au nombre d'observations n, ce qui fait donc $n-1$ possibles partitions pour une variable continue.

3.2.5. Obtention de l'arbre saturé

Un arbre saturé est le résultat de l'algorithme Figure 19 itéré jusqu'à ce que l'on ne puisse plus segmenter. C'est l'arbre étendu au maximum qui possède une feuille pour chaque profil, c'est-à-dire pour chaque combinaison de modalités des variables explicatives contenue dans les données d'apprentissage. Les valeurs associées à ces feuilles sont les moyennes empiriques et conditionnellement au profil.

Insistons sur le fait qu'en pratique, on instaure un critère d'arrêt à l'algorithme. On impose souvent que le nombre d'observations y_i à disposition dans le nœud fils, et participant au calcul de la moyenne, soit supérieur à un certain nombre donné. Ainsi un arbre saturé à 100 observations par feuille est l'arbre obtenu en itérant le procédé de segmentation jusqu'à ce que l'on ait plus que 100 observations par feuille. Il faut donc réaliser un arbitrage entre l'homogénéité des groupes (critère de précision) et la complexité de l'arbre, liée à sa taille et donc aux effectifs dans chaque sommet (critère de support).

Notons que si ce critère n'est pas imposé, l'arbre saturé obtenu n'aura pas forcément une seule observation par feuille. Prenons un exemple simple :

BASE DE DONNÉES SIMPLE	
X_1 (qualitative)	Y (continu)
rouge	5
rouge	3
bleu	5
vert	10
bleu	4
vert	5

On obtient l'arbre saturé suivant (construit « à la main » et à l'aide de R) donné page suivante.

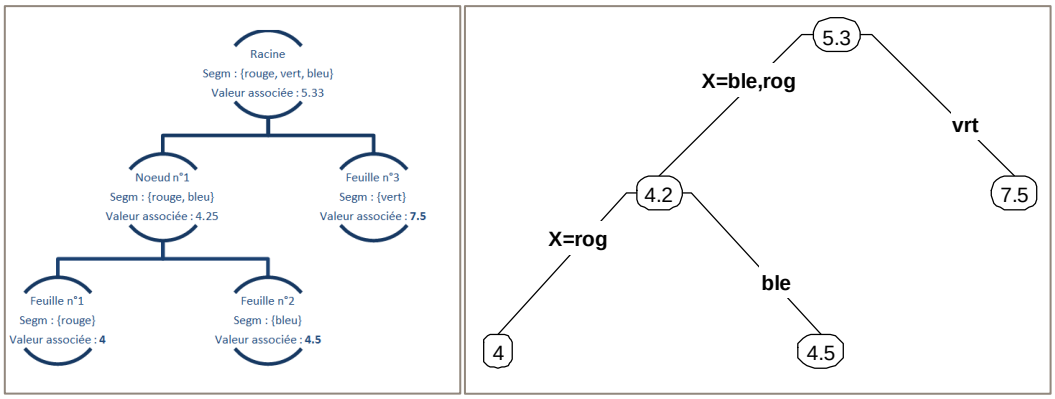


Figure 20 : Exemple d'un arbre CART saturé sans critère sur le nombre d'observations

On notera qu'il y a ici 2 observations par feuille. L'arbre s'interprète de la façon suivante : si un nouvel individu ayant la modalité {rouge} rejoint la base, alors il se verra attribuer la valeur 4.

Pour notre base automobile, voici l'arbre saturé obtenu sur la base d'apprentissage avec *buckets*.

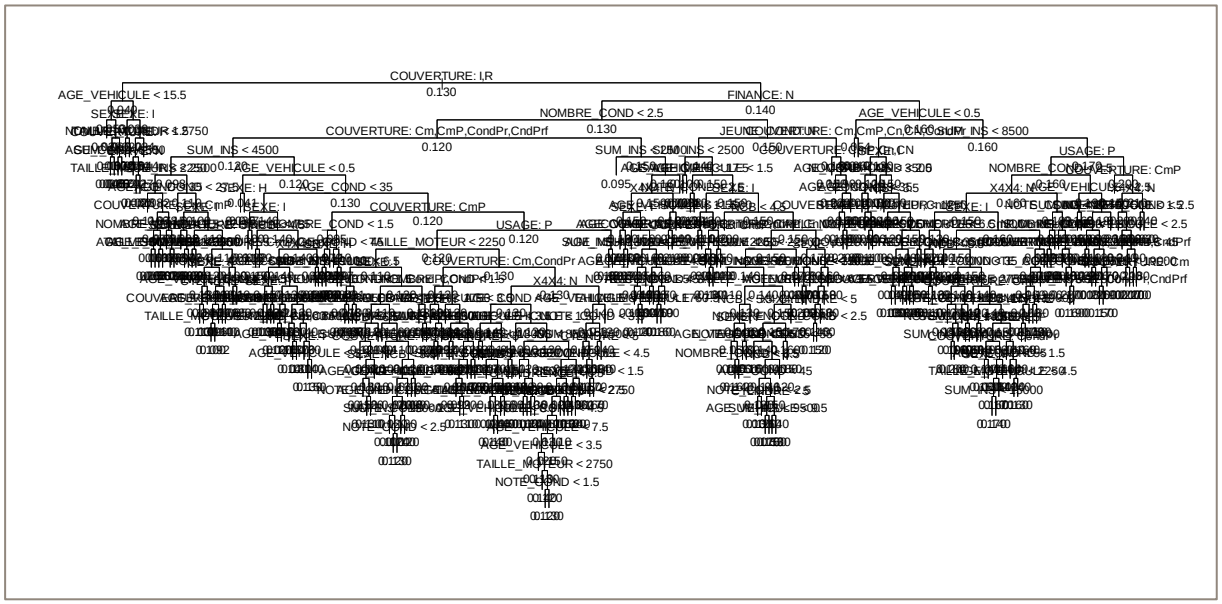


Figure 21 : Arbre saturé à 1000 observations par feuille

Il a été obtenu en imposant un nombre d'observations par feuilles minimal de 1000. Le problème est qu'il est illisible et « sur-appris ». Il n'est pas robuste eu égard à la base de validation ou la base de test. Il faut trouver une méthode pour le rendre optimal.

3.3. CRITÈRE D'OPTIMALITÉ

3.3.1. Principe

On utilise souvent le terme déviance pour déterminer la qualité de l'arbre ou comme fonction d'hétérogénéité, le terme déviance (comme utilisé dans la bibliothèque TREE de R ou dans la littérature) ne correspond pas à un calcul de déviance comme cité pour les modèles GLM (le double de la différence entre la log vraisemblance du modèle saturé et celle du modèle). Elle correspond en fait à la somme des carrés des écarts à la moyenne (MSE) quand utilisée avec une variable d'intérêt continue ou à la fonction d'entropie quand utilisée avec une variable d'intérêt discrète. Le lecteur pourra se reporter à l'introduction pour la justification de ce choix de l'erreur (équivalence (4)). Les erreurs sont calculées sur une autre base que celle d'apprentissage.

Le fait que les arbres soient des modèles statistiques non paramétriques, c'est-à-dire que l'on ne précise aucune loi pour Y, rend impossible le calcul de la vraisemblance et donc de la déviance. Néanmoins dans la partie 5 sur l'analyse des résultats, on utilisera la déviance d'une régression de Poisson donnée en équation (7) pour comparer les méthodes d'apprentissage avec le *GLM* (à titre purement indicatif).

Les performances des arbres de décision dépendent essentiellement de leur taille. Celle-ci est très importante car construire un arbre trop grand et donc très spécialisé peut conduire à des problèmes de sur-apprentissage et donc à des nœuds terminaux (ou feuilles) instables, car très dépendants des données d'apprentissage. On cherche à obtenir des arbres plus parcimonieux et donc plus robustes.

Il est facile de construire un arbre à partir de n'importe quel échantillon mais il est bien plus difficile de déterminer quel arbre choisir parmi tous ceux que l'on peut construire. Déterminer l'arbre optimal revient à trouver la taille optimale.

L'arbre saturé est celui qui donne les meilleures estimations sur la base d'apprentissage (erreur minimisée entre données prédites et valeurs observées) mais fera sans doute preuve d'un fort sur-apprentissage, et estimera mal des profils qui ne sont pas dans les données d'apprentissage. Il est donc capital de contrôler la taille de l'arbre pour obtenir les meilleures estimations possibles sur des données nouvelles.

Il existe deux techniques de contrôle de la taille :

- a priori : on impose certains critères avant de construire l'arbre que l'on appliquera lors de la construction, c'est-à-dire lors de la création de chaque nœud,

- a posteriori : cette méthode s'applique après la construction de l'arbre saturé. On le construit puis on remonte cet arbre en supprimant certains nœuds et donc couple de feuilles. On parle d'élagage. Celui-ci ne donne pas toujours l'arbre optimal mais tend à construire le meilleur arbre possible à partir des données disponibles.

3.3.2. Critère a priori

L'arbre saturé est l'arbre que l'on obtient si aucune restriction n'est imposée lors de la construction de l'arbre. En effet avant la création de celui-ci, on peut régler plusieurs paramètres pour limiter son développement : un nombre minimum d'observations par feuille, un nombre minimum d'observations dans un nœud pour qu'il soit divisé, ou une réduction minimale de l'hétérogénéité pour valider une division de nœud.

Le critère de réduction minimale de l'hétérogénéité peut être fixé arbitrairement : une première méthode consiste à imposer un certain pourcentage (par défaut 1 %). Ce pourcentage est la proportion minimale de l'hétérogénéité totale de l'arbre que doit produire la division. Ainsi lorsque la division optimale s^* a été choisie, on calcule l'hétérogénéité de l'arbre avant et après la division, et la différence des deux doit être supérieure à l'hétérogénéité de l'arbre avant division pondérée du coefficient.

Ce critère peut être quantifié par un test, par exemple pour une variable continue, on peut utiliser un ANOVA F test qui est de la forme : $F = \frac{\frac{(SSE_P - SSE_F)}{df_P - df_F}}{\frac{SEE_F}{df_F}}$, où ici SSE_P est l'hétérogénéité (variance intra pondérée) du nœud à diviser et SSE_F la somme de l'hétérogénéité des deux nœuds fils. Notons que

la grandeur df_P représente le nombre de degrés de liberté. Il est ici égal au nombre de feuilles, d'où $df_P = n - k$ où k est le nombre de feuilles, et $df_F = n - k - 2$. D'autres tests sont disponibles.

Ces critères sont pratiques car ils conduisent à des arbres plus parcimonieux et donc plus robustes, cependant ils peuvent aussi conduire à des arbres non optimaux.

En effet, par exemple en imposant une réduction minimale de l'hétérogénéité lors de la division d'un nœud, on peut empêcher la division d'un nœud qui ne va lui-même pas conduire à une forte réduction de l'hétérogénéité mais qui aurait conduit à des futurs nœuds performants.

Ce problème est une raison de l'utilisation de l'élagage, qui est une technique de contrôle de la taille de l'arbre a posteriori.

3.3.3. Critère a posteriori (élagage)

Le principe est de développer l'arbre au maximum, c'est-à-dire de construire l'arbre saturé puis de le remonter en partant des feuilles et de supprimer les nœuds dont la division n'améliore pas significativement l'arbre sur une base de validation. Réaliser cette sélection en partant du bas de l'arbre permet d'éviter le problème cité précédemment lorsque l'on impose une réduction minimale. En ce sens la méthode a posteriori est plus satisfaisante.

Il faut bien noter que l'arbre saturé est construit sur la base d'apprentissage puis ensuite élagué à l'aide d'une base de validation. Cette base de validation, indépendante, permet d'éviter le sur-apprentissage en autorisant un certain recul par rapport aux données.

Plus formellement un arbre T se traduit par une fonction $\phi_T: \mathbb{R}^p \to \mathbb{R}$ qui à un vecteur x de variables explicatives de taille p associe la valeur de sortie $y = \phi_T(x)$. Dans le cas d'une variable d'intérêt quantitative y, on peut définir une fonction d'erreur $R(T)$ comme la somme des écarts au carré (cf. partie 3.2.2), ainsi :

$$R(T) = \sum_{i=1}^{n} \left(y_i - \phi_T(x_i)\right)^2$$

où les $(x_i, y_i)_{i=1,\dots,n_v}$ appartiennent à la base – d'apprentissage ou de validation selon le contexte – de taille n.

L'algorithme est développé ci-dessous.

1. On construit l'arbre saturé T_{max} sur la base d'apprentissage, il minimise $R(T)$ sur cette base mais sûrement pas sur la base de validation. On note K son nombre de feuilles.
2. On détermine T_{K-1}, c'est l'arbre qui minimise $R(T)$ sur **la base de validation** parmi les arbres construits sur la base d'apprentissage ayant $K - 1$ feuilles.
3. On répète l'étape précédente pour obtenir la suite : $T_1 \subset \cdots \subset T_{K-1} \subset T_K$. On note à chaque étape la valeur de $R(T_k)$ en fonction de k.
4. On trace la courbe de $R(T_k)$ en fonction de $k \in \{1, \dots, K\}$. On choisit le k^*, et donc l'arbre correspondant T_{k^*}, qui minimise $R(T)$.

Figure 22 : Procédure d'élagage avec l'échantillon de validation

Rappelons qu'il est crucial dans l'étape 2 de ne pas utiliser les données d'apprentissage pour élaguer l'arbre, on utilise l'échantillon de validation.

En pratique, cet algorithme n'est pas scrupuleusement suivi. Plutôt que de trouver le nombre optimal de feuilles k^*, on préfère souvent trouver un coefficient réel positif α^* qui vient perturber notre critère de décision de création ou non d'un nœud lors de la construction. Concrètement, une fois l'optimisation sur la base de validation réalisée, il suffit de créer un arbre sur la base d'apprentissage avec un critère pénalisant par α^* le nombre de feuilles pour obtenir notre arbre optimal à k^* feuilles.

Mathématiquement, ce nouveau critère se traduit par une fonction objectif à minimiser qui s'écrit sous la forme :

$$R_\alpha(T) = R(T) + \alpha|T|$$

$|T|$ est le nombre de feuilles de l'arbre T
α un réel positif pénalisant la complexité de l'arbre
$R(T)$ représente l'erreur

Cette fonction $R_\alpha(T)$ est le plus souvent nommée qualité de l'arbre, ou erreur de l'arbre.

Construire la suite $T_1 \subset \cdots \subset T_{K-1} \subset T_K$ avec le critère $R(T)$ et déterminer le nombre de feuilles optimal k^* est équivalent à déterminer la suite $T_{\alpha_1} \subset \cdots \subset T_{\alpha_{K-1}} \subset T_{\alpha_K}$ avec le critère $R_\alpha(T)$ (où $\alpha_K = 0$ et α_1 tel que T_{α_1} soit l'arbre tronc, c'est à dire avec une feuille) et la pénalité de complexité optimale α^*.

En effet, l'idée est de noter que $R(T)$ correspond à notre critère d'hétérogénéité utilisé pour diviser un nœud, or entre un arbre à k feuilles (avant le nœud) et un arbre à $k+1$ feuilles (après le nœud) :

$$\begin{aligned} R_\alpha(T_{k+1}) &= R(T_{k+1}) + \alpha|T_{k+1}| \\ &= R(T_{k+1}) + \alpha(k+1) \end{aligned}$$

$$\begin{aligned} R_\alpha(T_k) &= R(T_k) + \alpha|T_k| \\ &= R(T_k) + \alpha k \end{aligned}$$

De là :

$$R_\alpha(T_k) - R_\alpha(T_{k+1}) > 0 \quad \Leftrightarrow \quad R(T_k) - R(T_{k+1}) > \alpha$$

Par suite, si l'on construit un arbre saturé sur les données d'apprentissage avec le critère $R_\alpha(T)$ au lieu de $R(T)$, en ayant spécifié le α à l'avance, alors l'arbre va cesser de se diviser lorsque l'écart d'hétérogénéité devient plus petit que α. Avec un « vrai » arbre saturé où $\alpha = 0$, l'arbre continue de se diviser tant que $R(T_k) - R(T_{k+1}) > 0$, autrement dit il achève le processus jusqu'au terme.

Si le modèle est correctement spécifié et que les données d'apprentissage et de validation partagent la même dynamique reliant la variable d'intérêt aux variables explicatives, alors la courbe $k \longmapsto R(T_k)$ ou bien la courbe $\alpha \longmapsto R(T_\alpha)$ est en « U » (stricte convexité) et admet un minimum global. Cet arbre solution correspond à un équilibre entre biais et variance : un modèle assez complexe pour bien différencier les observations et donner de bonnes prédictions mais pas trop complexe pour ne pas sur-interpréter les données d'apprentissage. On retrouve l'idée de la Figure 1.

En pratique, on ne pourra que rarement commencer notre processus d'élagage à partir de l'arbre saturé car pour un grand nombre d'observations, cela conduit à des arbres très grands, surtout si une des variables explicatives est continue car il existera alors autant de profils que d'observations. Les fonctions sous R ont des limites de « profondeur d'arbre », c'est-à-dire que les arbres construits ne peuvent avoir plus qu'un certain nombre d'étages, ce qui limite aussi le nombre de nœuds. Ce nombre d'étages limite dépend de la fonction R utilisée pour construire l'arbre.

Voici la courbe que nous obtenons sur la base automobile que nous étudions :

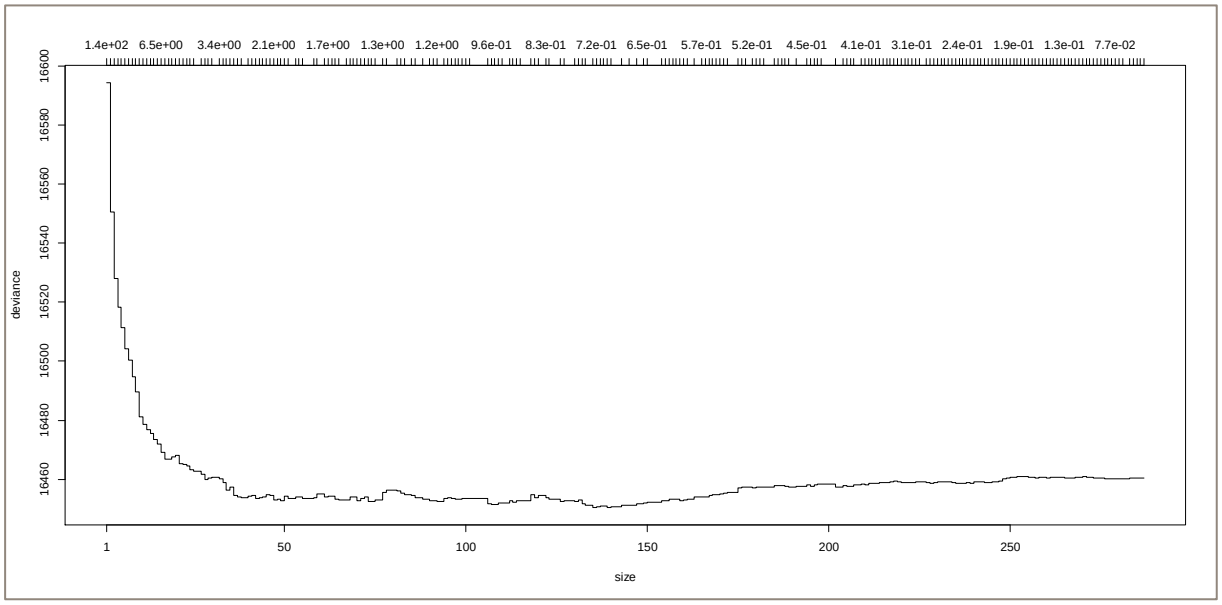


Figure 23 : Erreur sur la base de validation en fonction du nombre de feuilles (sans bucket)

Ce graphe a été obtenu sur les données sans *bucket*. Le minimum d'erreur est atteint pour $k = 139$. On retient alors l'arbre à 139 feuilles. Sur les données avec *buckets*, la courbe est la suivante :

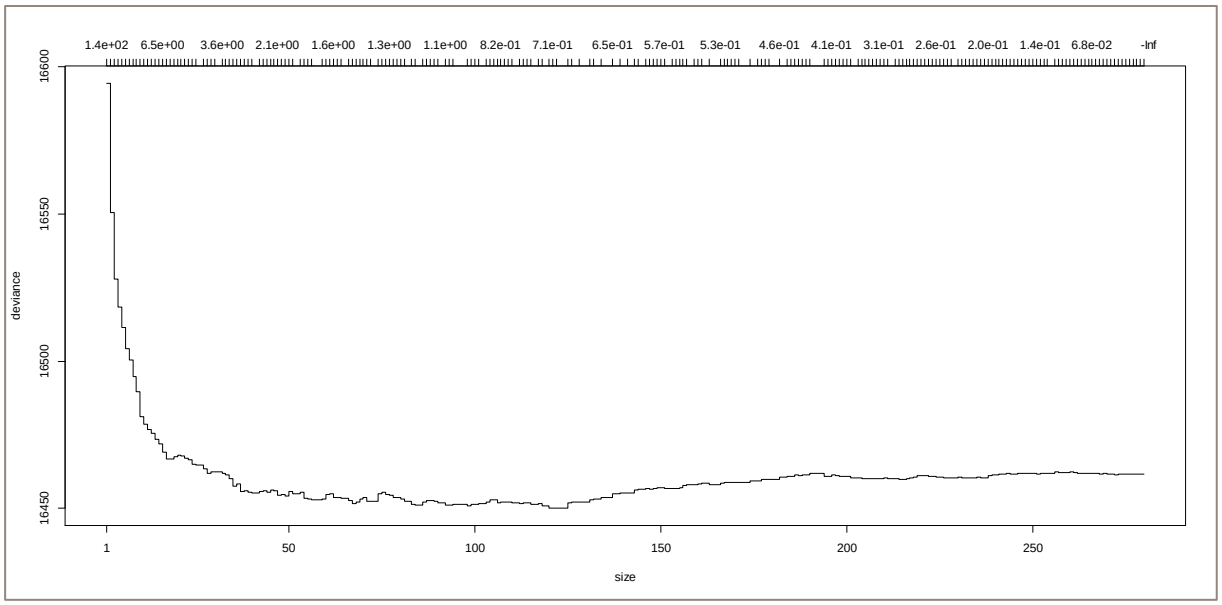


Figure 24 : Erreur sur la base de validation en fonction du nombre de feuilles (avec buckets)

On retient alors l'arbre à 122 feuilles. L'arbre est plus simple, voici à quoi il ressemble :

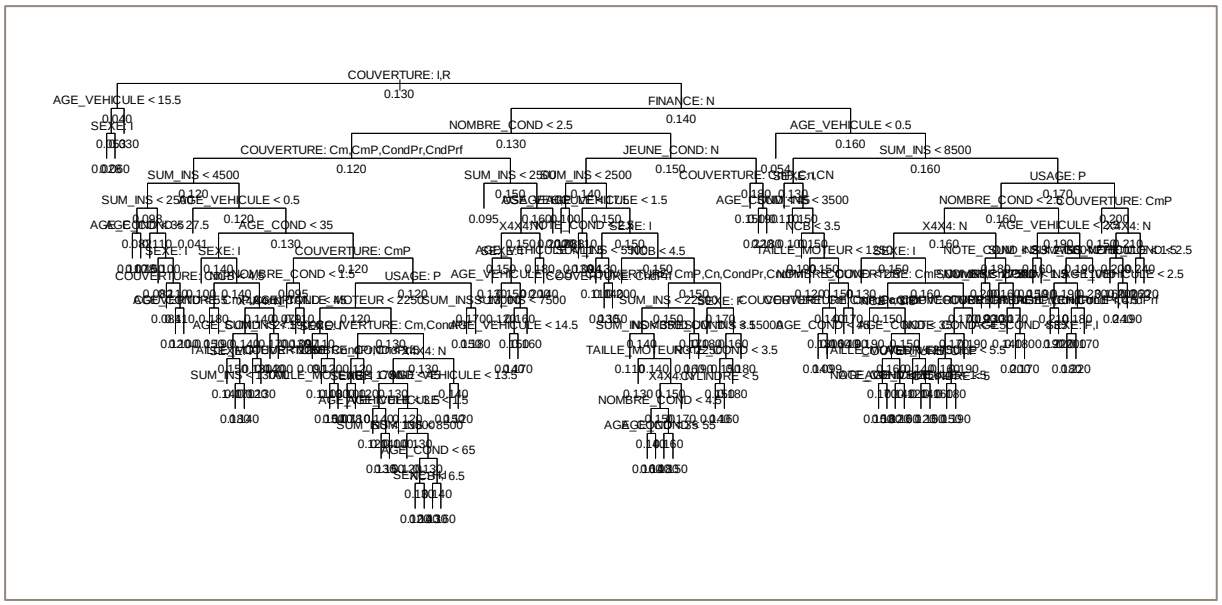


Figure 25 : Arbre optimal (122 feuilles) obtenu après élagage

On remarque notamment l'importance de la variable COUVERTURE. Nous avons de plus grandement gagné en simplicité par rapport à l'arbre saturé. On insiste aussi sur l'interprétation facilitée offerte par la représentation sous forme d'arbre, notamment en comparaison avec un *GLM*. **Nous retenons donc l'arbre optimal à 122 feuilles** sur les données avec *buckets*.

Néanmoins, nous le verrons dans l'analyse, même l'arbre optimal peut ne pas être un bon prédicteur eu égard à la base de test. Il existe des méthodes dites d'agrégation ou ensemblistes permettant de gagner en pouvoir prédictif.

Nous allons désormais mettre en œuvre les méthodes de *bagging*, de forêts aléatoires et de *boosting*.

4. MÉTHODES D'AGRÉGATION

Les arbres *CART* construits sont en général très instables et donc les estimations, ou prévisions produites, varient beaucoup. En effet plus l'arbre est étendu, plus les nœuds finaux (et donc les estimations) dépendent de certains profils et peuvent donner des estimations très bonnes comme très mauvaises lorsqu'il s'agit de prévoir des observations autres que celles qui ont été utilisées pour la construction de l'arbre.

L'idée essentielle des méthodes ensemblistes est d'offrir une meilleure stabilité et fiabilité aux résultats, en palliant le sur-apprentissage, principal défaut des arbres de décision. Les méthodes d'agrégation de modèles ont été développées pour répondre à ce problème : l'idée est de construire un grand nombre d'arbres puis de réunir les résultats pour obtenir une unique estimation. Cette procédure permet de réduire considérablement la variance de l'estimation et d'obtenir des estimateurs plus consistants. Nous étudierons ainsi des modèles d'agrégation dits parallèles, qui assemblent des arbres construits indépendamment des uns des autres, comme le *bagging* et les forêts aléatoires. Puis des modèles d'agrégation dits adaptatifs qui se différencient des méthodes précédentes par le fait que chaque arbre inclus dans le modèle dépend de l'arbre précédent et est construit en fonction de celui-ci. Ces méthodes sont multiples et réunies sous le nom de *boosting*.

L'objectif de cette section est d'étudier chacune de ces méthodes, en les appliquant à l'algorithme de construction d'arbre *CART*.

4.1. BAGGING

Le *bagging* (cf. BERK, 2004) est la contraction de *bootstrap aggregation*. L'algorithme propose de construire une multitude d'arbres par *bootstrap*, c'est-à-dire : si l'on dispose d'un échantillon de taille n, on tire aléatoirement avec remise des observations pour constituer un nouvel échantillon de taille n, on répète cette opération un grand nombre de fois pour ainsi obtenir B échantillons à partir desquels on va construire B arbres de taille maximale (c'est-à-dire que l'on n'élague pas les arbres construits). Puis pour chaque profil, on estime la variable d'intérêt en regroupant les B différents résultats obtenus à partir des B arbres construits :

- Si la variable d'intérêt est discrète, on examine dans quel profil l'individu a été classé au sein de chaque arbre, et on décide par vote à la majorité.

