



Institut de Statistique, Biostatistique et Actuariat

## AUTOCALIBRATION DES PRIMES PAR MACHINE LEARNING

Promoteur :  
Prof. **Michel DENUIT**  
*UCLouvain*

Mémoire présenté en vue de l'obtention du  
master en sciences actuarielles (à finalité  
spécialisé) par : **Carnot Dongmo Mbane**

Louvain-la-Neuve

Août 2022

# AUTOCALIBRATION DES PRIMES PAR MACHINE LEARNING

DONGMO MBANE CARNOT

Août 2022

# Table des matières

Liste des figures	v
Liste des tableaux	viii
Introduction	1
<b>1 Traitement des données</b>	<b>2</b>
1.1 Description de la base de données . . . . .	2
1.1.1 Statistiques descriptives sur le jeu de données . . . . .	3
1.1.2 La variable Exposure . . . . .	3
1.1.3 Nombre de sinistres . . . . .	4
1.1.4 La variable Power . . . . .	4
1.1.5 La variable CarAge . . . . .	6
1.1.6 La variable DriverAge : âge du conducteur . . . . .	6
1.1.7 La variable Brand . . . . .	7
1.1.8 La variable Gas . . . . .	8
1.1.9 La variable Density . . . . .	8
1.1.10 Les variables PolicyID et Region . . . . .	9
1.2 Analyse multivariée . . . . .	9
1.2.1 Liaison entre variable qualitative et variable quantitative . . . . .	9
1.2.2 Liaison entre deux variables quantitatives . . . . .	10
1.2.3 Liaison entre deux variables qualitatives . . . . .	10
<b>2 Contexte, définition et notation</b>	<b>11</b>
2.1 Fonction de régression . . . . .	11

---

2.2	Hypothèses techniques . . . . .	15
2.3	Notation . . . . .	16
2.4	Ordre convexe . . . . .	17
<b>3</b>	<b>Fonctions de perte cohérente pour la prime pure</b>	<b>18</b>
3.1	Fonctions de perte de Bregman . . . . .	18
3.2	Déviante de Tweedie . . . . .	20
3.2.1	Famille de la dispersion exponentielle (ED) . . . . .	20
3.2.2	Distributions Tweedie . . . . .	20
3.2.3	Deviance Tweedie . . . . .	21
3.3	Critère de comparaison . . . . .	22
<b>4</b>	<b>Evaluation des performances</b>	<b>24</b>
4.1	La dominance de Bregman . . . . .	24
4.2	La dominance Tweedie . . . . .	30
4.3	Application à la régression de Poisson . . . . .	35
<b>5</b>	<b>Restauration de l'équilibre à l'échelle globale et locale</b>	<b>37</b>
5.1	Autocalibrattion . . . . .	37
5.2	Autocalibrage d'un prédicteur donné . . . . .	40
5.3	Performance d'un prédicteur . . . . .	44
5.4	Comparaison des performances de deux prédicteurs . . . . .	51
	<b>Conclusion</b>	<b>53</b>
	<b>Bibliographie</b>	<b>54</b>
	<b>Annexe</b>	<b>56</b>

---

## Résumé

La technique de Boosting est une méthode de machine learning particulièrement efficace pour la tarification de l'assurance. Dans la pratique, il y a néanmoins souvent des débats sans fin sur le choix de la bonne fonction de perte à utiliser pour entraîner le modèle d'apprentissage automatique, ainsi que sur la métrique appropriée pour évaluer les performances des modèles concurrents. En outre, la somme des travaux ajustés peut différer dans une large mesure des totaux observés, ce qui déroute souvent les analystes actuariels.

Ce travail vise à étudier davantage ce phénomène lorsque l'apprentissage est effectué en minimisant la déviance Tweedie. On montre que la minimisation des déviations implique un compromis entre l'intégrale des différences pondérées des moments partiels inférieurs et le biais mesuré sur une échelle spécifique. L'autocalibration est alors proposée comme solution. Cette nouvelle méthode de correction du biais ajoute une étape supplémentaire de GLM local à l'analyse.

Sur le plan théorique, on montre qu'elle garantit que l'équilibre se maintient également sur l'échelle locale, et pas seulement au niveau du portefeuille comme avec les techniques existantes de correction du biais. L'ordre convexe semble être l'outil naturel pour comparer les modèles de compétence, ce qui donne un nouvel éclairage aux graphes de diagnostic à savoir les courbes de concentration et de Lorentz et aux métriques associées proposés par Denuit et al. (2019) qui est l'aire ABC (area between curves) entre les courbes de Concentration et de Lorentz.

---

## Remerciements

Je tiens à remercier tout l'ensemble des personnes qui ont contribué à la réussite du bon déroulement de ce travail, notamment mon promoteur Professeur Michel Denuit pour le suivi et les conseils tout au long de mon mémoire. J'aimerais aussi remercier tous mes professeurs et les membres de ma famille, ainsi que les gens qui ont pris le temps de lire et de commenter ce travail.

# Table des figures

1.1	Répartition en pourcentage de l'exposure par mois . . . . .	4
1.2	Proportion de polices par nombre de sinistres . . . . .	5
1.3	Nombre d'observation de puissance du véhicule par durée dans le portefeuille . . . . .	5
1.4	Rapport de la puissance avec la fréquence des sinistres du portefeuille. . . . .	6
1.5	Proportion de puissance par nombre de sinistres . . . . .	7
1.6	Courbe de variation de la fréquence des sinistres en fonction de l'âge . . . . .	7
1.7	Tableau de valeur des différentes marque de voiture . . . . .	8
1.8	Tableau de valeurs des différentes densité au km <sup>2</sup> . . . . .	8
1.9	Tableau de valeur des différentes regions . . . . .	9
1.10	Corrélation entre les variables quantitatives . . . . .	10
2.1	Répartition des primes avec les modèles GLM, GAM et GBM . . . . .	14
3.1	Distributions ED avec $\nu = 1$ . . . . .	21
3.2	Distributions Tweedie et paramètres de puissance correspondant . . . . .	21
5.1	Visualisation de $s \mapsto \mathbb{E}[Y \hat{\pi}(X) = s]$ où $s$ est un certain niveau de prime, pour les différents modèles avec 0 degré de liberté. . . . .	41
5.2	Visualisation de $u \mapsto \mathbb{E}[Y \hat{\pi}(X) = F_{\hat{\pi}}^{-1}(u)] - F_{\hat{\pi}}^{-1}(u)$ avec la ligne droite correspondant à $\mu = \hat{\pi}$ . . . . .	41
5.3	Répartition des primes autocalibrés $\hat{\pi}_{BC}$ avec les modèles GLM, GAM et GBM (boosting). . . . .	44
5.4	Corrélation de rang Spearman entre $\hat{\pi}^{glm}$ , $\hat{\pi}^{gam}$ , $\hat{\pi}^{gbm}$ et leurs valeurs autocalibrées homologues $\hat{\pi}_{bc}^{glm}$ , $\hat{\pi}_{bc}^{gam}$ et $\hat{\pi}_{bc}^{gbm}$ sur les données de validation. . . . .	45

---

5.5	Visualisation de $s \mapsto \mathbb{E}[Y \widehat{\pi}_{BC}(X) = s]$ où $s$ est un certain niveau de prime, pour les différents modèles avec 0 degré de liberté avec la fine courbe colorée représente la prime prédite $\widehat{\pi}$ . . . . .	45
5.6	Visualisation de $u \mapsto \mathbb{E}[Y \widehat{\pi}_{BC}(X) = F_{\widehat{\pi}_{BC}}^{-1}(u)] - F_{\widehat{\pi}_{BC}}^{-1}(u)$ avec la ligne droite correspondant à $\mu = \widehat{\pi}_{BC}$ où la fine courbe colorée représente la prime prédite $\widehat{\pi}$ . . . . .	46
5.7	Evolution de $CC[\mu, \widehat{\pi}; \alpha]$ cas du GLM . . . . .	47
5.8	Evolution de $CC[\mu, \widehat{\pi}; \alpha]$ cas du GAM . . . . .	47
5.9	Evolution de $CC[\mu, \widehat{\pi}; \alpha]$ cas du Boosting . . . . .	48
5.10	Evolution de $LC[\widehat{\pi}; \alpha]$ cas du GLM . . . . .	49
5.11	Evolution de $LC[\widehat{\pi}; \alpha]$ cas du GAM . . . . .	49
5.12	Evolution de $LC[\widehat{\pi}; \alpha]$ cas du Boosting . . . . .	50
5.13	Diagramme de la variable puissance . . . . .	56
5.14	Proportion d'âge de véhicule par durée dans le portefeuille . . . . .	56
5.15	Proportion d'âge de véhicule par durée dans le portefeuille . . . . .	57
5.16	Proportion d'âge d'assuré par durée dans le portefeuille . . . . .	57
5.17	Proportion marque de voiture par durée dans le portefeuille . . . . .	57
5.18	Proportion marque de voiture par sinistres dans le portefeuille . . . . .	58
5.19	Proportion du type par durée dans le portefeuille . . . . .	58
5.20	Proportion du type de carburant par durée dans le portefeuille . . . . .	58
5.21	densité par sinistre . . . . .	59
5.22	Interaction entre l'âge du conducteur et le type de carburant . . . . .	59
5.23	Interaction entre l'âge de la voiture et le type de carburant . . . . .	59
5.24	Summary du modèle GLM . . . . .	60
5.25	Summary du modèle GAM . . . . .	60
5.26	Summary du modèle GBM . . . . .	61
5.27	Dépendance partielle des variables dans le modèle GBM . . . . .	61
5.28	Visualisation de $u \mapsto \mathbb{E}[Y \widehat{\pi}_{BC}(X) = F_{\widehat{\pi}_{BC}}^{-1}(u)]$ avec la ligne droite correspondant à $\mu = \widehat{\pi}_{BC}$ où La fine courbe colorée représente la prime prédite $\widehat{\pi}$ . . . . .	62
5.29	Visualisation de $u \mapsto \mathbb{E}[Y \widehat{\pi}_{BC}(X) = F_{\widehat{\pi}_{BC}}^{-1}(u)]/F_{\widehat{\pi}_{BC}}^{-1}(u)$ avec la ligne droite correspondant à $\mu = \widehat{\pi}_{BC}$ où La fine courbe colorée représente la prime prédite $\widehat{\pi}$ . . . . .	62
5.30	QQ plot de $\widehat{\pi}_{BC}$ contre $\widehat{\pi}$ . . . . .	63

---

5.31 Nuage de points $\{\hat{\pi}(x_i), \hat{\pi}_{bc}(x_i)\}$ . . . . .	63
--	----

# Liste des tableaux

1.1	Correlation entre deux variables qualitatives . . . . .	10
5.1	Quartiles (10%, Mean, 90%) de chaque prédicteur. . . . .	43
5.2	Aire ABC des différentes prédictions . . . . .	51

---

# Introduction

De nos jours, les actuaires ont recours à des méthodes statistiques avancées pour pouvoir évaluer avec précision la prime pure attendue par les assurés. Les modèles actuariels de tarifications sont généralement calibrés de manière qu'une mesure statistique d'adéquation soit optimisée sur l'ensemble d'apprentissage, tandis que le réglage du modèle est effectué par la validation croisée. Les modèles concurrents sont comparés sur un ensemble de validation ou de test distincts. La fonction de perte utilisée pour l'apprentissage du modèle correspond généralement à la déviance ou à la vraisemblance logarithmique et appartient à la famille de Tweedie avec une fonction de variance de puissance. La métrique utilisée pour évaluer les performances du modèle diffère parfois de la fonction de perte utilisée pour la formation.

L'évolution des modèles s'est faite progressivement et la prudence sous-jacente des MMT à l'égard des GLM a été progressivement oubliée et les actuaires ont adopté massivement chaque nouvel outil étendant la machinerie des GLM. Dans ce travail, nous questionnons la pertinence de la déviance en tant que fonction objective, sans la contrainte de l'équilibre global. Il est démontré que la déviance peut conduire à des primes de candidats douteux car elle peut dévier fortement des pertes observées quand les totaux ne sont pas maintenus inchangés. Une solution naturelle est de restaurer l'équilibre global à chaque étape de la procédure itérative employée pour optimiser la fonction de perte. Cette solution rétablit l'équilibre global, mais ne garantit pas que l'équilibre financier se maintient dans les sous-portefeuilles importants.

Pour cette raison, nous proposons une nouvelle stratégie basée sur le concept d'autocalibration qui garantit l'équilibre global ainsi que l'équilibre local dans l'esprit du MMT au sein d'une analyse GLM locale. Plus précisément, après que l'analyse a été effectuée avec une méthode qui ne respecte pas nécessairement des totaux marginaux, un ajustement GLM local constant est réalisé afin de rétablir le lien avec la MMT. Cette approche GLM locale consiste à utiliser la prime candidate elle-même comme nouvelle caractéristique dans la dernière étape.

Dans ce mémoire, nous ferons dans la première partie, une présentation des données utilisées pour l'implémentation à travers des analyses descriptives. Dans la partie 2, nous exposons les différentes définitions tout en décrivant les trois modèles GLM, GAM et GBM à l'étude. Dans la troisième partie, nous rappelons la forme des fonctions de perte qui sont cohérentes pour la moyenne. Puis, à la partie 4, nous présentons les diverses évaluations de performances. Enfin la partie 5 traitera du rétablissement de l'équilibre aux niveaux global et local, sur la base du concept d'autocalibration.

# Chapitre 1

## Traitement des données

### 1.1 Description de la base de données

Tout au long de notre travail, la base de données sur laquelle nous appliquons nos différentes techniques est celle utilisée dans le cadre de la tarification organisée en novembre 2015 par l'institut français des actuaires. C'est l'une des nombreuses bases de données fournies par le package R Casdatasets construit par Christophe Dutang et Arthur Charpentier dont la version récente utilisée pour notre travail a été publiée le 11 décembre 2020. Cette base de données intitulé "French Motor TPL Insurance claims Data" en abrégée freMTPLfreq issue d'une collection de jeux de données à l'origine pour le livre "Computational Actuarial Science with R" de Charpentier (2014). Nous étudierons les spécificités de ce portefeuille afin d'affiner au mieux notre approche. Elle est constituée de 410864 polices de responsabilité civile automobile observées sur plus de deux ans. Ce jeu de données est constitué de 10 variables à savoir :

1. **PolicyID** : l'ID (identifiant) de la police est utilisé pour établir un lien avec l'ensemble des données sur les sinistres.
2. **ClaimNb** : nombre total des sinistres enregistré pendant la durée du contrat.
3. **Exposure** : la période (durée) d'exposition.
4. **Power** : la puissance de la voiture (catégorique ordonnée).
5. **CarAge** : l'âge du véhicule en années.
6. **DriverAge** : l'âge du conducteur en année (en France, l'âge minimum pour conduire est de 18 ans).

- 
7. **Brand** : Marque de la voiture.
  8. **Gas** : le type de carburant de la voiture (essence ou diesel).
  9. **Region** : les régions politiques en France (sur la base d'une classification française standard).
  10. **Density** : la densité d'habitants représentant le nombre d'habitants par  $km^2$  dans la ville où vit le conducteur.

Ces 10 variables sont regroupées en trois qui sont :

- **Variable réponse** : ClaimNb ;
- **Variables Numériques** : PolicyID, Exposure, CarAge, DriverAge, Density ;
- **Variables catégorielles** : Power, Brand, Gas, Region.

Nous examinerons en détail les deux principales composantes de notre tarification, les variables ClaimNb et Exposure, et leurs interactions possibles avec d'autres variables explicatives. Le traitement des autres variables n'est pas ignoré, mais pour faciliter la lecture de ce mémoire, nous ne présenterons qu'une synthèse des résultats. Les détails de ces analyses sont fournis à l'annexe. La variable d'intérêt ici est la fréquence annuelle des sinistres. Le modèle classique serait un modèle de Poisson sur la variable ClaimNb, avec Exposure et diverses variables explicatives (continues et catégorielles).

### 1.1.1 Statistiques descriptives sur le jeu de données

Dans cette partie, nous allons effectuer une statistique descriptive de chacune des 10 variables.

### 1.1.2 La variable Exposure

L'exposition révèle la durée de temps pendant laquelle le titulaire de la police se trouve dans le portefeuille. La durée minimale est d'un mois et la durée maximale est de 2 ans avec une moyenne de six mois. Ayant fait le constat que plus de 99 des assurés n'avaient qu'une durée d'un an dans le portefeuille, nous avons décidé de restreindre notre étude sur les assurés ayant séjourné au plus d'un an. Nous allons utiliser la fonction "cut" sur R (qui permet de caractériser une variable numérique) pour représenter en nombre de mois l'exposition pour une bonne étude et observation. Le tableau suivant nous présente cette répartition en mois.

Mois	1	2	3	4	5	6	7	8	9	10	11	12
Expo	62633	29216	33452	24213	19463	29565	18835	14438	21518	13653	12653	131456

De même, le diagramme à barre avec pourcentage de la figure (1.1) nous illustre cette répartition.

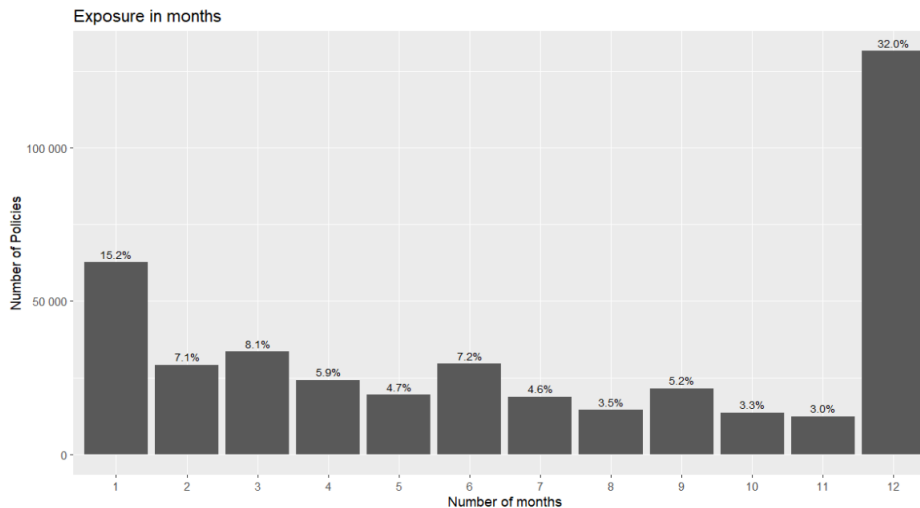


FIGURE 1.1 – Répartition en pourcentage de l'exposition par mois

### 1.1.3 Nombre de sinistres

La figure (1.2) montre la distribution du nombre de sinistres dans l'ensemble des données TPL avec une moyenne de 0.04 sinistres par assuré. La majorité des assurés à savoir 96.27% n'ont déclaré aucun sinistre au cours de leur période d'exposition ; un nombre assez important soit 3.55% ont déclaré un seul sinistre ; les autres assurés (0.18%) n'ont déclaré que deux sinistres et le reste des assurés ont déclaré au moins trois et au plus quatre sinistres. En outre, la fréquence moyenne des sinistres en fonction des différentes expositions dans ce jeu de données est de 7.01%.

### 1.1.4 La variable Power

La variable "Power" est une variable catégorielle, liée à la puissance de la voiture. Les niveaux de la variable sont ordonnés de manière catégorique avec des lettres alphabétique allant de "d" à "o". Nous savons qu'en assurance, les risques peuvent varier d'une personne à l'autre. Par conséquent, le graphique de la figure (5.13) en annexe ne mesure pas l'exposition pour chaque niveau de la variable

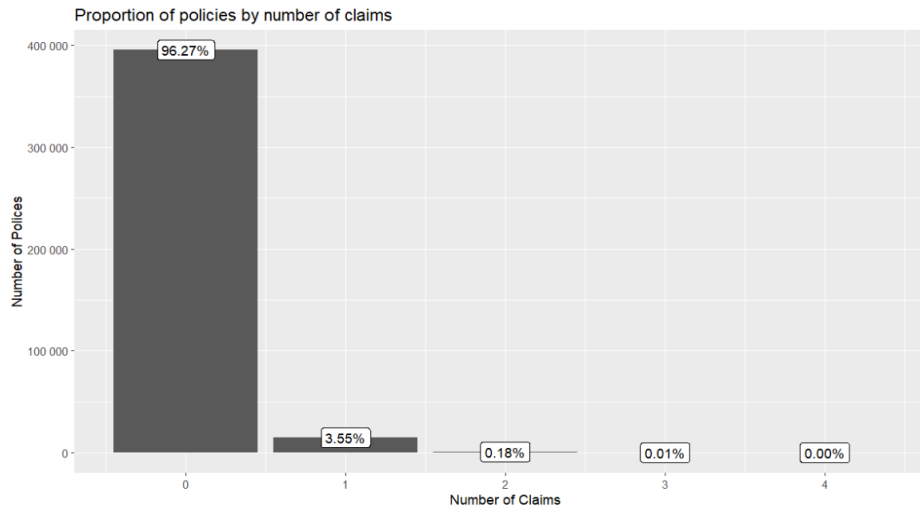


FIGURE 1.2 – Proportion de polices par nombre de sinistres

"Power". Sous R, nous avons utilisé les fonctions "group\_by" et "summarise" du package "dplyr" pour nous donner l'exposition pour chaque niveau de la variable comme nous pouvons l'observer sur la figure (1.3). Nous faisons la même examination pour la fréquence des sinistres dans chaque niveau. Sur la figure (1.4) nous observons que les catégories "d" et "g" sont en-dessous de la fréquence moyenne des sinistres du portefeuille et les catégories restantes sont au-dessus.

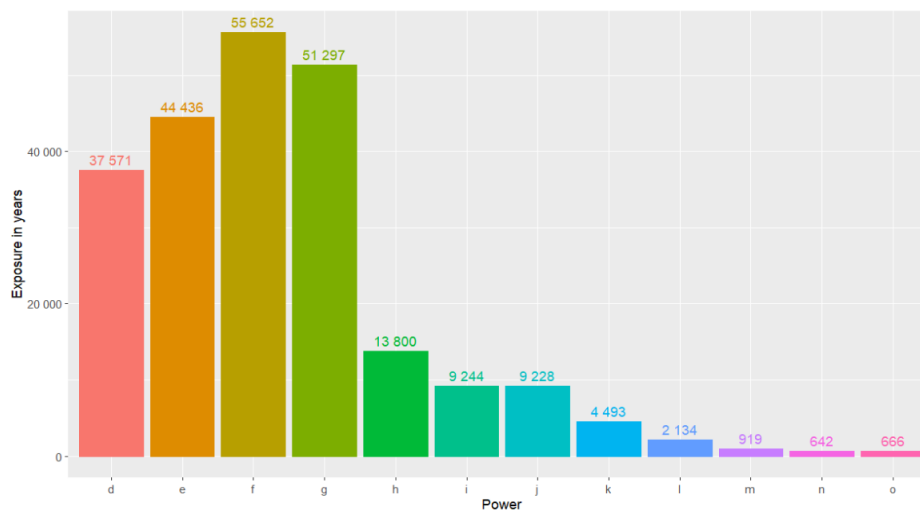


FIGURE 1.3 – Nombre d'observation de puissance du véhicule par durée dans le portefeuille

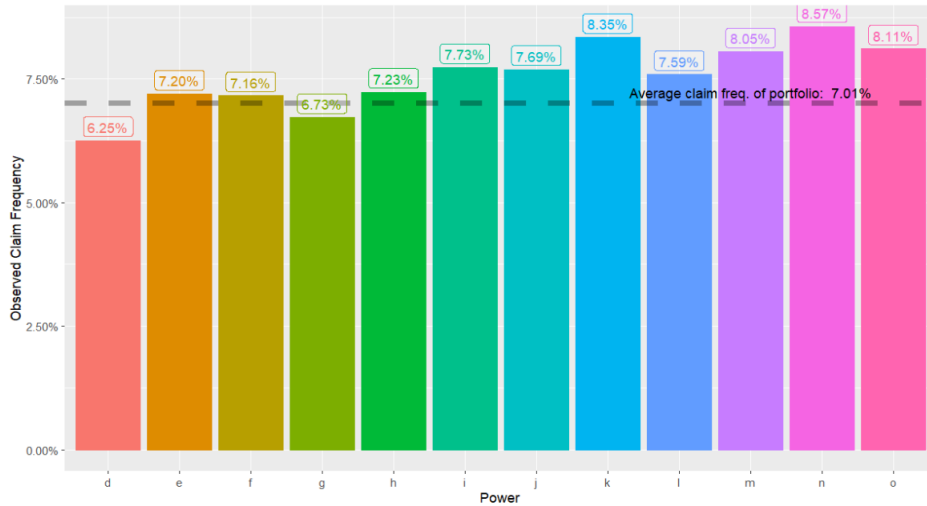


FIGURE 1.4 – Rapport de la puissance avec la fréquence des sinistres du portefeuille.

### 1.1.5 La variable CarAge

La variable de l'âge du véhicule en années est une variable continue bien qu'elle ne prend que des valeurs discrètes, avec une moyenne de 7.5 ans, une valeur minimale de zero et une valeur maximale de 100 ans. Nous utilisons "ddply" pour prendre en compte l'exposition dans notre graphique à barres. Les voitures récentes sont celles qui ont la plus grande exposition dans le portefeuille avec une grande représentativité dans l'intervalle d'âge de 0 à 4 ans puis s'en suit une forte décroissance telle qu'observé sur la figure (5.14). Suite à cette grande différence, nous avons porté un regard comme le présente la figure (5.15) en annexe, l'exposition pour les véhicules récents. Notre étude portera ainsi sur les véhicules d'âge inférieur à 25 ans. Nous constatons que les véhicule d'âge inférieur à 13 ans ont des fréquences avec le nombre de sinistres supérieur à la moyenne des sinistres du portefeuille et les voitures d'âge supérieur à 13 ans ont des fréquences bien inférieures comme nous présente la figure (1.5).

### 1.1.6 La variable DriverAge : âge du conducteur

La moyenne d'âge des assurés de notre portefeuille est de 45 ans avec les âges allant de 18 ans à 99 ans. Les assurés dont l'âge se situe entre 50 ans et 55 ans sont ceux qui restent le plus longtemps dans le portefeuille en terme de durée de contrat comme le présente en annexe la figure (5.16). En observant la figure (1.6), nous constatons que la fréquence des sinistres est plus élevée pour les jeunes conducteurs (ce qui est logique); il s'en suit une forte décroissance de la courbe

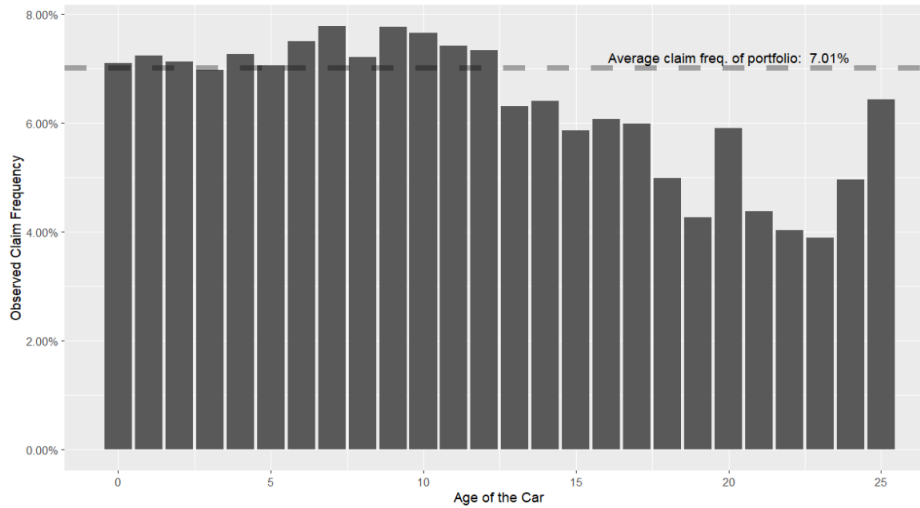


FIGURE 1.5 – Proportion de puissance par nombre de sinistres

de cette fréquence de sinistres en fonction de l'âge tendant vers une stabilisation autour de 5.2% correspondant à la tranche d'âge des assurés comprise entre 32 ans et 85 ans, avant de subir une forte variation pour les assurés âgés de plus de 85 ans.

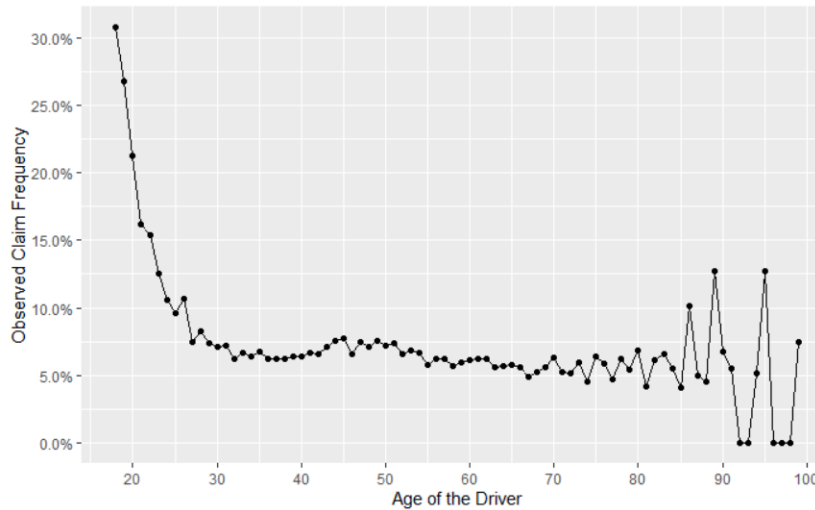


FIGURE 1.6 – Courbe de variation de la fréquence des sinistres en fonction de l'âge

### 1.1.7 La variable Brand

C'est une variable catégorielle liée à la marque de la voiture. Les assurés conduisant une voiture de marque française (Renault, Nissan ou Citroën) sont les plus représentés du portefeuille, ce sont les assurés qui ont la plus grande exposition soit 58.06% et ont le plus faible taux de fréquence des

sinistres (cause le moins de sinistres) par rapport aux autres marques de véhicule. Le tableau de la figure (1.7) nous donne un récapitulatif. Les figures (5.17) et (5.18) en annexe nous donne un meilleur aperçu.