- Pour une variable d'intérêt continue, on effectue la moyenne des différentes estimations données par les B arbres.

Ainsi, dans notre cas, nous obtenons avec le *bagging* une suite $(\phi_k)_{k=1,\dots,B}$ de B arbres et l'arbre final retenu est défini par $\phi = \frac{1}{B}\sum_{k=1}^{B}\phi_k$. Le *bagging* permet en agrégeant une multitude de modèles de réduire la variance de l'estimation. En effet, si l'on suppose que les arbres ϕ_k sont deux à deux corrélés par ρ (les échantillons utilisés se ressemblent à environ 50 %, cf. annexe G) et qu'ils ont une variance σ^2, alors :

$$
\begin{aligned}
V[\phi] &= V\left[\frac{1}{B}\sum_{k=1}^{B}\phi_k\right] \\
&= \frac{1}{B^2}V\left[\sum_{k=1}^{B}\phi_k\right] \\
&= \frac{1}{B^2}\sum_{k=1}^{B}\left(\sigma^2 + \sum_{k'=1,k'\neq k}^{B}\rho\sigma^2\right) \\
&= \frac{\sigma^2}{B^2}\sum_{k=1}^{B}(1+(B-1)\rho) \\
&= \frac{\sigma^2}{B^2}(1+(B-1)\rho) \\
&= \rho\sigma^2 + \frac{(1-\rho)\sigma^2}{B}
\end{aligned}
$$

La variance décroit en fonction du nombre d'itération.

L'algorithme est indiqué ci-dessous.

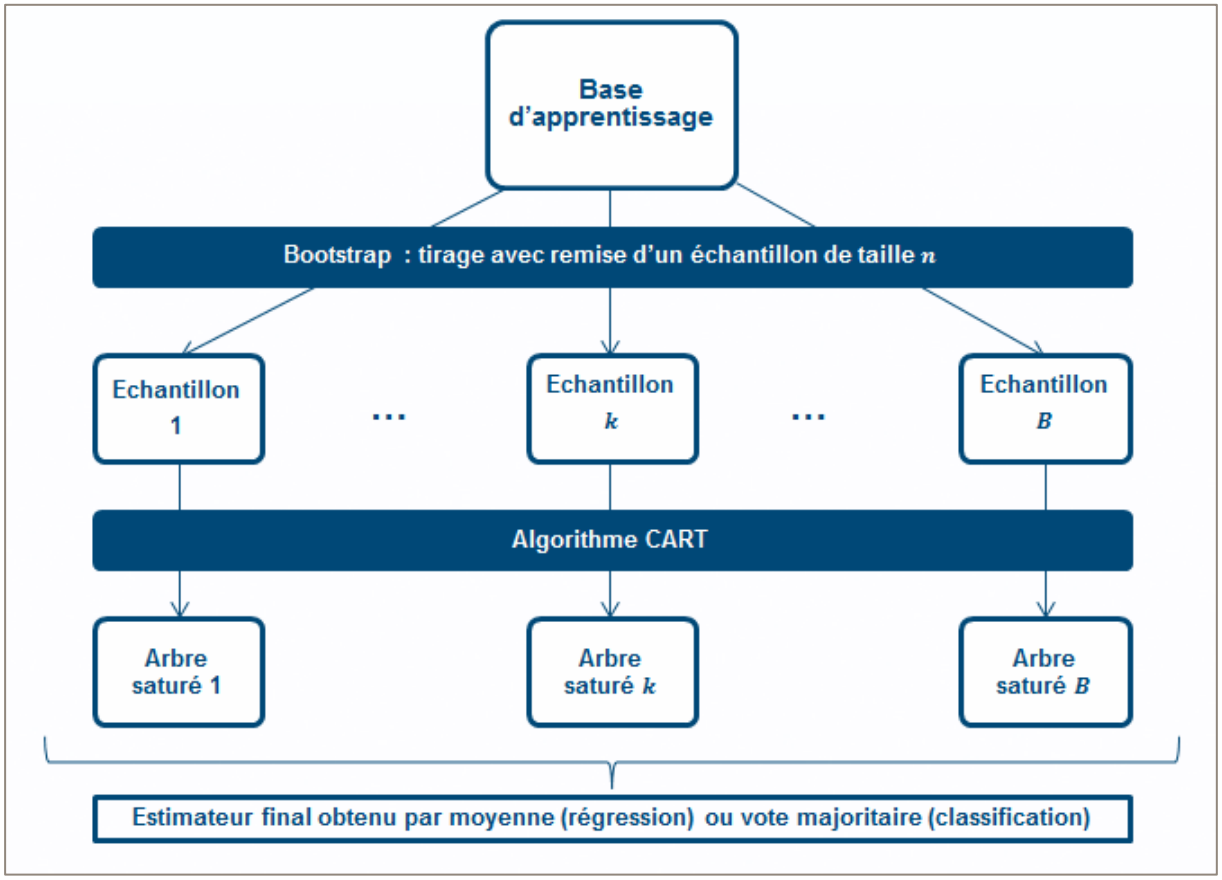


Figure 26 : Algorithme du bagging

Si cette méthode conduit à un classifieur plus robuste et précis, elle perd grandement en lisibilité : on obtient une moyenne de résultats et non une simple structure d'arbre, facilement interprétable.

On représente ici le graphique d'évolution de l'erreur sur la base de validation en fonction de B. On cherche le B optimal.

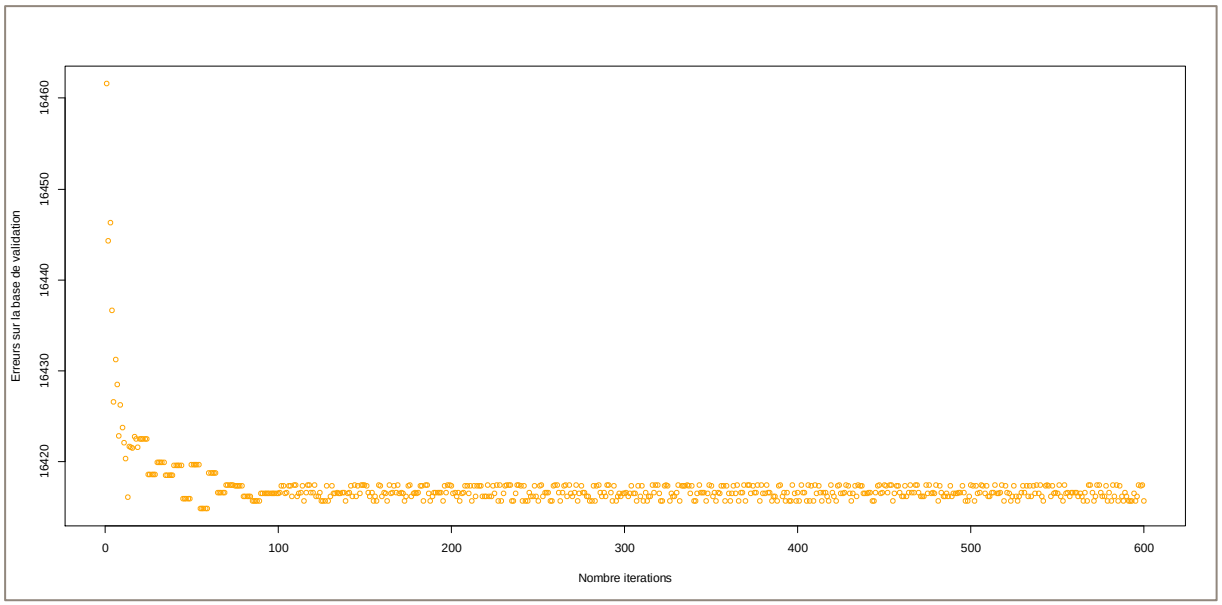


Figure 27 : Erreur de validation en fonction de B pour le bagging

On retient donc le nombre d'itérations $B^* = 55$, qui minimise l'erreur sur la base de validation. Il est à noter que la méthode étant stochastique, si on relance les algorithmes, nous obtiendrons une nouvelle courbe de l'erreur de validation en fonction de B, et donc un nouveau B^*. Toutefois cette remarque ne doit pas nous empêcher de conclure : **nous sélectionnons le *bagging* avec 55 itérations**.

La réduction de variance produite par le *bagging* est limitée. Il est possible d'introduire de l'aléatoire dans la procédure pour renforcer l'indépendance entre les modèles, cette idée est à l'origine des méthodes de forêts aléatoires.

4.2. RANDOM FOREST

La méthode des forêts aléatoires (cf. BREIMAN, 2001) est similaire au *bagging* dans la manière d'agréger les modèles. Cependant la différence se situe dans la construction des arbres qui vont être agrégés, en effet cette méthode se distingue du *bagging* par le fait qu'avant la division de chaque nœud, à la place de sélectionner la division optimale parmi les divisions possibles basées sur toutes les variables explicatives, on tire aléatoirement un certain nombre m de variables explicatives, et on considère les divisions possibles basées sur ce sous-ensemble.

Ajouter cet aléa dans la construction des arbres permet de rendre les arbres construits plus « indépendants » et de réduire donc la variance de l'estimation. De plus, dans le cas où il y a de nombreuses variables explicatives, et particulièrement des variables explicatives avec un grand nombre de modalités, la sélection aléatoire d'un sous-groupe de variables explicatives permet d'utiliser des variables qui n'auraient peut-être pas été sélectionnées si elles avaient été confrontées à toutes les variables.

Si on note p le nombre total de variables explicatives, le nombre de variables explicatives tirées aléatoirement est par défaut :

- $m \sim \sqrt{p}$ pour un arbre de classification,
- $m \sim \frac{p}{3}$ pour un arbre de régression.

On a vu dans la partie précédente qu'une approximation de la variance de l'estimateur basé sur le *bagging* est $\rho\sigma^2 + \frac{(1-\rho)\sigma^2}{B}$. Si les échantillons sont indépendants, alors $\rho = 0$ et la variance converge vers 0 quand B grandit. Les méthodes de forêts aléatoires permettent d'introduire plus d'indépendance entre les arbres construits, et donc de réduire le premier terme de la variance ci-dessus (indépendant de B et donc limitant la réduction de variance induite). L'algorithme est donné ci-dessous.

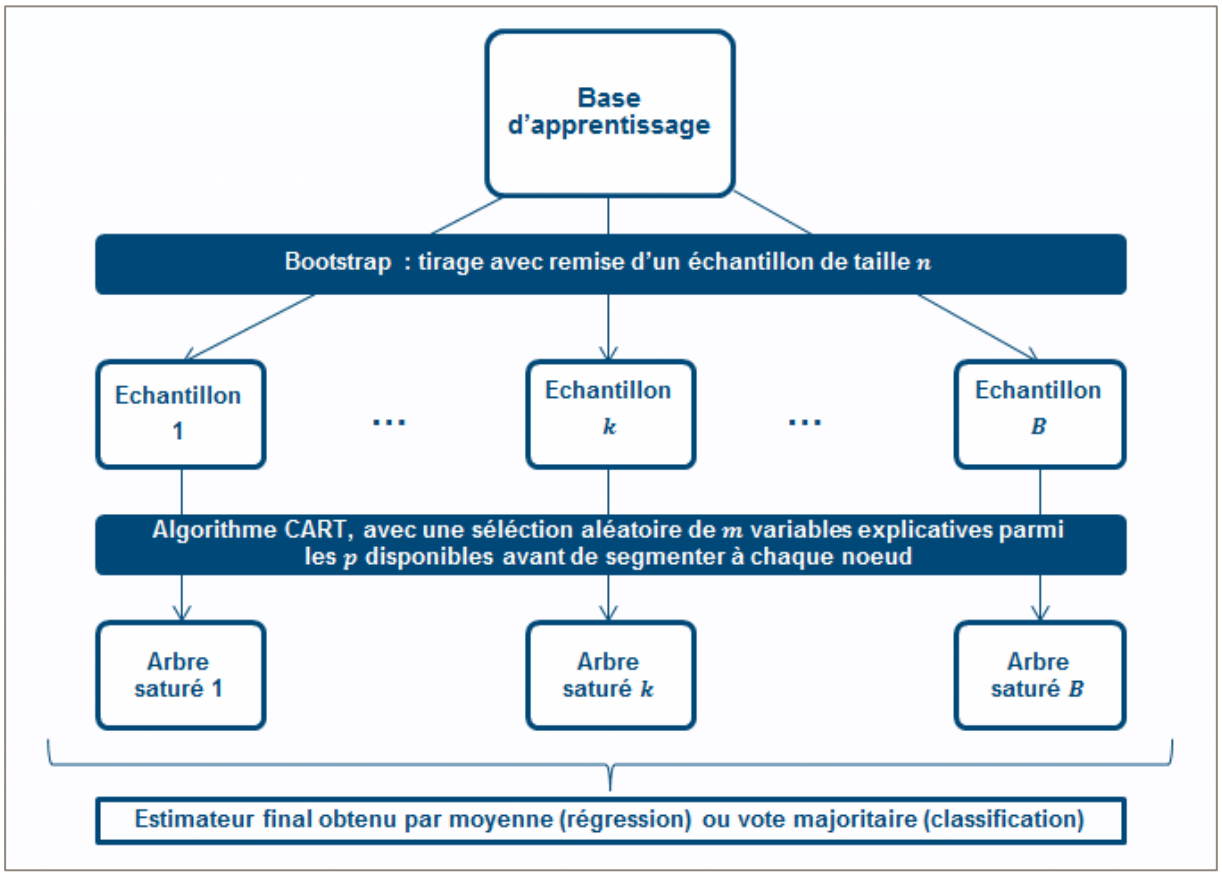


Figure 28 : Algorithme Random Forest

À nouveau, on peut tracer l'évolution de l'erreur en fonction du nombre d'itérations B.

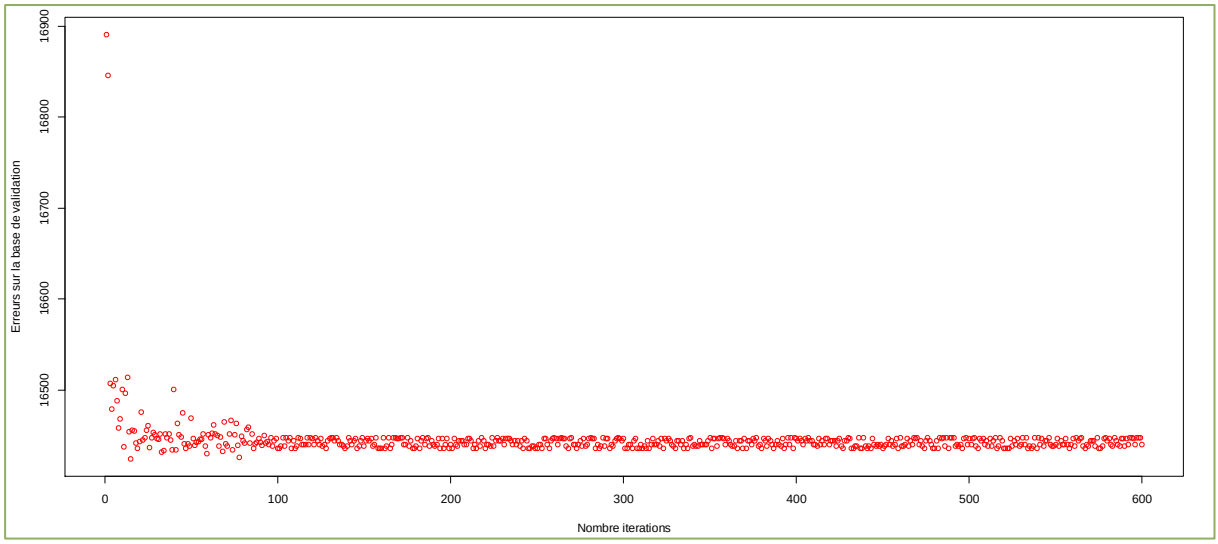


Figure 29 : Erreur de validation en fonction de B pour le random forest

La méthode *random forest* s'exécute plus rapidement que le *bagging* car les arbres se construisent nettement plus rapidement étant donné qu'on se limite à m variables explicatives à chaque nœud. Nous notons $B^* = 17$ et **nous sélectionnons donc le modèle *random forest* avec 17 itérations**. La même remarque que pour le *bagging* s'applique, eu égard au caractère stochastique : la convergence diffère selon les échantillons *bootstrap* et les variables tirées.

Le *bagging* et les forêts aléatoires sont des méthodes dites parallèles, on le conçoit bien en observant la Figure 26 et la Figure 28. Présentons à présent la méthode adaptative récursive qu'est le *boosting*.

4.3. BOOSTING

4.3.1. Principe

Nous avons vu jusqu'ici des techniques qui associent des arbres construits séparément les uns des autres, on pourrait aussi construire les arbres les uns après les autres, tel que chaque arbre introduit corrige les défauts du précédent. Cette idée a conduit aux multiples algorithmes, réunis sous le nom de *boosting*. Le *boosting* est particulièrement utile pour des classifieurs dits « faibles » dont la qualité de prévision peut être améliorée.

Les arbres du *bagging* sont identiquement distribués, et donc le biais de l'agrégation des arbres est le même que pour chaque arbre, ceci n'est plus vrai avec le *boosting*, et le biais est significativement réduit. Cela est dû au fait que chaque arbre est une version adaptative du précédent, en effet les observations mal prédites par le k-ième arbre vont être sur-pondérées pour la construction du prochaine arbre. Ainsi pas à pas, les arbres sont corrigés pour donner les meilleures estimations possibles.

Les algorithmes de *boosting* sont des méthodes de descente (cf. annexe F), plus l'algorithme avance, plus il est corrigé et devient particulièrement adapté aux données d'apprentissage. Ainsi considérer l'ensemble des arbres et trouver les paramètres tels que la convergence ne soit pas trop rapide, permet d'éviter le sur-apprentissage.

Il existe de multiples algorithmes de *boosting* (cf. FREUND et SCHAPIRE, 1996 et cf. RIDGEWAY, 1999) qui se distinguent par :

- La pondération des données à chaque itération,
- La pondération des arbres dans la formule de l'estimation finale,
- La fonction de perte permettant de mesurer l'erreur de prévision.

Les méthodes de *boosting* sont particulièrement intéressantes car si le classifieur faible utilisé est une souche ou un arbre tronc (avec une seule feuille, retournant la moyenne empirique de l'échantillon), on peut tout de même obtenir des résultats très bons, et même meilleurs que ceux obtenus sans *boosting* avec un arbre très sophistiqué alors que la quantité de calcul est similaire.

La première méthode à être présentée lorsque l'on parle de *boosting* est la méthode *Adaboost*. Nous ne la détaillerons pas car elle ne s'applique fondamentalement que pour des variables d'intérêt binaire. Nous présentons plutôt la méthode dite du *gradient boosting*.

4.3.2. Gradient boosting

L'idée du *gradient boosting* n'est pas simple et nous choisissons de revenir sur l'article original de Friedman (2001), en adoptant ses notations. Nous incitons le lecteur à faire le parallèle avec l'introduction, mais aussi à consulter l'annexe F sur l'algorithme du gradient dans le cas plus général d'optimisation de fonction.

Nous considérons un échantillon de p variables explicatives notées $x_{i,1}, \dots, x_{i,p}$ pour $i \in \{1, \dots, n\}$ où n correspond à la taille de l'échantillon. Pour plus de clarté, on notera $\boldsymbol{x}_i = (x_{i,1}, \dots, x_{i,p}) \in \mathbb{R}^p$. On note y_i la variable réponse. Plus généralement, on considérera que les $\boldsymbol{x}_i$ sont les réalisations d'une variable notée $\boldsymbol{x}$ et que les y_i sont les réalisations d'une variable notée y. On notera $\mathcal{F}$ l'espace des fonctions admissibles, inclus dans l'ensemble des fonctions définies sur $\mathbb{R}^p$ à valeurs dans $\mathbb{R}$.

Notre échantillon d'apprentissage correspond donc à la donnée des vecteurs $(\boldsymbol{x}_i, y_i)_{i \in \{1,\dots,n\}}$ pour chaque individu i. Etant donné cet échantillon connu, il s'agit de trouver une estimation $\hat{F}(\boldsymbol{x})$ de la fonction $F^*(\boldsymbol{x})$ permettant de relier les variables explicatives $\boldsymbol{x}$ à la réponse y. La fonction $F^*(\boldsymbol{x})$ est définie comme étant la fonction admissible qui minimise la moyenne d'une fonction de perte notée L sur l'ensemble $\mathcal{F}$ et sachant l'échantillon.

Mathématiquement, nous pouvons formaliser ceci en :

$$F^*(\boldsymbol{x}) = \arg\min_{F \in \mathcal{F}} E_{y,\boldsymbol{x}}\big[L\big(y, F(\boldsymbol{x})\big)\big] = \arg\min_{F \in \mathcal{F}} E\big[E\big[L\big(y, F(\boldsymbol{x})\big) \,\big|\, y\big] \,\big|\, \boldsymbol{x}\big]$$

Ainsi la « fonction réponse » peut être perçue comme celle qui, parmi toutes les fonctions admissibles, minimise un critère bien choisi. Dans la littérature, on retrouve différentes fonctions de perte :

- $L(y, F) = (y - F)^2$
- $L(y, F) = \big|y - F\big|$
- $L(y, F) = \ln(1 + e^{-2yF})$ quand $y = \pm 1$

Afin de faciliter la dérivation et l'optimisation, on préfère bien souvent la première fonction (dérivable et convexe), en lien avec l'analyse menée en introduction de ce mémoire.

Eu égard aux fonctions F admissibles, la procédure suppose bien souvent une vision paramétrique de la fonction, au sens où son évaluation en $\boldsymbol{x}$ dépend d'un jeu de paramètres $\boldsymbol{P} = (P_1, \dots, P_M) \in \mathbb{R}^M$. L'auteur se focalise dans son article sur un modèle additif :

$$F(\boldsymbol{x}; \boldsymbol{P}) = \sum_{m=1}^{M} \beta_m h(\boldsymbol{x}; a_m)$$

Il faut donc comprendre que les paramètres interviennent de façon additive, dans la mesure où il n'y a pas d'interaction entre des paramètres P_m (le terme ne dépend que de m). De façon générale, la fonction h introduite ici est une fonction reliant les variables d'entrées $\boldsymbol{x}$ aux paramètres $\boldsymbol{a} = (a_1, \dots, a_M)$. Ici, l'auteur s'intéresse au cas où chaque fonction h est un arbre de régression (type *CART*). Les paramètres $\boldsymbol{a}$ sont alors perçus comme les variables permettant d'obtenir les nœuds terminaux (les feuilles).

Le fait d'imposer une vision paramétrique de la fonction F permet transformer quelque peu le problème d'optimisation en le suivant :

$$\begin{aligned} F^*(\boldsymbol{x}) &= F(\boldsymbol{x}; \boldsymbol{P}^*) \\ \boldsymbol{P}^* &= \arg\min_{\boldsymbol{P} \in \mathbb{R}^M} \underbrace{E_{y,\boldsymbol{x}}\big[L\big(y, F(\boldsymbol{x}; \boldsymbol{P})\big)\big]}_{\Phi(\boldsymbol{P})} \end{aligned}$$

Des méthodes numériques peuvent être mises en œuvre pour résoudre ce problème, et cela suppose souvent sur un algorithme de résolution récursif convergeant vers la solution (type méthode de Newton par exemple). Dans la mesure où la relation de récurrence se traduit formellement par une égalité sur la dérivée discrète du type $u_{k+1} - u_k = z_k$, alors en intégrant on obtient une relation $u_K = u_0 + \sum_{k=1}^{K} z_k$.

En se basant sur cette remarque, nous choisissons de réécrire le jeu de paramètres optimal $\boldsymbol{P}^*$ par :

$$\boldsymbol{P}^* = \boldsymbol{p}_0 + \sum_{m=1}^{M} \boldsymbol{p}_m = \sum_{m=0}^{M} \boldsymbol{p}_m$$

Le $\boldsymbol{p}_0$ correspond à la première estimation (grossière) de $\boldsymbol{P}^*$, et les $(\boldsymbol{p}_m)_{m\in\{1,\dots,M\}}$ correspondent aux incréments successifs (les *boosts*) pour parvenir à l'estimation de la solution $\boldsymbol{P}^*$.

Revenons à présent à une approche non paramétrique. Nous avons :

$$F^*(\boldsymbol{x}) = \arg\min_{F\in\mathcal{F}} \underbrace{E_{y,\boldsymbol{x}}\big[L\big(y, F(\boldsymbol{x})\big)\big]}_{\Phi\left(F(\boldsymbol{x})\right)} = \arg\min_{F\in\mathcal{F}} \Phi\big(F(\boldsymbol{x})\big)$$

La même remarque que précédemment conduit à considérer l'écriture suivante :

$$F^*(\boldsymbol{x}) = \sum_{m=0}^{M} f_m(\boldsymbol{x})$$

De la même façon, le $f_0(\boldsymbol{x})$ correspond à la première et grossière estimation, tandis que les $(f_m)_{m\in\{1,\dots,M\}}$ correspondent aux incréments (les *boosts*) conduisant à la solution optimale. Il est intéressant de définir la notation suivante :

$$\forall m \in \{0; \dots; M\}, F_m(\boldsymbol{x}) = \sum_{i=0}^{m} f_i(\boldsymbol{x})$$

De sorte que $\hat{F}(\boldsymbol{x}) = F_M(\boldsymbol{x})$ et nous retrouvons la relation de récurrence $F_m(\boldsymbol{x}) - F_{m-1}(\boldsymbol{x}) = f_m(\boldsymbol{x})$.

Selon l'algorithme du gradient, nous avons :

$$f_m(\boldsymbol{x}) = -\rho_m g_m(\boldsymbol{x})$$

Avec :

$$\begin{aligned} g_m(\boldsymbol{x}) &= \left.\frac{\partial \Phi\big(F(\boldsymbol{x})\big)}{\partial F(\boldsymbol{x})}\right|_{F(\boldsymbol{x})=F_{m-1}(\boldsymbol{x})} \\ &= \left.\frac{\partial E\big[L\big(y, F(\boldsymbol{x})\big)\big]}{\partial F(\boldsymbol{x})}\right|_{F(\boldsymbol{x})=F_{m-1}(\boldsymbol{x})} \\ &= E\left[\left.\frac{\partial L\big(y, F(\boldsymbol{x})\big)}{\partial F(\boldsymbol{x})}\right|_{F(\boldsymbol{x})=F_{m-1}(\boldsymbol{x})} \Big| \boldsymbol{x}\right] \end{aligned}$$

Nous avons permuté l'espérance et la dérivation (les fonctions sont continues et la dérivée est intégrable). Dans le cas de l'algorithme de plus profonde descente, le facteur multiplicatif ρ_m est donné par la relation :

$$\rho_m = \arg\min_{\rho\in\mathbb{R}} E_{y,\boldsymbol{x}}\big[L\big(y, F_{m-1}(\boldsymbol{x}) - \rho g_m(\boldsymbol{x})\big)\big]$$

Nous pouvons à présent définir l'algorithme. Nous disposons de l'échantillon $(\boldsymbol{x}_i, y_i)_{i\in\{1,\dots,n\}}$.