Brand <fctr>	totalExposure <dbl>	Number.Observations <int>	Number.Claims <int>	Obs.Claim.Freq <dbl>
Fiat	9464.320	16653	714	0.07544123
Japanese (except Nissan) or Korean	31228.966	79031	2078	0.06654079
Mercedes, Chrysler or BMW	10392.166	19087	828	0.07967540
Opel, General Motors or Ford	21733.558	37287	1731	0.07964642
other	5676.076	9738	412	0.07258535
Renault, Nissan or Citroen	133460.245	216684	8905	0.06672399
Volkswagen, Audi, Skoda or Seat	18127.227	32384	1459	0.08048666

FIGURE 1.7 – Tableau de valeur des differentes marque de voiture

### 1.1.8 La variable Gas

C'est une variable catégorielle liée au type de carburant utilisé pour le véhicule. Dans notre jeu de données, les deux types utilisés sont Essence et Diesel. On a autant d'assurés qui conduisent une essence soit 50.2%, qu'une diesel. Les conducteurs d'essence ont la plus grande durée totale dans le portefeuille, soit 51.07% avec le plus faible taux de fréquence des sinistres à savoir 6.55% contre 7.49% ce qui est ce à quoi on s'attendait car en général, les véhicules utilisant le carburant diesel ont une fréquence des sinistres plus élevée.

### 1.1.9 La variable Density

Elle représente ici la densité de la population sur le lieu de résidence. Contrairement à l'âge du conducteur où à l'âge de la voiture, la densité a beaucoup de valeurs différentes. Dans ce cas, il est difficile de voir une tendance. Pour palier à cela, nous allons catégoriser la variable (figure (1.8)). Nous constatons que la densité  $I$  représente l'intervalle des assurés ayant la plus grande fréquence des sinistres soit 9.505%.

Region <fctr>	totalExposure <dbl>	Number.Observations <int>	Number.Claims <int>	Obs.Claim.Freq <dbl>
Aquitaine	14222.661	31211	1052	0.07396647
Basse-Normandie	6621.738	10848	451	0.06810900
Bretagne	27656.640	41986	1867	0.06750639
Centre	101843.464	159426	6460	0.06343068
Haute-Normandie	3147.222	8726	219	0.06958517
Ile-de-France	30016.992	69576	2575	0.08578474
Limousin	2376.002	4539	196	0.08249152
Nord-Pas-de-Calais	11346.794	27111	939	0.08275465
Pays-de-la-Loire	21791.752	38541	1569	0.07199972
Poitou-Charentes	11059.292	18900	799	0.07224694

FIGURE 1.8 – Tableau de valeurs des differentes densité au km<sup>2</sup>

---

### 1.1.10 Les variables PolicyID et Region

La variable PolicyID est une variable facteur à un seul niveau. En utilisant deux fois la fonction "table" sur R, nous n'obtenons que des "bons" (sans zéro) ce qui signifie que chaque PolicyID a une ligne unique correspondant à un identifiant unique de la police. Quant à la variable Region, c'est une variable catégorisée liée à la region du lieu de résidence.

Region <fctr>	totalExposure <dbl>	Number.Observations <int>	Number.Claims <int>	Obs.Claim.Freq <dbl>
Aquitaine	14322.167	31329	1055	0.07366204
Basse-Normandie	6657.901	10893	452	0.06788926
Bretagne	27753.464	42122	1871	0.06741501
Centre	102712.903	160601	6475	0.06303979
Haute-Normandie	3177.682	8784	220	0.06923285
Ile-de-France	30208.642	69791	2591	0.08577016
Limousin	2395.995	4567	197	0.08222055
Nord-Pas-de-Calais	11497.974	27285	944	0.08210142
Pays-de-la-Loire	21934.031	38751	1576	0.07185182
Poitou-Charentes	11163.409	19046	800	0.07166270

FIGURE 1.9 – Tableau de valeur des différentes regions

## 1.2 Analyse multivariée

Dans la sous-section précédente, nous avons présenté une analyse univariée des différentes variables et leurs interactions possibles avec les variables Claimb et Exposure. Nous allons actuellement présenter une possible interaction entre les autres variables.

### 1.2.1 Liaison entre variable qualitative et variable quantitative

Il est question d'observer les différentes interactions entre une variable qualitative et une variable quantitative. Pour une meilleure illustration, nous allons observer les différentes observations entre la qualité du carburant et l'âge de la voiture, aussi entre la qualité du carburant et l'âge du conducteur. Le graphique de la figure (5.22) donné en annexe nous montre que les véhicules de moins de 40 d'âge utilisent plus le carburant de type diesel et pour les plus de 40 ans d'âge, ils utilisent presque tous le carburant de type essence. En ce qui concerne le second graphique Figure (5.23), nous observons que les assurés âgés entre 20 et 23 ans, 90 et 98 ans utilisent uniquement le carburant de type essence et pour les autres âges, les assurés utilisent en grande partie le carburant diesel.

---

## 1.2.2 Liaison entre deux variables quantitatives

Nous travaillerons ici avec la matrice de corrélation de Pearson, car les modalités de chaque variable sont supérieures à 2. Cette analyse détermine si la corrélation entre deux variables données est linéaire, non linéaire, positive ou négative. La figure (1.10) nous présente les différentes corrélations entre les variables. On constate qu'il n'existe pas de corrélation entre les variables ClaimNb et CarAge; une corrélation négative entre les variables ClaimNb et DriverAge.

	ClaimNb	Exposure	CarAge	DriverAge	Density
ClaimNb	1.00	0.08	0.00	-0.01	0.01
Exposure	0.08	1.00	0.14	0.19	-0.11
CarAge	0.00	0.14	1.00	-0.05	-0.14
DriverAge	-0.01	0.19	-0.05	1.00	0.00
Density	0.01	-0.11	-0.14	0.00	1.00

FIGURE 1.10 – Corrélation entre les variables quantitatives

## 1.2.3 Liaison entre deux variables qualitatives

Nous allons étudier la relation entre différentes variables qualitatives. La corrélation de Cramer entre deux variables est donnée par le tableau suivant :

	Region	Brand	Gas	Power
Region	1	0, 19	0, 10	0, 05
Brand	0, 19	1	0, 10	0, 19
Gas	0, 10	0, 10	1	0, 33
Power	0, 05	0, 19	0, 33	1

TABLE 1.1 – Correlation entre deux variables qualitatives

On peut voir que toutes les variables sont peu corrélées et non liées entre elles.

# Chapitre 2

## Contexte, définition et notation

### 2.1 Fonction de régression

Considérons une réponse  $Y$  et un ensemble de caractéristiques  $X_1, \dots, X_p$  rassemblées dans le vecteur  $X \in \chi$  où  $\chi \subset \mathbb{R}^p$ . La réponse est typiquement le nombre de sinistres déclarés à la compagnie d'assurance par un assuré donné, la gravité moyenne des sinistres ou le montant total des sinistres en relation avec ce contrat. La structure de dépendance à l'intérieur du vecteur aléatoire  $(Y, X_1, \dots, X_p)$  est exploitée pour extraire l'information contenue dans  $X$  sur  $Y$ . Dans la tarification actuarielle, l'objectif est d'évaluer la prime pure aussi précisément que possible. Par conséquent, l'objectif est l'espérance conditionnelle  $\mu(X) = \mathbb{E}[Y|X]$  de la réponse  $Y$  (représentant le nombre de sinistres ou le montant du sinistre) compte tenu des informations disponibles  $X$ . Ainsi,  $\mu(X)$  est appelée la vraie prime (pure). Notons également que dans certaines applications,  $\mu(X)$  ne fait référence qu'à une seule composante de la prime pure. Dans le cas de la décomposition fréquence-sévérité de pertes d'assurance par exemple,  $\mu(X)$  peut être, soit le nombre attendu de sinistres assurés, soit le montant attendu des sinistres, soit la sévérité.

La fonction  $x \mapsto \mu(X) = \mathbb{E}[Y|X = x]$  est une fonction inconnue de l'actuaire et peut présenter un comportement complexe en  $x$ . C'est pourquoi cette fonction est approximée par une prime (de travail ou réelle)  $x \mapsto \pi(x)$  avec une structure plus simple. Lorsque l'analyste travaille dans le cadre de la décomposition fréquence-sévérité des pertes d'assurance,  $\pi(x)$  cible séparément le nombre attendu d'évènements assurés ou la gravité moyenne des sinistres. Une fois ajusté sur l'ensemble de données d'apprentissage à l'aide d'une procédure d'apprentissage appropriée, cela

---

produit des estimations  $\pi(x)$  pour  $\mu(x)$  ou des valeurs ajustées. Nous allons présenter et décrire les trois différents modèles que nous allons implémenter dans ce travail.

Les modèles linéaires généralisés (GLM) ont été introduits en 1974 par Nelder et Wedderburn et représente aujourd'hui le benchmark de la tarification non vie. Ces modèles doivent leur succès dans l'assurance à leur interprétabilité et à leur facile application dans les plans tarifaires. Ils permettent d'étudier la liaison entre  $Y$  et  $X$  via une fonction de lien entre la combinaison linéaire des variables explicatives et l'espérance conditionnelle sachant les valeurs prises par ces premiers prédictors. C'est-à-dire qu'il existe une fonction  $g$  monotone et différentiable telle que :

$$g(\mathbb{E}[Y|X_1 = x_1, \dots, X_p = x_p]) = \beta_0 + \beta_1 x_1 + \dots + \beta_p x_p + \epsilon.$$

La loi de  $Y$  doit appartenir à la famille de la dispersion exponentielle détaillée à la section 4.2.1 et chaque loi de cette famille possède une fonction de lien spécifique, appelée fonction de lien canonique et définie par :  $g(\mu) = (a')^{-1}(\mu)$ . L'estimation des coefficients  $\beta_i$ , s'effectue par la méthode de vraisemblance. Dans les modèles GLM, la composante déterministe qui est le score se calcule comme suit :

$$score_i = \beta_0 + \sum_{j=1}^p \beta_j x_{ij} = g(\mu_i), \quad i = 1, \dots, p.$$

Ainsi, le prédictor  $\mu_i$  est donné par :  $score_i = g(\mu_i) \iff \mu_i = g^{-1}(score_i)$ .

De nombreuses variables explicatives sont catégoriques et entrent efficacement dans les GLM après avoir été codées au moyen de variables binaires. Cependant, certaines variables d'évaluation importantes sont "continues". Ainsi, l'hypothèse de linéarité faite en générale sur la variable reponse est discutable, car les effets non-linéaires des facteurs de risque continues tels que la puissance du véhicule ou l'âge du conducteur doivent être introduits dans l'échelle du score pour capturer son effet. Les études sur la gestion de l'effet des variables continues à conduit dans les années 1980 à l'extension des GLM appelé modèle additifs généralisés (GAM). Ainsi un modèle GLM est un modèle GAM mais la réciprocity n'est pas vraie. En effet, d'après Denuit et al. 2009, la particularité des modèles GAM reside dans le fait qu'ils sont capables de maximiser la qualité de la prédiction d'une variable dépendante à partir de diverses distributions, en estimant des fonctions non-paramétriques des variables explicatives. Dans les modèles GAM, la composante déterministe qui est le score se calcule comme suit :

$$score_i = \beta_0 + \sum_{j=1}^{p_{cat}} \beta_j x_{ij} + \sum_{j=p_{cat}+1}^p \beta_j x_{ij} = g(\mu_i), \quad i = 1, \dots, p.$$

où  $p_{cat}$  est le nombre de variables catégorielles,  $p_{cont} = p - p_{cat}$  le nombre de variables continues et les fonctions  $f_j(\cdot)$ ,  $j = p_{cat}, \dots, p$  supposées régulières, traduisent l'influence des facteurs de risque continus  $x_{p_{cat}+1}, \dots, p$  sur la réponse  $y$ , elles sont non spécifiées et sont estimées à partir des données. La fonction de lien est la même que celle présentée précédemment dans le cas GLM compte tenue des fonctions continues.

Les modèles de Gradient Boosting (GBM) sont des modèles dits ensemblistes qui reposent sur une approche séquentielle dite adaptative qui consiste à combiner plusieurs modèles avec des pouvoirs de prédictions faibles (weak learners) pour obtenir un modèle avec un pouvoir de prédiction puissant. Dans cette approche, le score n'est pas estimé en une seule fois comme dans les GLM ou les GAM mais il est construit séquentiellement. Notons que plus cet algorithme de Boosting avance, plus il est corrigé et devient très adapté aux données. Notre objectif est de construire une estimation  $\hat{Y}_i$  de l'espérance  $\mathbb{E}[Y_i]$  sur la base des informations disponibles  $X_i$ . Cette estimation est une courbe non-linéaire de  $X_i$  à  $\hat{Y}_i$  :  $\hat{Y}_i = F(X_i)$ . Le meilleur estimateur  $F^*(\cdot)$  minimise la moyenne d'une fonction de perte  $L(\cdot)$  sur le portefeuille :

$$F^* = \operatorname{argmin}_F \frac{1}{n} \sum_{i=1}^n L(Y_i, F(X_i)).$$

La fonction de perte  $L(\cdot)$  est proportionnelle à la déviance. Une procédure courante consiste à restreindre  $F(\cdot)$  pour qu'elle appartienne à une classe paramétrique de fonctions. C'est-à-dire toute fonction  $F \in \mathcal{F}$  est un ensemble de  $M$  ajustements de base tels que :  $F(x) = F_0 + \sum_{m=1}^M \beta_m c_m(x)$ . Étant dans l'incapacité de trouver une solution globale, nous effectuons un algorithme itératif pour en déduire une approximation. La description de l'algorithme est donné comme suit :

- Initialisation : Choisir un estimateur initial  $F_0^*(x) = \beta_0^* = \operatorname{argmin}_\beta \frac{1}{n} \sum_{i=1}^n L(Y_i, \beta)$ .
- Itération : pour  $m = 1, \dots, M$ ,
  - Calcul du gradient négatif  $g = (g_i)_{i=1, \dots, n}$  de la fonction de perte :  $g_i = -\frac{1}{n} \frac{\partial L(Y_i, F_m(x_i))}{\partial F_m(x_i)} \Big|_{F_m(x_i)=F_m^*(x_i)}$  ;
  - Ajuster un arbre de regression  $c_m$  aux couples  $(x_i, g_i), i = 1, \dots, n$
  - Calcul  $\beta_m^*$  tel que :  $\beta_m^* = \operatorname{argmin}_\beta \frac{1}{n} \sum_{i=1}^n L(Y_i, \hat{F}_{m-1}(X_i) + \beta c_m(x_i))$  ;
  - Mise à jour :  $\hat{F}_m(X) = \hat{F}_{m-1}(X) + \beta_m^* c_m(X)$ .
- Résultat :  $\hat{F}(X) = \hat{F}_M(X)$ .

Le figure (2.1) suivante nous présente la répartition des primes en fonction des trois modèles à savoir le GLM, le GAM et le Boosting communément appelé GBM. Sur la figure (5.24) en annexe nous observons le niveau de significativité de chaque variable ; en utilisant la fonction "step()", on constate que toutes les variables sont significatives pour le modèle GLM. Il en est de même pour la régression GAM. Toutes les variables sont significatives avec une différence au niveau des sous-variables comme nous pouvons observer sur la figure (5.25) en annexe. En ce qui concerne la régression GBM avec une meilleure itération de validation croisée de 1352 pour 1500, la variable "DriverAge" est la plus influente des variables voir en annexe figure (5.26) et sur la figure (5.27) nous pouvons observer les dépendances partielles. Nous constatons que les primes sont plus dispersées et les coûts sont moins élevés dans le cas du GLM. Nous constatons aussi une grande similarité dans la répartition des primes entre les modèles GAM et GBM avec une répartition plus large pour le GBM. Comme référence, nous avons exécuté une régression simple de poisson de l'ordonnée à l'origine uniquement et on a obtenu  $\beta_0 = 2,662585$  qui correspond à une prédiction de base  $\bar{\pi} = 0,0697676$ .

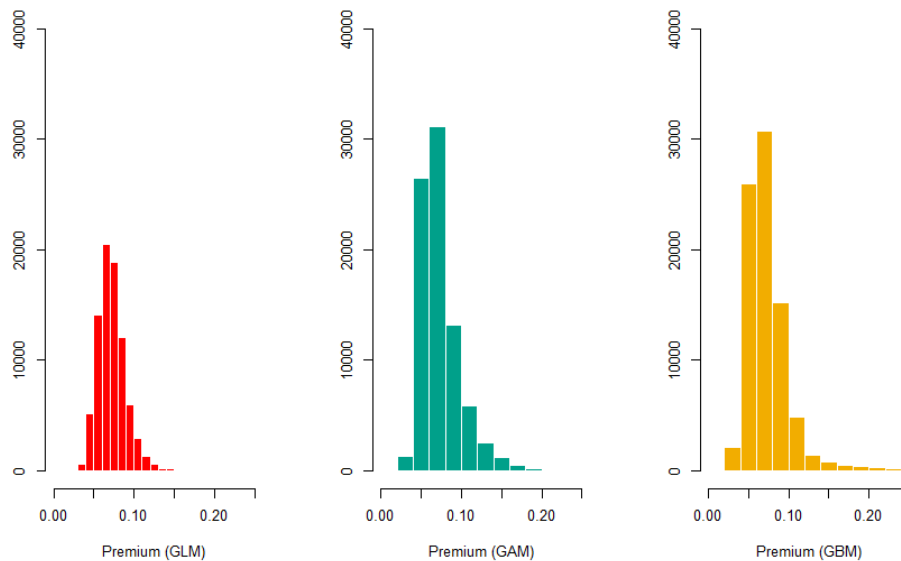


FIGURE 2.1 – Répartition des primes avec les modèles GLM, GAM et GBM

Nous travaillerons dans un contexte des totaux marginaux, un contexte dans lequel un équilibre global est souhaitable, c'est-à-dire où il est important que la somme des estimations ne s'écarte pas trop de la somme des observations réelles, à la fois au niveau de l'ensemble du portefeuille

---

et aussi plus localement, dans des classes significatives d'assurés. La raison en est évidente, car la somme des primes pures doit correspondre le plus exactement possible au total des sinistres afin que la compagnie d'assurance soit en mesure d'indemniser tous les tiers et bénéficiaires en exécution des contrats, sans excès ni déficit par application même de la définition de la prime pure qui stipule que les charges de frais et les charges de coûts du capital sont ajustées dans le calcul à un stade ultérieur, lors du passage aux primes commerciales. Cela se traduit naturellement par une contrainte d'équilibre global : en considérant que le total des sinistres est représentatif de l'expérience de l'année suivante, il est important que la somme des primes ajustées  $\hat{\pi}(x)$  corresponde le plus possible à la somme des réponses  $Y$  prises comme approximation du revenu total des primes (i.e. la somme des  $\mu(x)$ ). Mais l'équilibre local est également essentiel pour garantir une tarification compétitive.

Les mérites d'un outil de tarification donné peuvent être évalués en utilisant la paire  $(\mu(x), \hat{\pi}(x))$  de sorte que nous revenons au cas bivarié, même s'il y avait des milliers de caractéristiques comprises dans  $X$ . Ce qui compte vraiment, c'est la corrélation entre  $\hat{\pi}(x)$  et  $\mu(X)$  ; mais comme  $\mu(X)$  n'est pas observé, l'actuaire ne peut utiliser que sa version brutée  $Y$  pour révéler l'accord de la prime réelle  $\mu(X)$  avec sa contrepartie de travail  $\hat{\pi}(x)$ . Dans les applications d'assurance,  $\pi(X)$  est censé être utilisé comme une prime de sorte que la corrélation est importante, mais il est également essentiel que la somme des prévisions  $\hat{\pi}(x)$  corresponde le plus possible à la somme des pertes réelles, comme expliqué précédemment. Ceci est exprimé par la condition d'équilibre global et sa version locale.

## 2.2 Hypothèses techniques

En pratique, il y a au moins (ou presque toujours) une caractéristique continue contenue dans l'information disponible  $X$  et que la fonction  $\hat{\pi}$  est une fonction continue et croissante d'un score réel construit à partir de  $X$ . Par conséquent, l'hypothèse principale que nous ferons, sera de supposer que les prédicteurs  $\hat{\pi}(x)$  considérés, ainsi que l'espérance conditionnelle  $\mu(x)$  sont des variables aléatoires continues admettant des fonctions de densité de probabilité. Cependant, cela exclut les prédictions basées uniquement sur des caractéristiques discrètes, ainsi que les prédicteurs constants par morceaux, par exemple un arbre unique. En effet,  $\hat{\pi}(x)$  ne prend alors qu'un nombre limité de

valeurs. Puisque la tarification actuarielle est aujourd'hui basée sur des modèles plus sophistiqués (les arbres étant combinés en forêts aléatoires, par exemple), cette hypothèse de continuité ne restreint pas vraiment la généralité de l'approche. Nous supposons également que la réponse  $Y$  et le prédicteur  $\hat{\pi}$  sont non négatifs. Ces hypothèses sont retenues tout au long de ce document.

## 2.3 Notation

Nous notons par

$$F_{\hat{\pi}}(t) = P[\hat{\pi} \leq t], \quad \forall t \geq 0,$$

la fonction de repartition de  $\hat{\pi}(X)$ . Puisque  $f_{\hat{\pi}}$  correspond à la fonction de densité de probabilité,

$$F_{\hat{\pi}}(t) = \int_0^t f_{\hat{\pi}}(s) ds, \quad t \geq 0.$$

Soit  $F_{\hat{\pi}}^{-1}$  la fonction quantile (ou value-at-Risk) de  $\hat{\pi}$ , définie comme l'inverse généralisé de sa fonction de distribution  $F_{\hat{\pi}}$ , c'est-à-dire pour la probabilité  $\alpha$

$$F_{\hat{\pi}}^{-1}(\alpha) = \inf\{t | F_{\hat{\pi}}(t) \geq \alpha\}.$$

Nous considérons également la région basée sur le quantile comme des ensembles  $\chi$ , associés à  $\hat{\pi}$  comme

$$\chi_{\hat{\pi}, \underline{\alpha}\%} = \{x \in \chi : \hat{\pi} \leq F_{\hat{\pi}}^{-1}(\alpha)\},$$

où

$$\chi_{\hat{\pi}, \overline{\alpha}\%} = \{x \in \chi : \hat{\pi} > F_{\hat{\pi}}^{-1}(\alpha)\}.$$

D'un point de vue heuristique  $\chi_{\hat{\pi}, \underline{\alpha}\%}$  est l'ensemble des caractéristiques des assurés liées au risque le plus faible avec une proportion  $\alpha\%$ , tandis que  $\chi_{\hat{\pi}, \overline{\alpha}\%}$  est lié au plus grand risque avec la proportion  $1 - \alpha\%$  (où les petits et grands risques sont liés au prédicteur  $\hat{\pi}$ ). Bien entendu, ces ensembles peuvent également être définis pour la vraie prime  $\mu$ .

## 2.4 Ordre convexe

Il est à noter que plus  $\pi(X)$  est dispersé, puis il contient d'informations sur la prime réelle (à condition que la force de leur association reste la même). Le prédicteur constant  $\hat{\pi}(x) = \mathbb{E}[\mu]$ , le moins dispersé, n'apporte aucune information sur le risque relatif des différentes polices. Ainsi, la comparaison de la variabilité sous-jacente semble être importante dans le problème étudié.

L'ordre convexe est un outil probabiliste efficace pour évaluer la dispersion des variables aléatoires, au-delà des simples indicateurs tels que les écarts-types. Dans sa section 3.4, Denuit et al. (2019) nous rappelle qu'étant donné deux variables aléatoires  $Z$  et  $T$ ,  $T$  est plus petit que  $Z$  dans l'ordre convexe noté  $T \leq_{CX} Z$  si pour toute fonction convexe  $g$  telle que les espérances existent, on a :  $\mathbb{E}[g(T)] \leq \mathbb{E}[g(Z)]$ .

Une caractérisation importante de l'ordre convexe se fait par construction sur le même espace de probabilité en utilisant les attentes conditionnelles. Précisément, les variables aléatoires  $Z$  et  $T$  satisfont  $T \leq_{CX} Z$  si, et seulement s'il existe deux variables aléatoires  $\tilde{Z}$  et  $\tilde{T}$ , définie sur le même espace de probabilité, tel que  $\tilde{T}$  est distribué comme  $T$ ,  $\tilde{Z}$  est distribué comme  $Z$ , et  $\{\tilde{T}, \tilde{Z}\}$  est une martingale, c'est-à-dire  $\mathbb{E}[\tilde{Z}|\tilde{T}] = \tilde{T}$  se tient presque sûrement. Cela montre directement que l'augmentation du nombre de caractéristiques est bénéfique car  $X_1 \subseteq X_2 \implies \mu(X_1) \leq_{CX} \mu(X_2)$ . Passer de  $X_1$  à l'information plus riche  $X_2$  produit donc des primes plus dispersées  $\mu$ . En effet, soit  $\mathfrak{S}_i, i = 1, 2$  la filtration naturelle associée à  $X_i, i = 1, 2$  telle que  $\mathfrak{S}_1 \subset \mathfrak{S}_2$  (car par hypothèse  $X_1 \subset X_2$ ). Par définition,  $\mu(X_1) = \mathbb{E}[Y|X_1]$  et  $\mu(X_2) = \mathbb{E}[Y|X_2]$ ; alors posons  $\tilde{T} = \mathbb{E}[Y|\mathfrak{S}_1]$  et  $\tilde{Z} = \mathbb{E}[Y|\mathfrak{S}_2]$ . Nous avons donc,  $\tilde{T} =_d \mu(X_1)$ ,  $\tilde{Z} =_d \mu(X_2)$  et

$$\begin{aligned} \mathbb{E}[\tilde{Z}|\tilde{T}] &= \mathbb{E}[\mathbb{E}[Y|\mathfrak{S}_2]|\tilde{T}] \\ &= \mathbb{E}[Y|\tilde{T}] \\ &= \mathbb{E}[\mathbb{E}[Y|\mathfrak{S}_1]|\tilde{T}] \\ &= \mathbb{E}[\tilde{T}|\tilde{T}] = \tilde{T}. \end{aligned}$$

Donc  $\mu(X_1) \leq_{CX} \mu(X_2)$ .

# Chapitre 3

## Fonctions de perte cohérente pour la prime pure

### 3.1 Fonctions de perte de Bregman

Les primes pures correspondant à des attentes conditionnelles, elles ne peuvent être estimées de manière cohérente que si la perte attendue est minimale pour la réponse moyenne. Cela correspond au concept de fonction de perte cohérente. Précisément, une fonction de perte est cohérente pour la moyenne si aucune autre quantité ne conduit à une perte attendue plus faible que la moyenne. Les fonctions de perte de Bregman sont sous la forme

$$L(y, m) = l(y) - l(m) - l'(m)(y - m), \text{ pour une fonction convexe } l. \quad (3.1)$$

Les fonctions de perte de Bregman  $L(.,.)$  sont aussi appelées "q-loss function" données par Efron (1986). Ces fonctions de perte de Bregman mesurent l'écart entre  $y$  et  $m$  sur deux échelles distinctes ; sur l'échelle de  $l$  avec la différence  $l(y) - l(m)$  et sur l'échelle originale avec la différence  $y - m$ . Les deux différences sont ensuite combinées à l'aide du poids relatif  $l'(m)$ . Parfois, la relation (3.1) est définie en termes de fonction concave  $-l$ .

Sous réserve de faibles conditions de régularité, il a été prouvé dans la section 6 de Savage (1971) que pour toute fonction de perte donnée en (3.1) et toute réponse  $Y$ ,

$$\mathbb{E}[L(Y, \mathbb{E}[Y])] \leq \mathbb{E}[L(Y, m)], \text{ pour tout } m \text{ dans le support de } Y.$$

---

On dit alors que la classe des fonctions de perte de Bregman est cohérente pour la fonction moyenne (conditionnelle). Il convient de souligner que Savage (1971) se réfère à l'assurance pour motiver son approche en déclarant dans la section 2 que : "l'argent payable sous réserve d'une contingence, comme l'incendie accidentel d'une maison ou le résultat d'une course, peut-être considéré comme une marchandise. De telles marchandises sont explicitement traitées par les compagnies d'assurance et les bookmakers". Savage (1971) justifie ensuite l'utilisation de la fonction de perte de (3.1) par la révélation du taux de substitution individuel lorsqu'on lui présente un choix économique unique et relativement simple.

Les fonctions de perte de Bregman peuvent prendre une grande variété de formes : ces fonctions de perte peuvent être asymétriques, les sous-prédictions ou les sur-prédictions étant plus fortement pénalisées, et elles peuvent être strictement convexes ou avoir des segments concaves. Ainsi, en se limitant aux fonctions de perte qui génèrent la moyenne comme valeur optimale, ne nécessite par d'imposer la symétrie ou d'autres hypothèses sur la fonction de perte. En fait, parmi le nombre infini de fonctions de perte de Bregman, une seule est symétrique : la fonction de perte quadratique  $L(y, m) = (y - m)^2$  obtenue avec  $l(m) = m^2$ .

Dans les études actuarielles visant la prime pure, toute fonction de perte de Bregman serait admissible. Mais un choix judicieux de la fonction de perte reflétant la nature de la réponse étudiée ou sa relation moyenne-variance donne des résultats plus précis. Remarquez que le terme  $l(y)$  est parfois soustrait dans la relation (3.1). Cela ne modifie pas les prédictions et garantit que la fonction de perte soit définie sur toute la gamme de la réponse. Mentionnons également qu'il existe des classes de fonctions de perte qui sont compatibles avec d'autres indicateurs, comme les quantiles ou les expectibles par exemple. Une présentation plus accentuée et générale est donnée par Gneiting(2011). Les fonctions de perte sont parfois appelées fonctions de score dans la littérature statistique mais nous n'adoptons pas cette technologie afin d'éviter toute confusion avec le score impliqué dans  $\pi(x)$ .

---

## 3.2 Déviance de Tweedie

### 3.2.1 Famille de la dispersion exponentielle (ED)

Considérons la réponse  $Y$  évaluée sur un sous-ensemble  $S$  de la droite réelle  $(-\infty, +\infty)$ . Sa distribution est dite appartenir à la famille ED qu'on notera  $Y \sim ED(\theta, \frac{\phi}{\nu})$  si  $Y$  obéit à une fonction de masse de probabilité  $p_Y(y)$  ou à une fonction de densité de probabilité  $f_Y(y)$  de la forme

$$\left. \begin{array}{l} p_Y(y) \\ f_Y(y) \end{array} \right\} = \exp\left(\frac{y\theta - a(\theta)}{\frac{\phi}{\nu}}\right) C\left(y, \frac{\phi}{\nu}\right), \quad y \in S,$$

où

$\theta$  = paramètre de localisation à valeur réelle, appelé paramètre canonique

$a(\cdot)$  = fonction convexe monotone de  $\theta$

$\phi$  = paramètre d'échelle positif, appelé paramètre de dispersion

$\nu$  = constante positive connue, appelée le poids

$C(\cdot)$  = fonction de normalisation positive

$\mathbb{E}[Y] = \mu = \text{reponse moyenne} = a'(\theta)$

$\mathbb{V}[Y] = \frac{\phi}{\nu} a''(\theta) = \text{variance de la reponse}$ .

Dans la majorité des applications actuarielles, le poids correspond à une certaine mesure de volume, d'où la notation  $\nu$ .

Les paramètres  $\theta$  et  $\phi$  sont essentiellement des indicateurs de localisation et d'échelle, étendant la valeur moyenne  $\mu$  et la variance  $\sigma^2$  à toute la famille des distributions ED. Comme exemple de famille ED couramment utilisée et donnée dans le tableau (3.1), nous avons les distributions normale, binomiale, exponentielle, gamma, poisson qui correspondent au poids  $\nu = 1$ .

### 3.2.2 Distributions Tweedie

Les modèles de Tweedie sont des sous-classes la famille ED et y sont définis par une fonction de variance de puissance de la forme suivante :

$$\mathbb{V}(\mu) = \mu^\xi,$$

Dist.	$\theta$	$a(\theta)$	$\phi$	$\mu = E[Y]$	$V(\mu)$
$Ber(q)$	$\ln \frac{q}{1-q}$	$\ln(1 + \exp(\theta))$	1	$q$	$\mu(1 - \mu)$
$Bin(m, q)$	$\ln \frac{q}{1-q}$	$m \ln(1 + \exp(\theta))$	1	$mq$	$\mu \left(1 - \frac{\mu}{m}\right)$
$Geo(q)$	$\ln(1 - q)$	$-\ln(1 - \exp(\theta))$	1	$\frac{1-q}{q}$	$\mu(1 + \mu)$
$Pas(m, q)$	$\ln(1 - q)$	$-m \ln(1 - \exp(\theta))$	1	$m \frac{1-q}{q}$	$\mu \left(1 + \frac{\mu}{m}\right)$
$Poi(\mu)$	$\ln \mu$	$\exp(\theta)$	1	$\mu$	$\mu$
$Nor(\mu, \sigma^2)$	$\mu$	$\frac{\theta^2}{2}$	$\sigma^2$	$\mu$	1
$Exp(\mu)$	$-\frac{1}{\mu}$	$-\ln(-\theta)$	1	$\mu$	$\mu^2$
$Gam(\mu, \alpha)$	$-\frac{1}{\mu}$	$-\ln(-\theta)$	$\frac{1}{\alpha}$	$\mu$	$\mu^2$
$IGau(\mu, \alpha)$	$-\frac{1}{2\mu^2}$	$-\sqrt{-2\theta}$	$\frac{1}{\alpha}$	$\mu$	$\mu^3$

FIGURE 3.1 – Distributions ED avec  $\nu = 1$

où le paramètre de puissance  $\xi$  contrôle la forme de la distribution. Le tableau de la figure (3.2) nous présente toutes les distributions de Tweedie. Les valeurs négatives de  $\xi$  donnent des distributions continues sur tout l'axe réel. Pour  $0 < \xi < 1$ , il n'existe pas de distribution de dispersion exponentielle. Seuls les cas  $\xi \geq 1$  sont intéressants pour les applications en assurance.

	Type	Name
$\xi < 0$	Continuous	-
$\xi = 0$	Continuous	Normal
$0 < \xi < 1$	Non existing	-
$\xi = 1$	Discrete	Poisson
$1 < \xi < 2$	Mixed, non-negative	Compound Poisson sum with Gamma-distributed terms
$\xi = 2$	Continuous, positive	Gamma
$2 < \xi < 3$	Continuous, positive	-
$\xi = 3$	Continuous, positive	Inverse Gaussian
$\xi > 3$	Continuous, positive	-

FIGURE 3.2 – Distributions Tweedie et paramètres de puissance correspondant

### 3.2.3 Deviance Tweedie

Dans notre étude, nous supposons que la réponse obéit à une distribution de probabilité appartenant à la sous classe de tweedie. Pour le choix de la fonction appropriée  $l(m) = 2(m\theta_m - a(\theta_m))$  avec  $a'(\theta_m) = m$  dans (3.1), il résulte la déviance de la fonction de perte

$$\begin{aligned}
L(y, m) &= 2(y\theta_y - a(\theta_y)) - 2(m\theta_m - a(\theta_m)) - 2(y - m)\theta_m \\
&= 2(y\theta_y - a(\theta_y) - m\theta_m - a(\theta_m) - y\theta_m + m\theta_m) \\
&= 2(y\theta_y - y\theta_m - a(\theta_y) + a(\theta_m)) \\
&= 2[y(\theta_y - \theta_m) - a(\theta_y) + a(\theta_m)],
\end{aligned}$$

de la distribution de dispersion exponentielle où  $a'(\theta) = y$ . Notre travail étant centré sur des applications en assurance, nous nous limiterons dans ce document au cas  $\xi \geq 1$ . Pour ce cas, la déviance de la fonction de perte correspondante pour différents cas de  $\xi$  et en utilisant le tableau (3.1) est :

— Pour  $\xi = 1$  (cas Poisson) :

$$\theta_y = \ln(y), \theta_m = \ln(m), a(\theta_y) = e^{\theta_y} = y \text{ et } a(\theta_m) = m.$$

$$\text{Alors } L(y, m) = 2[y(\ln(y) - \ln(m)) - y + m] = 2[y\ln(\frac{y}{m}) - (y - m)].$$

— Pour  $\xi = 2$  (cas Gamma) :

$$\theta_y = -\frac{1}{y}, \theta_m = -\frac{1}{m}, a(\theta_y) = -\ln(-\theta_y) = \ln(y) \text{ et } a(\theta_m) = -\ln(-\theta_m) = \ln(m).$$

$$\text{Alors } L(y, m) = 2[y(-\frac{1}{y} + \frac{1}{m}) - \ln(y) + \ln(m)] = 2[-\ln(\frac{y}{m}) + \frac{y}{m} - 1].$$

— Pour  $\xi > 1$  et  $\xi \neq 2$  :

$$\theta_m = \frac{m^{1-\xi}}{1-\xi}, \theta_y = \frac{y^{1-\xi}}{1-\xi}, a(\theta_m) = \frac{m^{2-\xi}}{2-\xi} \text{ et } a(\theta_y) = \frac{y^{2-\xi}}{2-\xi}.$$

$$\text{Alors } L(y, m) = 2[y(\frac{y^{1-\xi}}{1-\xi} - \frac{m^{1-\xi}}{1-\xi}) - \frac{y^{2-\xi}}{2-\xi} + \frac{m^{2-\xi}}{2-\xi}] = 2(\frac{y^{2-\xi}}{(1-\xi)(2-\xi)} - \frac{ym^{1-\xi}}{1-\xi} + \frac{m^{2-\xi}}{2-\xi}).$$

Il en résulte,

$$L(y, m) = \begin{cases} 2(y\ln(\frac{y}{m}) - (y - m)) & \text{pour } \xi = 1, \\ 2(-\ln(\frac{y}{m}) + \frac{y}{m} - 1) & \text{pour } \xi = 2, \\ 2(\frac{y^{2-\xi}}{(1-\xi)(2-\xi)} - \frac{ym^{1-\xi}}{1-\xi} + \frac{m^{2-\xi}}{2-\xi}) & \text{pour } \xi > 1 \text{ et } \xi \neq 2. \end{cases} \quad (3.2)$$

### 3.3 Critère de comparaison

Soit  $\hat{\pi}(x)$  la réponse moyenne estimée de  $Y$  construite à partir d'un certain ensemble d'entraînement (toutes les formules de ce travail sont signifiées compte tenu de cet ensemble d'entraînement). Les performances respectives des modèles concurrents peuvent alors être évaluées sur la base d'un ensemble de validation  $\{(Y_i, X_i), i = 1, \dots, n\}$ , qui n'a pas été utilisé pour obtenir  $\hat{\pi}$ . Les perfor-

mances de  $\hat{\pi}$  sont généralement évaluées à l'aide de la déviance hors échantillon (d'entraînement), également appelée déviance prédictive et donné par :

$$D_n(\xi, \hat{\pi}) = \frac{1}{n} \sum_{i=1}^n L(Y_i, \hat{\pi}(X_i)) .$$

Lorsque  $n$  est assez grand, nous pouvons recourir à la valeur limite.

$$D_n(\xi, \hat{\pi}) \rightarrow \mathbb{E}[L(Y^{new}, \hat{\pi}(X^{new}))] \quad \text{quand } n \rightarrow \infty,$$

où  $(Y^{new}, \hat{\pi}(X^{new}))$  est une nouvelle observation indépendante et est distribuée comme  $(Y_i, X_i)$ , contenue dans l'ensemble d'apprentissage. Dans ce document, nous comparons les modèles sur la base de la version à grand échantillon de la déviance prédictive. Cette approche est utile dans les applications d'assurance ou l'analyse se trouve généralement dans une situation riche en données.

Dorénavant, nous comparons donc les modèles sur la base de la déviance prédictive de Tweedie

$$\mathbb{E}[L(Y^{new}, \hat{\pi}(X^{new}))] = \begin{cases} 2(\mathbb{E}[\hat{\pi}(X^{new}) - Y^{new} \ln(\hat{\pi}(X^{new}))] + \mathbb{E}[Y^{new} \ln(Y^{new}) - Y^{new}]) & \text{pour } \xi = 1 \\ 2\left(\mathbb{E}\left[\hat{\pi}(X^{new}) - \frac{Y^{new}}{\hat{\pi}(X^{new})}\right] - \mathbb{E}[\ln(Y^{new}) - 1]\right) & \text{pour } \xi = 2 \\ 2\left(\mathbb{E}\left[\frac{\hat{\pi}(X^{new})^{2-\xi}}{2-\xi} - \frac{Y^{new} \hat{\pi}(X^{new})^{1-\xi}}{1-\xi}\right] + \mathbb{E}\left[\frac{(Y^{new})^{2-\xi}}{(1-\xi)(2-\xi)}\right]\right) & \text{pour } \xi > 1, \xi \neq 2 \end{cases} \quad (3.3)$$

qui dépend du paramètre de puissance  $\xi$  et du prédicteur  $\hat{\pi}$  considéré. Remarquez que pour toute valeur de  $\xi$ , le dernier terme de (3.3) ne dépend pas de  $\hat{\pi}$  et n'est donc pas pertinent pour la comparaison de modèles, de sorte que dans le reste du document, nous utilisons réellement

$$D(\xi, \hat{\pi}) = \begin{cases} \mathbb{E}[\hat{\pi}(X^{new}) - Y^{new} \ln(\hat{\pi}(X^{new}))] & \text{pour } \xi = 1, \\ \mathbb{E}\left[\ln(\hat{\pi}(X^{new})) + \frac{Y^{new}}{\hat{\pi}(X^{new})}\right] & \text{pour } \xi = 2, \\ \mathbb{E}\left[\frac{\hat{\pi}(X^{new})^{2-\xi}}{2-\xi} - \frac{Y^{new} \hat{\pi}(X^{new})^{1-\xi}}{1-\xi}\right] & \text{pour } \xi > 1 \text{ et } \xi \neq 2. \end{cases} \quad (3.4)$$

pour évaluer les performances d'un prédicteur donné  $\hat{\pi}$ .

# Chapitre 4

## Evaluation des performances

### 4.1 La dominance de Bregman

#### Définition

La dominance de Bregman, également appelée dominance de la prévision est définie comme la dominance pour chaque fonction de perte de Bregman (3.1). Plus précisément  $\hat{\pi}_2$  est plus performante que  $\hat{\pi}_1$  en terme de dominance de Bregman si pour toute fonction de perte de Bregman  $L$ , l'inégalité suivante est vérifiée :

$$\mathbb{E}[L(Y, \hat{\pi}_2)] \leq \mathbb{E}[L(Y, \hat{\pi}_1)]$$

.

**Remarque :** Une présentation plus approfondie de ce concept de dominance de Bregman est donnée par Kruger et Zieger (2020).

La dominance de Bregman est donc une relation d'ordre stochastique particulière utilisée pour comparer les performances de deux estimateurs  $\hat{\pi}_1$  et  $\hat{\pi}_2$  pour les moyennes conditionnelles.

Il est bien connu que toute fonction convexe  $y \mapsto l(y)$  peut être représentée comme mélange de fonctions d'arrêt de perte ("stop-loss") :  $y \mapsto (y - t)_+$  qui sont des fonctions convexes.

**Théorème 4.1.1.** *Sous des conditions de régularité faible, que la fonction de perte de Bregman (3.1) admet une représentation de la forme*

$$L(y, m) = \int_{-\infty}^{+\infty} L_t(y, m) dH(t) \tag{4.1}$$

où  $H$  est une mesure non-négative et

$$L_t(y, m) = (y - t)_+ - (m - t)_+ - (y - m)I[m > t].$$

**Preuve 4.1.1.** *D'une part, on a :*

$$\begin{aligned} L_t(y, m) &= (y - t)_+ - (m - t)_+ - (y - m)I[m > t], \\ &= \begin{cases} t - m - y + m & \text{si } y \leq t < m, \\ y - t & \text{si } m \leq t < m, \\ 0 & \text{sinon,} \end{cases} \\ &= \begin{cases} t - y & \text{si } y \leq t < m, \\ y - t & \text{si } m \leq t < m, \\ 0 & \text{sinon.} \end{cases} \end{aligned}$$

*D'autre part, par définition de  $L(\cdot, \cdot)$  d'après (3.1),  $l$  est une fonction continue et dérivable.*

*Nous allons procéder suivant les différents cas de  $L_t$ .*

— *Pour  $y < m$ , on a :*

$$\begin{aligned} L(y, m) &= l(y) - l(m) - l'(m)(y - m) \\ &= \int_m^y l'(t)dt + l'(m)(m - y) \\ &= [(t - y)l'(t)]_{t=m}^y - \int_y^m l'(t)dt \\ &= \int_y^m (t - y)dl'(t), \text{ car intgration par parties} \\ &= \int_{-\infty}^{+\infty} L_t(y, m)dH(t). \end{aligned}$$

— *Pour  $y > m$ , on a :*

$$\begin{aligned} L(y, m) &= l(y) - l(m) - l'(m)(y - m) \\ &= -(l(m) - l(y)) - l'(m)(y - m) \\ &= \int_m^y l'(t)dt + [(y - t)l'(t)]_{t=m}^y \\ &= \int_y^m (y - t)dl'(t), \text{ car intgration par parties} \\ &= \int_{-\infty}^{+\infty} L_t(y, m)dH(t). \end{aligned}$$

— Pour  $m = y$ , on a :  $L(y, m) = 0 = \int_{-\infty}^{+\infty} L_t(y, m)dH(t)$ .

De plus, la mesure  $H$  est déterminée de façon unique car découle de  $L$ . Ce qui achève notre preuve.

Ainsi chaque fonction de notation cohérente pour la moyenne peut être écrite comme une moyenne pondérée sur les fonctions de perte ou extrémales  $L_t$ . Une conséquence importante est qu'une estimation qui est préférable en termes de toute fonction de perte de Bregman et est donc supérieure en termes de dominance de Bregman.

En se basant sur la représentation (4.1), Kruger et Ziegel (2020) ont établi dans leur théorème 2.1, la propriété suivante :

**Propriété 4.1.1.** *L'estimateur  $\hat{\pi}_2$  surpasse  $\hat{\pi}_1$  en termes de dominance de Bregman si et seulement si*

$$\mathbb{E}[(\hat{\pi}_2 - t)_+] + \mathbb{E}[(Y - \hat{\pi}_2)I[\hat{\pi}_2 > t]] \geq \mathbb{E}[(\hat{\pi}_1 - t)_+] + \mathbb{E}[(Y - \hat{\pi}_1)I[\hat{\pi}_1 > t]] \quad (4.2)$$

Les deux membre de l'inégalité (4.2) sont l'espérance de la fonction de perte élémentaire  $L_t(y, m)$  une fois que les primes stop-loss  $\mathbb{E}[(Y - t)_+]$  apparaissant dans les deux membres aient été annulées.

Pour la preuve de cette propriété, nous aurions besoin du résultat intermédiaire suivant :

**Corollaire 4.1.1.** *L'estimateur  $\hat{\pi}_2$  surpasse l'estimateur  $\hat{\pi}_1$  en terme de dominance de Bregman ssi  $\mathbb{E}[L_t(Y, \hat{\pi}_2)] \leq \mathbb{E}[L_t(Y, \hat{\pi}_1)]$ .*

**Preuve 4.1.2.** *Soit  $t \in \mathbb{R}$ . Par définition, la fonction  $L_t(\cdot, \cdot)$  est positive, alors d'après le théorème de Fubini-Lebesgue, l'échange de l'espérance et de l'intégration est permise. On a :*

$$\begin{aligned} \mathbb{E}[L_t(Y, \hat{\pi}_2)] &\leq \mathbb{E}[L_t(Y, \hat{\pi}_1)] \\ \iff \int_{-\infty}^{+\infty} \mathbb{E}[L_t(y, \hat{\pi}_2)]dH(t) &\leq \int_{-\infty}^{+\infty} \mathbb{E}[L_t(y, \hat{\pi}_1)]dH(t) \\ \iff \mathbb{E} \left[ \int_{-\infty}^{+\infty} L_t(y, \hat{\pi}_2)dH(t) \right] &\leq \mathbb{E} \left[ \int_{-\infty}^{+\infty} L_t(y, \hat{\pi}_1)dH(t) \right] \\ \iff \mathbb{E}[L(Y, \hat{\pi}_2)] &\leq \mathbb{E}[L(Y, \hat{\pi}_1)] \end{aligned}$$

---

$\iff \widehat{\pi}_2$  surpasse  $\widehat{\pi}_1$  en terme de dominance de Bregman.

D'où le resultat.

**Preuve de la propriété :** D'après le corollaire précédent,  $\widehat{\pi}_2$  surpasse  $\widehat{\pi}_1$  en terme de dominance de Bregman ssi

$$\mathbb{E}[L_t(Y, \widehat{\pi}_2)] \leq \mathbb{E}[L_t(Y, \widehat{\pi}_1)]$$

$$\iff \mathbb{E}[(y-t)_+] - \mathbb{E}[(\widehat{\pi}_2-t)_+] - \mathbb{E}[(y-\widehat{\pi}_2)I[\widehat{\pi}_2 > t]] \geq \mathbb{E}[(y-t)_+] - \mathbb{E}[(\widehat{\pi}_1-t)_+] - \mathbb{E}[(y-\widehat{\pi}_1)I[\widehat{\pi}_1 > t]]$$

$$\iff \mathbb{E}[(\widehat{\pi}_2-t)_+] - \mathbb{E}[(y-\widehat{\pi}_2)I[\widehat{\pi}_2 > t]] \geq \mathbb{E}[(\widehat{\pi}_1-t)_+] - \mathbb{E}[(y-\widehat{\pi}_1)I[\widehat{\pi}_1 > t]].$$

Ce qui achève la preuve.

**N.B :** L'inégalité (4.2) comprend un terme impliquant uniquement le prédicteur (sa transformée stop-loss par rapport au seuil  $t$ ) tandis que la distribution conjointe de la paire  $(Y, \widehat{\pi}_2)$  n'entre que dans le second terme. Cela fait écho aux conditions d'optimalité introduites dans Denuit et al. (2019) qui combinaient également les deux aspects : distribution marginale du prédicteur et distribution conjointe avec la réponse. Ce dernier découle des deux lemmes suivants qui entraine une généralisation de la proposition précédente.

**Lemme 4.1.1.** *Pour tout ensemble  $A \subset \mathbb{R}$  tribu borélienne,*

$$\mathbb{E}[(\widehat{\pi} - Y)_+ I[\widehat{\pi} \in A]] = \int_{-\infty}^{+\infty} \mathbb{P}[Y < \omega, \widehat{\pi} > \omega, \widehat{\pi} \in A] d\omega,$$

$$\mathbb{E}[(\widehat{\pi} - Y)_+ I[\widehat{\pi} \in A]] = \int_{-\infty}^{+\infty} \mathbb{P}[Y < \omega, \widehat{\pi} > \omega, \widehat{\pi} \in A] d\omega.$$

**Preuve 4.1.3.** *Soit  $F(\cdot | \widehat{\pi} = x)$  la fonction de repartition de la variable aléatoire conditionnelle*

$Y$  sachant  $\hat{\pi} = x$  et  $F_{\hat{\pi}}$  la fonction de repartition de  $\hat{\pi}$ . Par définition on a :

$$\begin{aligned}
\mathbb{E}[(\hat{\pi} - Y)_+ I[\hat{\pi} \in A]] &= \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x - y)_+ I[x \in A] dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \\
&= \int_{-\infty}^{+\infty} I[x \in A] \left( \int_{-\infty}^{+\infty} (x - y)_+ dF(y|\hat{\pi} = x) \right) dF_{\hat{\pi}}(x) \quad (\text{Fubini}) \\
&= \int_{-\infty}^{+\infty} I[x \in A] \int_{-\infty}^x (x - y) dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \\
&= \int_{-\infty}^{+\infty} I[x \in A] \int_{-\infty}^x \int_y^x d\omega dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \\
&= \int_{-\infty}^{+\infty} I[x \in A] \int_{\omega}^{+\infty} \int_{-\infty}^{\omega} d\omega dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \\
&= \int_{-\infty}^{+\infty} \left( \int_{\omega}^{+\infty} \int_{-\infty}^{\omega} I[x \in A] dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \right) d\omega \\
&= \int_{-\infty}^{+\infty} \left( \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} I[x \in A] I[x > \omega] I[y < \omega] dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \right) d\omega \\
&= \int_{-\infty}^{+\infty} \mathbb{E} \left[ I[\hat{\pi} \in A] I[\hat{\pi} > \omega] I[Y < \omega] \right] d\omega \\
&= \int_{-\infty}^{+\infty} \mathbb{P} \left[ Y < \omega, \hat{\pi} > \omega, \hat{\pi} \in A \right] d\omega.
\end{aligned}$$

De façon analogue, en appliquant le théorème de Fubini, on obtient :

$$\begin{aligned}
\mathbb{E}[(Y - \hat{\pi})_+ I[\hat{\pi} \in A]] &= \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (y - x)_+ I[x \in A] dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \\
&= \int_{-\infty}^{+\infty} I[x \in A] \int_x^{+\infty} (y - x) dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \\
&= \int_{-\infty}^{+\infty} I[x \in A] \int_x^{+\infty} \int_x^y d\omega dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \\
&= \int_{-\infty}^{+\infty} I[x \in A] \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} I[x \in (-\infty, \omega]] I[y \in [\omega, +\infty)] d\omega dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \\
&= \int_{-\infty}^{+\infty} \left( \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} I[x \in A] I[x \leq \omega] I[y \geq \omega] dF(y|\hat{\pi} = x) dF_{\hat{\pi}}(x) \right) d\omega \\
&= \int_{-\infty}^{+\infty} \mathbb{E} \left[ I[\hat{\pi} \in A] I[\hat{\pi} \leq \omega] I[Y \geq \omega] \right] d\omega \\
&= \int_{-\infty}^{+\infty} \mathbb{P} \left[ Y \geq \omega, \hat{\pi} \leq \omega, \hat{\pi} \in A \right] d\omega.
\end{aligned}$$

D'où le résultat.

**Lemme 4.1.2.** Soient  $X$  et  $Z$  deux variables aléatoires telles que  $\mathbb{E}[XZ]$  existe et est fini. Alors

$$\begin{aligned}\mathbb{E}[XZ] &= \int_0^{+\infty} \int_0^{+\infty} (H(x, z) - F(x) - G(z) + 1) dx dz + \int_{-\infty}^0 \int_{-\infty}^0 H(x, z) dx dz \\ &+ \int_{-\infty}^0 \int_0^{+\infty} (H(x, z) - G(z)) dx dz + \int_0^{+\infty} \int_{-\infty}^0 (H(x, z) - F(x)) dx dz,\end{aligned}$$

où  $H(x, z) = \mathbb{P}[X \leq x, Z \leq z]$ ,  $F(x) = \mathbb{P}[X \leq x]$ ,  $G(z) = \mathbb{P}[Z \leq z]$  sont les fonctions joint et de repartition de  $(X, Y)$ ,  $X$  et  $Z$  respectivement.

**Preuve 4.1.4.** Pour la variable aléatoire  $Y$ , les parties positives et négatives peuvent s'écrire :

$$Y_+ = \max\{Y, 0\} = \int_0^{+\infty} (1 - I[x \in [Y, \infty)]) dx,$$

$$Y_- = \max\{-Y, 0\} = \int_{-\infty}^0 I[x \in [Y, \infty)] dx.$$

Ainsi, on a :

$$\begin{aligned}(XZ)_+ &= X_+Z_+ + X_-Z_- \\ &= \int_0^{+\infty} \int_0^{+\infty} (1 - I[x \in [X, +\infty)])(1 - I[z \in [Z, +\infty)]) dx dz \\ &+ \int_{-\infty}^0 \int_{-\infty}^0 I[x \in [X, +\infty]] \times I[z \in [Z, +\infty]] dx dz \\ &= \int_0^{+\infty} \int_0^{+\infty} (1 - I[x \in [X, +\infty]] - I[z \in [Z, +\infty]] + I[x \in [X, +\infty]] \times I[z \in [Z, +\infty]]) dx dz \\ &+ \int_{-\infty}^0 \int_{-\infty}^0 I[x \in [X, +\infty]] \times I[z \in [Z, +\infty]] dx dz.\end{aligned}$$

En appliquant le théorème de Fubini sur cette dernière égalité, on obtient :

$$\begin{aligned}\mathbb{E}[(XZ)_+] &= \int_0^{+\infty} \int_0^{+\infty} \mathbb{E}[(1 - I[x \in [X, +\infty]] - I[z \in [Z, +\infty]] + I[x \in [X, +\infty]]I[z \in [Z, +\infty]])] dx dz \\ &+ \int_{-\infty}^0 \int_{-\infty}^0 \mathbb{E}[I[x \in [X, +\infty]] \times I[z \in [Z, +\infty]]] dx dz \\ &= \int_0^{+\infty} \int_0^{+\infty} (H(x, z) - F(x) - G(z) + 1) dx dz + \int_{-\infty}^0 \int_{-\infty}^0 H(x, z) dx dz.\end{aligned}$$

De même, on a :

$$\begin{aligned}
(XZ)_- &= X_+Z_- + X_-Z_+ \\
&= \int_0^{+\infty} \int_{-\infty}^0 (1 - I[x \in [X, +\infty]]) \times I[z \in [Z, +\infty]] dx dz \\
&+ \int_{-\infty}^0 \int_0^{+\infty} I[x \in [X, +\infty]] \times (1 - I[z \in [Z, +\infty]]) dx dz \\
&= \int_0^{+\infty} \int_{-\infty}^0 (I[z \in [Z, +\infty]] - I[x \in [X, +\infty]] \times I[z \in [Z, +\infty]]) dx dz \\
&+ \int_{-\infty}^0 \int_0^{+\infty} (I[x \in [X, +\infty]] - I[x \in [X, +\infty]] \times I[z \in [Z, +\infty]]) dx dz.
\end{aligned}$$

En appliquant le théorème de Fubini, on obtient :

$$\mathbb{E}[(XZ)_-] = \int_{-\infty}^0 \int_0^{+\infty} (H(x, z) - G(z)) dx dz + \int_0^{+\infty} \int_{-\infty}^0 (H(x, z) - F(x)) dx dz.$$

Puisque  $\mathbb{E}[XZ] = \mathbb{E}[(XZ)_+] + \mathbb{E}[(XZ)_-]$ , on a le résultat.

Ce qui achève notre preuve.

Nous allons énoncer le théorème généralisé suivant :

**Théorème 4.1.2.**  $\hat{\pi}_2$  surpasse  $\hat{\pi}_1$  en terme de dominance de Tweedie si et seulement si pour tout  $t \in \mathbb{R}$ ,  $\Psi_2(t) \geq \Psi_1(t)$ , où pour  $i \in \{1, 2\}$

$$\begin{aligned}
\Psi_i(t) &= \int_0^{+\infty} \left( \tau \mathbb{P}[\hat{\pi}_i > \omega, Y > \omega] + (1 - \tau) \mathbb{P}[\hat{\pi}_i > \omega, Y \leq \omega] \right) d\omega \\
&+ \tau \mathbb{E} \left[ (Y - \hat{\pi}_i)_+ I[\hat{\pi}_i > t] \right] - (1 - \tau) \mathbb{E} \left[ (\hat{\pi}_i - Y)_+ I[\hat{\pi}_i > t] \right], \quad \forall \tau \in (0, 1).
\end{aligned}$$

**NB :** Pour  $\tau = \frac{1}{2}$ , on obtient la proposition précédente.

## 4.2 La dominance Tweedie

Dans la tarification de l'assurance, il semble naturel de restreindre les fonctions de perte de Bregman aux déviations Tweedie.