Initialisation

$$F_0(\boldsymbol{x}) = \arg\min_{\rho\in\mathbb{R}} \sum_{i=1}^{n} L(y_i, \rho)$$

Dans le cas où $L(y, F) = (y - F)^2$, alors nous avons $F_0(\boldsymbol{x}) = \arg\min_\rho \sum_{i=1}^n (y_i - \rho)^2$. C'est une fonction convexe et dérivable en la variable ρ, le minimum global est donc atteint quand la dérivée s'annule, nous avons :

$$\begin{aligned}
\frac{\partial \sum_{i=1}^n L(y_i, \rho)}{\partial \rho} &= \sum_{i=1}^{n} \frac{\partial L(y_i, \rho)}{\partial \rho} \\
&= \sum_{i=1}^{n} \frac{\partial (y_i - \rho)^2}{\partial \rho} \\
&= -2 \sum_{i=1}^{n} (y_i - \rho) \\
&= -2 \sum_{i=1}^{n} y_i + 2n\rho
\end{aligned}$$

Ainsi :

$$\frac{\partial \sum_{i=1}^n L(y_i, \rho)}{\partial \rho} = 0 \quad \Leftrightarrow \quad \rho = \frac{1}{n} \sum_{i=1}^{n} y_i$$

Par conséquent nous avons :

$$F_0(\boldsymbol{x}) = \frac{1}{n} \sum_{i=1}^{n} y_i = \bar{y}$$

Autrement dit la moyenne empirique des $(y_i)_{i\in\{1,\dots,n\}}$ initialise l'algorithme, cela correspond à « l'arbre-tronc » avec une seule et unique feuille et qui associe la même valeur $\bar{y}$ à n'importe quelles valeurs d'entrée.

Itération

On considère un nombre fini d'itérations noté M. L'étape est notée $m \in \{1, \dots, M\}$. On calcule alors pour tout $i \in \{1, \dots, n\}$:

$$\tilde{y}_{i,m} = -\left.\frac{\partial L(y_i, F)}{\partial F}\right|_{F=F_{m-1}(\boldsymbol{x}_i)}$$

Ce qui conduit dans notre cas à :

$$\begin{aligned}
\tilde{y}_{i,m} &= -\left.\frac{\partial L(y_i, F)}{\partial F}\right|_{F=F_{m-1}(\boldsymbol{x}_i)} \\
&= -\left.\frac{\partial (y_i - F)^2}{\partial F}\right|_{F=F_{m-1}(\boldsymbol{x}_i)} \\
&= 2\big(y_i - F_{m-1}(\boldsymbol{x}_i)\big)
\end{aligned}$$

Ainsi pour la première étape nous avons $\tilde{y}_{i,1} = 2\big(y_i - F_0(\boldsymbol{x}_i)\big) = 2(y_i - \bar{y})$.

Nous calculons ensuite :

$$(\rho_m, a_m) = \arg\min_{a,\rho\in\mathbb{R}} \sum_{i=1}^{n} \Big(\tilde{y}_{i,m} - \rho h(\boldsymbol{x}_i; a)\Big)^2$$

Enfin nous pouvons évaluer :

$$F_m(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x}) + \rho_m h(\boldsymbol{x}; a_m)$$

Jusqu'à présent nous n'avons pas détaillé la fonction h. Nous considérons le cas particulier où elle traduit un arbre de régression à J feuilles. Nous pouvons alors écrire :

$$h\left(\boldsymbol{x}; \left(b_j, R_j\right)_{j\in\{1;\dots;J\}}\right) = \sum_{j=1}^{J} b_j \mathbb{1}_{\{\boldsymbol{x}\in R_j\}}$$

Autrement dit, les $\left(b_j\right)_{j\in\{1,\dots,J\}}$ correspondent aux valeurs associées aux valeurs d'entrées si celles-ci appartiennent à la classe notée $\left(R_j\right)_{j\in\{1,\dots,J\}}$. Les $\left(R_j\right)_{j\in\{1,\dots,J\}}$ forment une partition de l'espace des variables d'entrées.

La relation précédente devient ainsi :

$$F_m(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x}) + \rho_m \sum_{j=1}^{J_m} b_{j,m} \mathbb{1}_{\{\boldsymbol{x}\in R_{j,m}\}}$$

Les $\left(R_{j,m}\right)_{j\in\{1,\dots,J_m\}}$ sont les classes associées à l'arbre de régression des $\left(\tilde{y}_{i,m}\right)_{i\in\{1,\dots,n\}}$. Par suite, les coefficients $\left(b_{j,m}\right)_{j\in\{1,\dots,J_m\}}$ sont égaux aux moyennes des $\tilde{y}_{i,m}$ conditionnellement au fait que les variables d'entrées associées sont dans les classes $\left(R_{j,m}\right)_{j\in\{1,\dots,J_m\}}$. On peut donc écrire formellement :

$$b_{j,m} = E\left[\tilde{y}_m \,\middle|\, \boldsymbol{x} \in R_{j,m}\right]$$

A ce stade, nous réécrivons :

$$F_m(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x}) + \sum_{j=1}^{J_m} \underbrace{\rho_m b_{j,m}}_{\gamma_{j,m}} \mathbb{1}_{\{\boldsymbol{x}\in R_{j,m}\}}$$

On peut lire cette relation de la façon suivante : l'algorithme du *gradient boosting* consiste à ajouter à chaque étape des fonctions indicatrices (des feuilles d'une certaine façon). Nous pouvons perturber quelque peu l'estimation en considérant directement les $\gamma_{j,m}$ et en les cherchant de façon optimale.

Pour une étape donnée $m \in \{1, \dots, M\}$,

$$\left(\gamma_{1,m}, \dots, \gamma_{J_m,m}\right) = \arg\min_{\gamma_1,\dots,\gamma_{J_m}\in\mathbb{R}} \sum_{i=1}^{n} L\left(y_i, F_{m-1}(\boldsymbol{x_i}) + \sum_{j=1}^{J_m} \gamma_j \mathbb{1}_{\{\boldsymbol{x_i}\in R_{j,m}\}}\right)$$

Mais comme les classes $\left(R_{j,m}\right)_{j\in\{1,\dots,J_m\}}$ sont disjointes, cette relation se simplifie, et pour tout $j \in \{1, \dots, J_m\}$:

$$\gamma_{j,m} = \arg\min_{\gamma\in\mathbb{R}} \sum_{\boldsymbol{x_i}\in R_{j,m}} L(y_i, F_{m-1}(\boldsymbol{x_i}) + \gamma)$$

Dans notre cas,

$$\gamma_{j,m} = \arg\min_{\gamma\in\mathbb{R}} \sum_{\boldsymbol{x_i}\in R_{j,m}} \left(y_i - (F_{m-1}(\boldsymbol{x_i}) + \gamma)\right)^2$$

Or :

$$\begin{aligned}\frac{\partial \sum_{\boldsymbol{x}_i \in R_{j,m}} \left(y_i - (F_{m-1}(\boldsymbol{x}_i) + \gamma)\right)^2}{\partial \gamma} &= \sum_{\boldsymbol{x}_i \in R_{j,m}} \frac{\partial \left(y_i - (F_{m-1}(\boldsymbol{x}_i) + \gamma)\right)^2}{\partial \gamma} \\ &= \sum_{\boldsymbol{x}_i \in R_{j,m}} \frac{\partial (y_i - F_{m-1}(\boldsymbol{x}_i) - \gamma)^2}{\partial \gamma} \\ &= -2 \sum_{\boldsymbol{x}_i \in R_{j,m}} (y_i - F_{m-1}(\boldsymbol{x}_i) - \gamma) \\ &= 2\gamma \, card\left(R_{j,m}\right) - 2 \sum_{\boldsymbol{x}_i \in R_{j,m}} \left(y_i - F_{m-1}(\boldsymbol{x}_i)\right)\end{aligned}$$

Finalement :

$$\frac{\partial \sum_{\boldsymbol{x}_i \in R_{j,m}} \left(y_i - (F_{m-1}(\boldsymbol{x}_i) + \gamma)\right)^2}{\partial \gamma} = 0 \quad \Leftrightarrow \quad \gamma = \frac{\sum_{\boldsymbol{x}_i \in R_{j,m}} \left(y_i - F_{m-1}(\boldsymbol{x}_i)\right)}{card\left(R_{j,m}\right)}$$

Finalement nous obtenons :

$$\begin{aligned}\gamma_{j,m} &= \frac{\sum_{\boldsymbol{x}_i \in R_{j,m}} \left(y_i - F_{m-1}(\boldsymbol{x}_i)\right)}{card\left(R_{j,m}\right)} \\ &= \frac{1}{2} \frac{\sum_{\boldsymbol{x}_i \in R_{j,m}} \tilde{y}_{i,m}}{card\left(R_{j,m}\right)} \\ &= \frac{1}{2} E\left[\tilde{y}_m \,\middle|\, \boldsymbol{x} \in R_{j,m}\right]\end{aligned}$$

On notera que l'auteur suggère d'utiliser plutôt la fonction de perte suivante (en ajustant judicieusement δ) :

$$L(y, F) = \begin{cases} \frac{1}{2}(y - F)^2 & \text{si } |y - F| \leq \delta \\ \delta\left(|y - F| - \frac{\delta}{2}\right) & \text{si } |y - F| > \delta \end{cases}$$

Au terme de cette analyse, nous préférons choisir le critère $L(y, F) = \frac{1}{2}(y - F)^2$. Nous pouvons finalement conclure sur l'algorithme du *gradient boosting*.

1. $F_0(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^{n} y_i = \bar{y}$
2. Pour $m \in \{1, \ldots, M\}$
3. Pour $i \in \{1, \ldots n\}$, $\tilde{y}_{i,m} = -\frac{\partial L(y_i, F)}{\partial F}\Big|_{F = F_{m-1}(\boldsymbol{x}_i)} = y_i - F_{m-1}(\boldsymbol{x}_i)$
4. Évaluation des classes $\left(R_{j,m}\right)_{j \in \{1, \ldots, J_m\}}$ de l'arbre de régression à J_m feuilles des $\left(\tilde{y}_{i,m}\right)_{i \in \{1, \ldots, n\}}$
5. Pour $j \in \{1, \ldots, J_m\}$, $\gamma_{j,m} = \arg\min_{\gamma \in \mathbb{R}} \sum_{\boldsymbol{x}_i \in R_{j,m}} L(y_i, F_{m-1}(\boldsymbol{x}_i) + \gamma) = E\left[\tilde{y}_m \,\middle|\, \boldsymbol{x} \in R_{j,m}\right]$
6. $F_m(\boldsymbol{x}) = F_{m-1}(\boldsymbol{x}) + \sum_{j=1}^{J_m} \gamma_{j,m} \mathbb{1}_{\{\boldsymbol{x} \in R_{j,m}\}}$. Notons qu'un coefficient multiplicatif $0 < \nu \leq 1$ est parfois affecté à la somme. L'auteur montre qu'un coefficient $\nu \leq 0{,}1$ conduit à de meilleures estimations.

4.3.3. Algorithme retenu

On remarque que les étapes 4 et 5 peuvent être groupées car les $\gamma_{j,m}$ sont les résultats affichés dans les feuilles de l'arbre du point 4. De là, en revenant à des notations plus en lien avec ce mémoire, nous obtenons :

1. $\phi_0(x) = \frac{1}{n}\sum_{i=1}^{n} y_i = \bar{y}$
2. Pour $k \in \{1, \dots, B\}$
3. Pour $i \in \{1, \dots n\}$, $\tilde{y}_{i,k} = y_i - \phi_{k-1}(x_i)$
4. Évaluation de l'arbre de régression à J_k feuilles φ_k sur la base de données $\left(x_i, \tilde{y}_{i,k}\right)_{i\in\{1,\dots,n\}}$.
5. $\phi_k(x) = \phi_{k-1}(x) + \nu\, \varphi_k(x)$

Figure 30 : Algorithme du gradient boosting appliqué à la base automobile

Le coefficient ν est appelé *shrinkage coefficient*, il pénalise l'ajout d'un nouveau modèle, et ralentit l'ajustement. On choisira ici $\nu = 0{,}1$. Si le coefficient est trop faible, l'ajustement va être très lent, et le nombre d'arbres à construire sera alors très élevé pour avoir un modèle optimal. Au contraire, si le coefficient est plus élevé, l'ajustement va être rapide, mais on s'expose à des problèmes de sur-ajustement, car l'algorithme va rapidement converger et le poids des modèles parfaitement adaptés va être grand dans l'estimation finale.

Remarque

Classiquement, nous utilisons de petits arbres pour le *boosting* et de gros arbres pour le *bagging*. Dans l'article de Rosset (2005), l'auteur essaie de les connecter. Notamment, l'auteur montre que l'on peut voir la méthode du *bagging* comme un dérivé simplifié du *gradient boosting* dans le cas d'une fonction de perte linéaire en F. En ce cas, toutes les directions de descente deviennent identiques, et cela se réduit à réaliser des échantillons *bootstrap*. La fonction de perte linéaire du *bagging* traite de la même façon toutes les observations. En ce sens, le *bagging* est plus robuste et plus stable que le *boosting*, mais moins adapté aux données.

4.3.4. Résultats

On a le graphique suivant :

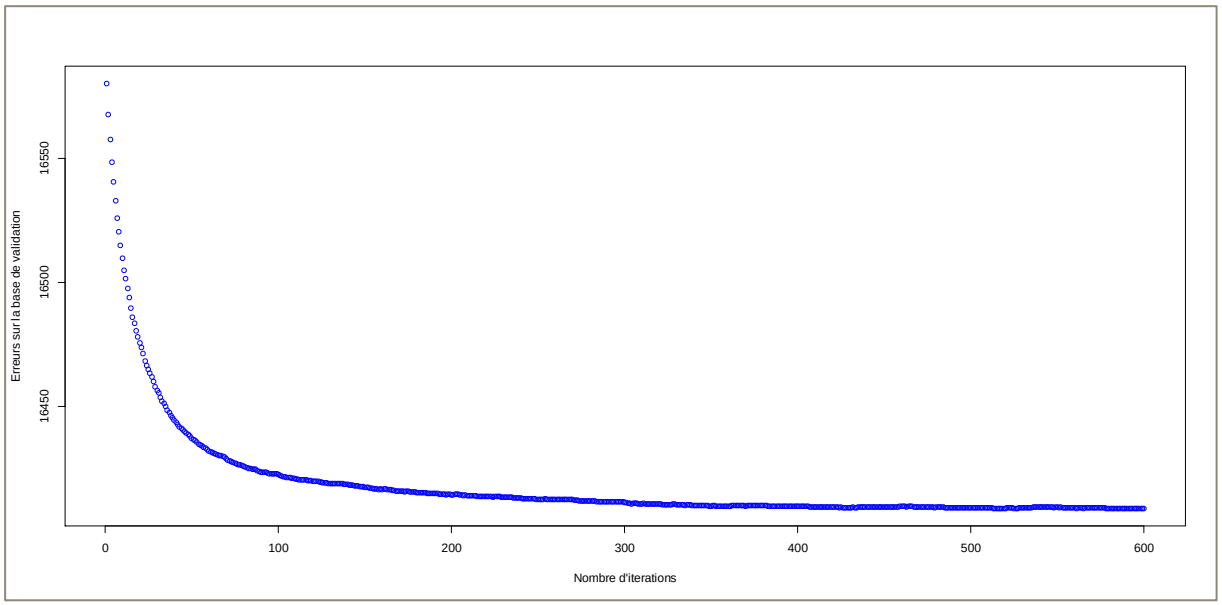


Figure 31 : Erreur de validation en fonction de B pour le gradient boosting

Ici, **nous retenons le modèle du *gradient boosting* avec 515 itérations**. C'est important de souligner que l'algorithme est déterministe. Autrement dit, il a l'avantage de toujours donner les mêmes résultats même une fois relancé, contrairement au *bagging* et au *random forest*.

4.3.5. Stochastic gradient boosting

L'idée de Friedman (2002) dans son article sur le *stochastic gradient boosting* est de combiner le *bootstrap* réalisé lors du *bagging* avec la méthode de plus profonde descente de l'algorithme du gradient.

La modification sur l'algorithme du *gradient boosting* est mineure. Il est en effet rigoureusement identique, à ceci près que les arbres sont construits à chaque itération sur un échantillon de taille plus petite que l'on a préalablement tiré sans remise. Notons qu'un échantillon *bootstrap* correspond à un tirage avec remise, de même taille n. Ici, on ne tire qu'un nombre $\tilde{n} < n$ et sans remise. Bien sûr, si $\tilde{n} = n$ alors cela ne perturbe pas la procédure puisque nous avons simplement effectué une permutation des données de l'échantillon. Afin de mieux manipuler ces notions, nous montrons en annexe G que $\tilde{n} = \frac{n}{2}$ est équivalent à effectuer un échantillon *bootstrap* (dans un sens que nous précisons). En vérité, on peut jouer sur le facteur $f = \frac{\tilde{n}}{n} \in]0,1[$.

Voici la convergence de l'algorithme :

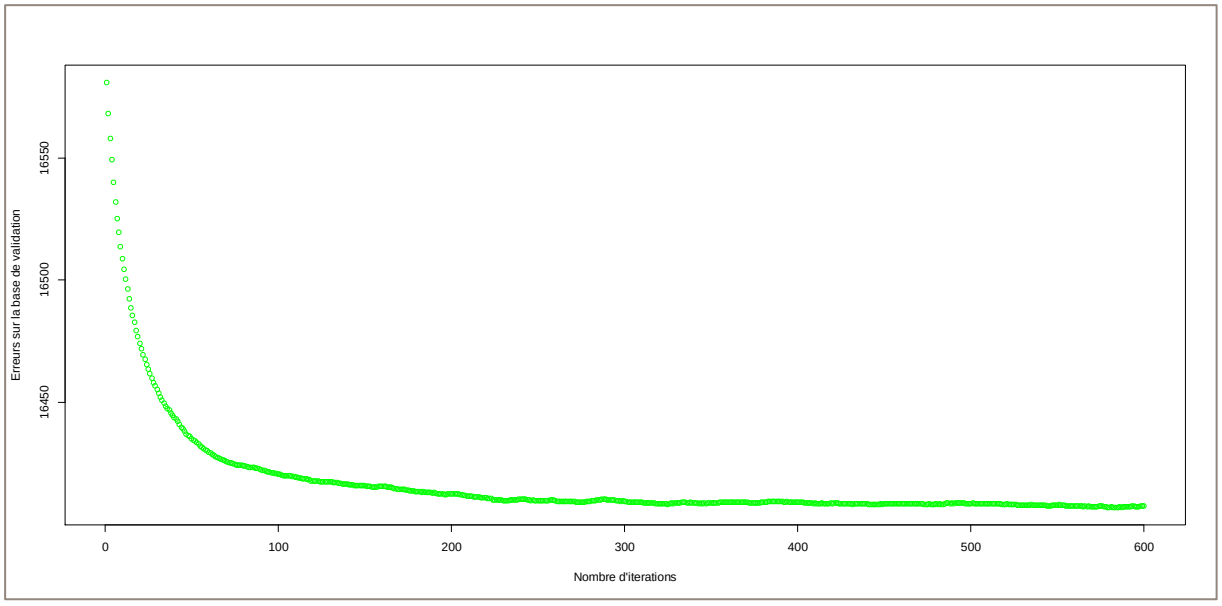


Figure 32 : Erreur de validation en fonction de B pour le stochastic gradient boosting

Nous retenons le modèle de *stochastic gradient boosting* avec 585 itérations.

À ce stade du mémoire, nous disposons :

- D'un modèle *GLM* (régression de Poisson),
- D'un modèle *CART* optimal à 122 feuilles,
- D'un modèle *bagging* avec 55 itérations,
- D'un modèle *random forest* avec 17 itérations,
- D'un modèle *gradient boosting* avec 515 itérations,
- D'un modèle *stochastic gradient boosting* avec 585 itérations.

Avant de tester les méthodes, il peut être intéressant de montrer les graphiques de convergence pour les 4 méthodes d'agrégation testées. Nous obtenons :

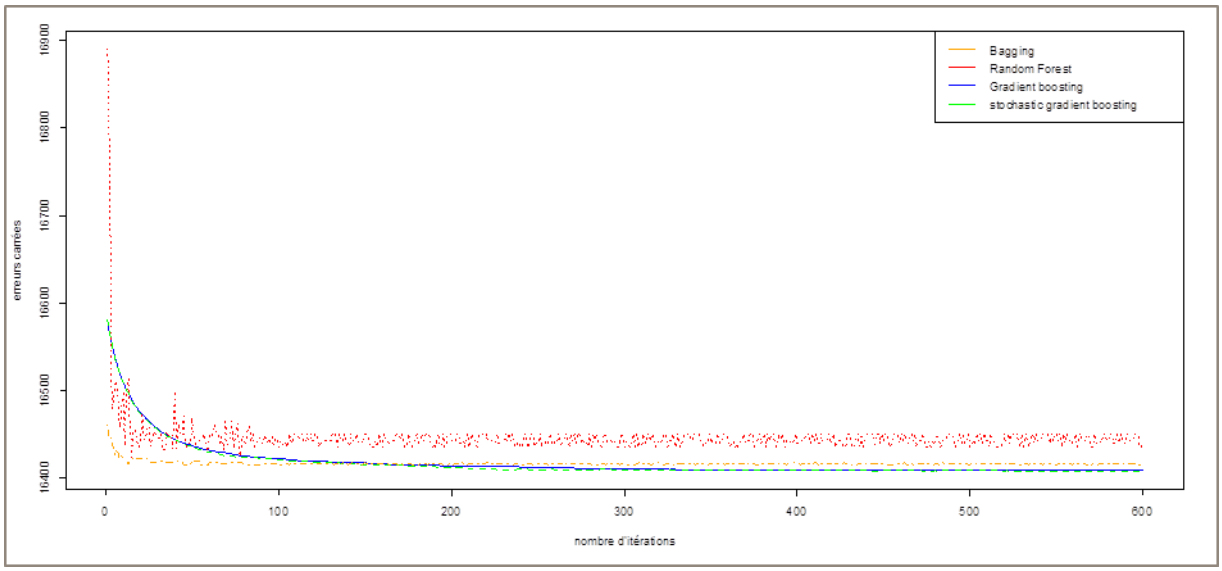


Figure 33 : Erreur de validation en fonction de B selon les méthodes

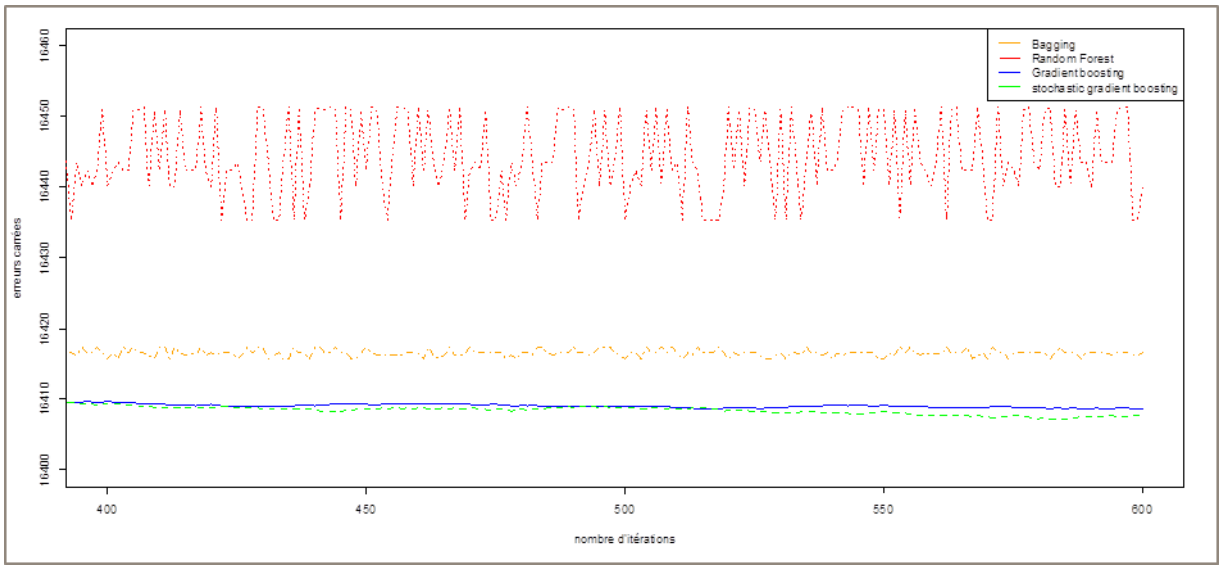


Figure 34 : Erreur de validation en fonction de B selon les méthodes (zoom)

On observe que l'algorithme du gradient converge plus lentement mais fournit de meilleurs résultats. De plus, il s'exécute par exemple beaucoup plus rapidement que le *bagging* qui nécessite de construire une série d'arbres saturés.

5. ANALYSE DES RÉSULTATS

Dans cette partie, nous souhaitons mener des analyses comparatives entre les différents modèles sélectionnés. Pour l'instant, seules les bases d'apprentissage et de validation ont été utilisées. Toutes les grandeurs et erreurs à venir ont été estimées sur **la base de test.**

5.1. POUVOIR DISCRIMINANT

Les courbes de *Lift* sont une mesure du pouvoir discriminant d'un modèle prédictif par rapport au modèle purement aléatoire uniforme. Nous présentons la théorie sous-jacente en annexe H et les courbes sont tracées Figure 35 page suivante.

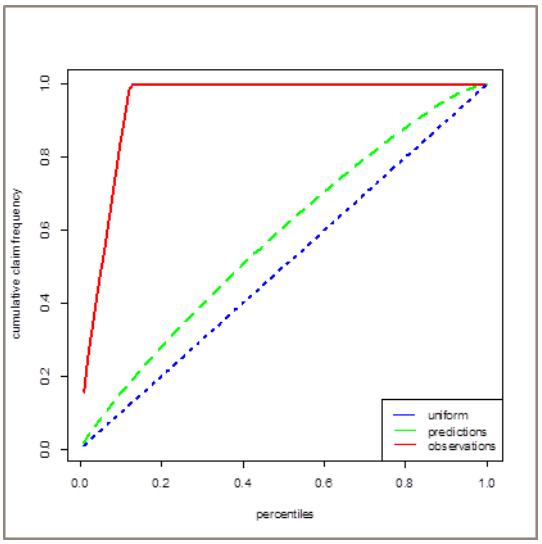
Les courbes seules, même si elles donnent des indications, ne suffisent pas. Nous les utilisons comme support pour le calcul des indices de Gini permettant de quantifier l'écart entre le modèle prédit et le modèle parfait. Voici les résultats obtenus sur la base de test :

INDICE DE GINI EN FONCTION DES MÉTHODES	
Méthode	**Indice de Gini**
GLM sélectionné	17,88 %
Arbre Optimal (122 feuilles)	19,33 %
Bagging (B=55)	2,35 %
Random Forest (B=17)	16,90 %
Gradient Boosting (B=515)	18,28 %
Stochastic Gradient Boosting (B=585)	18,63 %

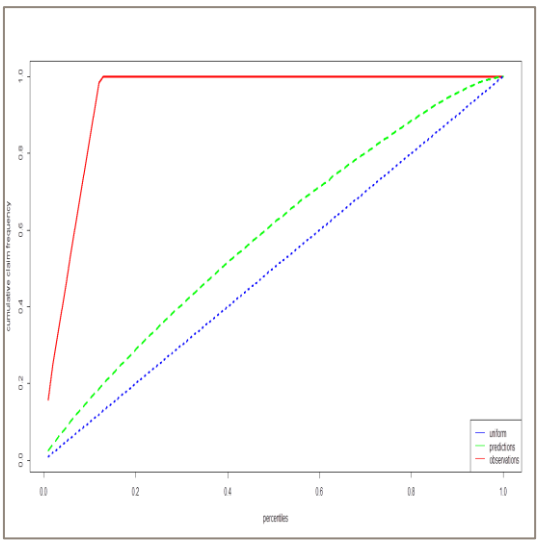
Nous notons que c'est l'arbre optimal *CART* qui présente le meilleur indice, tandis que la méthode du *bagging* délivre un indice très médiocre. Une interprétation possible est que le *bagging*, en réalisant une série d'échantillons *bootstrap*, ne considère qu'environ $50\,\%$ de la base. Les méthodes du *random forest* et du *stochastic gradient boosting* le font dans une moindre mesure, en forçant l'arbre à se diversifier en imposant des variables explicatives pour l'un et une direction de descente pour l'autre.