**Définition**  $\hat{\pi}_2$  surpasse  $\hat{\pi}_1$  en terme de dominance de Tweedie si l'inégalité :  $D(\xi, \hat{\pi}_2) \leq D(\xi, \hat{\pi}_1)$  est vérifiée pour tout paramètre de puissance  $\xi \neq 1$ .

Le résultat suivant fournit à l'actuaire une condition suffisante pour qu'un modèle surpasse un concurrent en termes de dominance de Tweedie.

**Proposition 4.2.1.** *Considérons*

$$\Psi_\xi(\pi) = \begin{cases} \ln(\pi) & \text{pour } \xi = 2, \\ \frac{\pi^2 - \xi}{2 - \xi} & \text{sinon.} \end{cases} \quad (4.3)$$

Alors  $\hat{\pi}_2$  est plus performant que  $\hat{\pi}$  en termes de dominance Tweedie si

$$\mathbb{E}[\Psi_\xi(\hat{\pi}_1(X^{new}))] \geq \mathbb{E}[\Psi_\xi(\hat{\pi}_2(X^{new}))] \quad \text{pour tout } \xi \geq 1, \quad (4.4)$$

$$\mathbb{E}[Y^{new} I[\hat{\pi}_1(X^{new}) \leq t]] \geq \mathbb{E}[Y^{new} I[\hat{\pi}_2(X^{new}) \leq t]], \quad \text{pour tout } t \geq 0. \quad (4.5)$$

**Preuve 4.2.1.** *Nous allons établir les relations (4.4) et (4.5) suivant les différentes valeurs de  $\xi$  données par (4.2) et en effectuant un changement de variable. Pour  $\xi = 1$ ,  $\hat{\pi}_2$  est supérieur à  $\hat{\pi}_1$  si*

$$\begin{aligned} \mathbb{E}[\hat{\pi}_1(X^{new})] - \mathbb{E}[Y^{new} \ln(\hat{\pi}_1(X^{new}))] &\geq \mathbb{E}[\hat{\pi}_2(X^{new})] - \mathbb{E}[Y^{new} \ln(\hat{\pi}_2(X^{new}))] \\ \iff \mathbb{E}[\hat{\pi}_1(X^{new})] - \mathbb{E}[\hat{\pi}] + \mathbb{E}[Y^{new} \ln(\hat{\pi}_2(X^{new}))] - \mathbb{E}[Y^{new} \ln(\hat{\pi}_1(X^{new}))] &\geq 0 \end{aligned} \quad (4.6)$$

Puisque l'identité,

$$\begin{aligned} \mathbb{E}[Y^{new} \ln(\hat{\pi}_1(X^{new}))] &= \int_0^\infty \mathbb{E}[Y^{new} I[\ln(\hat{\pi}(X^{new})) > t]] dt - \int_{-\infty}^0 \mathbb{E}[Y^{new} I[\ln(\hat{\pi}(X^{new})) \leq t]] dt \\ &= \int_0^\infty \mathbb{E}[Y^{new} I[\hat{\pi}(X^{new}) > e^t]] dt - \int_{-\infty}^0 \mathbb{E}[Y^{new} I[\hat{\pi}(X^{new}) \leq e^t]] dt \quad (\text{posons } s = e^t) \\ &= \int_1^\infty \mathbb{E}[Y^{new} I[\hat{\pi}(X^{new}) > s]] \frac{1}{s} ds - \int_0^1 \mathbb{E}[Y^{new} I[\hat{\pi}(X^{new}) \leq s]] \frac{1}{s} ds \end{aligned}$$

est vérifiée pour tout prédicteur  $\hat{\pi}$ , nous avons

$$\begin{aligned} &\mathbb{E}[Y^{new} \ln(\hat{\pi}_2(X^{new}))] - \mathbb{E}[Y^{new} \ln(\hat{\pi}_1(X^{new}))] \\ &= \int_1^\infty (\mathbb{E}[Y^{new} I[\hat{\pi}_2(X^{new}) > s]] - \mathbb{E}[Y^{new} I[\hat{\pi}_1(X^{new}) > s]]) \frac{1}{s} ds \\ &\quad - \int_0^1 (\mathbb{E}[Y^{new} I[\hat{\pi}_2(X^{new}) \leq s]] - \mathbb{E}[Y^{new} I[\hat{\pi}_1(X^{new}) \leq s]]) \frac{1}{s} ds \\ &= \int_1^\infty (\mathbb{E}[Y^{new} (1 - I[\hat{\pi}_2(X^{new}) \leq s])] - \mathbb{E}[Y^{new} (1 - I[\hat{\pi}_1(X^{new}) \leq s])]) \frac{1}{s} ds \\ &\quad + \int_0^1 (\mathbb{E}[Y^{new} I[\hat{\pi}_1(X^{new}) \leq s]] - \mathbb{E}[Y^{new} I[\hat{\pi}_2(X^{new}) \leq s]]) \frac{1}{s} ds \\ &= \int_0^\infty (\mathbb{E}[Y^{new} I[\hat{\pi}_1(X^{new}) \leq s]] - \mathbb{E}[Y^{new} I[\hat{\pi}_2(X^{new}) \leq s]]) \frac{1}{s} ds \geq 0 \quad (\text{car (4.5)}) \end{aligned}$$

De plus,  $\Psi_\xi(\pi) = \pi$  ( $\xi = 1$ ), alors d'après (4.4),  $\mathbb{E}[\widehat{\pi}_1(X^{new})] - \mathbb{E}[\widehat{\pi}_2(X^{new})] \geq 0$ . Par conséquent (4.6) est vérifié.

En ce qui concerne le cas  $\xi = 2$ ,  $\widehat{\pi}_2$  est supérieur à  $\widehat{\pi}_1$  si

$$\begin{aligned} \mathbb{E}[\ln(\widehat{\pi}_1(X^{new}))] + \mathbb{E}\left[\frac{Y^{new}}{\widehat{\pi}_1(X^{new})}\right] &\geq \mathbb{E}[\ln(\widehat{\pi}_2(X^{new}))] + \mathbb{E}\left[\frac{Y^{new}}{\widehat{\pi}_2(X^{new})}\right] \\ \iff \mathbb{E}[\ln(\widehat{\pi}_1(X^{new})) - \ln(\widehat{\pi}_2(X^{new}))] + \mathbb{E}\left[\frac{Y^{new}}{\widehat{\pi}_1(X^{new})}\right] - \mathbb{E}\left[\frac{Y^{new}}{\widehat{\pi}_2(X^{new})}\right] &\geq 0 \end{aligned} \quad (4.7)$$

Puisque  $\xi = 2$  alors  $\Psi_\xi(\pi) = \ln \pi$ . Par conséquent, on a d'après (4.4)

$$\mathbb{E}[\ln(\widehat{\pi}_1(X^{new})) - \ln(\widehat{\pi}_2(X^{new}))] \geq 0.$$

De plus, pour tout prédicteur  $\widehat{\pi}$ , on a :

$$\begin{aligned} \mathbb{E}\left[\frac{Y^{new}}{\widehat{\pi}(X^{new})}\right] &= \int_0^\infty \mathbb{E}[Y^{new} I[\frac{1}{\widehat{\pi}(X^{new})} \geq t]] dt \\ &= \int_0^\infty [\mathbb{E}[Y^{new} I[\widehat{\pi}(X^{new}) \leq \frac{1}{t}]]] dt \quad (\text{posons } s = \frac{1}{t}) \\ &= \int_0^\infty [\mathbb{E}[Y^{new} I[\widehat{\pi}(X^{new}) \leq s]]] \frac{1}{s^2} ds. \end{aligned}$$

Ainsi,

$$\begin{aligned} \mathbb{E}\left[\frac{Y^{new}}{\widehat{\pi}_1(X^{new})}\right] - \mathbb{E}\left[\frac{Y^{new}}{\widehat{\pi}_2(X^{new})}\right] &= \int_0^\infty ([\mathbb{E}[Y^{new} I[\widehat{\pi}_1(X^{new}) \leq s]] - \mathbb{E}[Y^{new} I[\widehat{\pi}_2(X^{new}) \leq s]]) \frac{1}{s^2} ds \\ &\geq 0 \quad \text{car (4.5)} \end{aligned}$$

Donc la relation (4.7) est vérifiée.

Pour le cas final c'est-à-dire  $\xi > 1$  et  $\xi \neq 2$ ,  $\widehat{\pi}_2$  est supérieure à  $\widehat{\pi}_1$  si

$$\begin{aligned} \mathbb{E}\left[\frac{\widehat{\pi}_1(X^{new})^{2-\xi}}{2-\xi} - \frac{Y^{new} \widehat{\pi}_1(X^{new})^{1-\xi}}{1-\xi}\right] &\geq \mathbb{E}\left[\frac{\widehat{\pi}_2(X^{new})^{2-\xi}}{2-\xi} - \frac{Y^{new} \widehat{\pi}_2(X^{new})^{1-\xi}}{1-\xi}\right] \\ \iff \mathbb{E}\left[\frac{\widehat{\pi}_1(X^{new})^{2-\xi}}{2-\xi}\right] - \mathbb{E}\left[\frac{\widehat{\pi}_2(X^{new})^{2-\xi}}{2-\xi}\right] + \frac{1}{\xi-1} (\mathbb{E}[Y^{new} \widehat{\pi}_1(X^{new})^{1-\xi}] - \mathbb{E}[Y^{new} \widehat{\pi}_2(X^{new})^{1-\xi}]) &\geq 0 \end{aligned} \quad (4.8)$$

D'après (4.4) la première différence est positive par définition de  $\Psi_\xi$  pour  $\xi > 1$  et  $\xi \neq 2$ . De plus, pour tout prédicteur  $\widehat{\pi}$ , nous pouvons écrire

$$\begin{aligned} \mathbb{E}[Y^{new} \widehat{\pi}(X^{new})^{1-\xi}] &= \int_0^\infty \mathbb{E}[Y^{new} I[\widehat{\pi}(X^{new})^{1-\xi} \geq t]] dt \\ &= \int_0^\infty \mathbb{E}[Y^{new} I[\widehat{\pi}(X^{new}) \leq \frac{1}{t^{\frac{1}{1-\xi}}}] ] dt \quad (\text{posons } s = \frac{1}{t^{\frac{1}{1-\xi}}}) \\ &= \int_0^\infty \mathbb{E}[Y^{new} I[\widehat{\pi}(X^{new}) \leq s]] \frac{\xi-1}{s^\xi} ds \end{aligned}$$

Ainsi, nous avons

$$\begin{aligned} & \mathbb{E}[Y^{new} \hat{\pi}_1 (X^{new})^{1-\xi}] - \mathbb{E}[Y^{new} \hat{\pi}_2 (X^{new})^{1-\xi}] \\ &= \int_0^\infty (\mathbb{E}[Y^{new} I[\hat{\pi}_1(X^{new}) \leq s]] - \mathbb{E}[Y^{new} I[\hat{\pi}_2(X^{new}) \leq s]]) \frac{\xi - 1}{s^\xi} ds \geq 0 \end{aligned}$$

car (4.6) et le fait que  $\xi > 1$ . Par conséquent, la relation (4.8) est vérifiée.

Ce qui conclure notre preuve.

**Remarque :** À partir de la preuve de la proposition précédente, on peut déduire une représentation mixte de la déviance prédictive (4.2) utilisée pour la comparaison des modèles, à savoir

$$D(\xi, \hat{\pi}) = \int_0^\infty \left( \frac{1}{t^{\xi-1}} P[\hat{\pi} \leq t] - \frac{1}{t^\xi} \mathbb{E}[Y I[\hat{\pi} \leq t]] \right) dt. \quad (4.9)$$

Cette représentation de  $D(\xi, \hat{\pi})$  permet de relier la dominance de Bregman à la dominance de Tweedie.

Rappelons que si  $\hat{\pi}_1$  est plus performant que  $\hat{\pi}_2$  en termes de dominance de Bregman, alors la perte attendue pour chaque fonction de perte  $L$  ayant la forme donnée dans (3.1) est plus petite pour  $\hat{\pi}_1$ , ce qui est notamment le cas pour les écarts de Tweedie prédictifs.

En effet, pour un certains  $\hat{\pi}$  fixé,

$$\begin{aligned} \mathbb{E}[(\hat{\pi} - t)_+] + \mathbb{E}[(Y - \hat{\pi}) I[\hat{\pi} > t]] &= \mathbb{E}[(\hat{\pi} - t) I[\hat{\pi} > t]] + \mathbb{E}[(Y - \hat{\pi}) I[\hat{\pi} > t]] \\ &= \mathbb{E}[(Y - t)(1 - \hat{\pi} \leq t)] \\ &= \mathbb{E}[Y] - t - \mathbb{E}[Y I[\hat{\pi} \leq t]] + tP[\hat{\pi} \leq t]. \end{aligned}$$

Ainsi l'inégalité (4.2) de la dominance de Bregman se réécrit comme suit :

$$\mathbb{E}[Y] - t - \mathbb{E}[Y I[\hat{\pi}_1 \leq t]] + tP[\hat{\pi}_1 \leq t] \leq \mathbb{E}[Y] - t - \mathbb{E}[Y I[\hat{\pi}_2 \leq t]] + tP[\hat{\pi}_2 \leq t]$$

équivalente sous la forme suivante :

$$tP[\hat{\pi}_1 \leq t] - \mathbb{E}[Y I[\hat{\pi}_1 \leq t]] \leq tP[\hat{\pi}_2 \leq t] - \mathbb{E}[Y I[\hat{\pi}_2 \leq t]] \quad (4.10)$$

En divisant les deux membres de l'inégalité (4.10) par  $t^\xi$ , on obtient

$$\frac{1}{t^{\xi-1}} P[\hat{\pi}_1 \leq t] - \frac{1}{t^\xi} \mathbb{E}[Y I[\hat{\pi}_1 \leq t]] \leq \frac{1}{t^{\xi-1}} P[\hat{\pi}_2 \leq t] - \frac{1}{t^\xi} \mathbb{E}[Y I[\hat{\pi}_2 \leq t]] \quad (4.11)$$

donc (4.8) donne  $D(\xi, \hat{\pi}_1) \leq D(\xi, \hat{\pi}_2)$  (dominance de Tweedie).

**Remarque :** Au lieu de la dominance de Tweedie, l'actuaire pourrait sélectionner un paramètre spécifique  $\xi$  uniquement. C'est généralement le cas avec  $\xi = 1$  (régression de poisson pour les dénombrements),  $\xi = 2$  ou  $\xi = 3$  (régression gamma ou inverse gaussienne pour les gravités de sinistre). Dans ce cas, la condition (4.4) dans la proposition précédente est imposée uniquement pour cette valeur spécifique de  $\xi$ .

L'idée principale de la preuve de la proposition précédente est l'intégrale de la différence des fonctions dans (4.5) pondérée par :

$$\frac{\xi - 1 + I[\xi = 1]}{s^\xi} = \frac{\xi - 1 + I[\xi = 1]}{V(s)}.$$

$V(\cdot)$  est la fonction de variance associée à la distribution de la réponse. Les pondérations semblent donc être inversement proportionnelles à la variabilité. Dans le cadre de la fonction de variance de Tweedie,  $V(s)$  est une puissance de  $s$ , de sorte que des poids plus faibles sont attribués aux différences pour des valeurs plus grandes de la réponse.

La proposition précédente montre que l'amélioration d'un prédicteur  $\hat{\pi}$  peut être obtenue en :

- (i) augmentant le biais global mesuré sur une échelle modifiée induite par la fonction auxiliaire  $\Psi_\xi$  définie dans (4.3). Cela peut aller à l'encontre de la conservation des totaux observés
- (ii) En augmentant la dépendance entre  $\hat{\pi}$  et la réponse, de façon à diminuer

$$\begin{aligned} \mathbb{E}[Y^{new} I[\hat{\pi}(X^{new}) \leq t]] &= \mathbb{E}[\mathbb{E}[Y^{new} I[\hat{\pi}(X^{new}) \leq t] | \hat{\pi}(X^{new}) \leq t]] \\ &= \mathbb{E}[Y^{new} | \hat{\pi}(X^{new}) \leq t] \times \mathbb{E}[I[\hat{\pi}(X^{new}) \leq t]] \\ &= \mathbb{E}[Y^{new} | \hat{\pi}(X^{new}) \leq t] \times P[\hat{\pi}(X^{new}) \leq t] \end{aligned}$$

$$\mathbb{E}[Y^{new} I[\hat{\pi}(X^{new}) \leq t]] = Cov[Y^{new}, I[\hat{\pi}(X^{new}) \leq t]] + \mathbb{E}[Y^{new}] \times P[\hat{\pi}(X^{new}) \leq t].$$

Ce dernier moment partiel inférieur dépend essentiellement de la structure de corrélation de la paire  $(Y^{new}, \hat{\pi}(X^{new}))$ . Remarquez que les quantités apparaissant dans la décomposition ci-dessus sont étroitement liées à la dépendance d'espérance telle que définie par Wright(1987).

Dans le contexte de la tarification de l'assurance, le moment partiel inférieur peut être interprété

comme le revenu de primes le mieux profilé

$$\mathbb{E}[Y^{new} I[\hat{\pi}(X^{new}) \leq t]] = \mathbb{E}[\mu(X^{new}) I[\hat{\pi}(X^{new}) \leq t]] \approx \frac{1}{n} \sum_{i|\hat{\pi}(X_i) \leq t} \mu(X_i).$$

Ici,  $\sum_{i|\hat{\pi}(X_i) \leq t} \mu(X_i)$  est la prime réelle du sous-portefeuille formé en rassemblant tous les assurés dont la prime prédite est au plus égale à  $t$ . Ces assurés présentent les meilleurs profils de risque selon la prime candidate  $\hat{\pi}(X_i)$ .

Bien sûr, il y a un compromis entre ces deux objectifs lors du calcul de  $D(\xi, \hat{\pi}(X_i))$ . Si le modèle est très flexible, il peut alors produire un prédicteur qui est fortement corrélé avec la réponse et le moment partiel inférieur peut-être diminué dans une large mesure par rapport aux modèles imposant une forme rigide pour le prédicteur (comme les GLM). Un biais plus important peut alors être autorisé.

**Remarque :** La discussion dans le présent document considère l'ensemble d'apprentissage comme fixe. Ceci est différent de la discussion consacrée à la stabilité du modèle, ou au compromis biais-variance. En effet, la variance est considérée comme une propriété faisant varier l'ensemble d'apprentissage, et il en va de même pour le biais. Les attentes et les variances sont donc prises par rapport à des ensembles d'entraînement variables lorsque le compromis biais-variance est discuté, alors qu'ici, l'ensemble d'entraînement est maintenu fixe et nous considérons qu'un ensemble de validation suffisamment grand est disponible. Nous pensons que cette approche est plus conforme à la pratique des assurances.

### 4.3 Application à la régression de Poisson

La déviance de Poisson est de loin la plus utilisée dans l'application d'assurance. Elle s'applique par exemple au nombre de sinistres en assurance dommages, au nombre de décès en assurance vie et au nombre de transitions en assurance santé. Nous supposons que nous avons affaire à une réponse  $Y$  obéissant à la distribution de Poisson. Un vecteur  $X$  de caractéristiques est disponible pour prédire la réponse moyenne  $\mu(X)$ . Pour faciliter la présentation, nous supposons que les expositions sont unitaires.

En considérant (4.3)-(4.4) avec  $\xi = 1$ , le biais est donc mesuré sur l'échelle de la réponse. La

---

condition (4.4) favorise  $\hat{\pi}_2(X_i)$  si

$$\mathbb{E}[\mu(X^{new})] - \mathbb{E}[\hat{\pi}_2(X^{new})] \geq \mathbb{E}[\mu(X^{new})] - \mathbb{E}[\hat{\pi}_1(X^{new})].$$

En effet, pour  $\xi = 1$  dans (4.4), on a

$$\mathbb{E}[\hat{\pi}_1(X^{new})] \geq \mathbb{E}[\hat{\pi}_2(X^{new})] \implies \mathbb{E}[\mu(X^{new})] - \mathbb{E}[\hat{\pi}_2(X^{new})] \geq \mathbb{E}[\mu(X^{new})] - \mathbb{E}[\hat{\pi}_1(X^{new})].$$

De ce dernier, on observe qu'un biais plus important est donc avantageux. Par conséquent, l'adoption de la déviance de Poisson comme fonction de perte en dehors des GLM peut créer un écart de revenu total des primes

$$\mathbb{E}[Y^{new}] - \mathbb{E}[\hat{\pi}(X^{new})] = \mathbb{E}[\mu(X^{new})] - \mathbb{E}[\hat{\pi}(X^{new})] \approx \frac{1}{n} \sum_{i=1}^n [\mu(X_i) - \hat{\pi}(X_i)],$$

où  $\sum_{i=1}^n \mu(X_i)$  est le revenu réel de la prime pour l'ensemble de la validation, tandis que  $\sum_{i=1}^n \hat{\pi}(X_i)$  est celui obtenu en adoptant le prédicteur  $\hat{\pi}$  pour le calcul de la prime. Cet écart peut devenir très important avec des modèles très flexibles comme les réseaux de neurones ou le boosting.

# Chapitre 5

## Restauration de l'équilibre à l'échelle globale et locale

Il est question pour nous de restaurer l'équilibre entre la prime pure et l'approximation (le prédicteur  $\hat{\pi}$ ) afin de corriger le fort biais observé et décrit dans la partie précédente. Cette restauration va se faire sur l'échelle globale et locale. Pour ce faire, nous allons procéder à une autocalibration du prédicteur  $\hat{\pi}$ .

### 5.1 Autocalibrattion

**Définition :** (Autocalibration)

Un prédicteur  $\hat{\pi}$  est une prévision autocalibrée de  $Y$  si  $\hat{\pi}(X) = \mathbb{E}[Y|\hat{\pi}(X)]$  presque sûrement.

La présentation générale de ce concept qui va suivre est donnée dans la partie 3 de Kruger and Ziegel (2020).

**Proposition 5.1.1.** *Si  $\hat{\pi}$  est autocalibré, alors  $\hat{\pi}(X)$  est plus petit que  $Y$  au sens de l'ordre convexe i.e. le prédicteur est moins variable que la réponse au sens de l'ordre convexe.*

**Preuve 5.1.1.** *Considérons une fonction convexe  $g$ .*

*Puisque  $\hat{\pi}$  est autocalibré (i.e.  $\hat{\pi}(X) = \mathbb{E}[Y|\hat{\pi}(X)]$ ), alors d'après l'inégalité de Jensen, on a :*

$$\mathbb{E}[g(\hat{\pi}(X))] = \mathbb{E}[g(\mathbb{E}[Y|\hat{\pi}(X)])] \leq \mathbb{E}[\mathbb{E}[g(Y)|\hat{\pi}(X)]] = \mathbb{E}[g(Y)]$$

*i.e.*  $\mathbb{E}[g(\hat{\pi}(X))] \leq \mathbb{E}[g(Y)]$ . De plus  $\mathbb{E}[\hat{\pi}(X)] = \mathbb{E}[Y]$ . L'inégalité précédente est de manière générale équivalente à :

$$\mathbb{E}[\hat{\pi}(X)] = \mathbb{E}[Y] \quad \text{et} \quad \mathbb{E}[(\hat{\pi}(X) - t)_+] \leq \mathbb{E}[(Y - t)_+], \quad \forall t \geq 0.$$

D'où le résultat.

**Remarque :** La dominance de Bregman se réduit à l'ordre convexe pour les prédicteurs autocalibrés comme l'ont souligné Kruger and Ziegel (2020). En effet, d'après la dominance de Bregman on a :

$$\mathbb{E}[(Y - t)_+] + \mathbb{E}[(Y - Y)I[\hat{\pi}_2 > t]] \geq \mathbb{E}[(\hat{\pi} - t)_+] + \mathbb{E}[(Y - \hat{\pi})I[\hat{\pi}_1 > t]].$$

Or

$$\begin{aligned} \mathbb{E}[(Y - \hat{\pi})I[\hat{\pi}_1 > t]] &= \mathbb{E}[(Y - \mathbb{E}[Y|\hat{\pi}])I[\hat{\pi}_1 > t]] \quad (\text{car } \hat{\pi} \text{ autocalibre}) \\ &= \mathbb{E}[YI[\hat{\pi}_1 > t]] - \mathbb{E}[Y|\hat{\pi}] \times \mathbb{E}[I[\hat{\pi}_1 > t]] \\ &= \mathbb{E}[Y|\hat{\pi}] \times \mathbb{E}[I[\hat{\pi}_1 > t]] - \mathbb{E}[Y|\hat{\pi}] \times \mathbb{E}[I[\hat{\pi}_1 > t]] = 0 \end{aligned}$$

on a donc  $\mathbb{E}[(Y - t)_+] \geq \mathbb{E}[(\hat{\pi} - t)_+]$ . De plus  $\mathbb{E}[\hat{\pi}(X)] = \mathbb{E}[Y]$ . Par conséquent, on a l'ordre convexe entre  $\hat{\pi}$  et  $Y$ .

Dans le cadre d'une exigence technique légère, un moyen simple de rétablir l'équilibre global consiste à passer de  $\hat{\pi}$  à sa version corrigée de l'équilibre

$$\hat{\pi}_{BC}(X) = E[Y|\hat{\pi}(X)],$$

qui fait la moyenne de  $\mathbb{E}[Y]$ , comme le montre le résultat suivant.

**Propriété 5.1.1.** *Si la fonction  $s \mapsto \mathbb{E}[Y|\hat{\pi}(X) = s]$  est continue et croissante, alors la version corrigée de l'équilibre  $\hat{\pi}_{BC}(X)$  de la prime candidate  $\hat{\pi}$  satisfait à la propriété d'autocalibrage.*

Pour la preuve de cette propriété, nous aurons besoin du lemme suivant dont la preuve est donnée par Shaked et al. (2012) (Lemme 2.2).

---

**Lemme 5.1.1.** Soient  $X$  et  $Y$  deux variables aléatoires et soit  $\phi$  une fonction continue et croissante sur le support de  $X$ . Alors

$$\mathbb{E}[Y|X] = \mathbb{E}[Y|\phi(X)].$$

**Preuve 5.1.2.** La fonction de répartition de la variable aléatoire  $Z = \mathbb{E}[Y|X]$  est donnée par :  
 $\forall t \in \mathbb{R}$

$$F_Z(t) = \mathbb{P}[Z \leq t] = \mathbb{P}[x : \mathbb{E}[Y|X = x] \leq t].$$

Considérons l'ensemble  $A_t$  défini par  $A_t = \{x : \mathbb{E}[Y|X = x] \leq t\}$  pour  $t$  fixé. En utilisant le fait que  $\phi$  est bijective, on a :

$$\begin{aligned} B_t \equiv \phi(A_t) &\equiv \{\phi(x) : \mathbb{E}[Y|X = x] \leq t\} \quad \text{posons } s = \phi(x) \\ &= \{s : \mathbb{E}[Y|X = \phi^{-1}(s)] \leq t\} \\ &= \{s : \mathbb{E}[Y|\phi(X) = s] \leq t\}. \end{aligned}$$

Ainsi  $\forall t \in \mathbb{R}$ ,

$$\begin{aligned} F_Z(t) &= \mathbb{P}\{X \in A_t\} \\ &= \mathbb{P}[\phi(X) \in \phi(A_t)] \\ &= \mathbb{P}[\phi(X) \in B_t] \\ &= F_{\mathbb{E}[Y|\phi(X)]}(t). \end{aligned}$$

i.e.  $\forall t, F_{\mathbb{E}[Y|X]}(t) = F_{\mathbb{E}[Y|\phi(X)]}(t)$ .

Donc  $\mathbb{E}[Y|X] = \mathbb{E}[Y|\phi(X)]$ .

Ce qui achève la preuve du lemme.

**Preuve de la propriété :** Par hypothèse,  $s \mapsto \mathbb{E}[Y|\hat{\pi}(X) = s]$  est continue et croissante c'est-à-dire  $\hat{\pi}_{BC}(X)$  est continue et croissante dans le support de  $\hat{\pi}(X)$ . Alors d'après le lemme précédent, nous pouvons écrire  $\mathbb{E}[Y|\hat{\pi}_{BC}(X)] = \mathbb{E}[Y|\hat{\pi}(X)] = \hat{\pi}_{BC}(X)$ . Donc  $\hat{\pi}_{BC}$  résultant est effectivement autocalibré. Ce qui achève la preuve  $\square$

Dans les applications d'assurance, l'autocalibrage induit un équilibre local et impose l'équilibre financier non seulement au niveau du portefeuille mais aussi dans tout sous-portefeuille suffisamment

grand. Ce concept semble donc particulièrement attrayant dans un contexte de tarification.