Il faut rappeler que l'indice de Gini n'est pas un critère pour juger la performance intrinsèque d'un modèle, pour cela nous nous reportons à l'équivalence (4) de l'introduction.

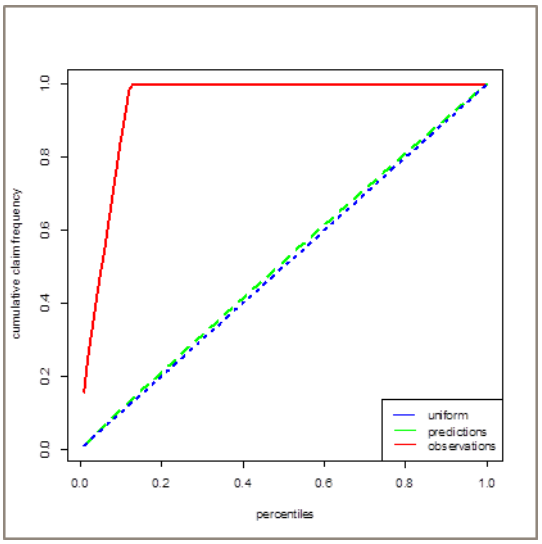
On notera de plus que ces indices sont globalement mauvais, cela est plus lié à la base de données qu'aux modèles, aussi nous ne commenterons pas plus leurs valeurs.



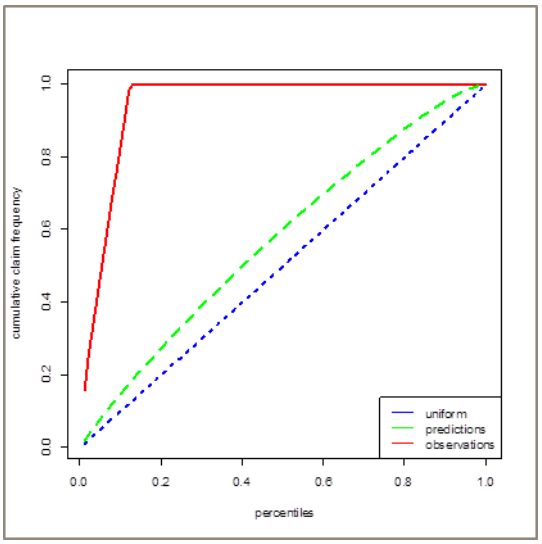
GLM



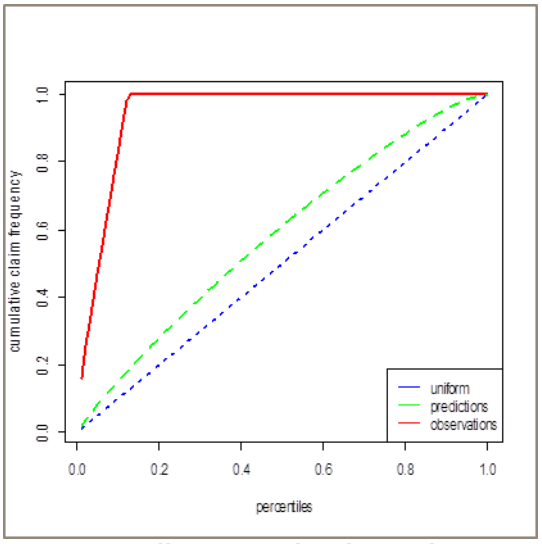
Arbre Optimal CART (122 feuilles)



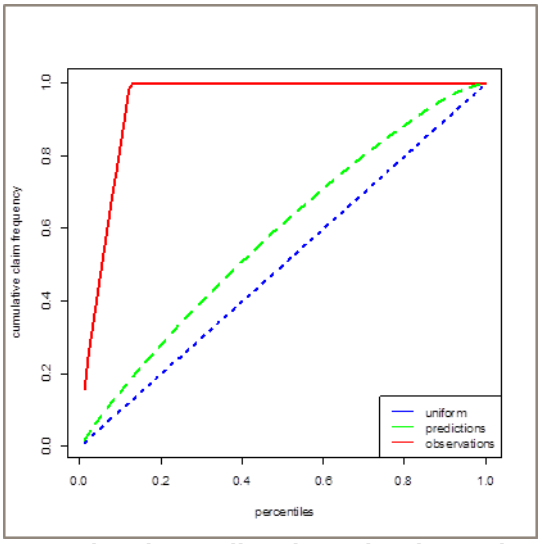
Bagging (B=55)



Random Forest (B=17)



Gradient Boosting (B=515)



Stochastic Gradient boosting (B=585)

Figure 35 : Courbes de Lift des différentes méthodes

5.2. POUVOIR DE PRÉDICTION

Dans cette partie, nous utilisons la base de test pour mesurer les écarts entre les valeurs prédites et les valeurs observées. Concrètement, nous disposons d'un échantillon test $(x_i, y_i)_{i=1,\dots,n}$ de taille n.

D'après l'équivalence (4) de l'introduction, nous souhaitons évaluer :

$$\text{erreur} = \sum_{i=1}^{n} \left(y_i - \hat{\phi}_{meth}(x_i)\right)^2$$

où $\hat{\phi}_{meth}$ est la fonction de prévision associée à la méthode en jeu. Nous l'avons dit, c'est ce critère qu'il importe de minimiser afin de trouver la meilleure méthode. Il est égal à la norme 2, à la racine près.

Néanmoins, afin de parfaire notre compréhension, nous proposons 3 autres mesures que sont la norme 1, la norme infinie et la déviance d'une loi de Poisson :

Norme 2, N^2

- Elle est définie par : $N^2(Y,\phi) = \|Y-\phi\|_2 = \sqrt{\sum_{i=1}^{n}(y_i-\phi(x_i))^2} = \sqrt{\text{erreur}}$.
- La norme 2 est donc le critère que nous souhaitons avant tout minimiser, en accord avec tout le contexte mathématique que nous avons dépeint.

Norme 1, N^1

- Elle est définie par : $N^1(Y,\phi) = \|Y-\phi\|_1 = \sum_{i=1}^{n}|y_i-\phi(x_i)|$.
- Nous revenons sur cette norme dans la partie 5.3.

Norme infinie, N^∞

- Elle est définie par : $N^\infty(Y,\phi) = \|Y-\phi\|_\infty = \max_{i=1,\dots,n}|y_i-\phi(x_i)|$.
- Elle permet d'évaluer l'erreur maximale que l'on commet avec ce modèle. La norme infinie peut être faible même s'il y a un grand nombre d'erreurs du moment qu'elles sont de faible amplitude.

Déviance, D

- Nous notons D la déviance d'une loi de Poisson donnée en équation (7).
- Elle s'écrit : $D(Y,\phi) = 2\sum_{i=1}^{n}\left(\phi(x_i) - y_i + y_i \ln\frac{y_i}{\phi(x_i)}\right)$.
- Nous introduisons cette grandeur issue de l'approche paramétrique car nous voulons comparer nos méthodes d'apprentissage avec le GLM. En pratique pour le calcul on discutera sur la nullité de y_i.

Une illustration classique des 3 premières normes dans le plan $\mathbb{R}^2$ consiste à représenter l'ensemble des points $x = (x_1, x_2)$ de norme 1 (boule unité), on obtient la représentation suivante :

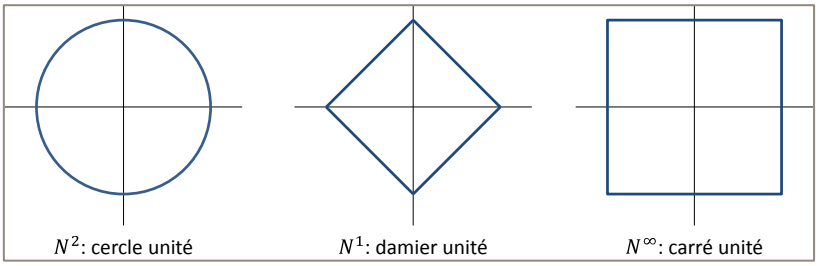


Figure 36 : Représentation des normes dans le plan

Si la norme infinie est petite, cela signifie qu'il n'y a pas de grands écarts de prévision pour chaque individu pris individuellement : cela se traduit bien géométriquement. En effet, l'espace ambiant est ici $\mathbb{R}^n$, la dimension associée (les axes) correspond donc aux individus. La boule unité, ou plus généralement une boule de rayon donné, est représentée par un « carré » de $\mathbb{R}^n$. Autrement dit on réalise bien que les individus sont traités indépendamment. Ce n'est en revanche pas le cas de la norme 2 où la forme de cercle suppose une implication jointe de tous les individus.

Rappelons que toutes ces distances sont uniquement calculées à titre indicatif, elles permettent d'avoir des informations complémentaires et d'ouvrir la discussion mais ne pourront véritablement servir à la décision. En effet, le juste critère à utiliser est celui de la norme 2 puisque c'est celui-là même qui a servi à la construction des modèles.

5.2.1. Influence des regroupements

Nous en avons parlé durant ce mémoire. Les *GLM* requièrent bien souvent de constituer des *buckets* afin de regrouper des modalités de variables. Ces regroupements ne sont pas aisés et parfois mêmes arbitraires. Nous testons ici l'influence de ces regroupements sur l'erreur générée par les arbres *CART* :

RÉSULTATS OBTENUS POUR LES ARBRES ENTRE DONNÉES AVEC ET SANS BUCKETS				
Buckets	**Méthode**	N^2	N^1	N^∞
avec	Arbre tronc	86,6518	11849,1700	4,8660
avec	Arbre saturé (1000 obs)	86,3184	11731,6500	4,8497
avec	Arbre saturé (500 obs)	86,3671	11719,6800	4,8697
avec	Arbre Optimal (122 feuilles)	86,2952	11733,8200	4,8497
avec	Sto. Gradient Boosting (B=585)	86,1568	11725,3900	4,8533
sans	Arbre tronc	86,6518	11849,1700	4,8660
sans	Arbre saturé (1000 obs)	86,3321	11734,4500	4,8497
sans	Arbre saturé (500 obs)	86,4213	11728,9800	4,8724
sans	CART Optimal (139 feuilles)	86,3138	11736,8800	4,8497
sans	Sto. Gradient Boosting (B=550)	86,1598	11725,2100	4,8559

Ce tableau permet de conclure que sur cette base de test et avec ces *buckets*, la base avec *buckets* est une meilleure source d'apprentissage pour les arbres *CART*. Toutefois nous sommes bien loin de pouvoir généraliser. Les méthodes d'agrégation, représentées ici par le *stochastic gradient boosting*, semblent montrer des résultats similaires que les données soient regroupées ou non.

Les méthodes ensemblistes semblent donc présenter l'avantage non négligeable de ne pas nécessiter de préciser a priori des *buckets*.

5.2.2. Influence de la base de test

Il faut nuancer tous les résultats obtenus dans cette partie, car ceux-ci sont intimement liés à l'homogénéité au sein de la base de test. Pour rappel, la séparation entre données d'apprentissage, de validation et de test (cf. introduction) est effectuée aléatoirement. Il est ainsi plus ou moins probable que la base de test soit plus ou moins sympathique. À titre d'exemple, voici quelques résultats obtenus à l'aide d'une autre base de test :

RÉSULTATS OBTENUS POUR LES ARBRES ET LE GLM AVEC DEUX BASES DE TEST DIFFÉRENTES				
Base de test	**Méthode**	N^2	N^1	N^∞
Actuelle	Arbre tronc	86,6518	11849,1700	4,8660
Actuelle	Arbre saturé (1000 obs)	86,3184	11731,6500	4,8497
Actuelle	Arbre optimal (122 feuilles)	86,2952	11733,8200	4,8497
Actuelle	GLM sélectionné	86,2150	11730,3100	4,8501
Autre	Arbre tronc	86,7694	11862,6300	5,8664
Autre	Arbre saturé (1000 obs)	86,4292	11728,8900	5,8045
Autre	Arbre optimal (114 feuilles)	86,3962	11732,1800	5,7897
Autre	GLM sélectionné	86,3798	11733,7800	5,8287

Ces résultats ne sont pas voués à être commentés et sont simplement présentés pour illustrer la dépendance des conclusions à la base de test.

5.2.3. Résultats

Nous présentons ci-dessous les résultats obtenus sur les données avec *buckets*, pour les différentes méthodes citées et testées au long de ce mémoire. À nouveau précisons que les erreurs sont évaluées sur la base de test.

RÉSULTATS OBTENUS				
Méthode	N^2	N^1	N^∞	D
Arbre tronc	86,6518	11849,1700	4,8660	29185,8800
Arbre saturé (1000 obs)	86,3184	11731,6500	4,8497	28699,2800
Arbre optimal (122 feuilles)	86,2952	11733,8200	4,8497	28670,7400
GLM sélectionné	86,2150	11730,3100	4,8501	28571,0600
Bagging (B=55)	86,1904	11729,9600	4,8432	28535,8200
Random Forest (B=17)	86,1893	11733,5000	4,8550	28537,0700
Gradient Boosting (B=515)	86,1511	11723,2400	4,8596	28488,4000
Sto. Gradient Boosting (B=585)	86,1568	11725,3900	4,8533	28494,6500

On remarque d'abord que les grandeurs se ressemblent beaucoup. Ceci est principalement dû au fait que nos prévisions correspondent à la fréquence moyenne de sinistres, de l'ordre 10 %, et que l'on compare ces valeurs prédites à des séquences de 0 et 1 principalement. On ne pourra jamais espérer se rapprocher de zéro pour l'erreur.

On décide pour faire apparaitre plus clairement les résultats de calculer les écarts relatifs par rapport au modèle trivial qui est l'arbre tronc. Pour rappel, l'arbre tronc associe à tous les individus la moyenne empirique de la base d'apprentissage. Nous obtenons le tableau ci-dessous.

ECARTS RELATIFS À L'ARBRE TRONC (ÉCARTS AU MODÈLE TRIVIAL)				
Méthode	N^2	N^1	N^∞	D
Arbre tronc	0,000 %	0,000 %	0,000 %	0,000 %
Arbre saturé (1000 obs)	0,385 %	0,992 %	0,334 %	1,667 %
Arbre optimal (122 feuilles)	0,412 %	0,973 %	0,334 %	1,765 %
GLM sélectionné	0,504 %	1,003 %	0,327 %	2,107 %
Bagging (B=55)	0,533 %	1,006 %	0,468 %	2,227 %
Random Forest (B=17)	0,534 %	0,976 %	0,226 %	2,223 %
Gradient Boosting (B=515)	0,578 %	1,063 %	0,131 %	2,390 %
Sto. Gradient Boosting (B=585)	0,571 %	1,045 %	0,261 %	2,368 %

Nous pouvons le représenter graphiquement :

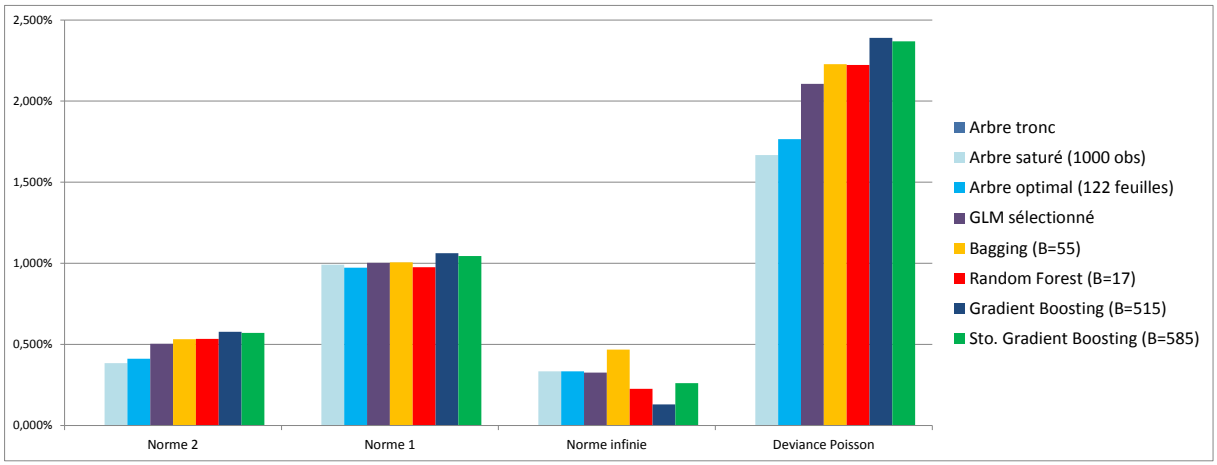


Figure 37 : Graphe des écarts relatifs des normes par rapport au modèle trivial

On conclut que le *gradient boosting* est la meilleure méthode au sens de la norme la plus importante, à savoir la norme 2. Elle l'est également au sens de la norme 1 et de la déviance.

Toutefois, on pourrait souhaiter comparer les méthodes d'apprentissage avec le *GLM* sélectionné. Pour cela, nous évaluons la grandeur suivante :

$$\frac{\frac{Norme(Meth) - Norme(Tronc)}{Norme(Tronc)}}{\frac{Norme(GLM) - Norme(Tronc)}{Norme(Tronc)}}$$

Nous obtenons le tableau page suivante.

RAPPORTS DES ECARTS RELATIFS ENTRE LE MODÈLE ET LE GLM (GAINS PAR RAPPORT AU GLM)				
Méthode	N^2	N^1	N^∞	D
Arbre optimal (122 feuilles)	81,642 %	97,047 %	102,215 %	83,787 %
GLM sélectionné	100,000 %	100,000 %	100,000 %	100,000 %
Bagging (B=55)	105,626 %	100,294 %	143,308 %	105,732 %
Random Forest (B=17)	105,871 %	97,316 %	69,181 %	105,528 %
Gradient Boosting (B=515)	114,620 %	105,948 %	39,999 %	113,445 %
Sto. Gradient Boosting (B=585)	113,324 %	104,139 %	79,884 %	112,428 %

Nous pouvons le représenter de cette façon :

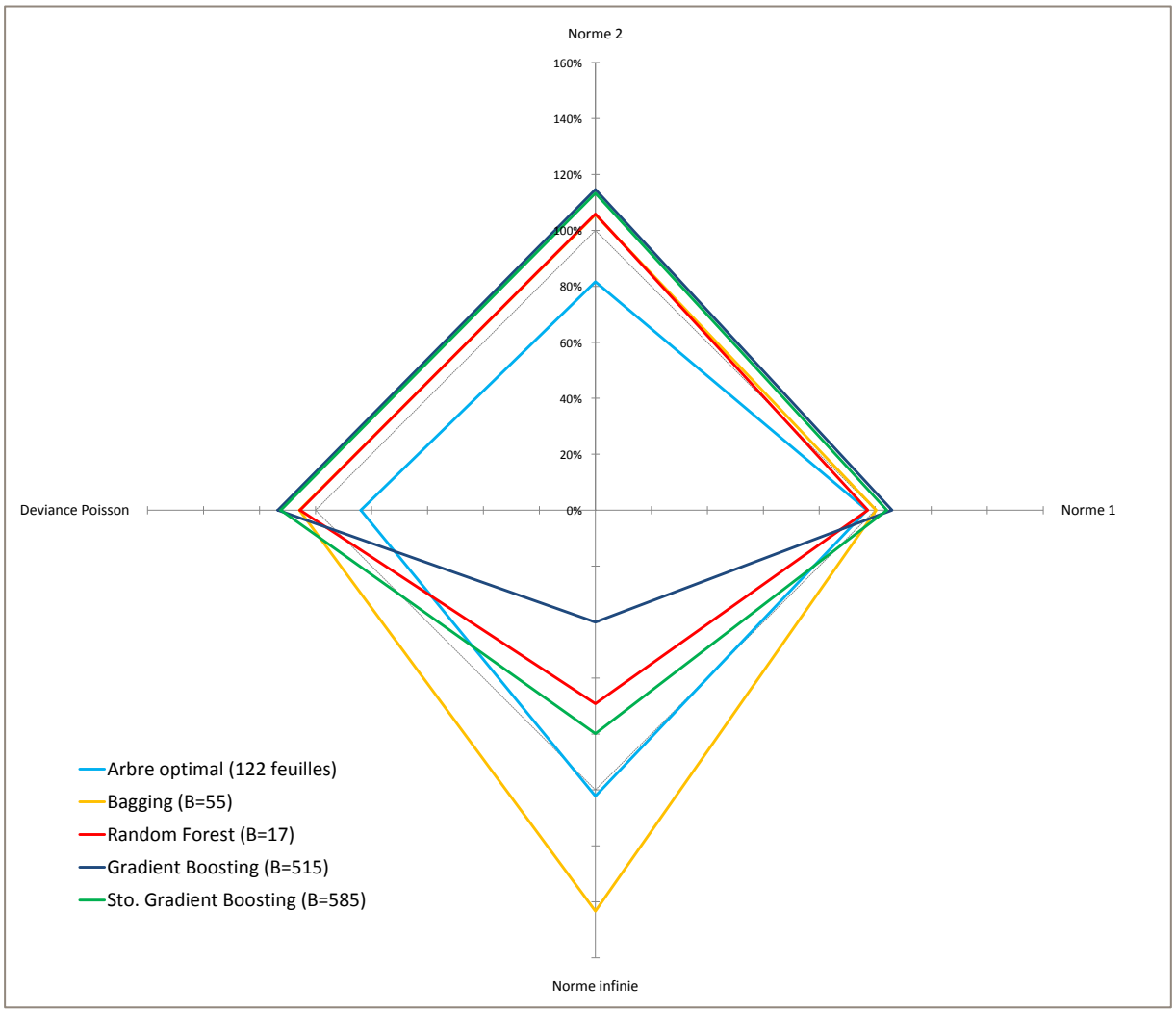


Figure 38 : Gains des modèles testés par rapport au modèle GLM

Si on oublie un instant la prépondérance du critère N^2, alors la meilleure méthode serait celle qui maximise ses valeurs sur les 4 axes. La méthode du *stochastic gradient boosting* ressort alors plus nettement que le gradient simple. En effet, pour des gains à peu près similaires par rapport au *GLM* sur les axes N^2, N^1, D, la méthode stochastique permet d'augmenter le gain sur N^∞. Aussi le *bagging*, bien que meilleur en norme infinie, ne convient pas car il est bien plus faible sur les autres normes. On conclut de cette étude que **les méthodes adaptatives (déterministe et stochastique) sont les meilleures testées**.

5.3. À PROPOS DE LA ROBUSTESSE

La théorie mathématique de la robustesse est souvent délaissée par les actuaires, le formalisme sous-jacent étant pourtant digne d'intérêt. En pratique, l'on se contente de dire qu'une méthode est peu robuste si un changement mineur dans les données provoque un changement important dans le modèle. C'est exact, mais légèrement incomplet. Nous jugeons intéressant, avant de conclure, de présenter les concepts mathématiques liés la théorie de la robustesse (cf. MARCEAU et RIOUX, 2001, et cf. BREHENY, 2010).

5.3.1. Contexte

La principale question est la suivante : qu'est-ce qu'un bon estimateur ? Quand le modèle est connu et correct, les réponses standards s'appliquent (méthode du maximum de vraisemblance par exemple) et sont valables. Cependant, si l'on considère le problème de la modélisation de la prime pure dans une approche paramétrique, alors les modèles choisis ne seront jamais parfaits. En statistique, les méthodes classiques reposent sur des hypothèses assez lourdes qui ne sont jamais vérifiées en pratique. Malheureusement, s'il y des données aberrantes (*outliers*) ou une mauvaise spécification du modèle, alors il est probable que les méthodes classiques aient une bien pauvre performance.

Pour quantifier la robustesse d'une méthode, il est nécessaire de définir des mesures associées. Les plus communes sont le point de rupture, la fonction de sensibilité et la fonction d'influence. La fonction d'influence a été introduite par Hampel (1974), pour étudier le comportement infinitésimal de fonctionnelles. Dans la suite, on considère une variable aléatoire X dont la fonction de répartition est notée F dépendant d'un paramètre θ, un échantillon $(X_i)_{i=1\dots n}$ et une réalisation $(x_i)_{i=1\dots n}$. On note $\hat{\theta}_n$ un estimateur sans biais de θ, c'est donc une variable aléatoire. Généralement, $\hat{\theta}_n$ est défini comme une fonction de $(X_i)_{i=1\dots n}$, pour faire plus de sens avec l'approche de Hampel, on présente $\hat{\theta}_n$ comme une fonction T de la fonction de répartition empirique F_n :

$$\begin{aligned} \theta &= T(F) \\ \hat{\theta}_n &= T(F_n) \end{aligned}$$

Un tel estimateur est qualifié d'estimateur *plug-in*. Notons que l'on peut écrire $\hat{\theta}_n((X_i)_{i=1\dots n}) = T(F_n)$ car les $(X_i)_{i=1\dots n}$ sont permutables et contiennent strictement la même information que F_n.

5.3.2. Point de rupture

Le point de rupture (*breakdown point*) se définit comme la plus grande proportion de données aberrantes ou corrompues qu'un estimateur peut supporter. Plus concrètement, le point de rupture d'un estimateur $\hat{\theta}_n$ est la proportion d'observations arbitrairement grandes que l'estimateur peut supporter avant de devenir lui aussi arbitrairement grand.

Plus ce point est haut, et plus l'estimateur est qualifié de robuste. Heuristiquement, on comprend que le point de rupture ne peut excéder $50\,\%$, car il y a alors plus de données contaminées que de données issues de la réelle distribution. Par suite, le point de rupture est toujours compris entre $0\,\%$ et $50\,\%$.

5.3.3. Fonction de sensibilité

La fonction de sensibilité (*sensitivity function*) est la fonction d'influence empirique définie ci-après. Elle renseigne sur le comportement de l'estimateur lorsque l'on perturbe l'échantillon en ajoutant une donnée arbitraire. Plus précisément, on ajoute la valeur x, une $n+1$ ième valeur à l'échantillon et on observe le changement.

La fonction de sensibilité d'une fonctionnelle T et d'un échantillon $(X_i)_{i=1\dots n}$ est définie par :

$$SF_{T,X}(x) = \frac{T_{n+1}(X_1, \dots, X_n, x) - T_n(X_1, \dots, X_n)}{\frac{1}{n+1}}$$

Cette définition est facile à interpréter, puisque l'on souhaite quantifier l'écart entre l'estimateur sur les données contaminées avec le « vrai » estimateur, on normalise toutefois par la taille du nouvel échantillon. Cette fonction présente deux avantages :

- Elle permet d'évaluer facilement la robustesse d'un estimateur,
- Elle conceptualise un cadre favorable à la compréhension de la définition de la fonction d'influence, en effet il suffit de laisser $n \to \infty$.

5.3.4. Fonction d'influence

La fonction d'influence est définie pour des lois continues, perturbées par l'ajout d'une masse de probabilité. Elle décrit l'impact d'une perturbation infinitésimale au point x des données, avec un poids λ. Un estimateur sera qualifié de robuste si la fonction d'influence est bornée, autrement dit qu'elle ne peut pas devenir arbitrairement grande à raison que x devient grand.

Si l'on considère une fonctionnelle T d'une distribution F, pour $x \in \mathbb{R}$, alors la fonction d'influence est définie par :

$$IF_{T,X}(x) = \lim_{\lambda \to 0^+} \frac{T(F_{\lambda,x}) - T(F)}{\lambda} = \left.\frac{dT(F_{\lambda,x})}{d\lambda}\right|_{\lambda=0}$$

Où $F_{\lambda,x}$ est définie pour $x \in \mathbb{R}$ et $\lambda \in]0,1[$ par $F_{\lambda,x}: \mathbb{R} \to \mathbb{R}, u \longmapsto F(u) + \lambda\left(\mathbb{1}_{\{x \le u\}} - F(u)\right)$.