De manière heuristique, la fonction  $s \mapsto \mathbb{E}[Y|\hat{\pi}(X) = s]$  est la valeur moyenne des points  $\mu(\hat{\pi}^{-1}(s))$  où  $\hat{\pi}^{-1}(s)$  désigne simplement l'ensemble des  $x$  remplissant cette condition i.e l'hypothèse de la propriété précédente. Visualisons cette situation avec notre jeu de données. Le but étant de vérifier si chaque prime  $\hat{\pi}_{BC}$  de la prime candidate  $\hat{\pi}$  de chacun de nos trois modèles satisfait à la propriété d'autocalibrage. Dans la figure (5.1), nous pouvons observer la variation de nos courbes lorsque  $s \in [0, 0.2]$  varie sur l'intervalle des primes possibles. En ce qui concerne le modèle linéaire généralisé et le modèle additif généralisé, il est difficile de se prononcer par contre pour le cas du modèle boosting, nous avons  $\mathbb{E}[Y|\hat{\pi}(X) = s] \sim s$  lorsque  $s < 0.1$  et  $\mathbb{E}[Y|\hat{\pi}(X) = s] < s$  à partir d'une certaine prime. Nous pouvons aussi remarquer que lorsque  $\hat{\pi}$  est proche de  $\mu$ , alors la courbe devrait être proche de la première bissectrice. Nous pouvons observer l'évolution de la différence entre  $\mathbb{E}[Y|\hat{\pi}(X) = F_{\hat{\pi}}^{-1}(u)]$  et  $F_{\hat{\pi}}^{-1}(u)$  lorsque  $u$  varie, qui est interprété comme le biais local additif sur le modèle  $\hat{\pi}$  et la version multiplicative est donné par la figure (5.29), avec  $u \mapsto \mathbb{E}[Y|\hat{\pi}(X) = F_{\hat{\pi}}^{-1}(u)]/F_{\hat{\pi}}^{-1}(u)$  qui est le multiplicateur utilisé pour passer de  $\hat{\pi}$  à  $\hat{\pi}_{BC}$ . La même information peut-être observée sur la figure (5.2) où l'axe des abscisses n'est plus une prime mais un quartile c'est-à-dire :  $u \mapsto \mathbb{E}[Y|\hat{\pi}(X) = F_{\hat{\pi}}^{-1}(u)]$ , nous observons une chute de la prime au niveau du quartile proche de 90%. Ainsi le biais positif observé respectivement sur les primes  $\hat{\pi}^{glm}$  et  $\hat{\pi}^{gam}$  montre que ces modèles sous-estiment le véritable prix du risque.

Il est à noter que si le correcteur a bien fonctionné, nous devons avoir une ligne (presque) horizontale,  $y = 0$  pour le biais additif et  $y = 1$  pour le biais multiplicatif.

## 5.2 Autocalibrage d'un prédicteur donné

Dans cette section, il est question pour nous d'observer les conditions d'autocalibration d'un prédicteur donné afin de rétablir l'équilibre. L'approche permettant de rétablir l'équilibre global ou de réduire le biais du prédicteur est de réconcilier le total prédit et observé sur l'ensemble d'apprentissage. En procédant ainsi, nous retrouvons la propriété d'équilibre global celle des totaux marginaux observés dans le cadre de la régression des modèles linéaire généralisés (GLM). Bien qu'ayant cet équilibre global imposé, on a pas de garantie que l'équilibre se maintient localement.

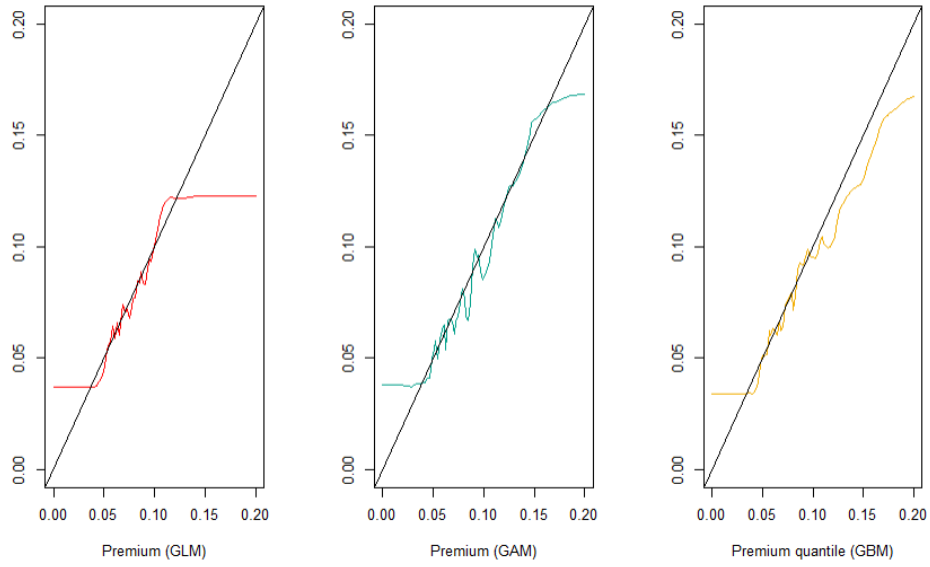


FIGURE 5.1 – Visualisation de  $s \mapsto \mathbb{E}[Y|\hat{\pi}(X) = s]$  où  $s$  est un certain niveau de prime, pour les différents modèles avec 0 degré de liberté.

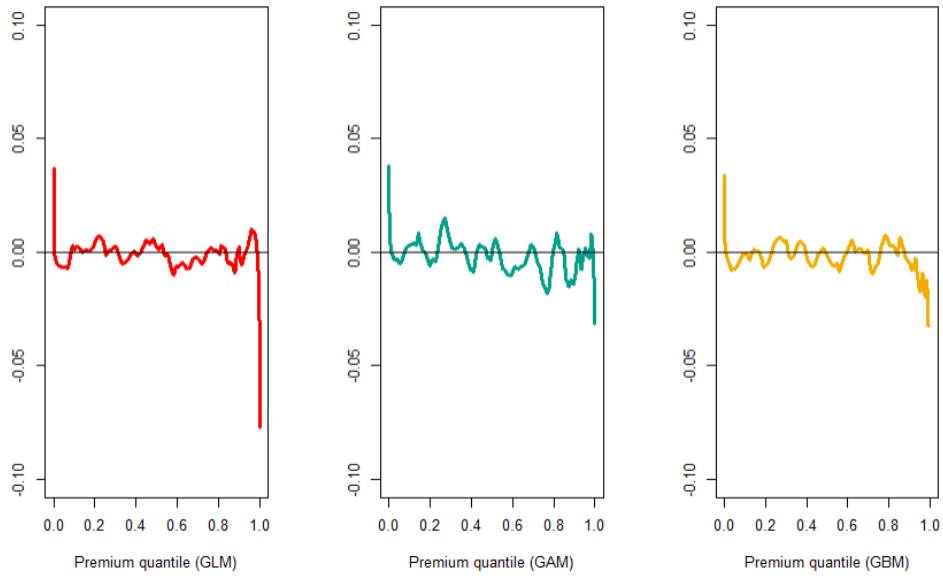


FIGURE 5.2 – Visualisation de  $u \mapsto \mathbb{E}[Y|\hat{\pi}(X) = F_{\hat{\pi}}^{-1}(u)] - F_{\hat{\pi}}^{-1}(u)$  avec la ligne droite correspondant à  $\mu = \hat{\pi}$

---

Il est question d'étendre les autres équations de vraisemblance imposées dans le cadre du GLM aux procédures générales d'apprentissage automatique. Pour ce fait, nous partons du principe que l'équilibre global n'est qu'une des équations de vraisemblance du GLM sous lien canonique, correspondant à l'intercept et devons imiter la manière dont le GLM local procède pour l'ajustement en définissant des voisinages significatifs ou un apprentissage statistique.

Pour y arriver, nous travaillons sur la fonction de lien canonique. La seule caractéristique entrant dans l'analyse est le logarithme du prédicteur, c'est-à-dire  $\ln[\hat{\pi}(x_i)]$ , de sorte que le GLM local soit ajusté à  $(Y_i, e_i, \hat{\pi}(x_i))$ ,  $i = 1, \dots, n$ , pour une certaine exposition pertinente  $e_i$  avec  $e_i = Y_i - \hat{\pi}(x_i)$ . Une solution intuitivement acceptable consisterait à imposer des contraintes marginales sur les voisinages locaux définis par la moyenne  $\hat{\pi}$ . Cela permet certains transferts locaux de sinistres et de primes des assurés voisins et met ainsi en oeuvre des conditions d'équilibre local dans les sous-portefeuilles correspondant à ces voisinages. D'après Loader, 1999, il s'agit essentiellement de l'approche GLM qui permet à l'actuaire de maintenir la relation avec le MMT.

Pour obtenir une version autocalibrée  $\hat{\pi}_{BC}$  de  $\hat{\pi}$ , considérons un profit de risque spécifique  $x$ . Considérons une fonction de poids  $\nu(\cdot)$  définie sur  $[-1, 1]$ , continue avec un pic à 0. À partir de cette fonction de poids, nous attribuons à chaque  $(Y_i, e_i, \hat{\pi}(x_i))$ ,  $i = 1, \dots, n$  un poids  $\nu_i(\hat{\pi}(x))$ . Ces fonctions de poids dépendent de la distance relative de  $\hat{\pi}(x_i)$  par rapport à  $\hat{\pi}(x)$ . Ici, nous avons fait le choix de la fonction de poids tricube pour laquelle  $\nu_i(\hat{\pi}(x))$  est plus grande pour les assurés  $i$  tels que  $\hat{\pi}(x_i)$  est proche de  $\hat{\pi}(x)$  et diminue lorsque  $\hat{\pi}(x_i)$  s'éloigne de  $\hat{\pi}(x)$ . Notamment, il existe plusieurs alternatives disponibles tels que les fonctions de poids rectangulaire (uniformes), gaussienne ou Epanechnikov.

La procédure d'autocalibration sur "R" s'est faite au moyen du package "locfit". Locfit dispose de deux fonctions, `locfit()` et `locfit.Raw()`, pour la régression locale. La fonction `locfit()` utilise le langage de modèle S pour spécifier le modèle de régression local, tandis que `locfit.raw()` à des arguments séparés pour les variables prédicteur et les variables réponses. Dans un autre sens, les deux fonctions sont identiques. Nous allons décrire quelques arguments de cette fonction :

- **Paramètre de lissage** : il est donné par  $\alpha$  contrôle la largeur de bande basée sur les plus proches voisins. Lorsque alpha est donné comme un nombre unique, il représente une

fraction de plus proche voisin. En général,  $\alpha$  peut être spécifié sous forme de vecteur à deux composantes. La seconde composante représente une largeur de bande constante, donc  $\alpha = c(0.1)$  signifie que  $h(x) = 1$  est utilisé partout. Si la composante du plus proche voisin et la composante fixe sont toutes deux non nulles les deux largeurs de bande sont calculées, et  $h(x)$  sera choisi comme la plus grande composante. Plus précisément, si  $\alpha = (\alpha_0, \alpha_1)$ , la largeur de bande  $h(x)$  est calculée comme suit :

- $k = \lfloor x \rfloor$  ;
- calculer  $d_i = |x - x_i|$ ,  $i = 1, \dots, n$  et trouver le  $k - ime$  plus petit  $d_k$  ;
- Résultat :  $h(x) = \max(d_k, \alpha_1)$ .

— **Degré du polynôme local.** Le degré du polynôme local est spécifié par le paramètre "deg" : deg=1 spécifie un ajustement linéaire local ; deg=2 espèces de quadratique local (par défaut). Pour les ajustements univariés, `locfit.raw()` supporte n'importe quel degré, bien qu'il y ait généralement peu de raisons d'utiliser des degrés supérieurs à 3. Pour les ajustements multivariés, deg=3 est le degré maximum.

— **La fonction de poids.** `Locfit` supporte un certain nombre de fonctions de poids, comme le présente Loader 1999. Ces derniers sont sélectionnés à l'aide de l'argument "kern" de la fonction `locfit.raw()`.

Comme nous pouvons le voir dans le tableau (5.1) suivant présentant les différents quartiles, on constate une amélioration des différents prédicteurs et la correction du biais global.

Prédicteur	mean	10%	90%	Prédicteur	mean	10%	90%
$\hat{\pi}_{glm}$	0.0724	0.0524	0.0944	$\hat{\pi}_{glm\_bc}$	0.0715	0.0542	0.0920
$\hat{\pi}_{gam}$	0.0732	0.0496	0.1059	$\hat{\pi}_{gam\_bc}$	0.0712	0.0501	0.0982
$\hat{\pi}_{gbm}$	0.0732	0.0483	0.1007	$\hat{\pi}_{gbm\_bc}$	0.0707	0.0468	0.0975

TABLE 5.1 – Quartiles (10%, Mean, 90%) de chaque prédicteur.

La figure (5.3) nous présente différent histogramme d'autocalibration des primes de chaque modèle. Nous pouvons observer sur la figure (2.1) que comparer à la figure (5.3), la répartition des primes par les trois modèles autocalibrés est presque identique.

Pour effectuer une comparaison entre  $\hat{\pi}$  et  $\hat{\pi}_{bc}$ , nous avons sur la figure (5.30) (en annexe), tracé QQ de  $\hat{\pi}_{bc}$  contre  $\hat{\pi}$  qui correspond au tracé de  $u \mapsto \{F_{\hat{\pi}}^{-1}(u), F_{\hat{\pi}_{bc}}^{-1}(u)\}$ . Sur la figure (5.31) (en annexe) un nuage de points de  $\{\hat{\pi}(x_i), \hat{\pi}_{bc}(x_i)\}$  pour un sous ensemble de la base de donnée de

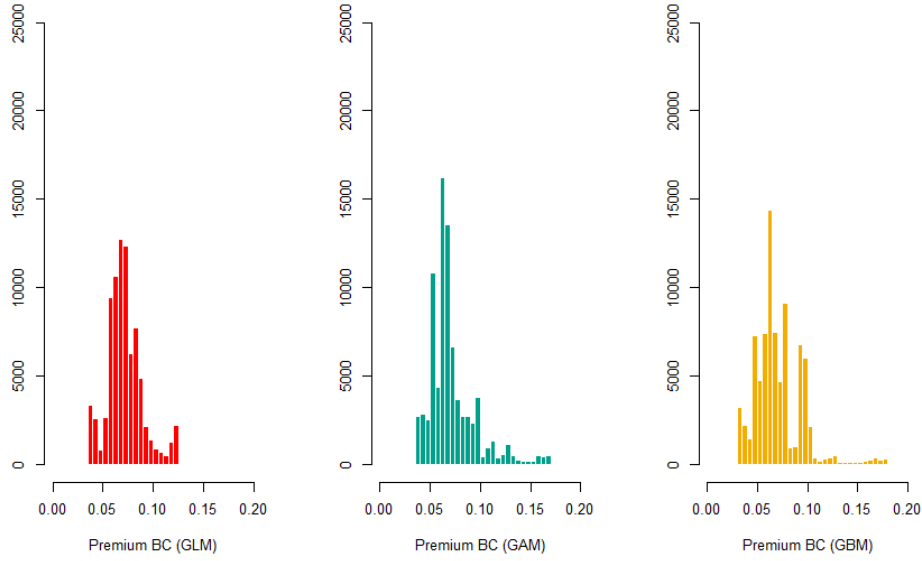


FIGURE 5.3 – Répartition des primes autocalibrés  $\hat{\pi}_{BC}$  avec les modèles GLM, GAM et GBM (boosting).

validation.

Comme nous avons pu l’observer avec les figures (2.1) et (5.3), lorsqu’on passe de  $\hat{\pi}$  à  $\hat{\pi}_{bc}$  la distribution des primes change substantiellement et comme le montre les figures (5.30) et (5.31), la transformation n’est pas (parfaitement) monotone. Néanmoins, la corrélation de rang entre  $\hat{\pi}$  et  $\hat{\pi}_{bc}$  est plutôt élevée supérieur à 90% pour les trois modèles comme le présente la figure (5.4)

Faisant suite à la courbe  $u \mapsto \mathbb{E}[Y | \hat{\pi}_{bc}(X) = F_{\hat{\pi}_{bc}}^{-1}(u)]$ , nous pouvons constater la correction du biais qui était visible sur les figures (5.2) et (5.1). Sur les figures (5.5) et (5.6) nous pouvons constater que  $\pi_{BC}$  en trait fort est proche de  $\mu$  dans les trois modèles. Ainsi, l’autocalibration de la prédiction du modèle GBM nous donne de meilleures primes.

Les mêmes conclusions sont visibles pour la fonction  $s \mapsto \mathbb{E}[Y | \hat{\pi}_{bc}(X) = s]$  est la valeur moyenne des points  $\mu(\hat{\pi}_{bc}^{-1}(s))$  comme le montre les figures (5.28) et (5.29) en annexe.

### 5.3 Performance d’un prédicteur

Bien qu’ayant un prédicteur, il est tout naturel de s’interroger sur la performance de ce dernier. Pour répondre à cette question, Denuit et al. (2019) ont suggéré de baser l’étude de la mesure de la performance d’un prédicteur sur la courbe de concentration et la courbe de Lorenz. une étude plus approfondie sur les courbes de concentration et de Lorenz a été effectuée dans le livre de Yitzhaki

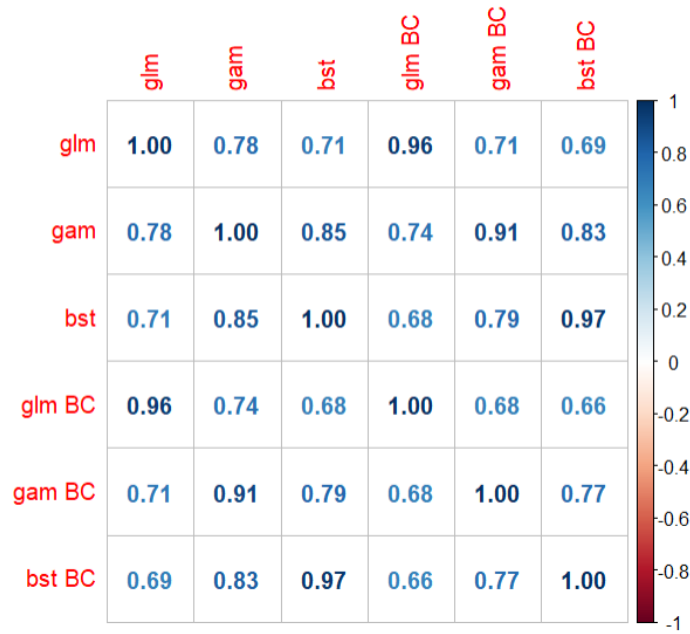


FIGURE 5.4 – Corrélation de rang Spearman entre  $\hat{\pi}^{glm}$ ,  $\hat{\pi}^{gam}$ ,  $\hat{\pi}^{gbm}$  et leurs valeurs autocalibrées homologues  $\hat{\pi}_{bc}^{glm}$ ,  $\hat{\pi}_{bc}^{gam}$  et  $\hat{\pi}_{bc}^{gbm}$  sur les données de validation.

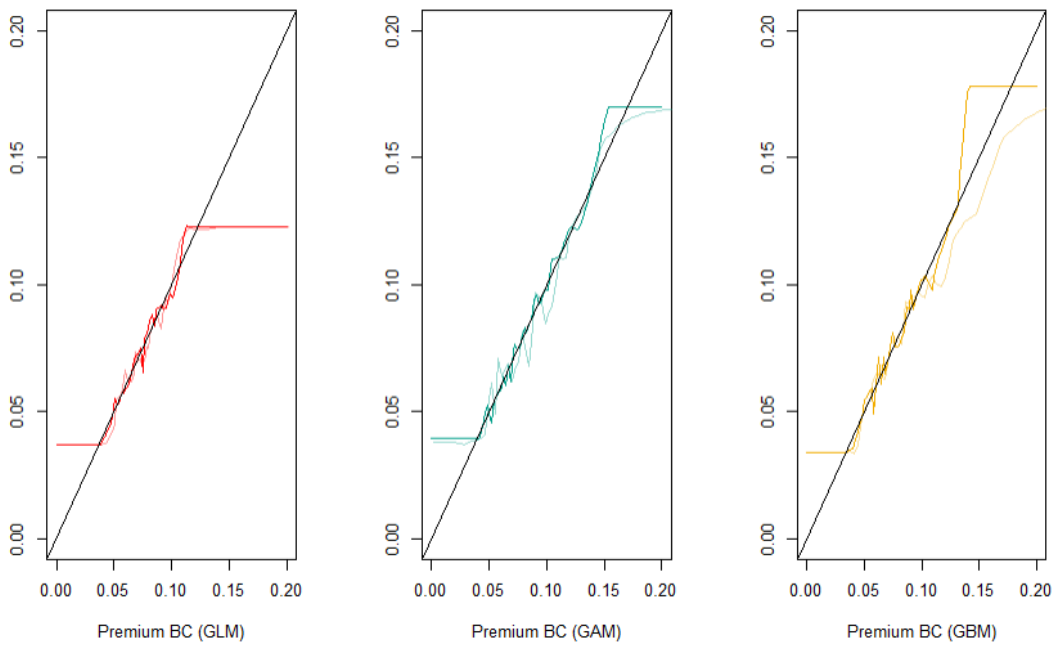


FIGURE 5.5 – Visualisation de  $s \mapsto \mathbb{E}[Y | \hat{\pi}_{BC}(X) = s]$  où  $s$  est un certain niveau de prime, pour les différents modèles avec 0 degré de liberté avec la fine courbe colorée représente la prime prédite  $\hat{\pi}$

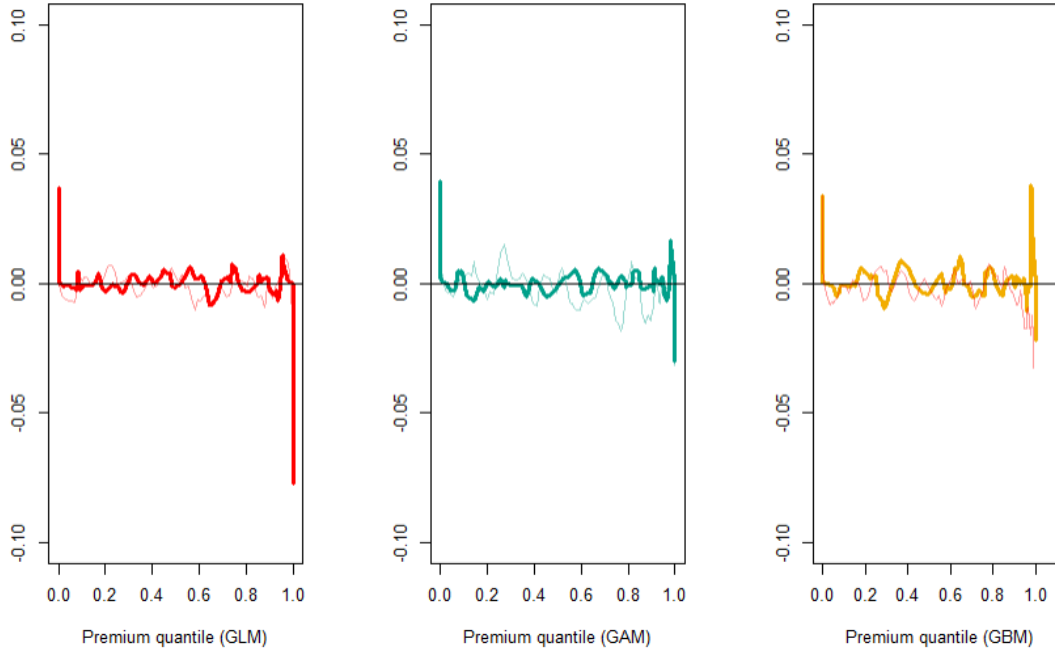


FIGURE 5.6 – Visualisation de  $u \mapsto \mathbb{E}[Y|\hat{\pi}_{BC}(X) = F_{\hat{\pi}_{BC}}^{-1}(u)] - F_{\hat{\pi}_{BC}}^{-1}(u)$  avec la ligne droite correspondant à  $\mu = \hat{\pi}_{BC}$  où la fine courbe colorée représente la prime prédite  $\hat{\pi}$

and Schechtman (2013). Dans notre étude, nous nous limiterons à la définition de ces deux notions et aux critères de comparaison.

**Definition :** La courbe de concentration de la prime réelle  $\mu(X)$  par rapport à la prime de travail  $\hat{\pi}$  basée sur les informations contenues dans le vecteur  $X$  est définie comme :

$$\alpha \mapsto CC[\mu(X), \hat{\pi}(X); \alpha] = \frac{\mathbb{E}[\mu(X)I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]]}{\mathbb{E}[\mu(X)]}.$$

Nous allons donner une brève signification de cette de concentration de  $\mu(X)$ . En considérant cette définition, nous observons que la courbe  $CC[\mu(X), \hat{\pi}(X); \alpha]$  représente le pourcentage du revenu total de la prime correspondant au sous-portefeuille  $\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)$ , c'est-à-dire au  $\alpha\%$  de contrats dont la prime est plus petite que  $\hat{\pi}$ . Puisque ces contrats ont un pôle de risque plus faible, ils risquent de quitter le portefeuille dû à la concurrence. Afin de les conserver, il est important de ne pas surfacturer ce groupe d'assurés. D'où l'importance de la courbe de concentration pour évaluer le bien-fondé de la prime envisagée.

Les fonctions  $\alpha \mapsto CC[\mu(X), \hat{\pi}(X); \alpha]$  peuvent être visualisées avec notre jeu de donnés. Sur

les figures (5.7) nous pouvons visualiser à gauche les deux modèles GLM et à droite l'aire entre nos deux différentes courbes. Sur la figure (5.8) nous visualisons les deux modèles GAM et sur la figure (5.9) nous avons les deux modèles du boosting. Nous pouvons constater que les courbes de concentration des primes prédites et autocalibrés par les modèles GLM et GBM sont proches en comparaison de celles données par le modèle GBM.

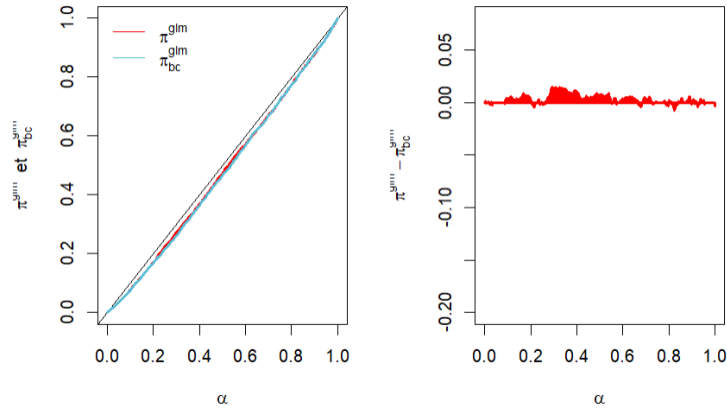


FIGURE 5.7 – Evolution de  $CC[\mu, \hat{\pi}; \alpha]$  cas du GLM

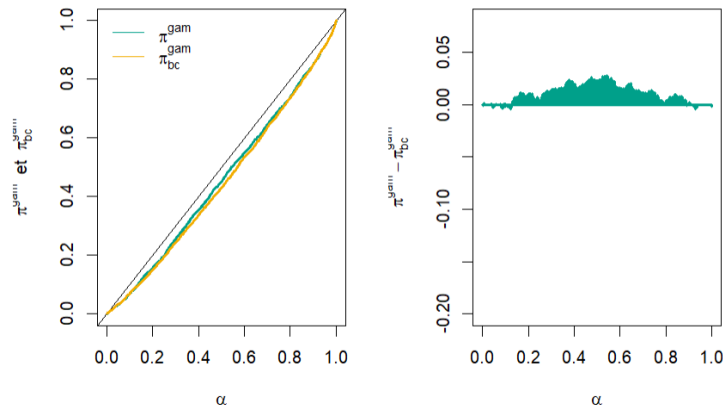


FIGURE 5.8 – Evolution de  $CC[\mu, \hat{\pi}; \alpha]$  cas du GAM

La courbe de concentration ne suffit pas à elle seule à évaluer les performances de  $\hat{\pi}$ . En effet, dans la tarification, nous devons en plus de la prise en compte de la prime réelle, prendre en compte le montant des primes  $\hat{\pi}$ . C'est pourquoi nous utilisons également la courbe de Lorenz du prédicteur, en plus de la courbe de concentration de la reponse par rapport au prédicteur.

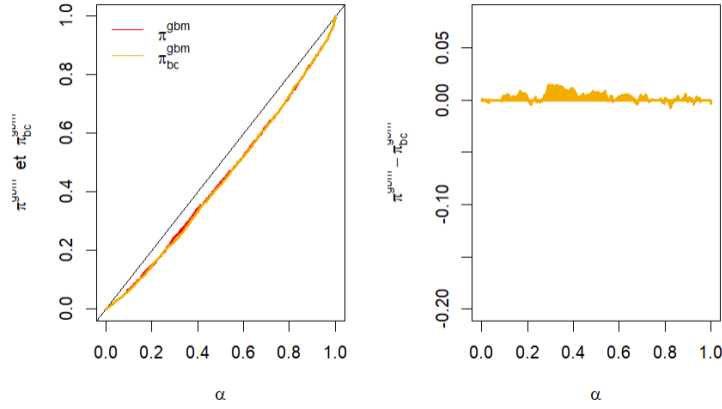


FIGURE 5.9 – Evolution de  $CC[\mu, \hat{\pi}; \alpha]$  cas du Boosting

**Définition :** La courbe de Lorenz associée à la prédiction  $\hat{\pi}(X)$  est donnée par :

$$\alpha \mapsto LC[\hat{\pi}(X); \alpha] = CC[\hat{\pi}(X), \hat{\pi}(X); \alpha] = \frac{\mathbb{E}[\hat{\pi}(X)I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]]}{\mathbb{E}[\hat{\pi}(X)]}.$$

En considérant la définition, on observe que la courbe de Lorenz est strictement liée à la dispersion (ou variabilité).

Les fonctions  $\alpha \mapsto LC[\hat{\pi}(X); \alpha]$  peuvent être visualisées sur nos modèles. Sur les figures (5.10) nous pouvons visualiser à gauche les deux modèles GLM et à droite l'aire entre nos deux différentes courbes. Sur les figures (5.11) et (5.12) nous visualisons respectivement les deux modèles GAM et boosting.

Les courbes de concentration et de Lorenz coïncident pour la vraie prime  $\mu(X)$ . Le résultat suivant nous montre qu'il en est de même pour les prédicteurs autocalibrés.

**Propriété 5.3.1.** *Si  $\hat{\pi}$  est autocalibré, alors pour tout niveau de probabilité  $\alpha$ ,*

$$CC[\mu(X), \hat{\pi}(X); \alpha] = LC[\hat{\pi}(X); \alpha].$$

**Preuve 5.3.1.** *Par hypothèse,  $\hat{\pi}$  est autocalibré, alors  $\hat{\pi}(X) = \mathbb{E}[Y|\hat{\pi}(X)]$ . On a :*

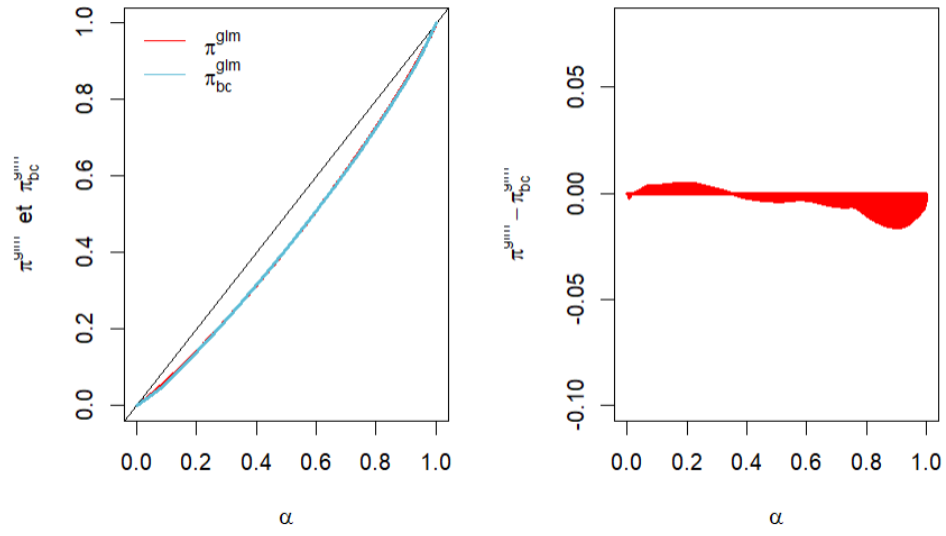


FIGURE 5.10 – Evolution de  $LC[\hat{\pi}; \alpha]$  cas du GLM

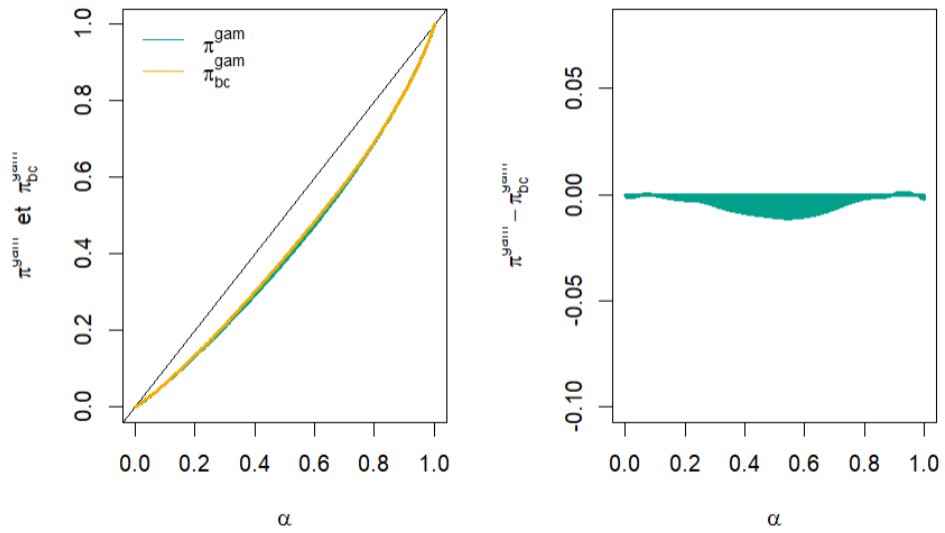


FIGURE 5.11 – Evolution de  $LC[\hat{\pi}; \alpha]$  cas du GAM

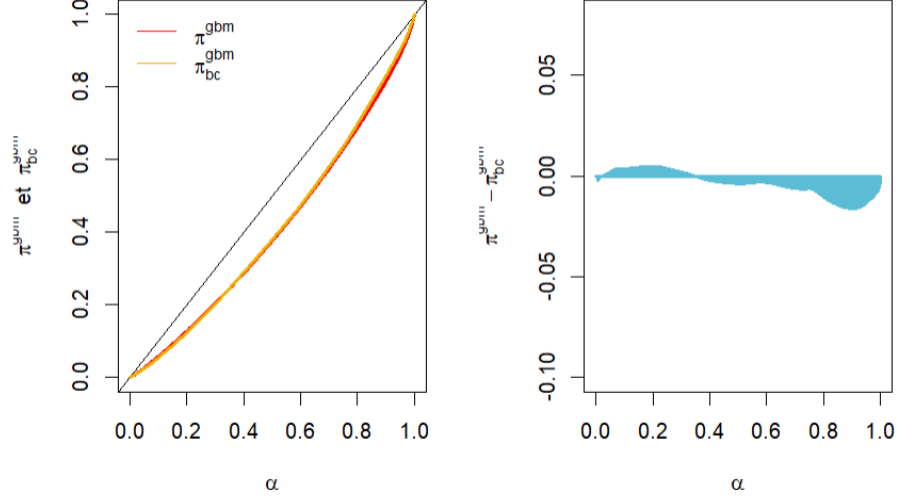


FIGURE 5.12 – Evolution de  $LC[\hat{\pi}; \alpha]$  cas du Boosting

$$\begin{aligned}
LC[\hat{\pi}(X); \alpha] &= \frac{\mathbb{E}[\hat{\pi}(X)I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]]}{\mathbb{E}[\hat{\pi}(X)]} \\
&= \frac{\mathbb{E}[\mathbb{E}[Y|\hat{\pi}(X)]I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]]}{\mathbb{E}[\hat{\pi}(X)]} \quad \text{car } \hat{\pi} \text{ autocalibre} \\
&= \frac{\mathbb{E}[\mathbb{E}[YI[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]|\hat{\pi}(X)]]}{\mathbb{E}[\hat{\pi}(X)]} \\
&= \frac{\mathbb{E}[YI[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]]}{\mathbb{E}[\hat{\pi}(X)]} \quad \text{car } \mathbb{E}[A] = \mathbb{E}[\mathbb{E}(A|B)] \\
&= \frac{\mathbb{E}[\mathbb{E}[YI[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]|X]]}{\mathbb{E}[\hat{\pi}(X)]} \\
&= \frac{\mathbb{E}[\mathbb{E}[Y|X]I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]]}{\mathbb{E}[\hat{\pi}(X)]} \quad \text{or } \mu(X) = \mathbb{E}[Y|X] \\
&= \frac{\mathbb{E}[\mu(X)I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]]}{\mathbb{E}[\hat{\pi}(X)]} \quad \text{or } \mathbb{E}[\hat{\pi}(X)] = \mathbb{E}[\mu(X)] \\
&= \frac{\mathbb{E}[\mu(X)I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\alpha)]]}{\mathbb{E}[\mu(X)]} \\
&= CC[\mu(X), \hat{\pi}(X); \alpha].
\end{aligned}$$

D'où le résultat.

La métrique ABC introduite par Denuit et al. (2019) et définie comme étant l'aire entre les deux courbes CC et LC, joue le rôle d'indicateur de performance d'un prédicteur. D'après la propriété précédente, on constate que pour tout prédicteur autocalibré,  $ABC[\hat{\pi}(X)] = 0$ . On conclut que la

métrique ABC est très utile pour la nécessité de mise en oeuvre d'un autocalibrage.

On constate que les courbes de concentration et de lorentz de la prime autocalibré du modèle boosting sont plus proches en comparaison des primes autocalibrés obtenues par les modèle GLM et GAM. Le tableau suivant nous présente la valeur de différentes aires ABC pour chaque modèle simple et pour chaque modèle autocalibré.

$ABC[\widehat{\pi}(X)]$	GLM	GLM <sub>BC</sub>	GAM	GAM <sub>BC</sub>	GBM	GBM <sub>BC</sub>
Valeur	0.03988	0.03861	0.04437	0.02989	0.03438	0.02801

TABLE 5.2 – Aire ABC des différentes prédictions

Nous venons de donner le critère de mesure de la performance d'un prédicteur, la question qui nous vient en tête est celle de savoir, que se passerait-il si on a deux prédicteurs et qu'on doit en choisir ? Dans la section suivante, nous essayerons de donner un critère de comparaison des deux prédicteurs en impliquant la dominance de Bregman.

## 5.4 Comparaison des performances de deux prédicteurs

**Definition :** Soit  $\widehat{\pi}_1$  et  $\widehat{\pi}_2$  deux prédicteurs autocalibrés. Alors  $\widehat{\pi}_2$  est plus performant que  $\widehat{\pi}_1$  en termes de dominance de Bregman si et seulement si,  $\widehat{\pi}_2$  domine  $\widehat{\pi}_1$  dans l'ordre convexes

$$\iff LC[\widehat{\pi}_2(X); \alpha] \leq LC[\widehat{\pi}_1(X); \alpha] \text{ pour tous les niveaux de probabilit } \alpha.$$

$$\iff CC[\mu(X), \widehat{\pi}_2(X); \alpha] \leq CC[\mu(X), \widehat{\pi}_1(X); \alpha] \text{ pour tous les niveaux de probabilit } \alpha.$$

L'ordre des courbes de concentration signifie que la part du revenu total des primes correspondant aux sous-portefeuilles rassemblent le  $\alpha\%$  des polices avec les plus petites primes est plus petit pour  $\widehat{\pi}_1$  par rapport à  $\widehat{\pi}_2$ .

Il est possible de faire face à une situation où deux prédicteurs n'ont pas un ordre constant. En effet, comme réléver par Denuit et al. (2019) dans la partie 4 de leurs documents, deux prédicteurs pourraient bien être incomparables du fait que leurs courbes de concentration ou de Lorenz respectives se croisent et conduit par exemple au phénomène suivant : un prédicteur est meilleur pour les risques faibles et pire pour les risques élevés. Dans un tel cas, on ne peut pas baser la comparaison

sur l'intégrale des courbes de concentration. Nous introduisons alors la courbe de concentration intégrée définie comme suit :

$$\begin{aligned}
ICC[\mu(X), \hat{\pi}(X); \alpha] &= \int_0^\alpha CC[\mu(X), \hat{\pi}(X); \xi] d\xi \\
&= \int_0^\alpha \left[ \frac{\mathbb{E}[\mu(X) I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\xi)]]}{\mathbb{E}[\mu(X)]} \right] d\xi \\
&= \frac{1}{\mathbb{E}[\mu(X)]} \int_0^\alpha \mathbb{E}[\mu(X) I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\xi)]] d\xi \\
&= \frac{1}{\mathbb{E}[\mu(X)]} \int_0^\alpha \mathbb{E}[\mu(X) I[F_{\hat{\pi}}(\hat{\pi}(X)) \leq \xi]] d\xi \\
&= \frac{\mathbb{E}[\mu(X)(\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+]}{\mathbb{E}[\mu(X)]} \\
&= \frac{Cov[\mu(X), (\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+]}{\mathbb{E}[\mu(X)]} + \mathbb{E}[(\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+].
\end{aligned}$$

Nous pouvons remplacer  $\mu(X)$  avec  $Y$  dans cette définition de la courbe de concentration intégrée. En effet,  $\mu(X) = \mathbb{E}[Y|X]$  et

$$\mathbb{E}[Y(\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+] = \mathbb{E}[\mathbb{E}[Y(\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+ | X]] = \mathbb{E}[\mu(X)(\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+].$$

Nous pouvons donc l'utiliser pour mesurer la performance de  $\hat{\pi}(X)$  dans la prédiction de la prime pure inconnue  $\mu(X)$ .

Grâce à la courbe de concentration intégrale, on peut mesurer la performance de  $\hat{\pi}(X)$  à prédire  $\mu(X)$ . Comme  $ICC$  dépend de  $Cov[\mu(X), (\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+]$  pour  $\alpha$  fixé, alors plus  $F_{\hat{\pi}}(\hat{\pi}(X))$  est petit par rapport de  $\alpha$ , plus  $\hat{\pi}(X)$  est petit.

Pour ce qui est de la courbe intégrale de Lorenz, elle est définie par :

$$\begin{aligned}
ILC[\hat{\pi}(X); \alpha] &= \int_0^\alpha CC[\hat{\pi}(X), \hat{\pi}(X); \xi] d\xi \\
&= \frac{1}{\mathbb{E}[\hat{\pi}(X)]} \int_0^\alpha \mathbb{E}[\hat{\pi}(X) I[\hat{\pi}(X) \leq F_{\hat{\pi}}^{-1}(\xi)]] d\xi \\
&= \frac{\mathbb{E}[\hat{\pi}(X)(\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+]}{\mathbb{E}[\hat{\pi}(X)]} \\
&= \frac{Cov[\hat{\pi}(X), (\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+]}{\mathbb{E}[\hat{\pi}(X)]} + \mathbb{E}[(\alpha - F_{\hat{\pi}}(\hat{\pi}(X)))_+].
\end{aligned}$$

---

## Conclusion

Les modèles d'apprentissage avancés sont capables de produire des scores qui ont une meilleure corrélation avec la réponse, ainsi qu'avec la prime réelle, par rapport aux GLM classiques. Cela est dû à la liberté supplémentaire obtenue en permettant aux scores de dépendre de manière flexible sur les caractéristiques disponibles, et non seulement linéairement. Mais la rupture de l'équilibre global est le prix à payer pour cette corrélation plus élevée. Étant donné qu'aucune contrainte sur la réplication de l'équilibre total ou global observé n'est imposée, les outils de machine learning sont également en mesure d'augmenter considérablement le biais global.

Pour éviter que cela n'arrive, la version corrigée de l'équilibre de tout prédicteur peut être obtenue par GLM local, en reconnaissant la réponse  $Y$ . La fonction de lien canonique est adoptée de façon à ce que les estimations du maximum de vraisemblance reproduisent les totaux observés. Dans cette approche, le score du modèle est suffisant pour calculer  $\hat{\pi}_{BC}$  de sorte que le modèle d'apprentissage est essentiellement utilisé pour évaluer la proximité entre les individus, avant d'effectuer une moyenne locale ; cette proximité est évaluée à l'aide de la valeur réelle  $\hat{\pi}$  produite par le modèle d'apprentissage. De plus, l'étape d'autocalibration supplémentaire proposée permet de contrer la surestimation potentielle en faisant la moyenne locale du prédicteur initial.

La relation établie avec la fonction de perte constante et la domination de Bregman paraît particulièrement instructive. Le législateur a mis en place un cadre juridique obligeant les assureurs à fournir des explications de manière compréhensive d'une part sur les raisons pour lesquelles il n'accorde pas de couverture et de l'autre sur le montant onéreux de la prime demandée. Un travail complémentaire consisterait à étudier la transparence des modèles à l'aide de diagrammes de dépendance partielle. Néanmoins, avec un minimum de biais, l'assureur peut expliquer que le montant de la prime facturée a été calculé afin de compenser les risques d'assurance au sein de sous-portefeuilles significatifs, en maintenant un équilibre global. Le mécanisme de correction d'équilibre proposé met en œuvre la même idée, à l'aide d'outils de machine learning pour définir des quartiers significatifs où les primes collectées doivent correspondre aux sinistres à indemniser. Cela est conforme au principe fondamental de mutualité qui est au cœur de l'assurance. Pour ce qui est des primes pures, la cohérence des fonctions de perte de Bregman permet aux actuaires sur le plan pratique de rassurer et de garantir aux assurés, au législateur et aux organisations de consommateurs que leur approche de l'algorithme de machine learning recherche la bonne prime.

---

## Bibliographie

Charpentier, A. (2014). Computational Actuarial Science with R. Chapman and Hall / CRC.

Denuit, M., Hainaut, D., Trufin, J. (2019). Effective Statistical Learning Methods for Actuaries III. Springer Nature Switzerland AG.

Denuit, M., Snajder, D., Trufin, J. (2019). Model selection based on Lorenz and Concentration curves, Gini indices and convex order. Insurance : Mathematics and Economics 89, 128-139.

Efron, B. (1986). How biased is the apparent error rate of a prediction rule? Journal of the American Statistical Association 81, 461-470.

Ehm, W., Gneiting, T., Jordan, A. Kruger, F. (2016). Of quantiles and expectiles : Consistent scoring functions, Choquet representations and forecast rankings. Journal of the Royal Statistical Society - Series B 78, 505-562.

Kruger, F., Ziegel, J.F. (2020). Generic conditions for forecast dominance. Journal of Business Economic Statistics, in press.

Loader, C. (1999). Local Regression and Likelihood. Springer, New York.

Savage, L. J. (1971). Elicitation of personal probabilities and expectations. Journal of the American Statistical Association 66, 783-810.

Shaked, M., Shanthikumar, J. G. (2007). Stochastic Orders. Springer, New York.

Shaked, M., Sordo, M. A., Suarez-Llorens, A. (2012). Global dependence stochastic orders. Methodology and Computing in Applied Probability 14, 617-648.

Wright, R. (1987). Expectation dependence of random variables with an application in portfolio theory. Theory and Decision 22, 179-202.

---

Yitzhaki, S., Schechtman, E. (2013). The Gini Methodology. A Primer Statistical Methodology. Springer.

Code R : <https://github.com/freakonometrics/autocalibration>.

## ANNEXE

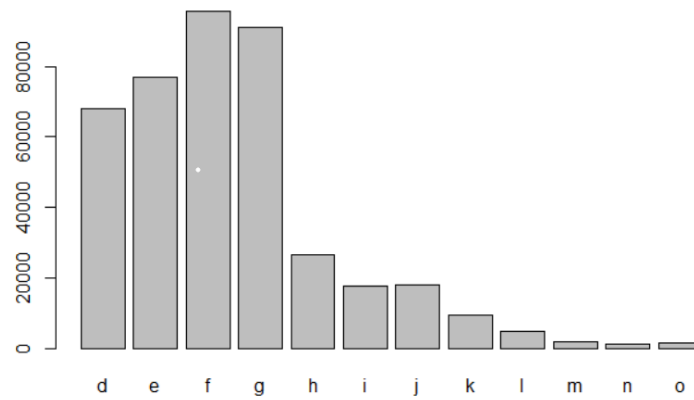


FIGURE 5.13 – Diagramme de la variable puissance

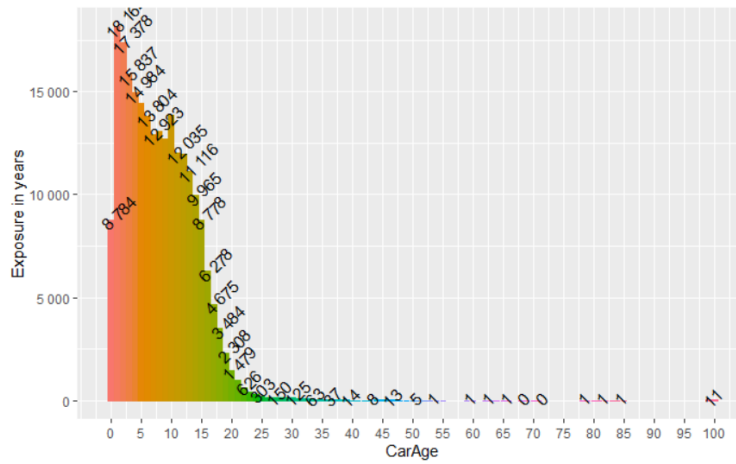


FIGURE 5.14 – Proportion d'âge de véhicule par durée dans le portefeuille

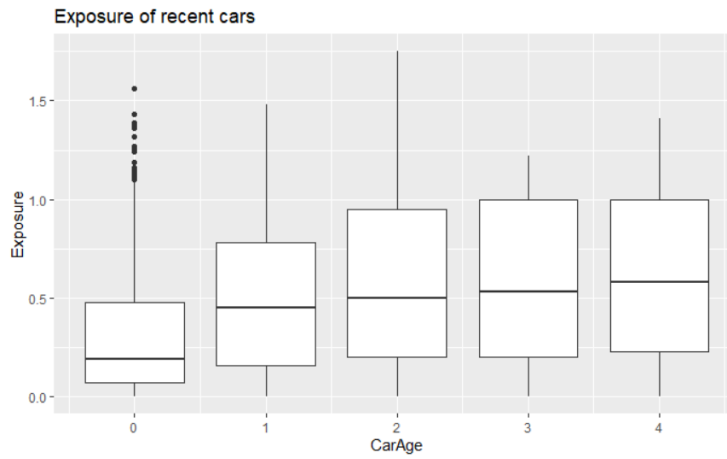


FIGURE 5.15 – Proportion d'âge de véhicule par durée dans le portefeuille

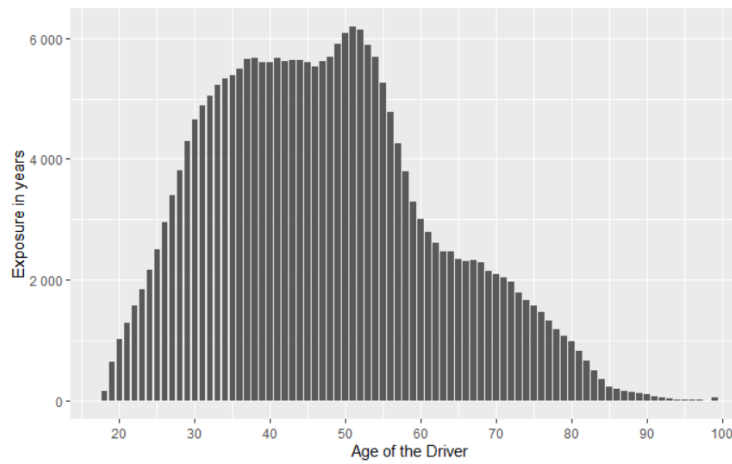


FIGURE 5.16 – Proportion d'âge d'assuré par durée dans le portefeuille

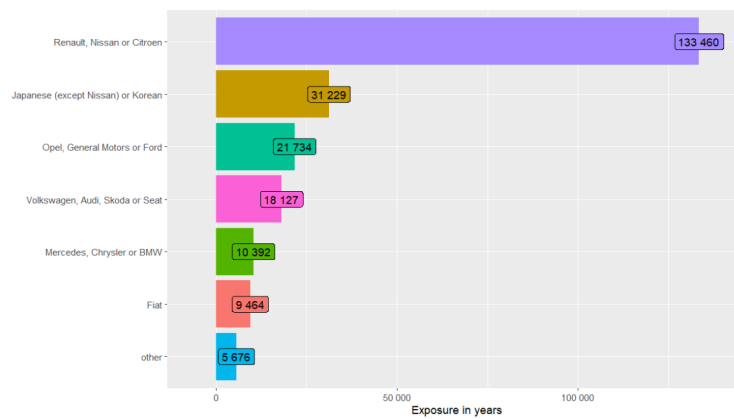


FIGURE 5.17 – Proportion marque de voiture par durée dans le portefeuille

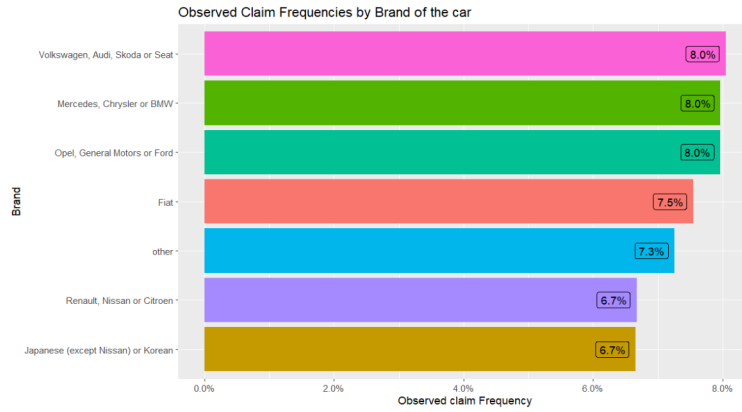


FIGURE 5.18 – Proportion marque de voiture par sinistres dans le portefeuille



FIGURE 5.19 – Proportion du type par durée dans le portefeuille

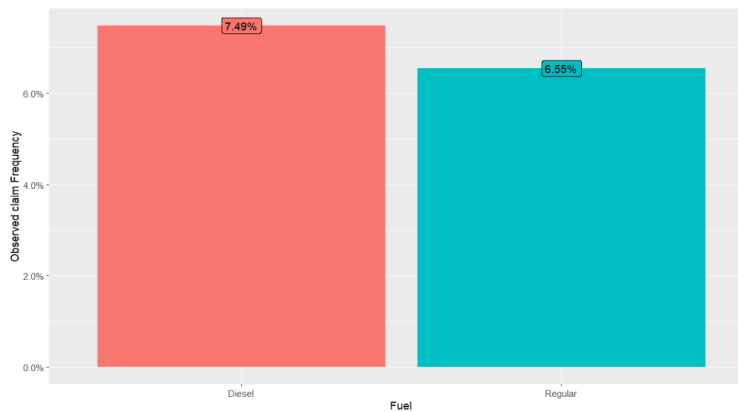


FIGURE 5.20 – Proportion du type de carburant par durée dans le portefeuille

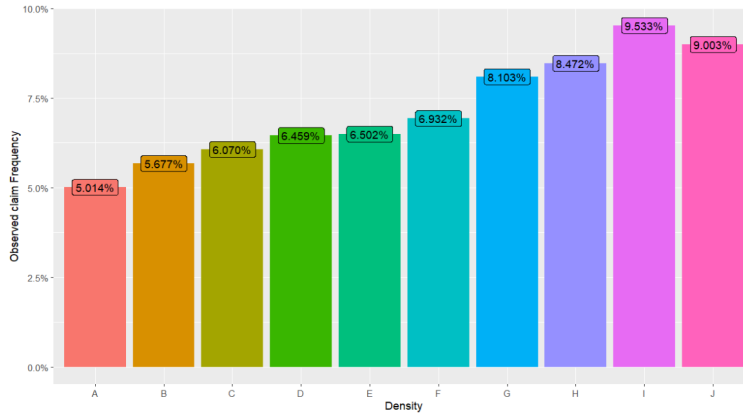


FIGURE 5.21 – densité par sinistre

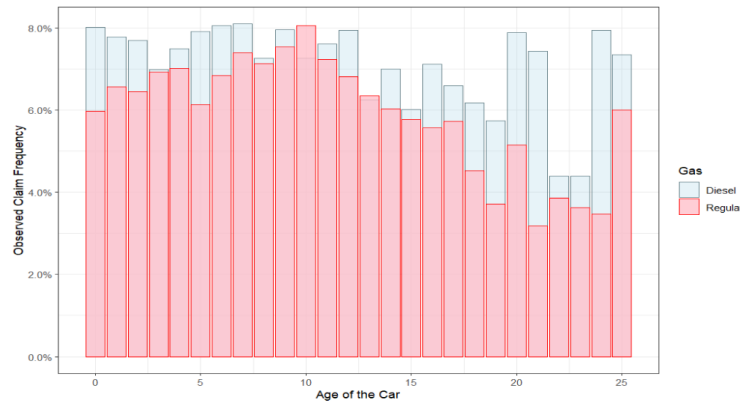


FIGURE 5.22 – Interaction entre l'âge du conducteur et le type de carburant

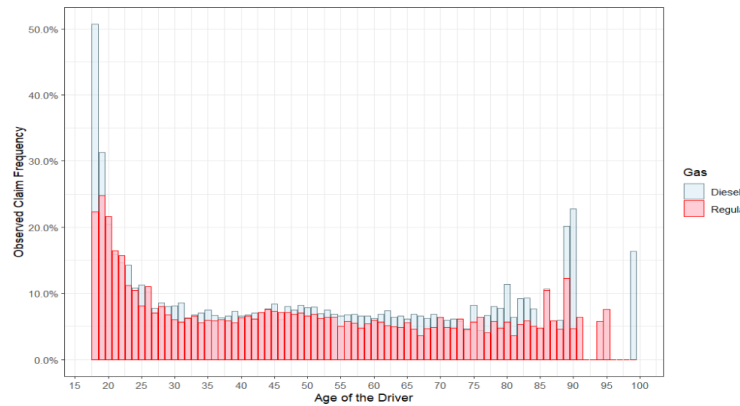


FIGURE 5.23 – Interaction entre l'âge de la voiture et le type de carburant

```

Call:
glm(Formula = ClaimNb ~ Power + CarAge + DriverAge + Density +
     Brand + Gas + Region + offset(log(Exposure)), family = poisson,
     data = train_LM)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.5907 -0.3442 -0.2672 -0.1498  6.5382

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.949e+00  6.487e-02 -30.042 < 2e-16 ***
Powere       6.087e-02  3.134e-02   1.942  0.052105 .
Powerf      6.882e-02  3.064e-02   2.246  0.024722 *
Powerg      4.513e-02  3.046e-02   1.481  0.138642
Powerh      1.070e-01  4.331e-02   2.471  0.013479 *
Poweri      1.678e-01  4.853e-02   3.458  0.000544 ***
Powerj      1.552e-01  4.929e-02   3.149  0.001640 **
Powerk      2.232e-01  6.388e-02   3.493  0.000477 ***
Powerl      5.400e-02  9.622e-02   0.561  0.574635
Powerm      2.240e-01  1.323e-01   1.694  0.090259 .
Powern      3.616e-01  1.475e-01   2.452  0.014205 *
Powero      1.256e-01  1.587e-01   0.792  0.428541
CarAge     -1.227e-02  1.873e-03  -6.552  5.66e-11 ***
DriverAge  -1.006e-02  6.404e-04 -15.716 < 2e-16 ***
Density     1.350e-05  2.177e-06   6.198  5.71e-10 ***
BrandJapanese (except Nissan) or Korean -3.510e-01  5.022e-02 -6.990 2.76e-12 ***
BrandMercedes, Chrysler or BMW        -9.701e-03  5.896e-02 -0.165  0.869296
BrandOpel, General Motors or Ford     -6.428e-03  4.976e-02 -0.129  0.897224
BrandOther                             -1.157e-01  6.980e-02 -1.658  0.097262 .
BrandRenault, Nissan or Citroen       -1.007e-01  4.321e-02 -2.330  0.019783 *
BrandVolkswagen, Audi, Skoda or Seat  -1.782e-02  5.097e-02 -0.350  0.726593
GasRegular                               -9.467e-02  1.931e-02 -4.904 9.40e-07 ***
RegionBasse-Normandie                 -1.267e-01  6.319e-02 -2.005  0.045015 *
RegionBretagne                        -1.393e-01  4.341e-02 -3.209  0.001332 **
RegionCentre                          -1.663e-01  3.753e-02 -4.429 9.48e-06 ***
RegionHaute-Normandie                 -1.244e-01  8.381e-02 -1.484  0.137834
RegionIle-de-France                   4.612e-02  4.494e-02   1.026  0.304769
RegionLimousin                        9.274e-02  8.787e-02   1.055  0.291243
RegionNord-Pas-de-Calais              2.552e-02  5.047e-02   0.506  0.613161
RegionPays-de-la-Loire                -5.857e-02  4.459e-02 -1.313  0.189061
RegionPoitou-Charentes                -5.504e-02  5.263e-02 -1.046  0.295578
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 84025 on 328690 degrees of freedom
Residual deviance: 83351 on 328660 degrees of freedom
AIC: 108257

Number of Fisher Scoring iterations: 6