On vérifie bien que $F_{\lambda,x} \to F$ quand $\lambda \to 0$. En général, $F_{\lambda,x}$ correspond à la distribution contaminée. On peut procéder à un développement de Taylor à l'ordre 1 :

$$T(F_{\lambda,x}) = T(F) + \lambda\, IF_{T,F}(x) + \cdots$$

À l'aide d'un développement de Von Mises (adapté aux fonctionnelles et généralisant l'équation ci-dessus), on peut montrer que :

$$T(F_n) = T(F) + \frac{1}{n}\sum_{i=1}^{n} IF_{T,X}(X_i) + o_{n\to\infty}(1)$$

Que l'on peut encore écrire :

$$\hat{\theta}_n - \theta = \frac{1}{n}\sum_{i=1}^{n} IF_{T,X}(X_i) + o_{n\to\infty}(1)$$

On reconnait la somme de n variables aléatoires i.i.d. de sorte que le théorème central limite permet de conclure sur un résultat asymptotique.

$$\sqrt{n}\left(\hat{\theta}_n - \theta\right) \to Nor\left(0, V\left[IF_{T,X}(X)\right]\right) \text{ en loi quand } n \to \infty$$

Il est important de rappeler que notre estimateur $\hat{\theta}_n$ est sans biais, ainsi nous devons avoir $E\left[IF_{T,X}(X)\right] = 0$, et de là $V\left[IF_{T,X}(X)\right] = E\left[IF_{T,X}(X)^2\right]$. Ce théorème permet d'obtenir des intervalles de confiance assez facilement, ce qui n'est pas aussi direct avec d'autres méthodes.

5.3.5. Exemples

5.3.5.1. Moyenne

L'estimateur empirique de la moyenne est donné par :

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Il est facile de voir que si l'on change arbitrairement un X_i donné, alors on peut rendre arbitrairement grande la valeur de l'estimateur. Le point de rupture est donc de $0\,\%$.

La fonction de sensibilité est donnée pour $x \in \mathbb{R}$ par :

$$\begin{aligned} SF_{T,X}(x) &= \frac{\frac{\sum_{i=1}^{n} X_i + x}{n+1} - \frac{\sum_{i=1}^{n} X_i}{n}}{\frac{1}{n+1}} \\ &= x - \frac{1}{n}\sum_{i=1}^{n} X_i \\ &= x - \bar{X}_n \end{aligned}$$

Par la loi des grands nombres, on remarque en particulier que $SF_{T,X}(x) \to x - E[X]$ quand $n \to \infty$.

Avant de nous pencher sur la fonction d'influence, il est nécessaire d'exprimer l'espérance comme une fonctionnelle de la fonction de répartition :

$$\begin{aligned} E[X] &= \int_{\mathbb{R}} x dF(x) \\ &= T(F) \end{aligned}$$

Ainsi $E[X] = \mu = T(F)$ et $\hat{\mu}_n = \bar{X}_n = T(F_n)$. La fonction d'influence est donnée par (on permutera dérivée et intégrale) :

$$\begin{aligned} E[X] &= \left.\frac{dT(F_{\lambda,x})}{d\lambda}\right|_{\lambda=0} \\ &= \left.\frac{d}{d\lambda}\left[\int_{\mathbb{R}} u dF_{\lambda,x}(u)\right]\right|_{\lambda=0} \\ &= \int_{\mathbb{R}} u d\left[\left.\frac{d}{d\lambda}F_{\lambda,x}(u)\right|_{\lambda=0}\right] \\ &= \int_{\mathbb{R}} u d\left[\left.\frac{d}{d\lambda}\Big(F(u) + \lambda\left(\mathbb{1}_{\{x\le u\}} - F(u)\right)\Big)\right|_{\lambda=0}\right] \\ &= \int_{\mathbb{R}} u d\left[\mathbb{1}_{\{x\le u\}} - F(u)\right] \\ &= \int_{\mathbb{R}} u d\left[\mathbb{1}_{\{x\le u\}}\right] - \int_{\mathbb{R}} u d[F(u)] \\ &= x - \mu \end{aligned}$$

On peut représenter la fonction, ci-après.

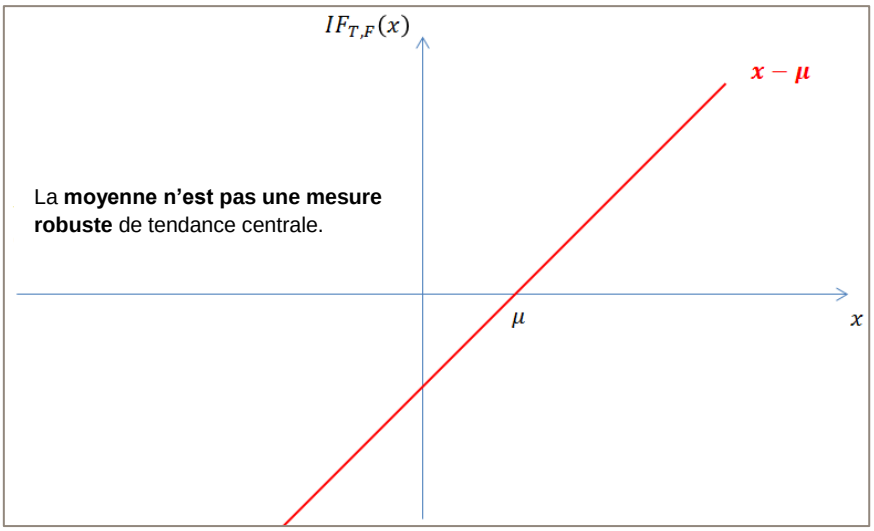


Figure 39 : Fonction d'influence de l'estimateur de la moyenne

Comme on le voit sur le graphe, la fonction d'influence n'est pas bornée.

On conclut que **la moyenne empirique n'est pas une mesure robuste** de valeur centrale. On pourra au passage instantanément s'assurer que $E\left[IF_{T,X}(X)\right] = 0$.

5.3.5.2. Médiane

Pour la médiane, définie comme étant le quantile à $50\,\%$ de X, il est plus difficile de calculer la fonction de sensibilité. Formalisons donc directement les concepts via la fonction d'influence. Nous pourrions utiliser l'inverse généralisé pour avoir une expression explicite mais cela complique grandement les calculs, on préfère utiliser des relations définissant implicitement la médiane notée $\pi = \pi[X] = VaR_{0,5}[X]$. On choisira tout de même pour simplifier une loi continue admettant une densité f pour simplifier les notations.

$$F\big(T(F)\big) = \frac{1}{2} \qquad \pi = T(F)$$
$$F_n\big(T(F_n)\big) = \frac{1}{2} \qquad \hat{\pi}_n = T(F_n)$$

Ces relations sont encore valables pour la distribution contaminée : $F_{\lambda,x}\left(T\left(F_{\lambda,x}\right)\right) = \frac{1}{2}$. Dérivons cette relation par rapport à λ :

$$\begin{aligned}
\frac{d}{d\lambda}\left[F_{\lambda,x}\left(T\left(F_{\lambda,x}\right)\right)\right] &= \frac{d}{d\lambda}\left[F\left(T\left(F_{\lambda,x}\right)\right) + \lambda\left(\mathbb{1}_{\{x \le T(F_{\lambda,x})\}} - F_{\lambda,x}\left(T\left(F_{\lambda,x}\right)\right)\right)\right] \\
&= \frac{d}{d\lambda}\left[T\left(F_{\lambda,x}\right)\right] f\left(T\left(F_{\lambda,x}\right)\right) + \left(\mathbb{1}_{\{x \le T(F_{\lambda,x})\}} - F_{\lambda,x}\left(T\left(F_{\lambda,x}\right)\right)\right) \\
&\quad + \lambda \frac{d}{d\lambda}\left[\mathbb{1}_{\{x \le T(F_{\lambda,x})\}} - F_{\lambda,x}\left(T\left(F_{\lambda,x}\right)\right)\right] \\
&= 0
\end{aligned}$$

Si on substitue $\lambda = 0$, alors on obtient $F_{0,x} = F$, $T\left(F_{0,x}\right) = \pi$ et $F_{0,x}\left(T\left(F_{0,x}\right)\right) = \frac{1}{2}$ de sorte que nous avons :

$$\left.\frac{dT\left(F_{\lambda,x}\right)}{d\lambda}\right|_{\lambda=0} f(\pi) + \left(\mathbb{1}_{\{x \le \pi\}} - \frac{1}{2}\right) = 0$$

Finalement, pour $x \in \mathbb{R}$,

$$IF_{T,X}(x) = \frac{\frac{1}{2} - \mathbb{1}_{\{x \leq \pi\}}}{f(\pi)}$$

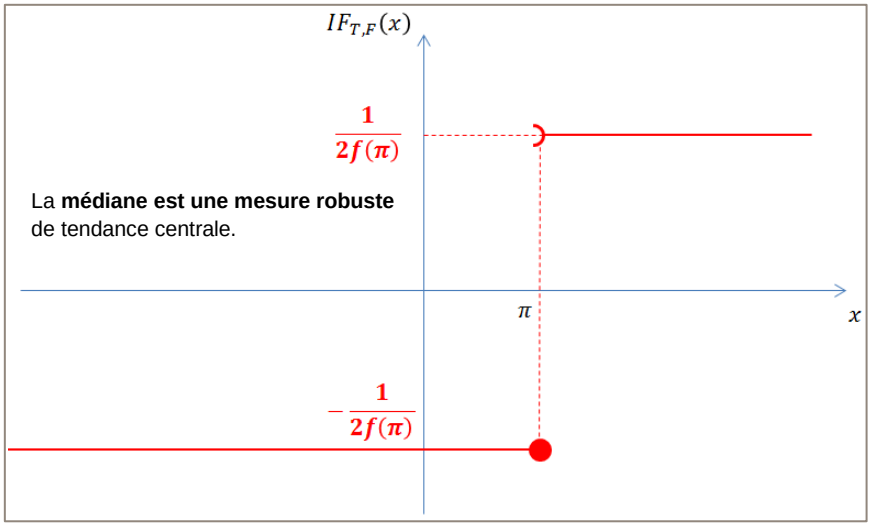


Figure 40 : Fonction d'influence de l'estimateur de la médiane

On remarque que la fonction d'influence est bornée : **l'estimateur plug-in de la médiane est robuste**. On notera de plus qu'un point à droite de la médiane a le même impact qu'un point à gauche, ce qui est intuitif. Le point de rupture est ici maximal, et égal à $50\,\%$.

On peut vérifier que :

$$\begin{aligned} E\left[IF_{T,X}(X)\right] &= \frac{\frac{1}{2} - E\left[\mathbb{1}_{\{X \leq \pi\}}\right]}{f(\pi)} \\ &= \frac{\frac{1}{2} - \mathbb{P}[X \leq \pi]}{f(\pi)} \\ &= 0 \end{aligned}$$

5.3.6. Applications

5.3.6.1. Arbres CART

Dans la mesure où la fonction d'influence a été présentée dans le cas paramétrique où la distribution dépendait d'un paramètre θ, il est délicat d'appliquer correctement la formalisation mathématique citée plus haut. Néanmoins, elle fournit des pistes de réflexion, c'est ainsi que BAR-HEN, GEY et POGGI (2011) proposent différentes fonctions d'influence. Nous laissons le soin au lecteur intéressé de s'y pencher, nous nous contenterons ici d'observer qualitativement les impacts de changement de la base d'apprentissage sur le résultat d'un *CART*.

Dans le cas d'algorithme d'apprentissage basé sur les arbres, le but est d'assigner à chaque vecteur d'entré une valeur donnée. La sensibilité peut dans ce contexte s'exprimer comme l'impact des variations sur la base d'apprentissage sur les estimations. La base d'apprentissage est la réalisation $(x_i, y_i)_{i=1\ldots n}$ avec $x_i \in \mathbb{R}^p$. La modification de l'échantillon se traduit par un nouvel échantillon de taille $n+1$ avec la nouvelle réalisation (x_{n+1}, y_{n+1}). On notera que les auteurs préfèrent en pratique considérer les arbres *jackknife* auxquels on retire une observation.

Pour un arbre donné, on peut isoler deux aspects :

- Les valeurs des prédictions,
- Les partitions réalisées par les différents nœuds (la structure de l'arbre).

Nous décidons de nous focaliser sur le second aspect, nous suggérons de créer les arbres à partir des échantillons privés de 1 % de la base totale d'apprentissage. Le problème est que l'on ne peut se permettre de tous les créer, nous nous contentons donc d'en faire 10 et nous relevons pour chaque arbre ainsi créé le nombre de feuilles de l'arbre optimal.

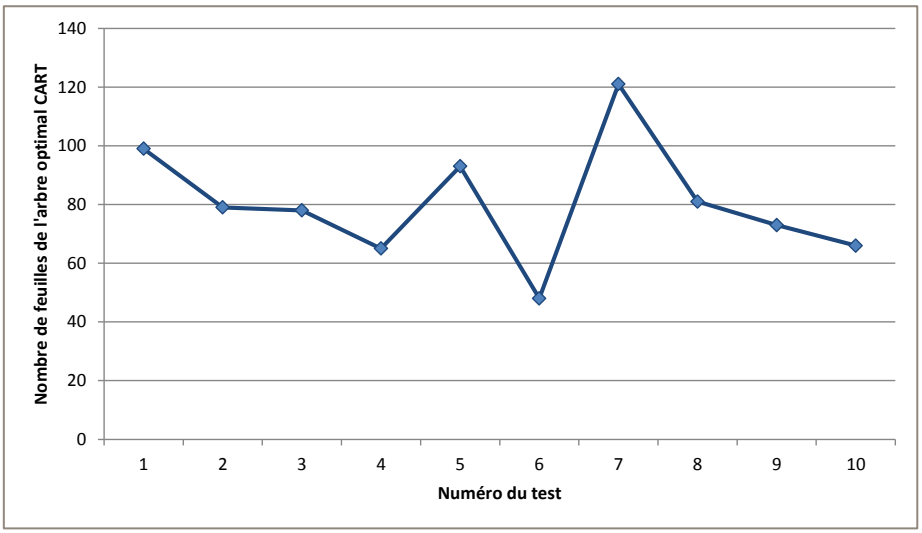


Figure 41 : Représentation du nombre de feuille du CART optimal en fonction du test

C'est un phénomène connu : la méthode *CART* apparait comme sensible aux perturbations des données d'apprentissage. En effet, le seul fait de retirer $1\,\%$ de la base d'apprentissage change grandement le nombre de feuilles. C'est d'ailleurs ce désavantage qui nous pousse vers des méthodes d'ensemble comme le *bagging*, les forêts aléatoires et le *boosting*.

5.3.6.2. De l'intérêt de la médiane et de la norme 1

Ce formalisme a permis de montrer que la moyenne n'est pas robuste, là où la médiane l'est. Il est dès lors légitime de s'interroger sur les choix de modélisation basés sur l'espérance. Nous l'avons vu dans l'introduction générale, c'est bien l'espérance conditionnelle qui apparait naturellement lorsque l'on opère dans l'espace de Hilbert des variables de carré intégrable $L^2(\Omega)$, ce cadre favorable permettant de transformer un problème de minimisation de distance en une projection. Nous ne remettons pas en question cette approche, tout à fait justifiée. Néanmoins il s'agit de s'interroger sur l'impact de données aberrantes sur nos estimations.

Nous l'avons vu la médiane semble généralement plus robuste. Soit $p \in]0,1[$ une probabilité fixée et posons $u = 2p - 1 \in]-1,1[$. On note $Q(u,t) = |t| + ut$ pour $u \in]-1,1[$ et $t \in \mathbb{R}$ la fonction de perte. Intéressons-nous au problème de minimisation suivant (cf. CHAOUCH, GANNOUN et SARACCO, 2009) :

$$\begin{aligned} \theta^* &= \arg\min_{\theta\in\mathbb{R}} E[Q(u, X-\theta)] \\ &= \arg\min_{\theta\in\mathbb{R}} \underbrace{E[|X-\theta| + u(X-\theta)]}_{g(\theta)} \end{aligned}$$

Bien sûr, la fonction n'est pas dérivable en $\theta = X$, toutefois en dehors $g'(\theta) = -sgn(X - \theta) - u$. On en déduit que θ^* est tel que

$$E[sgn(X - \theta^*)] + u = 0$$

De plus, on rappelle que $p = \frac{1+u}{2}$:

$$\begin{aligned} F_X\left(F_X^{-1}(p)\right) - p &= \mathbb{P}[X \leq F_X^{-1}(p)] - p \\ &= E\left[\mathbb{1}_{\{X \leq F_X^{-1}(p)\}}\right] - p \\ &= E\left[\mathbb{1}_{\left\{X \leq F_X^{-1}\left(\frac{1+u}{2}\right)\right\}}\right] - \frac{1+u}{2} \\ &= \frac{1}{2}\left(E\left[\underbrace{2\,\mathbb{1}_{\left\{F_X^{-1}\left(\frac{1+u}{2}\right) - X \geq 0\right\}} - 1}_{sgn\left(F_X^{-1}\left(\frac{1+u}{2}\right) - X\right)}\right] - u\right) \end{aligned}$$

Or par définition, et pour des lois continues, nous avons $F_X\left(F_X^{-1}(p)\right) - p = 0$, il reste donc :

$$E\left[-sgn\left(X - F_X^{-1}\left(\frac{1+u}{2}\right)\right)\right] - u = 0$$

On déduit que :

$$\theta^* = F_X^{-1}\left(\frac{1+u}{2}\right)$$

Par suite, si nous supposons $p = \frac{1}{2}$, alors $u = 0$ et on parvient à l'écriture suivante pour la médiane:

$$\pi[X] = VaR_{0,5}[X] = F_X^{-1}\left(\frac{1}{2}\right) = \arg\min_{\theta \in \mathbb{R}} E[|X - \theta|]$$

C'est un résultat intéressant, car on réalise le choix de la norme 1 au lieu de la norme 2 conduit à la médiane conditionnelle au lieu de l'espérance conditionnelle. Dans la mesure où la médiane est plus robuste que la moyenne, c'est un critère séduisant. On pourrait ainsi penser à utiliser la médiane conditionnelle en lieu et place de l'espérance conditionnelle lorsque les bases de données sont petites et où l'influence d'une ligne est importante.

5.3.6.3. Conclusions

La théorie de la robustesse, bien que méconnue, offre un cadre favorable et une conceptualisation appréciée pour l'étude de l'influence des observations sur la construction et l'apprentissage des arbres. Nous avons vu que les arbres de régression classiques ne peuvent pas être qualifiés de robustes, nous justifions donc a posteriori les méthodes ensemblistes. En effet, en introduisant de l'aléatoire celles-ci s'affranchissent un peu plus des données.

De plus, cette partie aura été l'occasion de se pencher sur l'intérêt de la norme 1, ayant un lien direct avec la médiane. Même si la norme 2 et la minimisation du carré des erreurs est le cadre théorique standard (vu en introduction), il peut être tout à fait cohérent de minimiser des écarts en valeurs absolues, ce que nous avons fait dans l'analyse quantitative.

CONCLUSION

Dans ce mémoire, nous avons appliqué de nouvelles méthodes de tarification sur une base de données automobile, en comparant les résultats obtenus avec ceux dérivant des modèles linéaires généralisés. La base de données a été subdivisée en trois parties : un échantillon d'apprentissage permettant de construire nos différents modèles, un échantillon de validation afin d'optimiser les méthodes basées sur les arbres et enfin un échantillon de test.

Nous avons discuté de la nécessité d'effectuer des *buckets*, ou regroupements de modalités, pour les *GLM*. En effet, les variables qualitatives à k modalités sont souvent décomposées en une série de $k-1$ variables binaires, ce qui peut conduire à un grand nombre de variables. Cela peut générer des problèmes liés au temps de calcul, de même qu'il devient difficile de gérer les interactions. La création de *buckets* est donc une étape importante mais qui n'est pas rigoureusement nécessaire si l'on applique la méthode *CART* et ses dérivés ensemblistes. C'est un avantage certain car ces regroupements traduisent d'emblée le caractère biaisé de l'étude.

La digression sur la théorie de la robustesse a posé les bases nécessaires pour la juste compréhension du traitement des données aberrantes par un arbre. C'est par l'expérience que nous montrons qu'ils sont relativement instables. En effet, des changements minimes peuvent induire ou non la sélection d'un attribut lors de l'apprentissage. Si celui-ci est près de la racine, alors la forme de l'arbre est grandement influencée. Nous justifions ainsi l'utilité des méthodes d'agrégation, afin d'augmenter la robustesse.

Ce sont sur les données de test que nous avons comparé les modèles, au sens de différentes normes. D'après la partie introductive, c'est la norme 2 qui est à considérer. À ce titre la méthode du *gradient boosting*, qu'elle soit stochastique ou non, offre les meilleurs résultats. On retiendra aussi que l'algorithme est rapide à mettre en œuvre, contrairement au *bagging* par exemple. Il est de plus crucial de rappeler que les conclusions que nous dressons sont indubitablement liées à la base de données. Nous ne pourrons certainement jamais affirmer que les méthodes d'apprentissage sont toujours meilleures.

Pour conclure, nous avons montré les limites des modèles linéaires généralisés pourtant largement utilisés en actuariat. Ces méthodes présupposent une forme particulière des risques et de leurs interactions, ce qui de fait limite la compréhension des subtilités du portefeuille. Nous avons ici proposé une approche nouvelle en tarification non-vie. Les résultats obtenus par les méthodes d'agrégation sont, dans notre cas, bien meilleurs. En outre, les arbres de régression offrent une vision synthétique de la base de données et leurs interprétations aisées peuvent enrichir la réflexion autour de la stratégie tarifaire.

Aujourd'hui, certaines études suggèrent de coupler les deux approches, à savoir *GLM* et *machine learning*, en appliquant un arbre *CART* sur les résidus. De plus en ouverture nous pouvons citer l'utilisation des modèles additifs généralisés (*GAM*) que nous avons volontairement écartés du périmètre de l'étude afin de nous concentrer sur les méthodes d'apprentissage. Ces modèles permettent de relâcher l'hypothèse de linéarité via des techniques de lissage. Enfin, les réseaux de neurones, bien que complexes conceptuellement, semblent également fournir de très bons résultats.

BIBLIOGRAPHIE

BAR-HEN A., GEY S., POGGI J.M. (2011) *Influence Functions for CART*. Mathématiques appliquées Paris 5, Laboratoire de Mathématiques d'Orsay

BERK R.A. (2004) *An introduction to Ensemble Methods for Data Analysis*. Department of Statistics UCLA

BREHENY P. (2010) *Statistical functionals and influence functions*.

BREIMAN L. (2001) *Random Forests*. Statistics Department, Berkeley

CHAOUCH M., GANNOUN A., SARACCO J. (2009) *Estimation de quantiles géométriques conditionnels et non conditionnels*. Journal de la Société Française de Statistique

DENUIT M., CHARPENTIER A. (2004) *Mathématiques de l'assurance non vie. Tome 1 : Principes fondamentaux de théorie du risque*. Economica

DENUIT M., CHARPENTIER A. (2005) *Mathématiques de l'assurance non vie. Tome 2 : Tarification et Provisionnement*. Economica

DREYFUS G., MARTINEZ J.M., SAMUELIDES M., GORDON M.B., BADRAN F., THIRIA S. (2008) *Apprentissage statistique*. Eyrolles

EMBRECHTS P., KLUPPELBERG C., MIKOSCH T. (1997) *Modelling extremal events for insurance and finance*. Springer

FREUND Y., SCHAPIRE R.E. (1996) *Experiments with a New Boosting Algorithm*. Machine Learning: Proceedings of the Thirteenth International Conference

FRIEDMAN J.H. (2001) *Greedy function approximation: a gradient boosting machine*. The annals of Statistics

FRIEDMAN J.H. (2002) *Stochastic gradient boosting*. Computational Statistics & Data Analysis

MARCEAU E., RIOUX J. (2001) *On robustness in risk theory*. Insurance: Mathematics and Economics

PAGLIA A., PHELIPPE-GUINVARC'H M.V. (2010) *Tarification des risques en assurance non-vie, une approche par modèle d'apprentissage statistique*. Bulletin Français d'Actuariat, Vol. 11, n°22

RIDGEWAY G. (1999) *The State of Boosting*. Computing Science and Statistics

ROSSET S. (2005) *Robust Boosting and its Relation to Bagging*. KDD

STROBL C., MALLEY J., TUTZ G. (2009) *An introduction to Recursive Partitioning : Rationale, Application and characteristics of classification and regression trees, bagging and random forests*. NIH

THERNEAU T.M., ATKINSON E.J. (1997) *An Introduction to Recursive Partitioning Using the RPART Routines*. Mayo Foundation

Package R 'gbm' (2013) *Generalized Boosted Regression Models*. CRAN

Package R 'ipred' (2013) *Improved Predictors*. CRAN

Package R 'randomForest' (2012) *Breiman and Cutler's random forests for classification and regression*. CRAN

Package R 'rpart' (2013) *Recursive Partitioning*. CRAN

Package R 'tree' (2013) *Classification and regression trees*. CRAN

TABLE DES FIGURES

NOTATIONS

X	Variable aléatoire (v.a.)
$E[X]$	Espérance
$V[X]$	Variance
$\sigma[X]$	Écart type
$\gamma[X]$	Coefficient d'asymétrie
$\pi[X]$	Médiane
$Cov[X,Y]$	Covariance entre X et Y
$\sigma(X)$	Tribu engendrée par X
$\mathbb{1}_A$	Fonction indicatrice sur l'ensemble A
$\sum_{i=1}^{0} = 0$	Convention adoptée
$VaR_\kappa[X]$	Value-at-Risk de niveau κ de X
F_X	Fonction de répartition de X
$S_X = \overline{F_X}$	Fonction de survie de X
$sgn(x) = \begin{cases} +1 \text{ si } x \geq 0 \\ -1 \text{ si } x < 0 \end{cases}$	Fonction « signe »

ANNEXES

A. NOTIONS DE PROBABILITÉS

Espace probabilisé

On considère une variable aléatoire X, i.e. le résultat d'une expérience ou d'un phénomène aléatoire. Formellement c'est une fonction qui à tout élément $\omega \in \Omega$ associe un nombre réel $X(\omega)$. Ω est appelé l'univers, cela correspond à l'ensemble des états du monde ω, en ce sens que ω correspond à une situation du monde. Néanmoins, il est très difficile pour ne pas dire impossible de décrire entièrement l'univers des possibles, et on préfère aborder les variables aléatoires par les probabilités.

On considère un ensemble $(\Omega, F, \mathbb{P})$ un espace probabilisé. L'ensemble Ω est l'univers des possibles, il est non vide. F est une tribu, autrement dit c'est une partie de $\mathcal{P}(\Omega)$ (un ensemble d'ensemble) qui est stable par passage au complémentaire et par union dénombrable. La probabilité $\mathbb{P}$ est une application de F dans $[0,1]$ qui à n'importe quel élément de l'ensemble des parties de Ω associe sa probabilité d'occurrence. Elle vérifie de plus $\mathbb{P}[\Omega] = 1$ et la propriété d'additivité pour les ensembles disjoints.

Si l'on décrit un phénomène par une variable aléatoire X, i.e. une application de Ω dans $\mathbb{R}$ (du moins en dimension 1), alors la tribu naturellement choisie sera la tribu engendrée par X, notée $\sigma(X)$. Cette tribu est la plus petite tribu sur Ω rendant la variable aléatoire X mesurable. Plus clairement, cette tribu est l'ensemble de tous les événements dont on peut évaluer la probabilité connaissant la loi de X.

Un processus stochastique $(X_t)_{t\in\mathbb{R}^+}$ est une suite de variables aléatoires indicée par le temps $t \in \mathbb{R}^+$ (on peut bien sûr tout aussi bien indicer par un temps discret). En finance, ces processus ont un réel sens et il est normal de considérer que l'information disponible est croissante avec le temps. Il est donc légitime de préciser que l'information disponible à la date t, la tribu $\mathcal{F}_t$, est engendrée par les précédentes variables $X_s, s \leq t$. En particulier, $\mathcal{F}_0 = \{\emptyset, \Omega\}$. La suite $(\mathcal{F}_t)_{t\geq 0}$ est croissante.

Généralités

On définit la fonction de répartition F_X de la variable X par :

$$F_X(x) = \mathbb{P}[X \leq x]$$

pour tout $x \in \mathbb{R}$. On parle également de loi ou de distribution, c'est une grandeur nécessaire et suffisante pour décrire entièrement la variable aléatoire X. La fonction de survie est définie par :

$$S_X(x) = \overline{F_X}(x) = 1 - F_X(x) = \mathbb{P}[X > x]$$

Considérons une fonction $\varphi: \mathbb{R} \to \mathbb{R}$ intégrable. Dans le cas d'une variable continue à valeurs dans $\mathbb{R}$, on peut définir la densité et l'espérance par :

$$f_X(x) = \frac{dF_X(x)}{dx} = -\frac{dS_X(x)}{dx}$$
$$E[\varphi(X)] = \int_{\mathbb{R}} \varphi(x) f_X(x) dx$$

Dans le cas d'une variable discrète à valeurs dans $\mathbb{Z}$, on peut définir la fonction de masse de probabilité et l'espérance par :

$$f_X(k) = \mathbb{P}[X = k]$$
$$E[\varphi(X)] = \sum_{k\in\mathbb{Z}} \varphi(k) f_X(k)$$

Moments

L'espérance d'une variable désigne sa valeur espérée, on la note $E[X]$. Si X représente l'encours d'un contrat pour une année, alors son espérance représente l'encours moyen et on peut logiquement s'attendre à ce que la prime chargée à l'assuré soit proche de cette moyenne, c'est la raison pour laquelle on parle de « prime pure ». La variance est définie par :

$$V[X] = E[(X - E[X]^2)] = E[X^2] - E[X]^2$$

Elle s'interprète comme une mesure de la variabilité de X par rapport à son espérance. Pour les gestionnaires, elle peut traduite une mesure de risque, si elle augmente, alors l'aléa également. On définit l'écart type par :

$$\sigma[X] = \sqrt{V[X]}$$

On définit également souvent le coefficient d'asymétrie (*skewness*) par :

$$\gamma[X] = \frac{E[(X - E[X]^3)]}{V[X]^{\frac{3}{2}}}$$

Celui-ci renseigne sur la symétrie de la distribution par rapport à sa moyenne. Ainsi, dans le cas où $\gamma[X] = 0$ alors la variable est symétrique par rapport à son espérance. C'est le cas de la loi normale. En général en actuariat, $\gamma[X] > 0$, i.e. qu'il y a de grandes chances que la variable prenne des valeurs plus élevées que sa moyenne, l'asymétrie est positive.

Fonction génératrice des moments

Il faut bien garder à l'esprit que ces grandeurs ne suffisent pas à déterminer entièrement la loi de X, même si ce sont de très bons indicateurs. En revanche, la fonction génératrice des moments contient cette information, elle est définie par :

$$M_X(t) = E[e^{tX}]$$

pour un réel t tel que la somme converge (la fonction génératrice n'existe pas pour tout t, ni même pour certaines lois paramétriques comme la loi de Pareto ou la loi lognormale). Notons au passage que l'on retrouve parfois dans la littérature une définition avec une exponentielle complexe, mais l'idée sous-jacente est la même. On peut en effet écrire[1] pour tout $n \in \mathbb{N}$:

$$\begin{aligned} \left.\frac{d^n M_X(t)}{dt^n}\right|_{t=0} &= \left.\frac{d^n E[e^{tX}]}{dt^n}\right|_{t=0} \\ &= E\left[\left.\frac{d^n e^{tX}}{dt^n}\right|_{t=0}\right] \\ &= E[X^n e^{tX}]|_{t=0} \\ &= E[X^n] \end{aligned}$$

On comprend dès lors l'origine du nom de cette fonction, puisque ses dérivées donnent accès à tous les moments d'ordre n de X.

Fonction inverse

On définit la fonction inverse de la fonction de répartition par :

$$VaR_u[X] = F_X^{-1}(u) = \inf\{x \in \mathbb{R} : F_X(x) \geq u\} = \sup\{x \in \mathbb{R} : F_X(x) < u\}$$

pour $u \in [0,1]$. On rappellera les conventions $\inf \emptyset = +\infty$ et $\sup \emptyset = -\infty$. La fonction inverse est croissante au sens large et semi-continue à gauche.

[1] On peut permuter la dérivée et la somme car $M_X(t)$ existe.

Distribution empirique

On considère une réalisation $\{x_1, \dots, x_n\}$ d'un échantillon i.i.d. $X_1, \dots, X_n$ ayant pour fonction de répartition parente F_X. On cherche à estimer F_X.

On définit la fonction de répartition empirique par :

$$\hat{F}_n(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{x_i \leq x\}}$$

pour $x \in \mathbb{R}$. Elle satisfait les propriétés d'une distribution. C'est une fonction en escalier.

D'après le théorème de Glivenko – Cantelli, on peut écrire :

$$\lim_{n\to\infty}\left(\sup_{x\in\mathbb{R}}\left|\hat{F}_n(x) - F_X(x)\right|\right) = 0 \text{ presque surement}$$

En ce sens, la distribution empirique fournit une bonne approximation lorsque n est grand, et on peut l'utiliser à des fins actuarielles. Le graphe de $\hat{F}_n$ épouse d'autant mieux celui de F que n est grand.

Théorème central limite (TCL)

Le théorème central limite est d'une grande importance pour proposer une approximation de la somme de variables aléatoires. On considère donc $X_1, X_2, \dots$ des v.a. i.i.d. de variable parente X. On définit :

$$S_n = \sum_{i=1}^{n} X_i$$

Le théorème assure alors que :

$$\frac{S_n - E[S_n]}{\sqrt{V[S_n]}} \to Z \text{ en loi quand } n \to \infty$$

Où $Z \sim Nor(0,1)$. La convergence en loi peut s'écrire :

$$\forall z \in \mathbb{R}, \lim_{n\to\infty} \mathbb{P}\left[\frac{S_n - E[S_n]}{\sqrt{V[S_n]}} \leq z\right] = \Phi(z)$$

Où Φ est la fonction de répartition de la loi normale centrée réduite Z. Ce théorème ne sera pas démontré ici.

Loi des grands nombres (LGN)

La loi des grands nombres reprend les hypothèses du TCL. Elle permet d'avoir une meilleure vision du comportement de la variable $\bar{X}_n := \frac{S_n}{n}$. Le théorème assure que :

$$\bar{X}_n \to E[X] \text{ en probabilité quand } n \to \infty$$

La convergence en probabilité peut s'écrire :

$$\forall \epsilon > 0, \lim_{n\to\infty} \mathbb{P}[|\bar{X}_n - E[X]| > \epsilon] = 0$$

Ce théorème ne sera pas démontré ici.

B. LOIS USUELLES

Afin d'accompagner le lecteur dans les notations choisies pour ce mémoire, nous présentons les lois usuelles abordées en actuariat. On rappelle avant tout les définitions de la fonction de répartition d'une loi normale centrée réduite et la fonction gamma :

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-\frac{t^2}{2}} dt \text{ pour } x \in \mathbb{R}$$

$$\Gamma(x) = \int_{0}^{\infty} t^{x-1} e^{-t} dt \text{ pour } x > 0$$

Lois continues

Lois	Notation	Distribution	Densité	Espérance	Variance	FGM
Normale	$Nor(\mu, s^2)$ $\mu \in \mathbb{R}$ $s \in \mathbb{R}^+$	$\Phi\left(\frac{x-\mu}{s}\right)$ $x \in \mathbb{R}$	$\frac{1}{\sqrt{2\pi}s} e^{-\frac{(x-\mu)^2}{2s^2}}$ $x \in \mathbb{R}$	μ	s^2	$e^{t\mu + \frac{t^2 s^2}{2}}$
Log-normale	$LNor(\mu, s^2)$ $\mu \in \mathbb{R}$ $s > 0$	$\Phi\left(\frac{\ln x - \mu}{s}\right)$ $x > 0$	$\frac{1}{x\sqrt{2\pi}s} e^{-\frac{(\ln x-\mu)^2}{2s^2}}$ $x > 0$	$e^{\mu + \frac{s^2}{2}}$	$e^{2\mu+s^2}\left(e^{s^2} - 1\right)$	Non analytique
Exponentielle	$Exp(\lambda)$ $\lambda > 0$	$1 - e^{-\lambda x}$ $x > 0$	$\lambda e^{-\lambda x}$ $x > 0$	$\frac{1}{\lambda}$	$\frac{1}{\lambda^2}$	$\left(1 - \frac{t}{\lambda}\right)^{-1}$
Gamma	$Gam(\alpha, \beta)$ $\alpha > 0$ $\beta > 0$ $\theta = \frac{1}{\beta}$	Non analytique	$\frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$ $x > 0$	$\frac{\alpha}{\beta} = \alpha\theta$	$\frac{\alpha}{\beta^2} = \alpha\theta^2$	$\left(1 - \frac{t}{\beta}\right)^{-\alpha}$
Pareto	$Par(\alpha, \lambda)$ $\alpha > 0$ $\lambda > 0$	$1 - \left(\frac{\lambda}{\lambda + x}\right)^\alpha$ $x > 0$	$\frac{\alpha\lambda^\alpha}{(\lambda + x)^{\alpha+1}}$ $x > 0$	$\frac{\lambda}{\alpha - 1}$ *si* $\alpha > 1$	$\frac{\alpha\lambda^2}{(\alpha-1)^2(\alpha-2)}$ *si* $\alpha > 2$	N'existe pas
Pareto généralisée	$GPar(\alpha, \lambda, \tau)$ $\alpha > 0$ $\lambda > 0$ $\tau > 0$	$\beta\left(\tau, \alpha; \frac{x}{\lambda + x}\right)$ $x > 0$	$\frac{\Gamma(\alpha+\tau)\lambda^\alpha x^{\tau-1}}{\Gamma(\alpha)\Gamma(\tau)(\lambda + x)^{\alpha+\tau}}$ $x > 0$			N'existe pas
Weibull	$Wei(\tau, \lambda)$ $\tau > 0$ $\lambda > 0$	$1 - e^{-\lambda x^\tau}$ $x > 0$	$\tau\lambda x^{\tau-1} e^{-\lambda x^\tau}$ $x > 0$	$\frac{\Gamma\left(1 + \frac{1}{\tau}\right)}{\lambda^{\frac{1}{\tau}}}$	$\frac{\Gamma\left(1 + \frac{2}{\tau}\right) - \left(\Gamma\left(1 + \frac{1}{\tau}\right)\right)^2}{\lambda^{\frac{2}{\tau}}}$	Non analytique
Uniforme	$Uni([a, b])$ $a < b \in \mathbb{R}$	$\begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \le x \le b \\ 1 & x > b \end{cases}$	$\begin{cases} 0 & x < a \\ \frac{1}{b-a} & a \le x \le b \\ 0 & x > b \end{cases}$	$\frac{a+b}{2}$	$\frac{(b-a)^2}{12}$	$\frac{e^{bt} - e^{at}}{(b-a)t}$

Lois discrètes

Lois	Notation	Masse de probabilité	Espérance	Variance	FGM
Binomiale	$Bin(n, p)$ $n \in \mathbb{N}^*$ $p \in]0; 1[$	$\binom{n}{k} p^k (1-p)^{n-k}$ $k \in \{0, \dots, n\}$	np	$np(1-p)$	$(pe^t + 1 - p)^n$
Poisson	$Poi(\lambda)$ $\lambda > 0$	$\frac{e^{-\lambda}\lambda^k}{k!}$ $k \in \mathbb{N}$	λ	λ	$\exp\left(\lambda(e^t - 1)\right)$
Binomiale négative 1ère paramétrisation	$NBin(r, p)$ $r > 0$ $p \in]0; 1[$ $p = (1+\beta)^{-1}$	$\frac{\Gamma(r+k)}{\Gamma(r)k!} p^r (1-p)^k$ $k \in \mathbb{N}$	$r\frac{1-p}{p}$	$r\frac{1-p}{p^2}$	$\left(\frac{p}{1-(1-p)e^t}\right)^r$
Binomiale négative 2ème paramétrisation	$NBin(r, \beta)$ $r > 0$ $\beta > 0$ $\beta = (1-p)p^{-1}$	$\frac{\Gamma(r+k)}{\Gamma(r)k!}\left(\frac{1}{1+\beta}\right)^r \left(\frac{\beta}{1+\beta}\right)^k$ $k \in \mathbb{N}$	$r\beta$	$r\beta(1+\beta)$	$\left(1 - \beta(e^t - 1)\right)^{-r}$

C. ESPÉRANCE CONDITIONNELLE

Soit $(X_t)_t$ un processus stochastique, soit $s \leq t$, alors l'espérance conditionnelle de X_t par rapport à la tribu $\mathcal{F}_s$ issue de la filtration naturelle est définie comme étant l'unique[1] variable aléatoire Z vérifiant :

$$\forall\, G \in \mathcal{F}_s, \qquad \int_G Z d\mathbb{P} = \int_G X_t d\mathbb{P}$$

En d'autres termes, $E[X_t|X_s] = E[X_t|\mathcal{F}_s]$ est la meilleure estimation de X_t que l'on puisse faire sachant que l'on a l'historique jusqu'à X_s (que l'on dispose de toute l'information contenue dans la tribu $\mathcal{F}_s$).

Pour X et Y v.a. intégrables et de produit intégrable, $\mathcal{G}$ sous-tribu de $\mathcal{F}$, $\mathcal{H}$ sous-tribu de $\mathcal{G}$ alors :

- Si $\mathcal{G} = \mathcal{F}$ alors $E[X|\mathcal{G}] = X$
- Si $\mathcal{G} = \{\emptyset, \Omega\}$ alors $E[X|\mathcal{G}] = E[X]$
- $E[E[X|\mathcal{G}]|\mathcal{H}] = E[E[X|\mathcal{H}]|\mathcal{G}] = E[X|\mathcal{H}]$
- $E\big[E[X|\mathcal{G}]\big] = E[X]$
- Si X est $\mathcal{G}$-mesurable, $E[XY|\mathcal{G}] = XE[Y|\mathcal{G}]$, en particulier $E[X|\mathcal{G}] = X$
- Si X est indépendante de $\mathcal{G}$, $E[X|\mathcal{G}] = E[X]$

On considère deux variables aléatoires discrètes (le raisonnement dans le cas continu est exactement le même mais les notations sont plus difficiles d'accès) notées X et Y. La fonction masse de probabilité conditionnelle de X sachant $Y = y$, pour $x \in A_X$ et $y \in A_Y$ où A_X et A_Y représentent l'ensemble des valeurs prises par X et Y, est donnée par :

$$\begin{aligned} p_{X|Y=y}(x) &= \mathbb{P}[X = x|Y = y] \\ &= \frac{\mathbb{P}[X = x, Y = y]}{\mathbb{P}[Y = y]} \end{aligned}$$

Pour un sous-ensemble $a_Y \subset A_Y$, alors :

$$\begin{aligned} p_{X|Y\in a_Y}(x) &= \mathbb{P}[X = x|Y \in a_Y] \\ &= \frac{\mathbb{P}[X = x, Y \in a_Y]}{\mathbb{P}[Y \in a_Y]} \\ &= \frac{\sum_{y\in a_Y} \mathbb{P}[X = x, Y = y]}{\sum_{y\in a_Y} \mathbb{P}[Y = y]} \end{aligned}$$

A ce stade, nous sommes en mesure de calculer l'espérance de X conditionnellement à $Y \in a_Y$:

$$\begin{aligned} E[X|Y \in a_Y] &= \sum_{x\in A_X} x\, p_{X|Y\in a_Y}(x) \\ &= \sum_{x\in A_X} x \frac{\mathbb{P}[X = x, Y \in a_Y]}{\mathbb{P}[Y \in a_Y]} \\ &= \sum_{x\in A_X} x \frac{\sum_{y\in a_Y} \mathbb{P}[X = x, Y = y]}{\sum_{y\in a_Y} \mathbb{P}[Y = y]} \end{aligned}$$

La formule de transfert s'applique toujours et pour des fonctions φ boréliennes, on peut écrire :

$$\begin{aligned} E[\varphi(X)|Y \in a_Y] &= \sum_{x\in A_X} \varphi(x)\, p_{X|Y\in a_Y}(x) \\ &= \sum_{x\in A_X} \varphi(x) \frac{\sum_{y\in a_Y} \mathbb{P}[X = x, Y = y]}{\sum_{y\in a_Y} \mathbb{P}[Y = y]} \end{aligned}$$

[1] Aux ensembles négligeables près.

Dans l'introduction, nous utilisons la topologie de l'espace $L^2(\Omega)$ pour écrire l'espérance conditionnelle comme une projection et donc la solution d'un problème de minimisation de normale associée au produit scalaire. Détaillons justement ce point.

Ici, on considère un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$ et on note L^2 l'ensemble des variables aléatoires réelles X de carré intégrable (vérifiant $E[X^2] < \infty$). Cet espace est muni du produit scalaire suivant :

$$< X, Y >= E[XY]$$

Et de la norme associée :

$$\|X\|_2 = \sqrt{< X, X >} = \sqrt{E[X^2]}$$

On peut montrer que l'espace L^2 muni de cette norme est un espace de Hilbert (complet pour la norme associée au produit scalaire). On se donne un $X \in L^2$ et on note $\mathcal{G} = \sigma(X)$ la tribu engendrée par X. $L^2(\mathcal{G})$ est l'ensemble des variables aléatoires $\mathcal{G}$-mesurables. On peut alors décomposer l'espace de cette façon :

$$L^2 = L^2(\mathcal{G}) \oplus L^2(\mathcal{G})^{\perp}$$

Cette décomposition est possible car $L^2(\mathcal{G})$ est un sous-espace vectoriel fermé de L^2 espace de Hilbert[1].

Autrement dit,

$$\forall Y \in L^2, \exists! \left(Y_g, Y_{g^{\perp}}\right) : Y = Y_g + Y_{g^{\perp}}$$

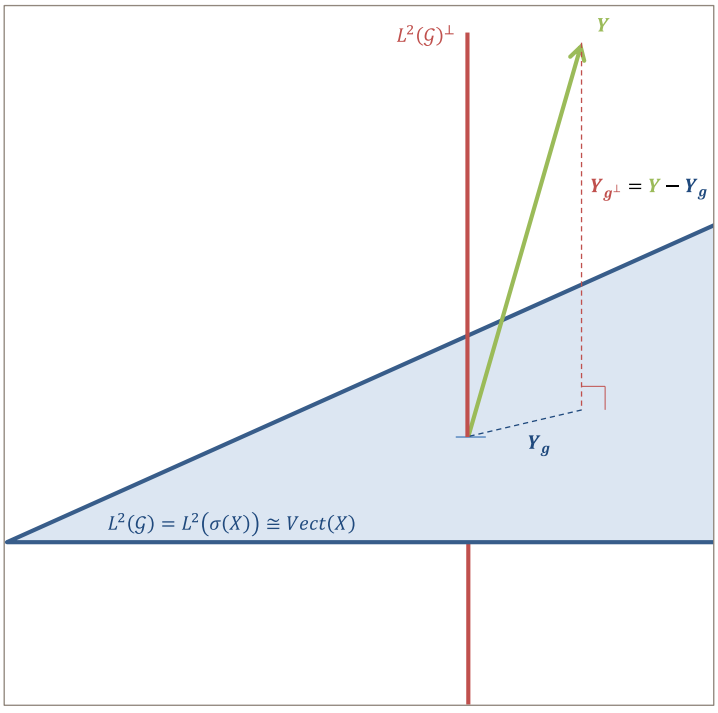


Figure 42 : Projection dans l'ensemble des fonctions de carré intégrable

[1] Dans un cadre moins favorable cette décomposition n'est pas innée, elle l'est toutefois en dimension finie.

La variable Y_g est la projection orthogonale de Y sur $L^2(\mathcal{G})$. Ainsi :

- Y_g est $\mathcal{G}$-mesurable,

- $Y_{g^\perp} = Y - Y_g \in L^2(\mathcal{G})^\perp$, en conséquence $\forall W \in L^2(\mathcal{G}), < W, Y - Y_g >= 0$ i.e. $E[WY] = E\left[WY_g\right]$.

Le théorème de projection assure que $E[Y|X]$ est la projection orthogonale de Y sur $L^2(\mathcal{G})$. Or l'on sait que dans le cadre d'un espace de Hilbert la projection peut être vue comme solution d'un problème d'optimisation (minimisation de distance) :

$$\begin{aligned} E[Y|X] &= \arg\min_{W \in L^2(\mathcal{G})} \|Y - W\|_2 \\ &= \arg\min_{W \in L^2(\mathcal{G})} \|Y - W\|_2^2 \\ &= \arg\min_{W \in L^2(\mathcal{G})} E[(Y - W)^2] \end{aligned}$$

Notons $\phi_X : L^2 \to L^2\big(\sigma(X)\big), Y \mapsto E[Y|X]$. Par définition, $\phi_X(Y)$ est bien une variable aléatoire $\sigma(X)$-mesurable et par propriété, nous savons que :

- $\forall \lambda \in \mathbb{R}, \forall U, V \in L^2, \phi_X(\lambda U + V) = E[\lambda U + V|X] = \lambda E[U|X] + E[V|X] = \lambda \phi_X(U) + \phi_X(V)$
- $\forall U \in L^2, \phi_X^2(U) = \phi_X(E[U|X]) = E\left[\underbrace{E[U|X]}_{\sigma(X)\text{-mesurable}} |X\right] = E[U|X] = \phi_X(U)$

L'application ϕ_X est donc linéaire, et c'est de plus un projecteur car elle est idempotente ($\phi_X^2 = \phi_X$). C'est l'application associée au projecteur sur $L^2(\mathcal{G})$.

On écrira donc :

$$E[Y|X] = \arg\min_{W \in L^2(\mathcal{G})} \|Y - W\|_2 = \arg\min_{W \in L^2(\mathcal{G})} E[(Y - W)^2]$$

C'est l'équation (1).

D. MODÈLE COLLECTIF

Pour le calcul de la prime pure, nous n'avons eu besoin que du premier moment de la sinistralité S. En revanche, un chargement de sécurité basé sur une mesure de risque peut nécessiter des moments d'ordre supérieurs ou bien des quantiles de la distribution de S. On montre ici comment obtenir les 3 premiers moments de S sans supposer une loi a priori pour N.

On considère le modèle suivant :

$$S = \sum_{i=1}^{N} W_i$$

S est la charge sinistre totale
N est le nombre total de sinistres survenus
W_i est le montant du sinistre numéro i
$W_1, W_2, \ldots$ forment un échantillon i.i.d. de loi parente W
(W_i) et N sont des v.a. indépendantes

Avant de supposer une loi de comptage, intéressons-nous au deux premiers moments de S. Tout d'abord remarquons que :

$$E[S|N=n] = E\left[\sum_{i=1}^{n} W_i\right] = \sum_{i=1}^{n} E[W_i] = n\,E[W] \Rightarrow E[S|N] = N\,E[W]$$
$$V[S|N=n] = V\left[\sum_{i=1}^{n} W_i\right] = \sum_{i=1}^{n} V[W_i] = n\,V[W] \Longrightarrow V[S|N] = N\,V[W]$$

On peut alors facilement en déduire :

$$\begin{aligned} E[S] &= E\big[E[S|N]\big] \\ &= E\big[N\,E[W]\big] \\ &= E[N]E[W] \end{aligned}$$

$$\begin{aligned} V[S] &= E\big[V[S|N]\big] + V\big[E[S|N]\big] \\ &= E\big[N\,V[W]\big] + V\big[N\,E[W]\big] \\ &= V[W]E[N] + E[W]^2V[N] \end{aligned}$$

Il est également très intéressant de se pencher sur la fonction génératrice des moments $M_S(t)$. Nous pouvons écrire pour $t \in \mathbb{R}$ tel que la somme converge :

$$\begin{aligned} M_S(t) &= E[e^{tS}] \\ &= E\left[E\left[e^{t\sum_{i=1}^{N} W_i}|N\right]\right] \\ &= E\left[E[\Pi_{i=1}^{N} e^{tW_i}|N]\right] \\ &= E[E[e^{tW}]^N] \\ &= E[M_W(t)^N] \\ &= E\left[e^{N\ln M_W(t)}\right] \\ &= M_N(\ln M_W(t)) \end{aligned}$$

Nous pouvons dès lors évaluer les dérivées successives de $M_S(t)$, en posant $f = M_W(t)$ et $g = M_N(t)$ afin de ne pas alourdir les notations, i.e. $M_S(t) = g(\ln f)$:

$$
\begin{aligned}
\frac{dM_X(t)}{dt} &= \frac{f'}{f} g'(\ln f) \\
\frac{d^2M_X(t)}{dt^2} &= \underbrace{\frac{f''f - f'^2}{f^2}}_{A_1} \underbrace{g'(\ln f)}_{A_2} + \underbrace{\left(\frac{f'}{f}\right)^2}_{A_3} \underbrace{g''(\ln f)}_{A_4} \\
\frac{d^3M_X(t)}{dt^3} &= A_1'A_2 + A_1A_2' + A_3'A_4 + A_3A_4' \\
A_1{}' &= \frac{f'''f - 3f''f'f + 2f'^3}{f^3} \\
A_2' &= \frac{f'}{f} g''(\ln f) \\
A_3' &= 2\frac{f'}{f}\frac{f''f - f'^2}{f^2} \\
A_4' &= \frac{f'}{f} g'''(\ln f)
\end{aligned}
$$

Ces formules se simplifient en les évaluant en $t = 0$, car en effet $f = 1, f' = E[W]$, etc. :

$$
\begin{aligned}
\frac{dM_X(t)}{dt} &= E[W]E[N] \\
\frac{d^2M_X(t)}{dt^2} &= (E[W^2] - E[W]^2)E[N] + E[W]^2E[N^2] \\
\frac{d^3M_X(t)}{dt^3} &= A_1'A_2 + A_1A_2' + A_3'A_4 + A_3A_4' \\
A_1{}' &= E[W^3] - 3E[W]E[W^2] + 2E[W]^3 \\
A_2' &= E[W]E[N^2] \\
A_3' &= 2E[W](E[W^2] - E[W]^2) \\
A_4' &= E[W]E[N^3] \\
A_1 &= E[W^2] - E[W]^2 \\
A_2 &= E[N] \\
A_3 &= E[W]^2 \\
A_4 &= E[N^2]
\end{aligned}
$$

Nous pouvons finalement en déduire :

$$
\begin{aligned}
E[S] &= E[W]E[N] \\
E[S^2] &= E[W]^2(E[N^2] - E[N]) + E[W^2]E[N] \\
E[S^3] &= E[W]^3(2E[N] - 3E[N^2] + E[N^3]) + 3E[W]E[W^2](E[N^2] - E[N]) + E[W^3]E[N]
\end{aligned}
$$

Ce sont les 3 premiers moments de S.

E. EXEMPLE SIMPLE D'UN CART

On génère aléatoirement un échantillon de taille $n = 30$ observations $(x_1, x_2, y)_{i=1,\dots,n}$ où x_1 est une variable entière prenant des valeurs entre 0 et 10, et x_2 une variable discrète prenant les valeurs A, B ou C. On pose :

$$Y = f(x_1, x_2) = \begin{cases} -x_1 & \text{si } x_2 = A \\ x_1 & \text{si } x_2 = B \\ x_1^2 & \text{si } x_2 = C \end{cases}$$

On connait donc exactement la réponse Y (ce qui n'est jamais le cas en pratique), l'avantage est que l'on pourra interpréter facilement l'arbre obtenu. Notons que Y étant continue, nous sommes dans le cas de fonction d'hétérogénéité de type variance (cf. partie 3.2.2). On parlera aussi de MSE ou de déviance, ou même d'erreur puisque celle-ci est également la somme des écarts au carré.