```

FIGURE 5.24 – Summary du modèle GLM

```

Family: poisson
Link function: log

Formula:
ClaimNb ~ Power + bs(CarAge) + bs(DriverAge) + bs(Density) +
Brand + Gas + Region + offset(log(Exposure))

Parametric coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.660325  0.077721 -21.363 < 2e-16 ***
Powere       0.060345  0.031356   1.925  0.05429 .
Powerf      0.068044  0.030794   2.210  0.02713 *
Powerg      0.067968  0.030542   2.225  0.02606 *
Powerh      0.151921  0.043370   3.503  0.00046 ***
Poweri      0.202053  0.048603   4.157  3.22e-05 ***
Powerj      0.193857  0.049399   3.924  8.70e-05 ***
Powerk      0.287481  0.064009   4.491  7.08e-06 ***
Powerl      0.123260  0.096309   1.280  0.20060
Powerm      0.280505  0.132357   2.119  0.03407 *
Powern      0.390662  0.147537   2.648  0.00810 **
Powero      0.207731  0.158714   1.309  0.19059
bs(CarAge)1  0.125573  0.107764   1.165  0.24391
bs(CarAge)2  0.028195  0.122241   0.231  0.81758
bs(CarAge)3 -0.667189  0.163982  -4.069 4.73e-05 ***
bs(DriverAge)1 -2.635005  0.144758 -18.203 < 2e-16 ***
bs(DriverAge)2  0.691276  0.154166   4.484  7.33e-06 ***
bs(DriverAge)3 -2.506916  0.227764 -11.007 < 2e-16 ***
bs(Density)1  1.453336  0.106048  13.705 < 2e-16 ***
bs(Density)2 -0.698335  0.221900  -3.147  0.00165 **
bs(Density)3  0.423325  0.063458   6.671  2.54e-11 ***
BrandJapanese (except Nissan) or Korean -0.307159  0.051014 -6.021 1.73e-09 ***
BrandMercedes, Chrysler or BMW        -0.006369  0.059058 -0.108  0.91412
BrandOpel, General Motors or Ford     -0.004640  0.049765 -0.093  0.92571
BrandOther                             -0.093929  0.069790 -1.346  0.17834
BrandRenault, Nissan or Citroen       -0.091396  0.043261 -2.113  0.03463 *
BrandVolkswagen, Audi, Skoda or Seat  -0.018608  0.051007 -0.365  0.71526
GasRegular                               -0.137915  0.019422  -7.101 1.24e-12 ***
RegionBasse-Normandie                 -0.098884  0.063276  -1.563  0.11811
RegionBretagne                        -0.086249  0.043642  -1.976  0.04812 *
RegionCentre                          -0.100606  0.038035  -2.645  0.00817 **
RegionHaute-Normandie                 -0.112186  0.083830  -1.338  0.18082
RegionIle-de-France                   -0.007754  0.045528 -0.170  0.86477
RegionLimousin                        0.210731  0.088249   2.388  0.01694 *
RegionNord-Pas-de-Calais              0.001386  0.050509   0.027  0.97811
RegionPays-de-la-Loire                -0.031936  0.044622  -0.716  0.47417
RegionPoitou-Charentes                0.007369  0.052924   0.139  0.88927
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.00715  Deviance explained = 1.44%
UBRE = -0.74781  Scale est. = 1          n = 328691

```

FIGURE 5.25 – Summary du modèle GAM

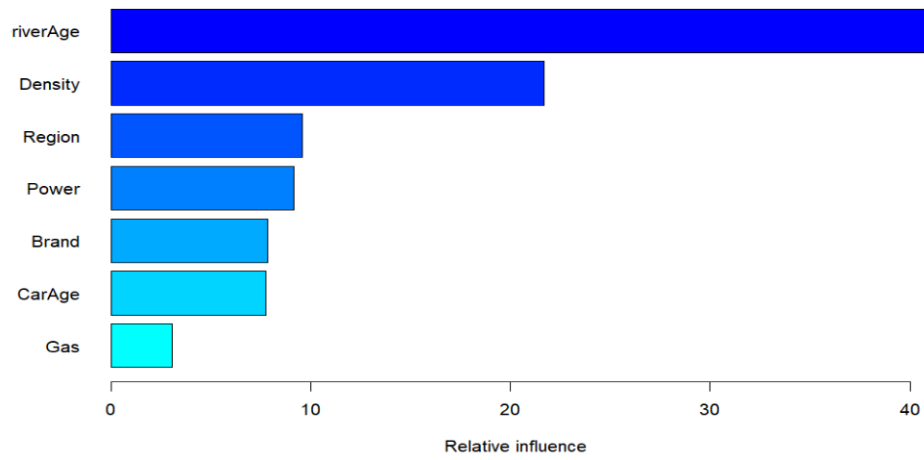


FIGURE 5.26 – Summary du modèle GBM

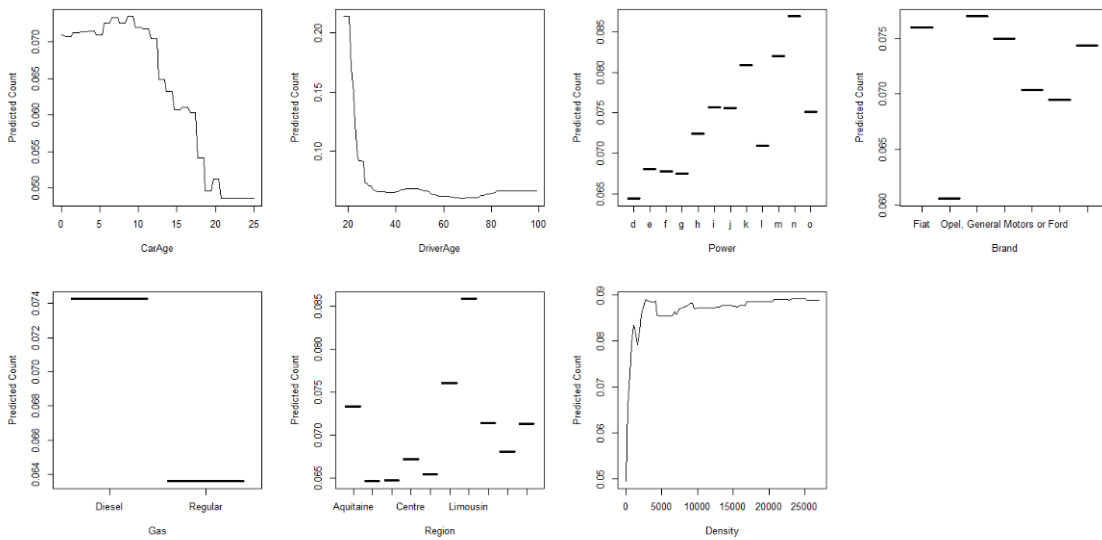


FIGURE 5.27 – Dépendance partielle des variables dans le modèle GBM

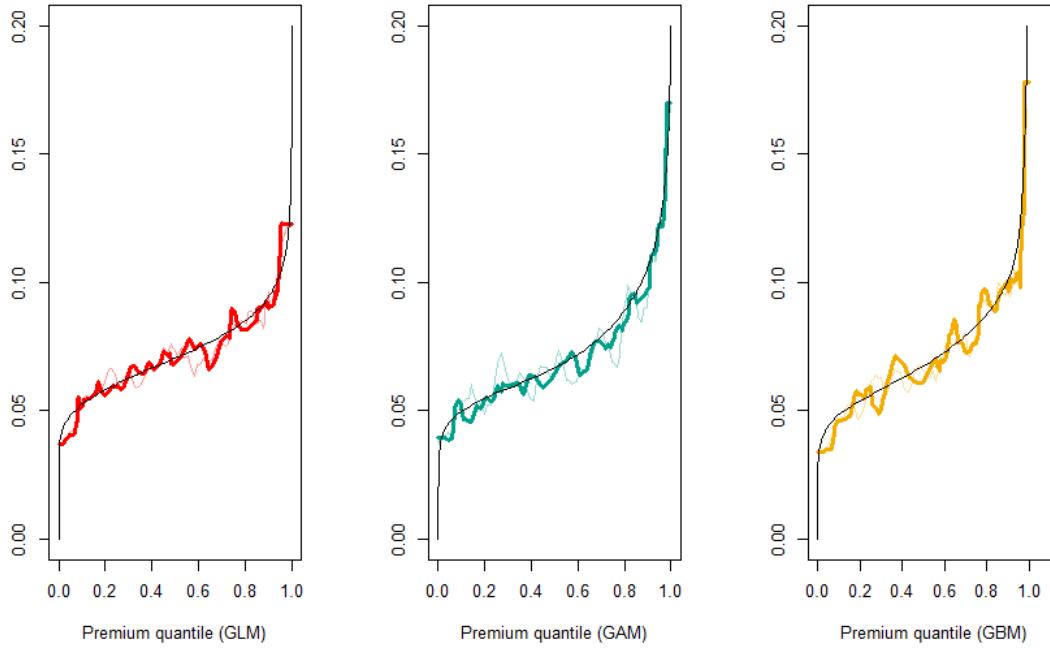


FIGURE 5.28 – Visualisation de  $u \mapsto \mathbb{E}[Y|\hat{\pi}_{BC}(X) = F_{\hat{\pi}_{BC}}^{-1}(u)]$  avec la ligne droite correspondant à  $\mu = \hat{\pi}_{BC}$  où La fine courbe colorée représente la prime prédite  $\hat{\pi}$

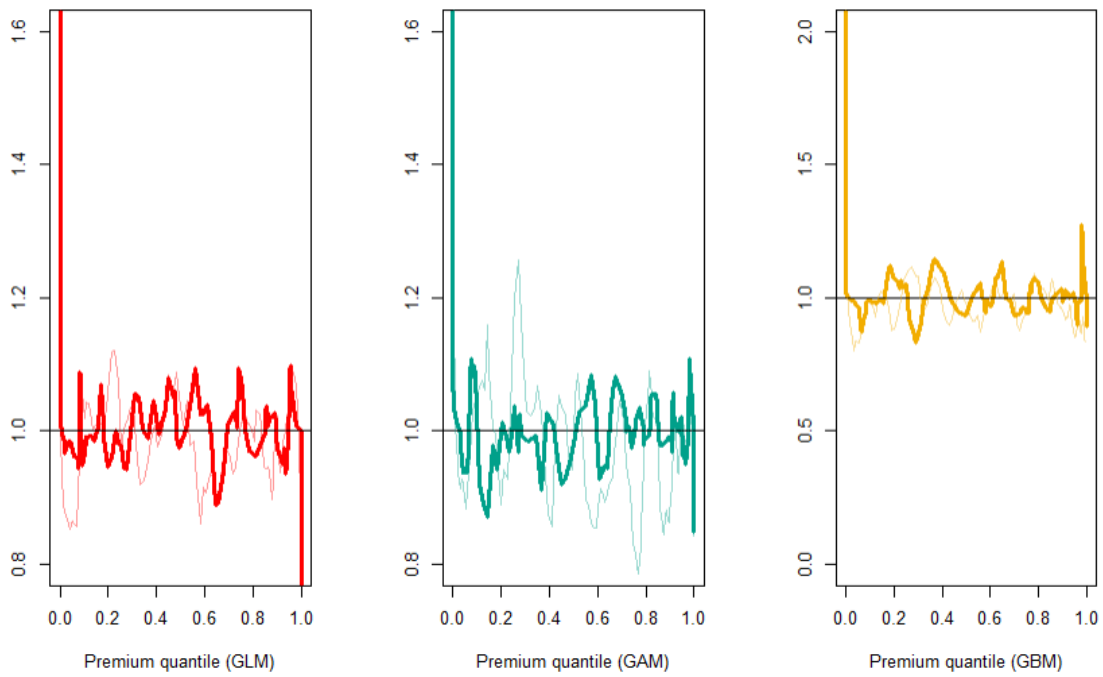


FIGURE 5.29 – Visualisation de  $u \mapsto \mathbb{E}[Y|\hat{\pi}_{BC}(X) = F_{\hat{\pi}_{BC}}^{-1}(u)]/F_{\hat{\pi}_{BC}}^{-1}(u)$  avec la ligne droite correspondant à  $\mu = \hat{\pi}_{BC}$  où La fine courbe colorée représente la prime prédite  $\hat{\pi}$

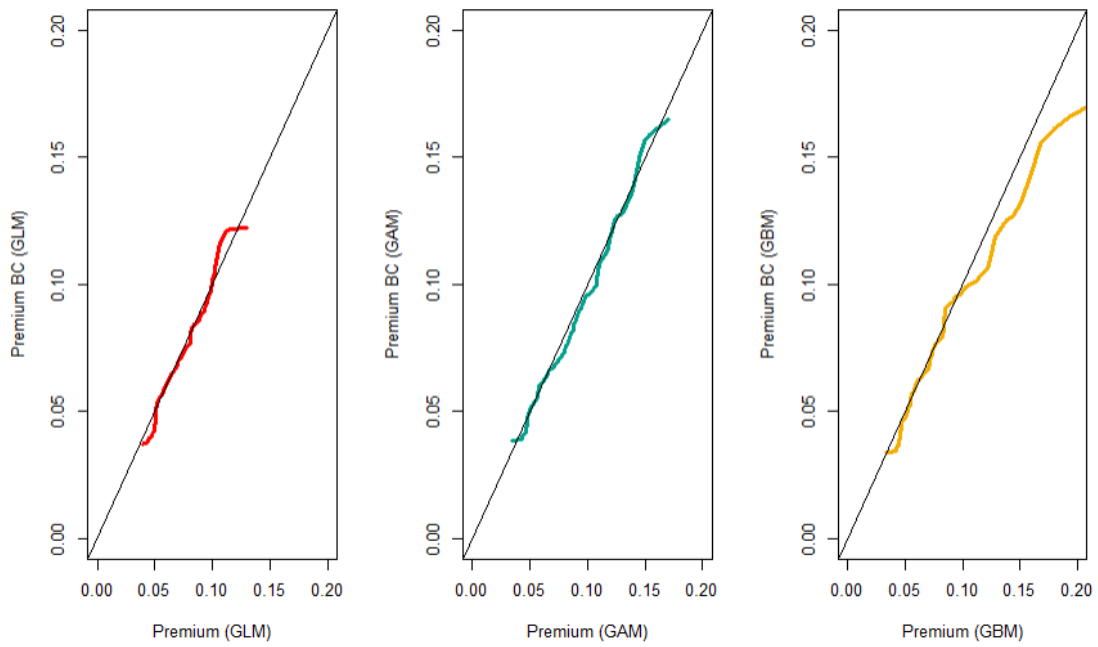


FIGURE 5.30 – QQ plot de  $\hat{\pi}_{BC}$  contre  $\hat{\pi}$

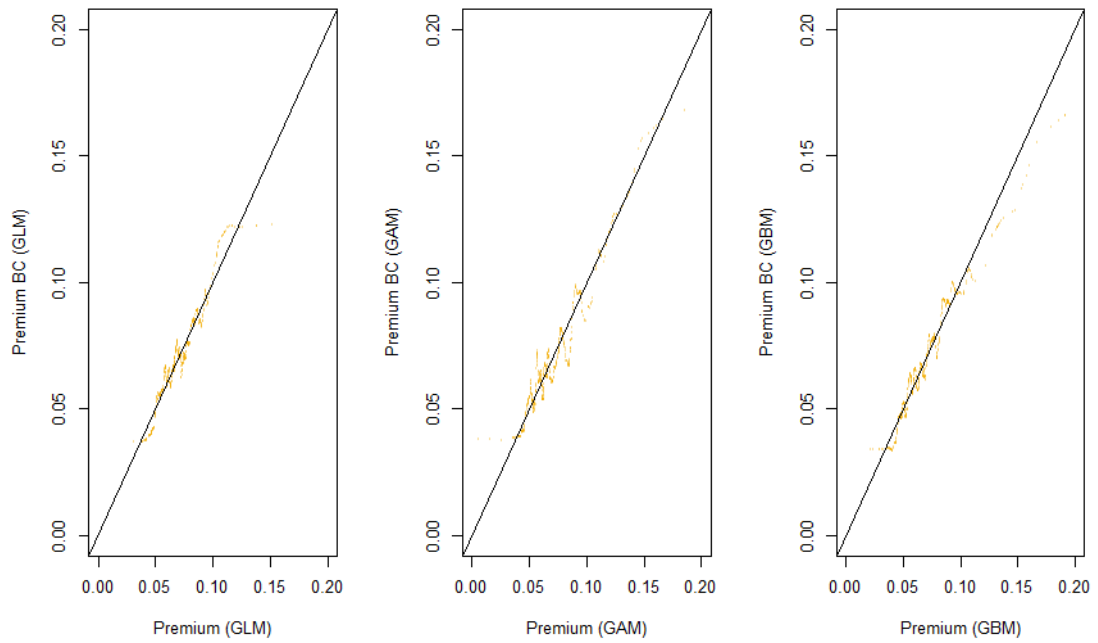


FIGURE 5.31 – Nuage de points  $\{\hat{\pi}(x_i), \hat{\pi}_{bc}(x_i)\}$

```
---
title: "Autocalibration des primes par Machine learning"
author: "Carnot"
date: '2022-08-14'
output: html_document
---
```

```
# Chargement des différents package
```

```
```{r, include=TRUE, message=FALSE, warning=FALSE}
options(encoding = 'UTF-8')
```

```
if (!require("xts")) install.packages("xts")
if (!require("CASdatasets")) install.packages("CASdatasets", repos =
"http://cas.uqam.ca/pub/", type="source")
if (!require("caret")) install.packages("caret")
if (!require("ggplot2")) install.packages("ggplot2")
if (!require("mgcv")) install.packages("mgcv")
if (!require("dplyr")) install.packages("dplyr")
if (!require("gridExtra")) install.packages("gridExtra")
if (!require("visreg")) install.packages("visreg")
if (!require("MASS")) install.packages("MASS")
if (!require("plotrix")) install.packages("plotrix")
if (!require("rgeos")) install.packages("rgeos", type="source")
if (!require("rgdal")) install.packages("rgdal", type="source")
if (!require("xtable")) install.packages("xtable")
if (!require("mapproj")) install.packages("mapproj")
if (!require("scales")) install.packages("scales")
if (!require("broom")) install.packages("broom")
if (!require("stringi")) install.packages("stringi")
```

```
require("CASdatasets")
require("ggplot2")
require("mgcv")
require("caret")
require("gridExtra")
require("dplyr")
require("visreg")
require("MASS")
require("plotrix")
require("rgdal")
require("rgeos")
require("xtable")
require("mapproj")
require("scales")
require("broom")
require("stringi")
```
```

```
```{r set-options, echo=FALSE, cache=FALSE}
options(width = 75)
```
```

```
# Chargement des données
```

```
```{r, tidy=TRUE}
```

```

## Loading the dataset
require("CASdatasets")
data("freMTPLfreq")

dataset1 = freMTPLfreq %>% filter(Exposure<=1 & Exposure >= 0 & CarAge<=25
)

saveRDS(dataset1, file = "../dataset.Rds")

...

Vérifier si le jeu de données a été parfaitement chargé.

```{r, tidy=TRUE}
head(dataset1)
...
```{r, tidy=TRUE}
summary(dataset1)
```

# Chapitre 1 : Statistique descriptive du portefeuille

## PolicyID
La variable *PolicyID* correspond à un identifiant unique de la politique.
Nous pouvons vérifier que chaque politique n'apparaît qu'une seule fois
dans l'ensemble de données.

```{r, tidy=TRUE}
length(unique(dataset1$PolicyID)) == nrow(dataset1)
```

## Variable Exposure en mois

On peut afficher le nombre de mois d'exposition sur un tableau. La fonction
*cut* permet de catégoriser (bin) une variable numérique. Nous pouvons
spécifier où " casser " et donner un nom à chaque niveau en utilisant
l'argument *labels*. La sortie est une variable factorielle.

```{r, fig.align='center', tidy=TRUE}
table(cut(dataset1$Exposure, breaks = seq(from = 0, to = 1,by = 1/12),
labels = 1:12))
```

En utilisant la fonction *prop.table*, il est possible de représenter ces
informations en termes relatifs d'afficher le nombre de mois d'exposition
sur un tableau.

```{r, fig.align='center', tidy=TRUE}
Exposures_prop = prop.table(table(cut(dataset1$Exposure, breaks = seq(from
= 0, to = 1,by = 1/12), labels = 1:12)))
round(Exposures_prop, 4 )
```

Utilisation de barplot pour affichage
```{r,fig.align='center', cache=TRUE, tidy=TRUE, dpi=500}

```

```

ggplot(dataset1)+geom_bar(aes(x=cut(Exposure,
                                breaks = seq(from = 0, to = 1,by =
1/12), labels = 1:12))) +
  scale_x_discrete(name = "Number of months") +
  scale_y_continuous(name = "Number of Policies", label = label_number()) +
  ggtitle("Exposure in months")
...

```

si nous voulons aussi montrer le pourcentage sur les barres

```

```{r,fig.align='center', cache=TRUE, tidy=TRUE, dpi=500}
ggplot(dataset1, aes(x=cut(Exposure, breaks = seq(from = 0, to = 1,by =
1/12), labels = 1:12),
                    label = scales::percent(prop.table(stat(count)),
accuracy = 0.1)))+
  geom_bar() +
  geom_text(stat = 'count',
            vjust = -0.5,
            size = 3) +
  scale_x_discrete(name = "Number of months") +
  scale_y_continuous(name = "Number of Policies",
                    label = label_number()) +
  ggtitle("Exposure in months")
...

```

```

```{r}
ggplot(dataset1, aes(x=Exposure))+geom_histogram(binwidth =1/12,
fill='gray', color='white') +
  scale_x_continuous(name = "Exposure in fraction of years",
breaks=seq(0,1,1/12), labels = round(seq(0,1,1/12), 3))+
  scale_y_continuous(name = 'Number of Polices', labels = label_number()) +
  ggtitle("Exposure in fraction of years")
...

```

## Nombre de sinistre : ClaimNb

```

```{r,fig.align='center', cache=TRUE, tidy=TRUE, dpi=500}
ggplot(dataset1, aes(x=ClaimNb))+
  geom_bar()+
  geom_label(stat='count',
            aes(label = percent(prop.table(after_stat(count)),
accuracy = 0.01)),
            vjust = 0.5)+
  scale_x_continuous(name = "Number of Claims")+
  scale_y_continuous(name = "Number of Polices",
                    labels = label_number())+
  ggtitle("Proportion of policies by number of claims")
...

```

Nous pouvons calculer la fréquence moyenne des sinistres dans ce portefeuille, en tenant compte des différentes expositions.

```

```{r, results='asis',eval=FALSE, tidy=TRUE}
scales::percent(sum(dataset1$ClaimNb) / sum(dataset1$Exposure),
accuracy = 0.01)
...

```

```
## Variable puissance: Power
```

Les niveaux de la variable sont ordonnés de manière catégorique.

```
```{r, echo=TRUE, cache=TRUE, tidy=TRUE}
levels(dataset1$Power)
table(dataset1$Power)
```

Affichage
```{r, tidy=TRUE, results='hide', message=FALSE}
plot(dataset1$Power)
```
```

le tableau ci-dessus ne mesure PAS l'exposition pour chaque niveau de la variable \*Power\*. Nous allons donner l'exposition à chaque niveau de la variable.

```
```{r, tidy=TRUE}
Power.summary = dataset1 %>% group_by(Power) %>% summarise(totalExposure =
sum(Exposure),
                                                                    Number.Observations
= length(Exposure))
```

Affichage
```{r, fig.align='center', dpi=500, tidy=TRUE}
ggplot(Power.summary, aes(x=Power,
                           y=totalExposure,
                           fill=Power,
                           color=Power,
                           label=scales::number(totalExposure))) +
  geom_bar(stat="identity")+
  geom_text(stat='identity', vjust=-0.5)+
  scale_y_continuous(name = "Exposure in years", labels = scales::number)+
  theme(legend.position = 'none')
```
```

Examinons maintenant la fréquence observée des demandes dans chaque niveau.

```
```{r, tidy=TRUE, results='hide', message=FALSE}
Power.summary = dataset1 %>% group_by(Power) %>% summarise(totalExposure =
sum(Exposure),
                                                                    Number.Observations
= length(Exposure),
                                                                    Number.Claims =
sum(ClaimNb),
                                                                    Obs.Claim.Frequency
= sum(ClaimNb)/sum(Exposure))
Power.summary
```
```

Nous pouvons calculer le rapport avec la fréquence des sinistres du portefeuille.

```
```{r, fig.align='center', dpi=500, tidy=TRUE}
portfolio.cf = sum(dataset1$ClaimNb) / sum(dataset1$Exposure)
# Can also be written as
portfolio.cf = with(dataset1, sum(ClaimNb)/sum(Exposure))

ggplot(Power.summary, aes(x=Power,
                           y=Obs.Claim.Frequency,
                           color=Power,
```

```

        fill=Power,
        label = percent(Obs.Claim.Frequency, accuracy =
0.01))) +
  geom_bar(stat="identity") +
  geom_hline(aes(yintercept=portfolio.cf), color="black", size= 2,
linetype="dashed", alpha = 0.33) +
  geom_label(vjust=-0.21, fill="white", alpha = 0.25)+
  annotate(geom="text",
          x='m', y=portfolio.cf,
          vjust=-0.5,
          label=paste("Average claim freq. of portfolio: ",
percent(portfolio.cf, accuracy = 0.01)),
          color="black")+
  scale_y_continuous(name = "Observed Claim Frequency", labels =
percent_format(accuracy = 0.01))+
  theme(legend.position = 'none')
...

## Variable Âge du véhicule: CarAge
```{r, fig.align='center', cache=TRUE, tidy=TRUE, dpi=500}
ggplot(dataset1,
        aes(x=CarAge)) + geom_bar() +
  scale_x_continuous(name = "Age of the Car", breaks=seq(0,100,5))+
  scale_y_continuous(name = "Number of Polices", labels=label_number())
...

Prise en compte de l'exposure
```{r, tidy=TRUE, results='hide'}
CarAge.summary = dataset1 %>% group_by(CarAge) %>% summarise(totalExposure
= sum(Exposure),
                                                    Number.Observations
= length(Exposure))
CarAge.summary
...
```{r, fig.align='center', dpi=500, tidy=TRUE}
ggplot(CarAge.summary, aes(x=CarAge,
                           y=totalExposure,
                           fill=factor(CarAge),
                           color=factor(CarAge),
                           label=label_number(accuracy=1)(totalExposure))) +
  geom_bar(stat="identity")+
  geom_text(stat='identity', color="black", hjust=0.25, vjust=0.5,
angle=45, check_overlap=TRUE)+
  scale_x_continuous(breaks = seq(0,100,5))+
  scale_y_continuous(name = "Exposure in years", labels = label_number()+
  theme(legend.position = 'none')
...