Voici la base d'apprentissage utilisée :

BASE D'APPRENTISSAGE DE TAILLE $n = 30$		
X1	**X2**	**Y**
1	C	1
0	B	0
6	B	6
7	B	7
5	B	5
2	C	4
4	A	-4
3	B	3
7	A	-7
3	B	3
8	B	8
2	C	4
4	A	-4
3	A	-3
0	B	0
3	C	9
3	C	9
2	C	4
0	A	0
1	C	1
2	C	4
5	A	-5
4	B	4
1	C	1
1	C	1
7	A	-7
10	B	10
1	A	-1
3	C	9
9	A	-9

On notera qu'il y a 18 profils différents :

NOMBRE DE PROFILS DIFFÉRENTS SUR LA BASE	
Profil	**Nombre**
X1=1,X2=C	4
X1=0,X2=B	2
X1=6,X2=B	1
X1=7,X2=B	1
X1=5,X2=B	1
X1=2,X2=C	4

X1=4,X2=A	2
X1=3,X2=B	2
X1=7,X2=A	2
X1=8,X2=B	1
X1=3,X2=A	1
X1=3,X2=C	3
X1=0,X2=A	1
X1=5,X2=A	1
X1=4,X2=B	1
X1=10,X2=B	1
X1=1,X2=A	1
X1=9,X2=A	1

Division du premier nœud

Dans cette partie, nous décrivons les calculs qui permettent de sélectionner quelle division choisir pour diviser le nœud tronc ou racine, puis le processus qui conduit à un arbre.

Il y a 13 divisions possibles. En effet :

- X_1 est ordinale à $k = 10$ modalités, il y a donc $k = 10$ divisions possibles.
- X_2 est purement qualitative avec $k = 3$ modalités, il y a donc $2^{k-1} - 1 = 3$ divisions possibles.

DÉTAILS DU CALCUL DE LA RÉDUCTION D'HÉTÉROGÉNÉITÉ POUR CHAQUE DIVISION POSSIBLE DU TRONC										
Index	**Variable testée**	N_G	N_D	$E[Y]$	$E[Y\|X \in N_G]$	$E[Y\|X \in N_D]$	**MSE Total (1)**	**MSE Gauche (2)**	**MSE Droite (3)**	**Réduction hétérogénéité (1)-(2)-(3)**
1	X2	{A;B}	{C}	1,8	0,3	4,3	771,4	552,1	110,2	109,1
2	X2	{A;C}	{B}	1,8	0,4	4,6	771,4	554,6	96,4	120,4
3	X2	{B;C}	[A]	1,8	4,4	-4,4	771,4	207,1	68,2	496,0
4	X1	<0,5	>0,5	1,8	0,0	2,0	771,4	0,0	761,0	10,4
5	X1	<1,5	>1,5	1,8	0,4	2,3	771,4	3,9	746,4	21,1
6	X1	<2,5	>2,5	1,8	1,6	1,9	771,4	38,9	731,8	0,7
7	X1	<3,5	>3,5	1,8	2,7	0,3	771,4	205,6	524,7	41,1
8	X1	<4,5	>4,5	1,8	2,1	0,9	771,4	290,6	470,9	9,9
9	X1	<5,5	>5,5	1,8	2,0	1,1	771,4	349,0	418,9	3,6
10	X1	<6,5	>6,5	1,8	2,1	0,3	771,4	364,6	391,3	15,4
11	X1	<7,5	>7,5	1,8	1,6	3,0	771,4	548,3	218,0	5,1
12	X1	<8,5	>8,5	1,8	1,9	0,5	771,4	587,4	180,5	3,4
13	X1	<9,5	>9,5	1,8	1,5	10,0	771,4	701,2	0,0	70,1

Si on trace la réduction d'hétérogénéité (que nous avons appelé $\widehat{\Delta}$ dans la partie 3.2.2) en fonction de l'identifiant de division, on obtient :

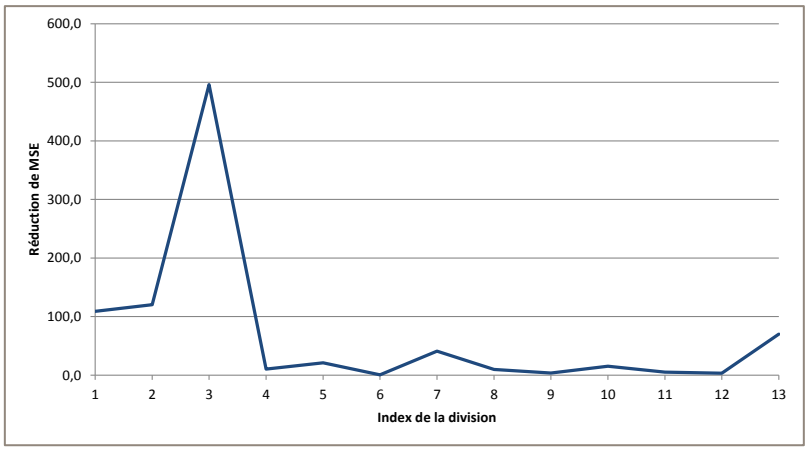


Figure 43 : Évolution de la réduction d'hétérogénéité en fonction de la division

Pour choisir la division, on choisit celle qui conduit à la plus grosse réduction d'hétérogénéité (ici, donc de MSE), avec le graphique et le tableau ci-dessus, on remarque que c'est la division numéro 3, qui divise les données selon la variable explicative X_2 entre les observations ayant pour modalité $X_2 = \{A\}$ et $X_2 = \{B; C\}$.

Construction de l'arbre

Maintenant que la première division est choisie, nous appliquons la même procédure au sein des deux bases ainsi créées. Nous obtenons alors l'arbre saturé suivant :

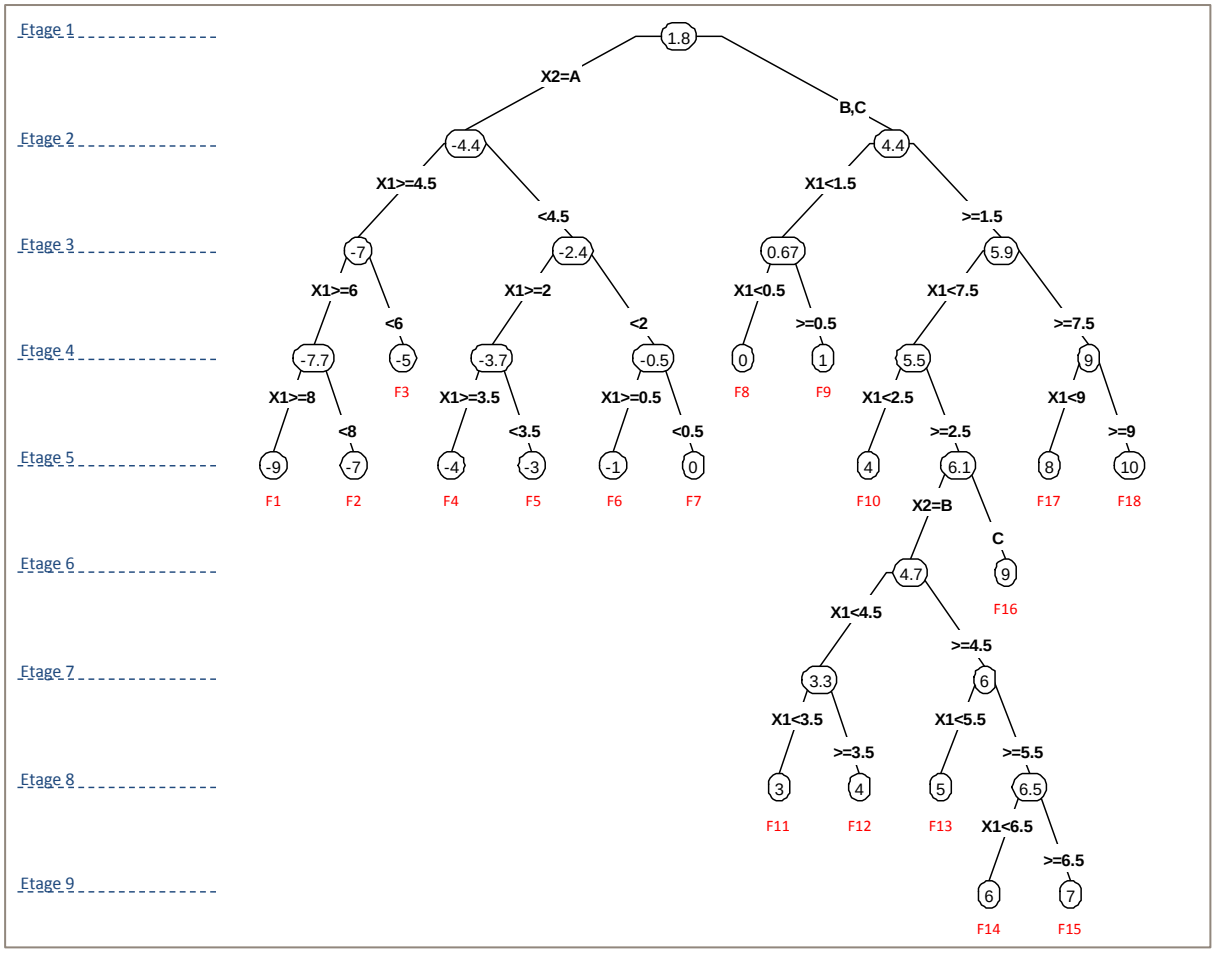


Figure 44 : Arbre saturé construit sur la base d'apprentissage

Nous avons matérialisé en bleu les étages et en rouge les feuilles. En particulier on vérifie que nous avons bien 18 feuilles, i.e. autant que de profils différents : l'arbre est saturé. Si nous prenons l'exemple du profil n°1 (X1=1,X2=C), nous avons 4 individus et la moyenne restreinte à ce profil est égale à 1, c'est bien la valeur affichée par la feuille F9.

On remarque que la première division est basée sur X_2, ce qui parait logique puisque les différentes valeurs de X_2 conduisent à des fonctions différentes de X_1 pour Y.

Toute construction d'arbre est accompagnée d'un tableau décrivant les différents nœuds représentés sur l'arbre. Le voici ci-dessous :

RÉSULTAT DE L'ALGORITHME CART					
Numéro du nœud	**Division**	**Effectif**	**MSE**	**Espérance sachant le nœud**	**Feuille**
1	racine	30	771,37	1,77	
2	X2=A	9	68,22	-4,44	
4	X1>=4,5	4	8,00	-7,00	
8	X1>=6	3	2,67	-7,67	
16	X1>=8	1	-	-9,00	F1
17	X1<8	2	-	-7,00	F2
9	X1<6	1	-	-5,00	F3
5	X1<4,5	5	13,20	-2,40	
10	X1>=2	3	0,67	-3,67	
20	X1>=3,5	2	-	-4,00	F4
21	X1<3,5	1	-	-3,00	F5
11	X1<2	2	0,50	-0,50	
22	X1>=0,5	1	-	-1,00	F6
23	X1<0,5	1	-	-	F7
3	X2=B	21	207,14	4,43	
6	X1<1,5	6	1,33	0,67	
12	X1<0,5	2	-	-	F8
13	X1>=0,5	4	-	1,00	F9
7	X1>=1,5	15	86,93	5,93	
14	X1<7,5	13	63,23	5,46	
28	X1<2,5	4	-	4,00	F10
29	X1>=2,5	9	50,89	6,11	
58	X2=B	6	13,33	4,67	
116	X1<4,5	3	0,67	3,33	
232	X1<3,5	2	-	3,00	F11
233	X1>=3,5	1	-	4,00	F12
117	X1>=4,5	3	2,00	6,00	
234	X1<5,5	1	-	5,00	F13
235	X1>=5,5	2	0,50	6,50	
470	X1<6,5	1	-	6,00	F14
471	X1>=6,5	1	-	7,00	F15
59	X2=C	3	-	9,00	F16
15	X1>=7,5	2	2,00	9,00	
30	X1<9	1	-	8,00	F17
31	X1>=9	1	-	10,00	F18

Élagage

Comme on l'a précisé précédemment, il est important d'utiliser un échantillon de validation pour élaguer notre arbre. Ici, la variable d'intérêt Y étant une simple fonction des variables explicatives, on peut facilement simuler un nouvel échantillon.

On construit ainsi l'échantillon de validation de 5 observations (soit 1/6 ième de la taille de l'échantillon d'apprentissage) suivant :

BASE DE VALIDATION DE TAILLE $n = 5$		
X1	**X2**	**Y**
1	C	1
1	B	1
9	B	9
5	C	25
2	C	4

On applique alors l'algorithme présenté Figure 22 partie 3.3.3. Le nombre K est égal à 18, i.e. le nombre de feuilles de l'arbre saturé.

À partir de l'échantillon d'apprentissage, on détermine la suite $T_1 \subset \cdots \subset T_{K-1} \subset T_K$, qui minimise la valeur de $R(T_k) = MSE$ sur la base de validation pour $k = 1 \ldots K$. La représentation de ces erreurs de validation en fonction de k est donnée ci-dessous :

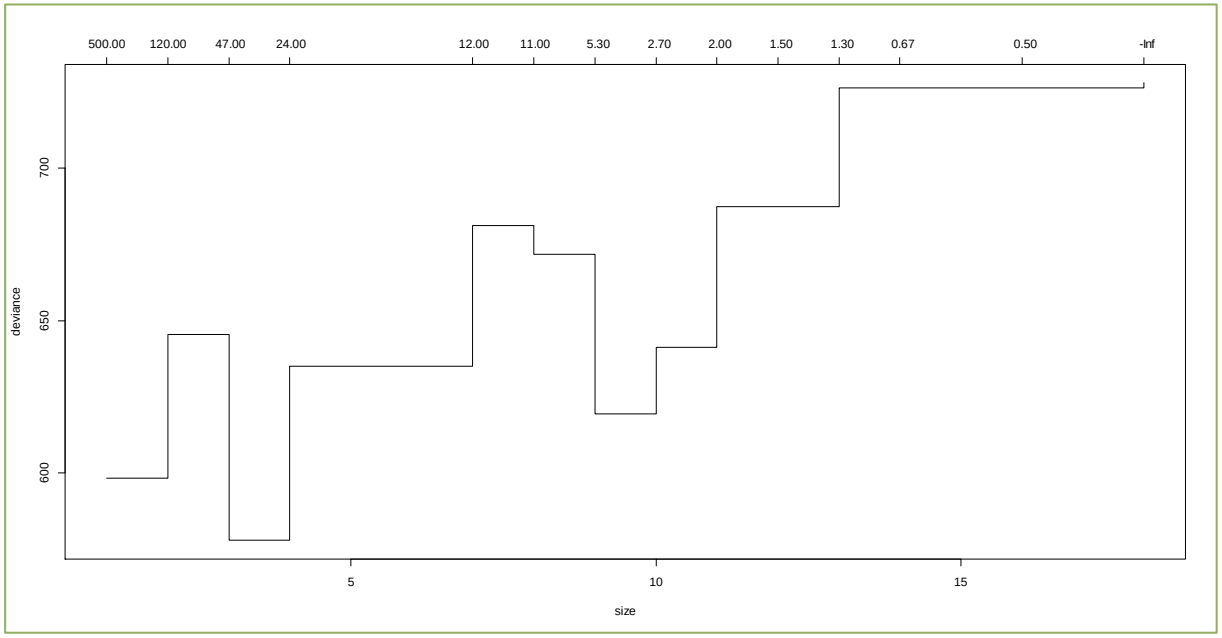


Figure 45 : Erreur MSE sur la base de validation en fonction du nombre de feuilles

Ne nous formalisons pas sur la forme de la courbe : elle est due au faible nombre de données que nous avons. Fondamentalement, nous avons plutôt une forme en « U ». Nous remarquons que pour un arbre à 3 feuilles, le MSE (ou l'erreur, ou la déviance, ce sont des notions équivalentes) est minimisé. Nous obtenons donc $k^* = 3$. Nous pouvons alors construire l'arbre optimal :

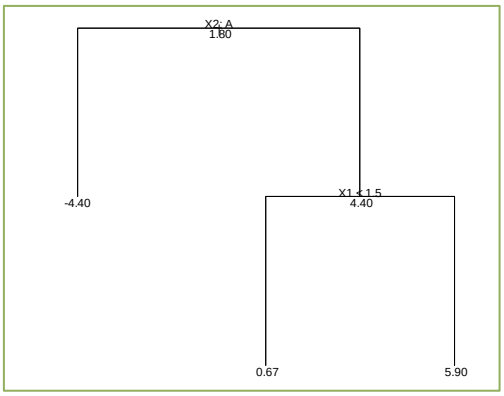


Figure 46 : Arbre optimal obtenu après élagage

On peut reconnaitre cet arbre sur notre arbre saturé :

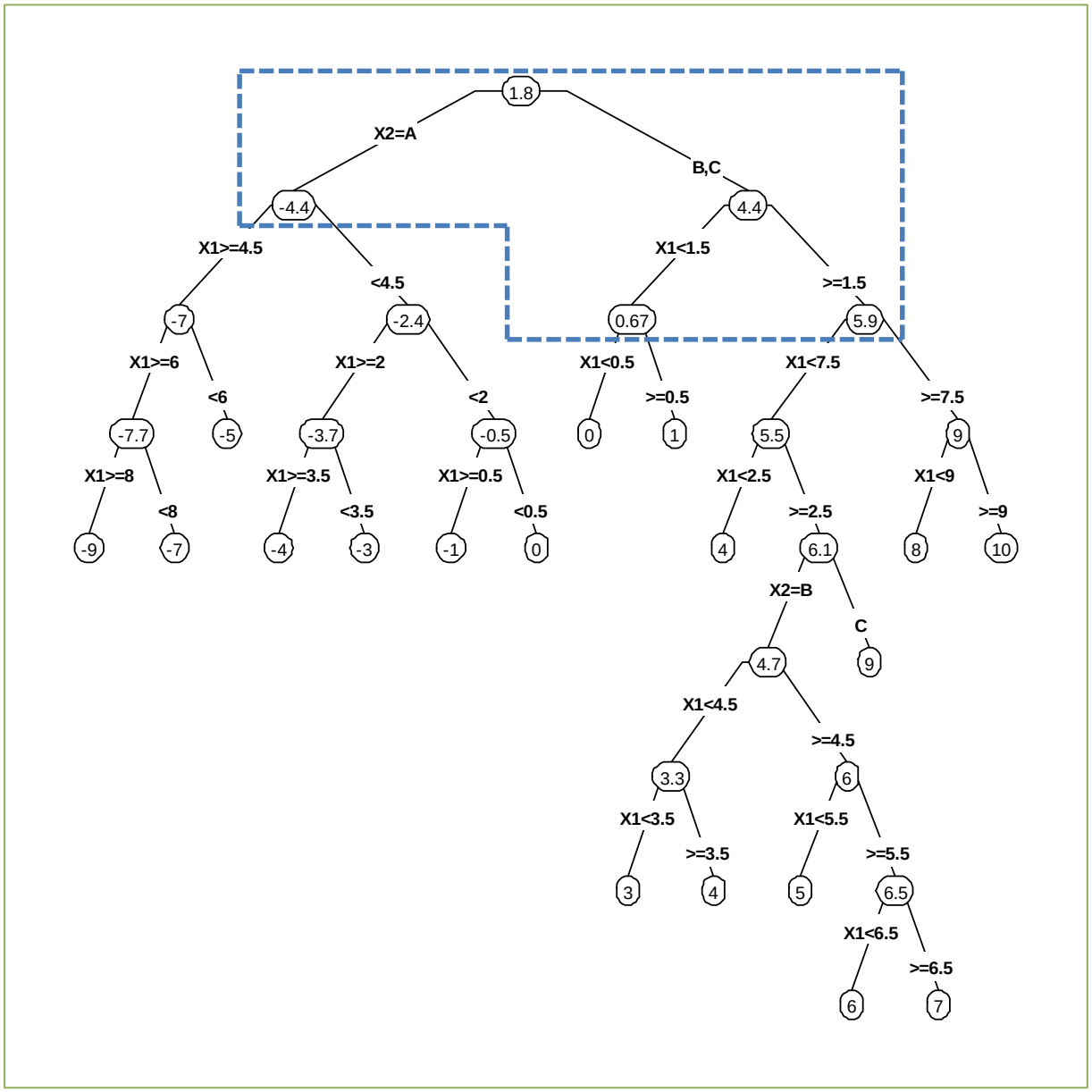


Figure 47 : Arbre optimal à 3 feuilles mis en évidence sur l'arbre saturé

Il est à ce stade légitime de s'interroger sur la place du coefficient α mentionné en partie 3.3.3. Il est indiqué sur l'échelle supérieure de la Figure 45. Nous remarquons bel et bien que pour $\alpha = 0$ nous obtenons l'arbre saturé à 18 feuilles. Toujours d'après la courbe, nous avons la valeur optimale $\alpha^* \cong 47$ correspondant au $k^* = 3$.

F. ALGORITHMES DE DESCENTE

Rappels théoriques

On considère pour cela des $\mathbb{R}$-espaces vectoriels notés E, F, G, H de dimensions finies et normés par $\| \ \|$. On note U, V des ouverts non vides de E et de F. On note la fonction $f: U \subset E \to F$, on choisit $a \in U$.

Définition : On dit que f est différentiable en a si f adment un développement limité à l'ordre 1 en a, autrement dit s'il existe une fonction $df_a \in \mathcal{L}(E, F)$ telle que

$$f(a+h) = f(a) + df_a(h) + o_{0_E}(\|h\|)$$

L'application linéaire df_a est alors appelée la différentielle de f en a.

Exemple : Soit $f: E \to F$ constante et $a \in E$. Alors $\forall h \in E, f(a+h) = f(a)$. Ainsi, en posant $df_a: E \to F, h \mapsto df_a(h) = 0$ (application linéaire), nous avons bien l'égalité ci-dessus. En conclusion, la différentielle de f est la fonction nulle.
Dans la suite, on supposera $F = \mathbb{R}$, on considère donc à présent des fonctions à valeurs réelles. Pour plus de clarté, on supposera également $E = \mathbb{R}^n$. Ainsi, E est un espace vectoriel muni du produit scalaire euclidien noté $< x, y >= \sum_{i=1}^{n} x_i y_i$, il est de plus muni de la base canonique $(e_i)_{i \in \{1,\dots,n\}}$.

Théorème : Si $f: U \subset \mathbb{R}^n \to \mathbb{R}$ est de classe C^1, alors pour tout élément $a \in U$ il existe un unique vecteur noté $grad\ f(a) \in \mathbb{R}^n$ et appelé gradient de f en a, tel que

$$\forall h \in \mathbb{R}^n, df_a(h) =< grad\ f(a), h >$$
$$grad\ f(a) = \begin{pmatrix} \partial_1 f(a) \\ \vdots \\ \partial_n f(a) \end{pmatrix}$$

Ce résultat est issu de la représentation de forme linéaire dans un espace euclidien, en effet df_a est ici une forme linéaire puisque $F = \mathbb{R}$.

Exemple 1 : Soit $f: \mathbb{R}^2 \to \mathbb{R}, (x_1, x_2) \longmapsto x_1^3 + 5x_1x_2$. La fonction est bien évidemment de classe C^1. De plus, pour $a = (a_1, a_2) \in \mathbb{R}^2$ les dérivées partielles sont :

$$\frac{\partial f}{\partial x_1}(a_1, a_2) = 3a_1^2 + 5a_2$$
$$\frac{\partial f}{\partial x_2}(a_1, a_2) = 5a_1$$

Ainsi :

$$grad\ f(a) = \begin{pmatrix} 3a_1^2 + 5a_2 \\ 5a_1 \end{pmatrix} \in \mathbb{R}^2$$

Exemple 2 : Soit $f: \mathbb{R} \to \mathbb{R}, x \longmapsto x^2$. La fonction est à nouveau de classe C^1 et l'on peut évaluer son gradient pour un $a \in \mathbb{R}$: $grad\ f(a) = f'(a)$. Ainsi nous avons :

$$\forall h \in \mathbb{R}^n, df_a(h) =< f'(a), h >= f'(a)h$$

On remarque en particulier que $df_a(1) = f'(a)$.

Pour simplifier, on supposera ici $U = E = \mathbb{R}^n$ (ce qui est vrai la plupart du temps). Rappelons qu'en général U est un ouvert inclus dans E. Un développement limité à l'ordre 1 conduit à écrire pour $a \in U$ et $h \in \mathbb{R}^n$:

$$df_a(h) = \lim_{t \to 0} \frac{f(a+th) - f(a)}{t}$$

C'est la dérivée en a selon le vecteur h, la fonction $t \longmapsto f(a+th)$est définie au voisinage de zéro et étudie les valeurs prises par f sur la droite $a + Vect(h)$. On comprend cette quantité comme la pente de f dans la direction de h. Comme $df_a(h) =< grad\, f(a), h >$, la pente est maximale quand h a le sens et la direction de $grad\, f(a)$. Par suite, le vecteur $grad\, f(a)$ renseigne sur la direction de la plus grande pente, on connait ainsi à la fois le sens de progression sur la pente et aussi la valeur de la pente maximale ($\|grad\, f(a)\|_2$). Ce cadre théorique est particulièrement adapté à la recherche d'extremum.

On rappelle que $f : U \subset \mathbb{R}^n \to \mathbb{R}$ est de classe C^1.

Définition : On dit que f admet un minimum global en $a \in U$ si $\forall x \in U, f(x) \geq f(a)$.
On dit que f admet un minimum local en $a \in U$ si $\exists \alpha > 0, \forall x \in U \cap B(a, \alpha), f(x) \geq f(a)$.

Définition : On dit que $a \in U$ est un point critique si $df_a = 0$, i.e. $\forall h \in \mathbb{R}^n, df_a(h) = 0$. Cela équivaut à $grad\, f(a) = 0$.

Théorème : Si f admet un minimum local en $a \in U$, alors a est un point critique. Autrement dit tout minimum annule le gradient.

On notera l'importance du caractère ouvert de U. En effet, la fonction $f : [0,1] \to \mathbb{R}, x \longmapsto x$ adment un minimum en $a = 0$ pourtant la dérivée ne s'y annule pas. Surtout, la réciproque de ce théorème est fausse. La simple recherche des points critiques ne suffit pas en général à trouver les extremums, même si cela indique où les chercher. De plus, le simple gradient ne suffit pas à décider s'il s'agit d'un minimum ou maximum. On peut néanmoins développer à l'ordre 2 la fonction pour obtenir cette information, et faire apparaitre la matrice hessienne.

Théorème : Si $f : U \subset \mathbb{R}^n \to \mathbb{R}$ est de classe C^2 alors pour $a \in U$:

$$f(a+h) = f(a) + df_a(h) + \frac{1}{2} q_a(h) + o_{0_E}(\|h\|_2^2)$$

Où q_a est la forme quadratique associée à la matrice hessienne de f. La matrice hessienne est symétrique et est définie par :

$$H_a = [q_a] = \left(\partial_i \partial_j f(a)\right)_{i,j \in \{1,\dots,n\}}$$

Théorème : à propos des conditions nécessaires et suffisantes d'extremum

- Condition nécessaire d'extremum local :
  - Si a est un minimum local de f, alors c'est un point critique et la hessienne est positive.
  - Si a est un maximum local de f, alors c'est un point critique et la hessienne est négative.
- Condition suffisante d'extremum local :
  - Si la hessienne est définie positive, le point critique a est un minimum local.
  - Si la hessienne est définie négative, le point critique a est un maximum local.

Généralités sur les méthodes de descente

Soit $f: \mathbb{R}^n \to \mathbb{R}$ une fonction convexe et C^2. On s'intéresse à la classe de méthodes de minimisation sur $U = E = \mathbb{R}^n$ s'écrivant sous la forme suivante :

$$\begin{aligned} &x^{(0)} \in \mathbb{R}^n \\ &x^{(k+1)} = x^{(k)} + \alpha_k p^{(k)}, k \in \mathbb{N} \\ &p^{(k)} \in \mathbb{R}^n \text{ donnant la direction de descente} \\ &\alpha_k \in \mathbb{R} \text{ tel que } f\left(x^{(k+1)}\right) \leq f\left(x^{(k)}\right) \end{aligned}$$

Lorsque $x^{(k)}$ et $p^{(k)}$ sont fixés dans $\mathbb{R}^n$ et $\mathbb{R}^{n*}$, on peut chercher le réel α_k satisfaisant au mieux l'inégalité. Cela s'appelle la recherche linéaire, cette partie de l'algorithme permet de renforcer la convergence globale de la méthode de descente. On a donc :

$$\alpha_k^* = \arg\min_{\alpha \in \mathbb{R}} \underbrace{f\left(\underbrace{x^{(k)} + \alpha p^{(k)}}_{g^{(k)}(\alpha)}\right)}_{J^{(k)}(\alpha)}$$

On peut s'intéresser à la dérivée de $J^{(k)}(\alpha)$ en fonction de α :

$$\begin{aligned} J^{(k)\prime}(\alpha) &= dJ^{(k)}{}_{\alpha}(1) \\ &= d\left(f \circ g^{(k)}\right)_{\alpha}(1) \\ &= df_{g^{(k)}(\alpha)}\left(dg^{(k)}{}_{\alpha}(1)\right) \\ &= df_{g^{(k)}(\alpha)}\left(p^{(k)}\right) \\ &= < grad\, f\left(x^{(k)} + \alpha p^{(k)}\right), p^{(k)} > \end{aligned}$$

Par conséquent que la solution α_k^* vérifie :

$$< grad\, f\left(x^{(k)} + \alpha_k^*\, p^{(k)}\right), p^{(k)} >= 0$$

Algorithme du gradient à pas optimal

C'est un algorithme de descente où $p^{(k)} = -grad\, f\left(x^{(k)}\right)$ correspond à la direction de plus forte descente, on a donc :

$$\begin{aligned} &x^{(0)} \in \mathbb{R}^n \\ &x^{(k+1)} = x^{(k)} - \alpha_k\, grad\, f\left(x^{(k)}\right), k \in \mathbb{N} \\ &\alpha_k^* = \arg\min_{\alpha \in \mathbb{R}} f\left(x^{(k)} - \alpha\, grad\, f\left(x^{(k)}\right)\right) \end{aligned}$$

Dans ce cas, nous avons :

$$< grad\, f\left(x^{(k)} + \alpha_k^*\, grad\, f\left(x^{(k)}\right)\right), grad\, f\left(x^{(k)}\right) >= 0$$

Et ainsi :

$$< grad\, f\left(x^{(k+1)}\right), grad\, f\left(x^{(k)}\right) >= 0$$

Deux directions de descente successives calculées par l'algorithme de plus profonde descente à pas optimal sont orthogonales.

La condition d'arrêt de l'algorithme est donné par $\left\|grad\, f\left(x^{(k)}\right)\right\| < \epsilon$ pour un $\epsilon > 0$ fixé en amont. La suite $\left(f\left(x^{(k)}\right)\right)_{k \in \mathbb{N}}$ est décroissante, et même strictement décroissante quand le gradient est non nul. La suite converge vers un minimum local de f. Dans le cas où f est strictement convexe, la suite converge vers un minimum global.

G. ÉCHANTILLON BOOTSTRAP

Le *bootstrap* consiste généralement en un tirage avec remise de l'échantillon de départ. Certaines observations sont dupliquées tandis que d'autres sont absentes, ce qui introduit une notion d'aléatoire et rend les méthodes d'estimation plus robustes. L'intérêt de la méthode est de répéter la procédure afin d'obtenir plusieurs réalisations de la statistique estimée.

On souhaite évaluer la proportion d'un échantillon *bootstrap* dans l'échantillon initial. On cherche donc une fonction permettant de dire si les deux échantillons sont identiques, se ressemblent ou s'ils sont complétement différents. On considère un échantillon initial noté $(X_i)_{\{1,\dots,n\}}$ et un échantillon *bootstrap* (tirage aléatoire et avec remise parmi (X_i)) noté $(Y_i)_{i\in\{1,\dots,n\}}$. A des fins d'illustration, nous remarquons que nous pouvons remplacer l'échantillon $(X_i)_{\{1,\dots,n\}}$ par $(i)_{\{1,\dots,n\}}$: en effet il suffit d'identifier le numéro de la ligne. L'échantillon $(Y_i)_{i\in\{1,\dots,n\}}$ est ainsi créé en tirant aléatoirement et de façon indépendante un entier entre $\{1,\dots,n\}$. Dans le cas $n = 10$, voici une illustration :

X_i	Y_i
1	3
2	8
3	8
4	1
5	3
6	8
7	5
8	10
9	7
10	5

On cherche à quantifier la « proportion de Y dans X », une idée assez naturelle est de compter dans l'échantillon (Y_i) le nombre de fois où chaque i apparait. Dans notre exemple, cela donne :

X_i	N_i
1	1
2	0
3	2
4	0
5	2
6	0
7	1
8	3
9	0
10	1

Formellement, pour un i donné, le nombre de $Y_j = X_i$ est donné par :

$$N_i = \sum_{j=1}^{n} \mathbb{1}_{\{Y_j = X_i\}} = \sum_{j=1}^{n} \mathbb{1}_{\{Y_j = i\}}$$

- Si tous les N_i étaient égaux à 1, alors on aurait les mêmes échantillons.
- Si tous les N_i étaient nuls sauf 1, alors les échantillons seraient aussi éloignés que possible.

On notera au passage que $\sum_{i=1}^{n} N_i = \sum_{i=1}^{n}\sum_{j=1}^{n} \mathbb{1}_{\{Y_j = X_i\}} = n$, en particulier les N_i ne sont donc pas des variables indépendantes.

Dans la suite, on notera $N_Y = (N_1, \dots, N_n) \in \mathbb{R}^{n+}$ associé à l'échantillon *bootstrap* et $N_X = (1, \dots, 1) \in \mathbb{R}^{n+}$ associé à l'échantillon initial.

Suite à notre remarque précédente, nous nous intéressons à la fonction ci-dessous :

$$f : \mathbb{R}_+^n \to \left[\frac{1}{n}, 1\right], x \mapsto \frac{\|x\|_2^2}{\|x\|_1^2}$$

De façon moins formelle, nous avons :

$$f(x) = \frac{\sum_{i=1}^{n} x_i^2}{(\sum_{i=1}^{n} x_i)^2}$$

Les encadrements classiques de normes de $\mathbb{R}^n$ (type relation d'équivalence) montrent que la fonction est bien définie. De plus, les bornes sont atteintes :

- f est minimale et égale à $\frac{1}{n}$ quand tous les x_i sont égaux, on comprend que la diversification est alors la plus grande car tout est équitablement réparti.

- f est maximale et égale à 1 quand tous les x_i sont nuls sauf 1, on comprend que ce n'est pas du tout diversifié.

On réalise que f satisfait aux critères que nous voulions et nous sommes amenés à évaluer :

$$f(N) = \frac{\sum_{i=1}^{n} N_i^2}{(\sum_{i=1}^{n} N_i)^2}$$

Dans la mesure où nous cherchons à quantifier l'écart par rapport à l'échantillon initial dont l'image par f donne $\frac{1}{n}$, l'indicateur mathématique permettant de quantifier le degré de similitude entre l'échantillon initial et l'échantillon *bootsrap* est ainsi donné par :

$$\begin{aligned}
\alpha_n &= \frac{f(N_X)}{f(N_Y)} \\
&= f(N_X)\frac{1}{f(N_Y)} \\
&= \frac{\sum_{i=1}^{n} 1^2}{(\sum_{i=1}^{n} 1)^2}\frac{1}{\frac{\sum_{i=1}^{n} N_i^2}{(\sum_{i=1}^{n} N_i)^2}} \\
&= \frac{1}{n}\frac{1}{\frac{\sum_{i=1}^{n} N_i^2}{(\sum_{i=1}^{n} N_i)^2}} \\
&= \frac{1}{n}\frac{1}{\frac{\sum_{i=1}^{n} N_i^2}{n^2}} \\
&= \frac{1}{\frac{1}{n}\sum_{i=1}^{n} N_i^2}
\end{aligned}$$

A ce stade, on peut vérifier que :

- α_n est maximum et vaut 1 si tous les N_i sont égaux à 1, c'est le cas où l'échantillon $(Y_i)_i = (X_i)_i$,

- α_n est minimum et vaut $\frac{1}{n}$ si tous les N_i sont nuls sauf 1 qui vaut n, c'est le cas où l'échantillon $(Y_i)_i$ est le plus éloigné de $(X_i)_i$ car il a tiré n fois la même valeur.

On conclut que α_n est une bonne mesure du degré de similitude entre $(Y_i)_i$ et $(X_i)_i$.

On s'intéresse à la limite de $E[\alpha_n]$ quand $n \to \infty$. Pour cela on remarquera que pour $i \in \{1, \dots, n\}$ la v.a. N_i suit une loi binomiale $Bin\left(n, \frac{1}{n}\right)$, on notera en particulier que $V[N_i] = 1 - \frac{1}{n}$, $E[N_i] = 1$ et $E[N_i^2] = 2 - \frac{1}{n}$. On a :

$$\alpha_n = \frac{1}{\frac{1}{n}\sum_{i=1}^{n} N_i^2} = \frac{1}{Z_n} = g(Z_n)$$

Avec $Z_n = \frac{1}{n}\sum_{i=1}^{n} N_i^2 \in [1, n]$ et $g: [1, \infty[\to \mathbb{R}, x \mapsto \frac{1}{x}$. De plus, on estime[1] que pour n grand :

$$Y_n = \sum_{i=1}^{n} \frac{(N_i - E[N_i])^2}{V[N_i]} = \sum_{i=1}^{n} \frac{(N_i - 1)^2}{1 - \frac{1}{n}} = \frac{n}{n-1}\left(\sum_{i=1}^{n} N_i^2 - n\right) \sim \chi^2(n-1)$$

Comme la variance d'une $\chi^2(n-1)$ est donnée par $2(n-1)$, ainsi on retiendra que :

$$V\left[\sum_{i=1}^{n} N_i^2\right] \sim \frac{2(n-1)^3}{n^2} \text{ quand } n \to \infty$$

Observons désormais l'espérance et la variance de Z_n :

$$\begin{aligned} E[Z_n] &= E\left[\frac{1}{n}\sum_{i=1}^{n} N_i^2\right] \\ &= \frac{1}{n}\sum_{i=1}^{n} E[N_i^2] \\ &= 2 - \frac{1}{n} \end{aligned}$$

$$\begin{aligned} V[Z_n] &= V\left[\frac{1}{n}\sum_{i=1}^{n} N_i^2\right] \\ &= \frac{1}{n^2} V\left[\sum_{i=1}^{n} N_i^2\right] \\ &\sim \frac{1}{n^2}\frac{2(n-1)^3}{n^2} \text{ quand } n \to \infty \\ &\sim \frac{2(n-1)^3}{n^4} \text{ quand } n \to \infty \\ &\to 0 \text{ quand } n \to \infty \end{aligned}$$

Revenons à présent sur la fonction g, c'est une fonction C^∞ qui admet pour tout $x \geq 1$ le développement de Taylor reste intégral à l'ordre 1 en $\mu_n = E[Z_n]$:

$$g(x) = g(\mu_n) + g'(\mu_n)(x - \mu_n) + \underbrace{\int_{\mu_n}^{x} g''(t)(x-t)dt}_{R_n(x)}$$

[1] Nous n'en présentons pas la preuve.

La dérivée seconde est strictement décroissante, donc $\sup_{t\geq 1}|g''(t)| = |g''(1)| = 2$, ainsi la formule de Taylor – Lagrange fournit :

$$g(x) = g(\mu_n) + g'(\mu_n)(x-\mu_n) + R_n(x)$$
$$|R_n(x)| \leq (x-\mu_n)^2$$

En particulier cela est vrai pour $x = Z_n \geq 1$:

$$g(Z_n) = g(\mu_n) + g'(\mu_n)(Z_n-\mu_n) + R_n(Z_n)$$
$$|R_n(Z_n)| \leq (Z_n-\mu_n)^2$$

En prenant l'espérance (on rappelle que $E[Z_n-\mu_n]=0$) ,

$$\begin{aligned} E[g(Z_n)] &= g(\mu_n) + E[R_n(Z_n)] \\ |E[R_n(Z_n)]| &\leq E[|R_n(Z_n)|] \\ &\leq E[(Z_n-\mu_n)^2] \\ &\leq V[Z_n] \to 0 \text{ quand } n\to\infty \\ |E[g(Z_n)] - g(\mu_n)| &\to 0 \text{ quand } n\to\infty \end{aligned}$$

Or $g(\mu_n) = \frac{1}{\mu_n} = \frac{1}{E[Z_n]} = \frac{1}{2-\frac{1}{n}} \to \frac{1}{2}$ quand $n\to\infty$, finalement :

$$\left|E[\alpha_n] - \frac{1}{2}\right| = \left|E[g(Z_n)] - \frac{1}{2}\right| \leq \underbrace{|E[g(Z_n)] - g(\mu_n)|}_{\to 0 \text{ quand } n\to\infty} + \underbrace{\left|g(\mu_n) - \frac{1}{2}\right|}_{\to 0 \text{ quand } n\to\infty}$$

Finalement :

$$E[\alpha_n] \to \frac{1}{2} \text{ quand } n\to\infty$$

Nous pouvons finalement conclure que **le *bootstrap* d'un échantillon est plus ou moins équivalent à sélectionner aléatoirement et sans remise $50\,\%$ de l'échantillon initial**.

H. COURBES DE LIFT

Définitions

La « courbe de *Lift* » fournit un résumé visuel de l'information apportée par un modèle statistique. D'une certaine façon, la courbe synthétise les gains auxquels on peut s'attendre en utilisant le modèle prédictif. La courbe de Lift s'applique à la plupart des méthodes statistiques et est souvent utilisée en data mining. C'est une variante des courbes ROC. On parlera plus volontiers de courbe de Lorenz en économétrie. La courbe de concentration de Lorenz est en effet un moyen de mesurer les inégalités de possession de richesse.

On supposera donc ici que X représente la richesse des individus de la population. Concrètement, nous disposons d'un échantillon $(X_1, \dots, X_n)$ et d'une réalisation de cet échantillon $(x_1, \dots, x_n)$.

On note F la fonction de répartition de X, ainsi $F(x) = \mathbb{P}[X \leq x]$. La quantité $F(x)$ représente la proportion d'individu ayant une richesse X inférieure ou égale à x. On rappelle que $F(x) \in [0,1]$.

On note $FQ(x)$ la proportion de richesse des individus ayant une richesse $X \leq x$. De même, $FQ(x) \in [0,1]$.

La courbe de Lorenz est la courbe joignant l'ensemble des points $\big(F(x), FQ(x)\big)$ du plan. C'est donc une courbe paramétrée $\mathcal{L}: \mathbb{R} \to [0,1]^2, x \mapsto \big(F(x), FQ(x)\big)$ (ou nuage de points plutôt). On notera que la courbe est inscrite dans le carré $[0,1]^2$, et passe par les points $(0,0)$ et $(1,1)$.

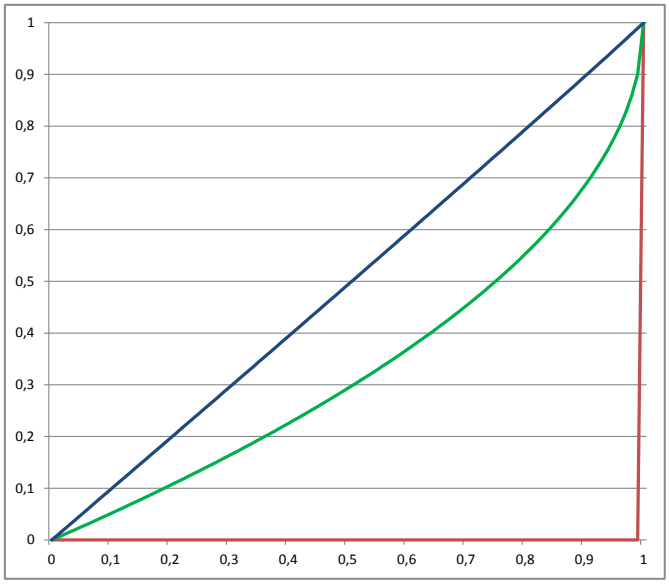


Figure 48 : Différentes courbes de Lorenz

La courbe bleue correspond au modèle aléatoire, la courbe verte au modèle statistique obtenu et la courbe rouge au modèle idéal (dans le cas où le nombre de données est suffisamment grand).

Si la variable aléatoire X ne prend qu'une seule valeur (i.e. $\forall i, x_i = x_0$), alors nous obtenons la courbe bleue. C'est un cas d'égalité parfaite où tout le monde possède la même richesse. Pour tout x, $F(x) = FQ(x)$, autrement dit u % des individus possèdent u % des richesses. À l'inverse, si la variable aléatoire X ne prend que deux valeurs 0 et 1, avec un seul individu ayant la valeur 1 et $n-1$ individus ayant la valeur 0 alors nous obtenons la courbe rouge (à la limite). C'est un cas d'inégalité totale.

Plus généralement, plus la courbe verte est proche de la courbe bleue, et plus le système est égalitaire. L'indice de Gini permet de quantifier cet écart en évaluant des quotients d'aires. Afin de

pouvoir tracer la courbe, il est nécessaire de se baser sur notre réalisation de l'échantillon, et donc sur des estimateurs F_n et FQ_n de F et FQ. Nous avons :

$$\begin{aligned} F_n(x) &= \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{x_i \leq x\}} \\ &\rightarrow E\left[\mathbb{1}_{\{X \leq x\}}\right] = F(x) \text{ quand } n \rightarrow \infty \end{aligned}$$

Et

$$\begin{aligned} FQ_n(x) &= \frac{1}{\sum_{i=1}^{n} x_i}\sum_{i=1}^{n} \mathbb{1}_{\{x_i \leq x\}} x_i \\ &= \frac{1}{\frac{1}{n}\sum_{i=1}^{n} x_i}\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{x_i \leq x\}} x_i \\ &= \frac{\frac{1}{n}\sum_{i=1}^{n} \mathbb{1}_{\{x_i \leq x\}} x_i}{\bar{X}_n} \\ &\rightarrow \frac{E\left[\mathbb{1}_{\{X \leq x\}} X\right]}{E[X]} \text{ quand } n \rightarrow \infty \end{aligned}$$

Plus formellement, la courbe de Lorenz associe donc à l'abscisse $F(x)$ la valeur $\frac{E\left[\mathbb{1}_{\{X \leq x\}} X\right]}{E[X]}$.

Dans le cas d'une loi continue et positive, si l'on pose $\kappa = F_X(x)$, la courbe associe la valeur :

$$\begin{aligned} \frac{E\left[\mathbb{1}_{\{X \leq x\}} X\right]}{E[X]} &= \frac{E\left[\mathbb{1}_{\{X \leq F_X^{-1}(\kappa)\}} X\right]}{E[X]} \\ &= \frac{1}{E[X]}\int_0^{F_X^{-1}(\kappa)} t dF_X(t) \\ &= \frac{1}{E[X]}\int_0^{\kappa} VaR_u[X] du \end{aligned}$$

La courbe est alors la fonction : $[0,1] \rightarrow [0,1], \kappa \longmapsto \frac{1}{E[X]}\int_0^{\kappa} VaR_u[X] du$. C'est une fonction croissante et convexe.

Exemple d'une loi de Pareto

À titre d'exemple, considérons le cas où X suit une loi de Pareto de paramètres $x_m > 0$ et $\alpha > 1$. La densité est donnée par :

$$f(x) = \frac{\alpha x_m^{\alpha}}{x^{\alpha+1}} \mathbb{1}_{\{x \geq x_m\}}$$

Et la fonction de répartition par :

$$F(x) = \left(1 - \left(\frac{x_m}{x}\right)^{\alpha}\right) \mathbb{1}_{\{x \geq x_m\}}$$

On peut alors montrer que :

$$E[X] = \frac{\alpha x_m}{\alpha - 1}$$

De plus, pour $x \geq x_m$:

$$\begin{aligned} FQ(x) &= \frac{E\left[1_{\{X \leq x\}} X\right]}{E[X]} \\ &= \frac{\int_{\mathbb{R}} \mathbb{1}_{\{t \leq x\}} t f(t) dt}{\frac{\alpha x_m}{\alpha - 1}} \\ &= \frac{(\alpha - 1) \int_{\mathbb{R}} \mathbb{1}_{\{t \leq x\}} \frac{\alpha x_m^{\alpha}}{t^{\alpha+1}} \mathbb{1}_{\{t \geq x_m\}} t dt}{\alpha x_m} \\ &= (\alpha - 1) x_m^{\alpha-1} \int_{x_m}^{x} t^{-\alpha} dt \\ &= (\alpha - 1) x_m^{\alpha-1} \frac{1}{1 - \alpha} (x^{1-\alpha} - x_m^{1-\alpha}) \\ &= 1 - x_m^{\alpha-1} x^{1-\alpha} \\ &= 1 - \left(\frac{x_m}{x}\right)^{\alpha-1} \end{aligned}$$

Or, quand x décrit $[x_m, +\infty[$, $F(x)$ décrit $[0,1[$. On procède donc à changement de variable en posant $\kappa = F(x)$ afin de se ramener à une courbe classique. On a donc $x = F^{-1}(\kappa)$. Cependant, si $x \geq x_m$:

$$F(x) = 1 - \left(\frac{x_m}{x}\right)^{\alpha} \Leftrightarrow x = x_m \left(1 - F(x)\right)^{-\frac{1}{\alpha}}$$

La quantité $FQ(x)$ devient alors :

$$\begin{aligned} FQ(x) &= 1 - \left(\frac{x_m}{x}\right)^{\alpha-1} \\ &= 1 - \left(\frac{x_m}{x_m (1 - \kappa)^{-\frac{1}{\alpha}}}\right)^{\alpha-1} \\ &= 1 - (1 - \kappa)^{1-\frac{1}{\alpha}} \end{aligned}$$

On conclut donc que la courbe de Lorenz est donnée par l'application définie de la façon suivante $[0,1] \to [0,1], \kappa \mapsto \begin{cases} 1 - (1 - \kappa)^{1-\frac{1}{\alpha}} & si\ \kappa < 1 \\ 1 & si\ \kappa = 1 \end{cases}$. Voici les courbes obtenues selon les valeurs de α.

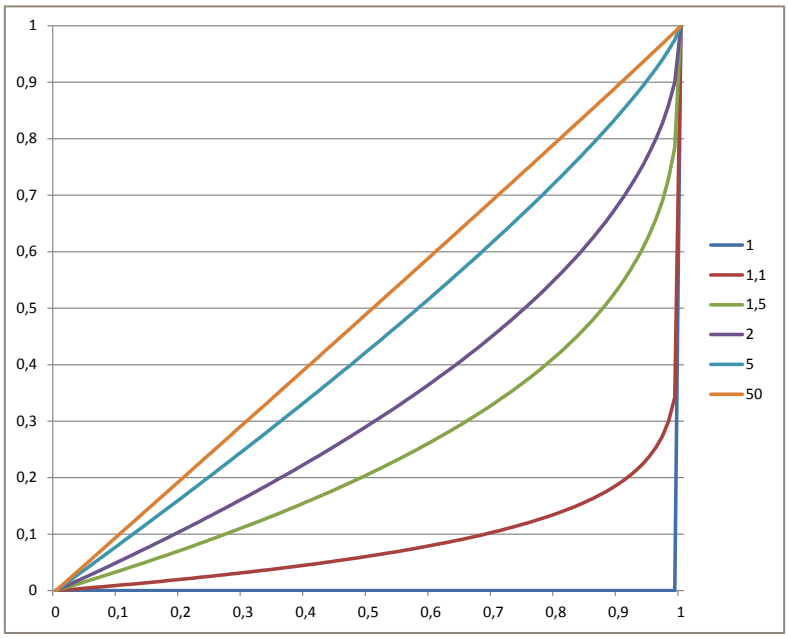


Figure 49 : Courbes de Lorenz de la loi de Pareto en fonction de α

Indice de Gini et application

On note $I = F(\mathbb{R})$ et on choisit $\kappa = F(x) \in I$. Ainsi nous avons $F^{-1}(\kappa) = VaR_\kappa[X] = x$ (inverse généralisé, i.e. $F^{-1}(\kappa) = \inf\{y : F(y) \geq \kappa\}$). De plus :

$$FQ_n(x) = \frac{1}{\sum_{i=1}^{n} x_i} \sum_{i=1}^{n} \mathbb{1}_{\{x_i \leq VaR_\kappa[X]\}} x_i$$

La représentation de la courbe de Lorenz devient :

$$\left(\kappa \ , \ \frac{1}{\sum_{i=1}^{n} x_i} \sum_{i=1}^{n} \mathbb{1}_{\{x_i \leq VaR_\kappa[X]\}} x_i \right)$$

En pratique, on préfère donc discrétiser l'axe $[0,1]$ selon des valeurs de κ (déciles, centiles) puis associer la valeur correspondante. De plus, on tracera parfois plus volontiers le nuage de points $\big(1 - F(x), 1 - FQ(x)\big) = \Big(\mathbb{P}[X > x] \ , \ E\Big[\frac{X}{E[X]} \mathbb{1}_{\{X > x\}}\Big]\Big)$. L'interprétation change alors quelque peu : on observe en abscisse la proportion d'individu ayant une richesse $X > x$ et en ordonnée la proportion de richesse de ces individus. On retrouve la définition de la courbe de lift que l'on décrit dans la littérature.

Il est toutefois important de noter que seuls les κ admissibles (autrement dit appartenant à l'image de F) ont un réel sens, la courbe n'est donc strictement définie que sur I. Ainsi, sur un échantillon de taille 10, cela ne ferait pas de sens de parler de $35\,\%$ de la population. Cette courbe peut être utilisée pour quantifier le pouvoir explicatif d'un modèle. On obtient ce type de courbe :

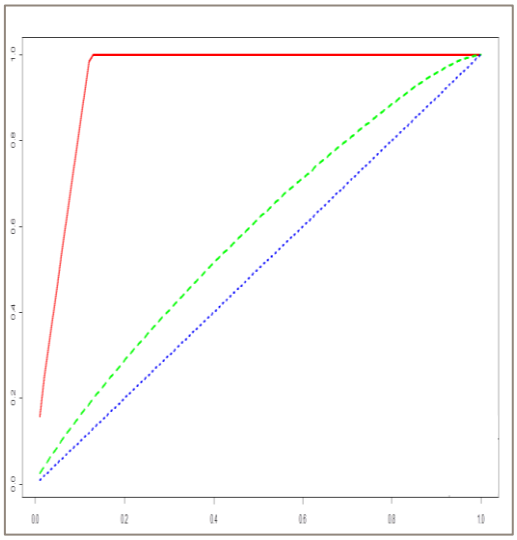


Figure 50 : Courbes de Lift

En rouge on retrouve le lift optimal, en bleu le lift aléatoire (uniforme) et en vert le lift estimé résultat du *scoring*.

On peut par exemple considérer qu'un bon modèle de score permet de toucher $60\,\%$ des sinistres avec $30\,\%$ des individus. Plus précisément, on sait que la courbe de lift de notre modèle doit tendre vers le modèle idéal dans le cas parfait. Ainsi un bon indicateur de l'écart entre les courbes est le rapport d'aire suivant :

$$\frac{\text{surface entre lift estimé et lift aléatoire}}{\text{surface entre lift optimal et lift aléatoire}}$$

C'est ce qu'on appelle l'indice de Gini. Plus il sera proche de 1 et plus le modèle sera discriminant.