Regardons l'exposition pour les véhicules récents, en utilisant un boxplot
par exemple.
```{r, fig.align='center', dpi=500, tidy=TRUE}
ggplot(dataset1[dataset1$CarAge<5,], aes(x=CarAge, y=Exposure,
group=CarAge)) +
  geom_boxplot() +
  ggtitle("Exposure of recent cars")
...

Calculons maintenant aussi la fréquence des sinistres par âge de la voiture
```{r, tidy=TRUE, results='hide'}
CarAge.summary = dataset1 %>% group_by(CarAge) %>% summarise(totalExposure
= sum(Exposure),

```

```

= length(Exposure),
= sum(ClaimNb),
= sum(ClaimNb)/sum(Exposure)
CarAge.summary
```
```{r, fig.align='center', dpi=500, tidy=TRUE}
portfolio.cf = with(dataset1, sum(ClaimNb)/ sum(Exposure))

ggplot(CarAge.summary, aes(x=CarAge,
                           y=Obs.Claim.Freq,
                           label = percent(Obs.Claim.Freq, accuracy =
0.01))) +
  geom_bar(stat="identity") +
  geom_hline(yintercept=portfolio.cf, color="black", size= 2,
linetype="dashed", alpha = 0.33) +
  annotate(geom="text",
          x=20, y=portfolio.cf,
          vjust=-0.5,
          label=paste("Average claim freq. of portfolio: ",
percent(portfolio.cf, accuracy = 0.01)),
          color="black")+
  scale_x_continuous(name = "Age of the Car", breaks=seq(0,100,5))+
  scale_y_continuous(name = "Observed Claim Frequency", labels =
percent_format(accuracy = 0.01))+
  theme(legend.position = 'none')
```
```
## Variable de l'âge du conducteur: DriverAge
```{r, fig.align='center', cache=TRUE, tidy=TRUE, dpi=500, results='hide'}
DriverAge.summary = dataset1 %>% group_by(DriverAge) %>%
summarise(totalExposure = sum(Exposure),
= length(Exposure),
= sum(ClaimNb),
= sum(ClaimNb)/sum(Exposure))
head(DriverAge.summary,9)
```
les expositions en fonction de l'âge du conducteur
```{r, fig.align='center', dpi=500, tidy=TRUE}
ggplot(DriverAge.summary, aes(x=DriverAge, y=totalExposure)) +
  geom_bar(stat='identity', width=0.8) +
  scale_y_continuous(name="Exposure in years", labels = label_number()+
  scale_x_continuous(name="Age of the Driver", breaks = seq(10,150,10))
```
Fréquence de sinistres par Âge
```{r, fig.align="center", dpi=500, tidy=TRUE}
ggplot(DriverAge.summary, aes(x=DriverAge, y=Obs.Claim.Freq)) +
  geom_line()+geom_point()+
  scale_y_continuous(name="Observed Claim Frequency", labels = percent,
breaks = seq(0,0.50,0.05))+
  scale_x_continuous(name="Age of the Driver", breaks = seq(10,150,10))
```
## Variable de la marque de véhicule: Brand
Number.Observations
Number.Claims
Obs.Claim.Freq

```

```

```{r, tidy=TRUE}
levels(dataset1$Brand)
```

```{r, echo=TRUE, tidy=TRUE, results='hide'}
Brand.summary = dataset1 %>% group_by(Brand) %>% summarise(totalExposure =
sum(Exposure),
Number.Observations =
=length(Exposure),
Number.Claims =
sum(ClaimNb),
Obs.Claim.Freq =
sum(ClaimNb)/sum(Exposure))

Brand.summary
```

```{r, results='asis', warning=FALSE, echo=FALSE}
require(xtable)
print(xtable(Brand.summary, digits=c(0,1,2,0,0,5)), type = "html",
include.rownames=FALSE)
```

```{r, fig.align='center', dpi=500, tidy=TRUE}
ggplot(Brand.summary, aes(x=reorder(Brand,totalExposure),
y=totalExposure,
fill=Brand,
label = label_number()(totalExposure))) +
  geom_bar(stat='identity') +
  coord_flip()+guides(fill='none')+
  scale_x_discrete(name = "") +
  scale_y_continuous("Exposure in years", labels = label_number(), expand =
c(0.10, 0))+
  geom_label()
```

Frequence des sinistres en fonction de l'âge du conducteur
```{r, fig.align='center', dpi=500, tidy=TRUE}
ggplot(Brand.summary, aes(x=reorder(Brand,Obs.Claim.Freq),
y=Obs.Claim.Freq,
fill=Brand,
label = percent(Obs.Claim.Freq, accuracy=0.1))) +
  geom_bar(stat='identity') +
  geom_label(hjust=+1.2)+
  coord_flip()+guides(fill='none')+ ggtitle("Observed Claim Frequencies by
Brand of the car")+
  scale_x_discrete(name = "Brand") +
  scale_y_continuous("Observed claim Frequency", labels = percent)
```

## variable type de carburant: Gas
```{r, echo=TRUE, cache=TRUE, tidy=TRUE}
levels(dataset1$Gas)
```

```{r, echo=TRUE, fig.align='center', dpi=500, tidy=TRUE}
Gas.summary = dataset1 %>% group_by(Gas) %>% summarise(totalExposure =
sum(Exposure),
Number.Observations =
length(Exposure),

```

```

sum(ClaimNb),
sum(ClaimNb)/sum(Exposure))
ggplot(Gas.summary, aes(x=Gas, y=totalExposure, fill=Gas, label =
number(totalExposure))) +
  geom_bar(stat="identity") +
  geom_label()+
  guides(fill='none')+
  scale_x_discrete(name = "Fuel")+
  scale_y_continuous(name='Total Exposure (in years)', labels = number)

plot(dataset1$Gas)
table(dataset1$Gas)
```
Number.Claims =
Obs.Claim.Freq =

```{r, fig.align='center', dpi=500, tidy=TRUE}
ggplot(Gas.summary,
  aes(x=Gas, y=Obs.Claim.Freq,
  fill=Gas,
  label = percent(Obs.Claim.Freq))) +
  geom_bar(stat="identity") +
  geom_label()+ guides(fill='none')+
  scale_x_discrete(name = "Fuel") +
  scale_y_continuous("Observed claim Frequency", labels = percent)
```

## variable region des assurées: Region
```{r, echo=TRUE, cache=TRUE, tidy=TRUE}
levels(dataset1$Region)
```

Quelles sont les expositions dans chaque région ? Quelles sont les
fréquences de sinistres observées ?

```{r, tidy=TRUE, results='hide'}
Region.summary = dataset1 %>% group_by(Region) %>% summarize(totalExposure
= sum(Exposure),
= length(Exposure),
sum(ClaimNb),
sum(ClaimNb)/sum(Exposure))
Region.summary
```

Number.Observations
Number.Claims =
Obs.Claim.Freq =

```{r, results='asis', echo=FALSE}
print(xtable(Region.summary, digits=c(0,1,2,0,0,5)), type="html",
include.rownames=FALSE)
```

```{r, message=FALSE, fig.align='center', warning=FALSE, dpi=500,
tidy=TRUE}
area <- rgdal::readOGR("shapefiles/FRA_adm1.shp", use_iconv = TRUE,
encoding = "UTF-8") # From http://www.diva-gis.org/gData

# Note that the tidy function will remove the data.
area_tidy = tidy(area) # package broom

#Plot an "empty" map

```

```

ggplot(area_tidy, aes(x = long, y = lat, group = group)) +
  geom_polygon(color = "black", size = 0.1, fill = "lightgrey") +
  coord_equal() + theme_minimal()
...

```{r, message=FALSE, fig.align='center', warning=FALSE, dpi=500,
tidy=TRUE}
#First we re-include the data (that disappeared with the tidy function)
area$id <- row.names(area)
area_tidy2 <- area_tidy %>% full_join(area@data, by="id")

# Because of accents ...
area_tidy2$NAME_1 = stri_trans_general(str = area_tidy2$NAME_1,
                                     id = "Latin-ASCII")

# Which computed data do we want ?
data_to_add = Region.summary[,c("Region", "totalExposure",
"Obs.Claim.Freq")]
# Merge it
area_tidy2 <- area_tidy2 %>% full_join(data_to_add, by = c("NAME_1" =
"Region"))

# Very important: Do not forget to sort by "order" variable.
# area_tidy2 = area_tidy2[order(area_tidy2$order),]
# Easier with dplyr:
area_tidy2 = area_tidy2 %>% arrange(order)
...

## variable Densité: Density

```{r, fig.align='center', dpi=500, tidy=TRUE}
summary(dataset1$Density)
ggplot(dataset1, aes(Density)) + geom_histogram(bins=200)
...

```{r, tidy=TRUE, results='hide'}
Density.summary = dataset1 %>% group_by(Density) %>%
  summarise(totalExposure = sum(Exposure),
            Number.Observations = length(Exposure),
            Number.Claims = sum(ClaimNb),
            Obs.Claim.Freq = sum(ClaimNb)/
sum(Exposure))
head(Density.summary)
...

```{r, results='asis', echo=FALSE, fig.align='center'}
print(xtable(head(Density.summary), digits=c(0,1,2,0,0,5)), type="html",
include.rownames=FALSE)
...

Nous pouvons tracer les fréquences de réclamation observées...
```{r, fig.align='center', dpi=500, tidy=TRUE}
ggplot(Density.summary, aes(x=Density, y=Obs.Claim.Freq)) + geom_point()
...

catégorisation de la variable

```{r, tidy=TRUE}

```

```

dataset1$DensityCAT = cut(dataset1$Density, breaks =
quantile(dataset1$Density, probs = seq(from = 0, to = 1, by=0.1)),
include.lowest = TRUE)
table(dataset1$DensityCAT)
levels(dataset1$DensityCAT) <- LETTERS[1:10]
```

```{r, tidy=TRUE, results='hide'}
Density.summary = dataset1 %>% group_by(DensityCAT) %>%
  summarise(totalExposure = sum(Exposure),
            Number.Observations = length(Exposure),
            Number.Claims = sum(ClaimNb),
            Obs.Claim.Freq = sum(ClaimNb)/sum(Exposure))
Density.summary
```

```{r, results='asis', echo=FALSE}
print(xtable(Density.summary, digits=c(0,1,2,0,0,5)), type="html",
include.rownames=FALSE)
```

```{r, fig.align='center', dpi=500, tidy=TRUE}
ggplot(Density.summary,
  aes(x=DensityCAT,
      y=Obs.Claim.Freq,
      fill=DensityCAT,
      label = percent(Obs.Claim.Freq))) +
  geom_bar(stat="identity") +
  geom_label()+ guides(fill='none')+
  scale_x_discrete(name = "Density") +
  scale_y_continuous("Observed claim Frequency", labels = percent)
```

## Effet combiné

### Fuel and CarAge

```{r}
CarAge.Fuel.summary = dataset1 %>% group_by(CarAge, Gas) %>%
  summarise(totalExposure = sum(Exposure),
            Number.Observations = length(Exposure),
            Number.Claims = sum(ClaimNb),
            Obs.Claim.Freq = sum(ClaimNb)/
sum(Exposure))
ggplot(CarAge.Fuel.summary, aes(x=CarAge,
  y=Obs.Claim.Freq)) +
  facet_wrap(~Gas)+
  geom_bar(stat="identity") +
  scale_x_continuous(name = "Age of the Car", breaks=seq(0,100,5))+
  scale_y_continuous(name = "Observed Claim Frequency", labels =
percent_format(accuracy = 0.01))+
  theme(legend.position = 'none')
```

### Gas et CarAge

```{r, tidy=TRUE}
DriverAge.Fuel.summary = dataset1 %>% group_by(CarAge, Gas) %>%

```

```

        summarize(Obs.Claim.Freq = sum(ClaimNb)/
sum(Exposure))

ggplot(data=CarAge.Fuel.summary, aes(x=CarAge,
                                     y=Obs.Claim.Freq,
                                     fill=Gas,
                                     color=Gas,
                                     alpha=Gas)) +
  geom_bar(stat="identity", position = "identity") +
  scale_x_continuous(name = "Age of the Car", breaks = seq(0,100,5))+
  scale_y_continuous(name = "Observed Claim Frequency", labels =
label_percent()+
  scale_colour_manual(values=c("lightblue4", "red")) +
  scale_fill_manual(values=c("lightblue", "pink")) +
  scale_alpha_manual(values=c(.3, .8))+
  theme_bw()
...

### Fuel et DriverAge

```{r, tidy=TRUE}
DriverAge.Fuel.summary = dataset1 %>% group_by(DriverAge, Gas) %>%
  summarize(Obs.Claim.Freq = sum(ClaimNb)/
sum(Exposure))

ggplot(data=DriverAge.Fuel.summary, aes(x=DriverAge,
                                     y=Obs.Claim.Freq,
                                     fill=Gas,
                                     color=Gas,
                                     alpha=Gas)) +
  geom_bar(stat="identity", position = "identity") +
  scale_x_continuous(name = "Age of the Driver", breaks = seq(0,100,5))+
  scale_y_continuous(name = "Observed Claim Frequency", labels =
label_percent()+
  scale_colour_manual(values=c("lightblue4", "red")) +
  scale_fill_manual(values=c("lightblue", "pink")) +
  scale_alpha_manual(values=c(.3, .8))+
  theme_bw()
...

## correlation entre variables explicatives quantitatives

```{r, tidy=TRUE}
correl = round(cor(dataset1[,c("ClaimNb", "Exposure", "CarAge",
"DriverAge", "Density")]), 2)

correl
...

```{r, tidy=TRUE}
dataset1[,c("ClaimNb", "PolicyID", "Exposure", "CarAge", "DriverAge",
"Density")]
...

# correlation entre variable explicative qualitative

```

```

```{r, tidy=TRUE}
library(DescTools)
matable1=table(dataset1$Region,dataset1$Brand)
CramerV(matable1) #0.190095
matable2=table(dataset1$Region,dataset1$Gas)
CramerV(matable2) #0.1034957
matable3=table(dataset1$Region,dataset1$Power)
CramerV(matable3) #0.04831579
matable4=table(dataset1$Brand, dataset1$Power)
CramerV(matable4) #0.1926802
matable5=table(dataset1$Brand,dataset1$Gas)
CramerV(matable5) #0.1005371
matable6=table(dataset1$Power,dataset1$Gas)
CramerV(matable6) #0.3327472

tab=table(dataset1$Region,dataset1$Region)
CramerV(tab)

...

# Application aux données: Chapitre 5

## palette de couleurs pour les graphes

```{r, include=TRUE, message=FALSE, warning=FALSE}
options(encoding = 'UTF-8')
library("wesanderson")
library(scales)
clrpal = wes_palette("Zissou1", 22, type = "continuous")
clrpallow = scales::alpha(clrpal,.4)
bleurouge = clrpal[c(1,22)]
bleurougepal = clrpallow[c(1,22)]
clrpal4 = colr = wes_palette("Darjeeling1")[c(1,2,3,5)]
clrpal6 = wes_palette("Zissou1", 6, type = "continuous")

...

```{r, include=TRUE, message=FALSE, warning=FALSE}
library(h2o)
#h2o.shutdown(prompt = TRUE)
h2o.init(nthreads = -1)

h2o.no_progress()
```

## separation de la base de données

```{r, tidy=TRUE}
library(CASdatasets)
data(freMTPLfreq)
str(freMTPLfreq)

dataset2 = freMTPLfreq %>% filter(Exposure<=1 & Exposure >= 0 & CarAge<=25)

LM = model.matrix(ClaimNb ~
Power+CarAge+DriverAge+Brand+Density+Region+Gas+Exposure, data=dataset2)
LM = cbind(ClaimNb=dataset2$ClaimNb,LM)

```

```

LM = as.data.frame(LM)
...

```{r, tidy=TRUE}
TAUX = c(.8,.2)
idx_train = sample(1:nrow(LM),size = round(nrow(LM)*TAUX[1]))
idx_reste = (1:nrow(LM))[-idx_train]
idx_correct = sample(idx_reste,size = round(nrow(LM)*TAUX[2]))
train_LM = dataset2[idx_train,]
valid_LM = dataset2[idx_correct,]
valid_LM_1 = valid_LM
valid_LM_1$Exposure =1

...

```{r, tidy=TRUE}

summary(train_LM)
...

## Models

### GLM

#### GLM sans rien
```{r, tidy=TRUE}
library(splines)
reg0 = glm(ClaimNb~1+offset(log(Exposure)),family=poisson,data=train_LM)

summary(reg0)
...

#### GLM
```{r, tidy=TRUE}
reg_glm =
glm(ClaimNb~Power+CarAge+DriverAge+Density+Brand+Gas+Region+offset(log(Exposure)),family=
summary(reg_glm)
...

### GAM

```{r, tidy=TRUE}
library(mgcv)

reg_gam = gam(ClaimNb~Power+bs(CarAge)+bs(DriverAge)+bs(Density)
+Brand+Gas+Region+offset(log(Exposure)),family=poisson,data=train_LM)

summary(reg_gam)
...

### GBM

```

Nous pouvons commencer par charger les paquets.

```
```{r, warning=FALSE, message=FALSE, tidy=TRUE}
require(devtools)
install_github("gbm-developers/gbm3")

library(gbm3)
library(caret)
library(mgcv)
library(reshape2)
library(ggplot2)
library(dplyr)
```

#### Cross Validation pour la recherche des paramètres optimaux
il a pris près 36 heures

```{r}

trainControl <- trainControl(method = "cv",
                             number = 10,
                             returnResamp="all", ### use "all" to return all
cross-validated metrics
                             search = "grid")

tuneGrid1 <- expand.grid(
  n.trees = c(1000,1500,5000, 10000),
  interaction.depth = c(5,8,3,1),
  shrinkage = c(0.1,0.01, 0.001),
  min_num_obs_in_node=1000
)

tuneGrid1
```

```{r}

gbm_op1 = list()

for (it in 1:nrow(tuneGrid1)) {
  set.seed(89)
  gbm_op1[[it]] = gbmt(ClaimNb ~ offset(log(Exposure)) + CarAge + DriverAge +
    Power + Brand + Gas + Region + Density,
    data = train_LM,
    distribution = gbm_dist("Poisson"),
    train_params = training_params(num_trees = tuneGrid1[it,]$n.trees,
    shrinkage = tuneGrid1[it,]
    $shrinkage, #Default is 0.001
    interaction_depth = tuneGrid1[it,]
    $interaction.depth, #the max number of non - terminal nodes in each tree
    min_num_obs_in_node =tuneGrid1[it,]
    $min_num_obs_in_node,
```

```

        bag_fraction = 0.5,
        num_train = 1*nrow(train_LM)),
    is_verbose = TRUE,
    cv_folds = 5,
    keep_gbm_data = TRUE,
    par_details = gbmParallel(num_threads = 3)) #Parallel computing
}

gbm_op1
...

#### calcule de l'erreur pour déterminer le meilleur modèle

```{r}
err = c()
for (it in 1:36) {
  err[it] = gbm_op1[[it]]$cv_error[gbmt_performance(gbm_op1[[it]],
method="cv")]
}

err
...

Effectuons maintenant la validation croisée avec 1500 arbres. paramètre
optimal

```{r, cache=TRUE, tidy=FALSE, fig.align='center', dpi=500}
set.seed(89)
m0_gbm = gbmt(ClaimNb ~ offset(log(Exposure)) + CarAge + DriverAge + Power
+ Brand + Gas + Region + Density,
  data = train_LM,
  distribution = gbm_dist("Poisson"),
  train_params = training_params(num_trees = 1500,
                                shrinkage = 0.01, #Default is 0.001
                                interaction_depth = 5, #the max
                                number of non - terminal nodes in each tree
                                min_num_obs_in_node = 1000,
                                bag_fraction = 0.5,
                                num_train = 1*nrow(train_LM)),
  is_verbose = TRUE,
  cv_folds = 5,
  keep_gbm_data = TRUE,
  par_details = gbmParallel(num_threads = 3)) #Parallel computing
m0_gbm
```

Affichage des performance
```{r, tidy=TRUE, fig.align='center', dpi=500}
plot(gbmt_performance(m0_gbm), method="cv")
```

```{r, tidy=TRUE, fig.align='center', dpi=500}

```

```

best_iter = gbmt_performance(m0_gbm, "cv")
best_iter
```

Observation des variables importantes
```{r, tidy=TRUE, fig.align='center', dpi=500}
summary(m0_gbm)
```

Dépendance partielle

```{r, fig.height=5, fig.width = 10, tidy=TRUE, fig.align='center',
dpi=500}
par(mfrow=c(2,4))
for (i in 1:7){
  plot(m0_gbm, var_index = i, num_trees = best_iter, type="response")
}
```

## Construction des prédictions de primes simples

```{r, warning=FALSE, tidy=TRUE}
vx = seq(0,.2,length=351)
library(locfit)
valid_prime_glm = predict(reg_glm,newdata=valid_LM_1,type="response")
valid_prime_gam = predict(reg_gam,newdata=valid_LM_1,type="response")
valid_prime_bst = predict(m0_gbm,newdata = valid_LM_1,n.trees=best_iter,
type="response")

q_glm = q_gam = q_bst = rep(NA,length(vx))
for(i in 1:length(vx)){
  q_glm[i] = mean(valid_prime_glm<=vx[i])
  q_gam[i] = mean(valid_prime_gam<=vx[i])
  q_bst[i] = mean(valid_prime_bst<=vx[i])
}

```

Quartile des predicteurs

```{r, tidy=TRUE}
desc = function(y){
  m=c(mean(y),quantile(y,c(.1,.9)))
  m=round(m,4)
  names(m) = c("mean","10%","90%")
  m
}
desc(valid_prime_glm)
desc(valid_prime_gam)
desc(valid_prime_bst)

```

```{r, tidy=TRUE}
mean(predict(reg0,newdata=valid_LM_1,type="response"))

```

```

...

Representation
```{r, tidy=TRUE}
par(mfrow=c(1,3))
hist(valid_prime_glm, xlim=c(0,.25),col=clrpal4[1],border="white",
      main="",xlab="Premium (GLM)",ylab="",ylim=c(0,40000))

hist(valid_prime_gam, xlim=c(0,.25),col=clrpal4[2],border="white",
      main="",xlab="Premium (GAM)",ylab="",ylim=c(0,40000))

hist(valid_prime_bst, xlim=c(0,.25),col=clrpal4[3],border="white",
      main="",xlab="Premium (GLM)",ylab="",ylim=c(0,40000))

...

##### Recherche du paramètres optimales de la bande de voisinage Alpha

```{r, tidy=TRUE}

fit_glmt = locfit.raw(x=valid_prime_glm,
                     y=valid_LM[, "ClaimNb"]/valid_LM[, "Exposure"],
                     weights=valid_LM[, "Exposure"],
                     kern="rect",deg=0,alpha=c(0, 0.01, 0.02, 0.03,
0.05, 0.06))

plot(fit_glmt, m=300, get.data=T)

res=residuals(fit_glmt)

res

...

##### Choix porter sur Alpha = 0.05

```{r, tidy=TRUE}
fit_loc_glm = locfit.raw(x=valid_prime_glm,
                        y=valid_LM[, "ClaimNb"]/valid_LM[, "Exposure"],
                        weights=valid_LM[, "Exposure"],
                        kern="rect",deg=0,alpha=.05)
fit_loc_gam = locfit.raw(x=valid_prime_gam,
                        y=valid_LM[, "ClaimNb"]/valid_LM[, "Exposure"],
                        weights=valid_LM[, "Exposure"],
                        kern="rect",deg=0,alpha=.05)

fit_loc_bst = locfit.raw(x=valid_prime_bst,
                        y=valid_LM[, "ClaimNb"]/valid_LM[, "Exposure"],
                        weights=valid_LM[, "Exposure"],
                        kern="rect",deg=0,alpha=.05)

...

Figure 5.1

```{r, tidy=TRUE}
par(mfrow=c(1,3))
plot(fit_loc_glm,lwd=3,col=clrpal4[1],xlim=c(0,.2),xlab="Premium
(GLM)",ylab="",ylim=c(0,.2))

```

```

abline(a=0,b=1,lwd=.4)
plot(fit_loc_gam,lwd=3,col=clrpal4[2],xlim=c(0,.2),xlab="Premium
(GAM)",ylab="",ylim=c(0,.2))
abline(a=0,b=1,lwd=.4)

plot(fit_loc_bst,lwd=3,col=clrpal4[3],xlim=c(0,.2),xlab="Premium quantile
(GBM)",ylab="",ylim=c(0,.2))
abline(a=0,b=1,lwd=.4)

...

Figure 5.2
```{r, tidy=TRUE}

par(mfrow=c(1,3))
vy_glm = predict(fit_loc_glm, newdata=vx)
vy_gam = predict(fit_loc_gam, newdata=vx)
vy_bst = predict(fit_loc_bst, newdata=vx)

plot(q_glm,vy_glm-vx,lwd=3,col=clrpal4[1],xlim=c(0,1),xlab="Premium
quantile (GLM)",ylab="",ylim=c(-.1,.1),type="l")
abline(h=0,lwd=.4)
plot(q_gam,vy_gam-vx,lwd=3,col=clrpal4[2],xlim=c(0,1),xlab="Premium
quantile (GAM)",ylab="",ylim=c(-.1,.1),type="l")
abline(h=0,lwd=.4)
plot(q_bst,vy_bst-vx,lwd=3,col=clrpal4[3],xlim=c(0,1),xlab="Premium
quantile (GBM)",ylab="",ylim=c(-.1,.1),type="l")
abline(h=0,lwd=.4)

...

```{r, tidy=TRUE}
  par(mfrow=c(1,3))
plot(q_glm,vy_glm/vx,lwd=3,col=clrpal4[1],xlim=c(0,1),xlab="Premium
quantile (GLM)",ylab="",ylim=c(.8,1.6),type="l")
abline(h=1,lwd=.4)
plot(q_gam,vy_gam/vx,lwd=3,col=clrpal4[2],xlim=c(0,1),xlab="Premium
quantile (GAM)",ylab="",ylim=c(.8,1.6),type="l")
abline(h=1,lwd=.4)
plot(q_bst,vy_bst/vx,lwd=3,col=clrpal4[3],xlim=c(0,1),xlab="Premium
quantile (GBM)",ylab="",ylim=c(.5,1.6),type="l")
abline(h=1,lwd=.4)

...

## Correction des primes

### primes autocalibrés et quartiles

```{r, tidy=TRUE}
p_glm = predict(fit_loc_glm,newdata = valid_prime_glm)
p_gam = predict(fit_loc_gam,newdata = valid_prime_gam)
p_bst = predict(fit_loc_bst,newdata = valid_prime_bst)
vx = seq(0,.2,length=351)

library(locfit)

valid_prime_glm_bc = valid_prime_glm * (p_glm/valid_prime_glm)
valid_prime_gam_bc = valid_prime_gam * (p_gam/valid_prime_gam)

```



```
kern="rect",deg=0,alpha=.05)
```

```
...
```

```
### Comaraison des deux modèles avec des graphiques
```

```
```{r, tidy=TRUE}
vu=(0:300)/300
Q_glm = quantile(valid_prime_glm, probs = vu)[3:299]
Q_gam = quantile(valid_prime_gam, probs = vu)[3:299]
Q_bst = quantile(valid_prime_bst, probs = vu)[3:299]

Q_glm_bc = quantile(valid_prime_glm_bc, probs = vu)[3:299]
Q_gam_bc = quantile(valid_prime_gam_bc, probs = vu)[3:299]
Q_bst_bc = quantile(valid_prime_bst_bc, probs = vu)[3:299]

idxs = sample(1:nrow(valid_LM),size=500)
idxs1 = sample(1:nrow(valid_LM),size=500)
```

```
par(mfrow=c(1,3))
plot(Q_glm, Q_glm_bc, lwd=3, col=clrpal4[1], type="l",
     xlab="Premium (GLM)",ylab="Premium BC (GLM)",
     xlim=c(0, .2),ylim=c(0, .2))
abline(a=0,b=1,lwd=.4)
plot(Q_gam, Q_gam_bc, lwd=3, col=clrpal4[2], type="l",
     xlab="Premium (GAM)",ylab="Premium BC (GAM)",
     xlim=c(0, .2),ylim=c(0, .2))
abline(a=0,b=1,lwd=.4)
plot(Q_bst, Q_bst_bc, lwd=3, col=clrpal4[3], type="l",
     xlab="Premium (GBM)",ylab="Premium BC (GBM)",
     xlim=c(0, .2),ylim=c(0, .2))
abline(a=0,b=1,lwd=.4)
```

```
...
```

```
```{r, tidy=TRUE}
par(mfrow=c(1,3))
plot(Q_glm, Q_glm_bc, lwd=3, col="white", type="l",
     xlab="Premium (GLM)",ylab="Premium BC (GLM)",
     xlim=c(0, .2),ylim=c(0, .2))
points(valid_prime_glm[idxs],valid_prime_glm_bc[idxs],pch=1,cex=.
4,col=scales::alpha(clrpal4[3],.4))
abline(a=0,b=1,lwd=.4)
plot(Q_gam, Q_gam_bc, lwd=3, col="white", type="l",
     xlab="Premium (GAM)",ylab="Premium BC (GAM)",
     xlim=c(0, .2),ylim=c(0, .2))
points(valid_prime_gam[idxs],valid_prime_gam_bc[idxs],pch=1,cex=.
4,col=scales::alpha(clrpal4[3],.4))
abline(a=0,b=1,lwd=.4)

plot(Q_bst, Q_bst_bc, lwd=3, col="white", type="l",
     xlab="Premium (GBM)",ylab="Premium BC (GBM)",
     xlim=c(0, .2),ylim=c(0, .2))
points(valid_prime_bst[idxs],valid_prime_bst_bc[idxs],pch=1,cex=.
4,col=scales::alpha(clrpal4[3],.4))
abline(a=0,b=1,lwd=.4)
```

```
...
```

```
```{r, tidy=TRUE}
par(mfrow=c(1,3))
plot(fit_loc_glm_bc,lwd=3,col=clrpal4[1],xlim=c(0,.2),xlab="Premium BC
(GLM)",ylab="",ylim=c(0,.2))
lines(fit_loc_glm,col=scales::alpha(clrpal4[1],.4))
abline(a=0,b=1,lwd=.4)
plot(fit_loc_gam_bc,lwd=3,col=clrpal4[2],xlim=c(0,.2),xlab="Premium BC
(GAM)",ylab="",ylim=c(0,.2))
lines(fit_loc_gam,col=scales::alpha(clrpal4[2],.4))
abline(a=0,b=1,lwd=.4)
plot(fit_loc_bst_bc,lwd=3,col=clrpal4[3],xlim=c(0,.2),xlab="Premium BC
(GBM)",ylab="",ylim=c(0,.2))
lines(fit_loc_bst,col=scales::alpha(clrpal4[3],.4))
abline(a=0,b=1,lwd=.4)
```

```
...
```

Figure 5.8

```
```{r, tidy=TRUE}

par(mfrow=c(1,3))
vy_glm_bc = predict(fit_loc_glm_bc, newdata=vx)
vy_gam_bc = predict(fit_loc_gam_bc, newdata=vx)
vy_bst_bc = predict(fit_loc_bst_bc, newdata=vx)

plot(q_glm_bc,vy_glm_bc,lwd=3,col=clrpal4[1],xlim=c(0,1),xlab="Premium
quantile (GLM)",ylab="",ylim=c(0,.2),type="l")
lines(q_glm,vx,lwd=.4)
lines(q_glm,vy_glm,col=scales::alpha(clrpal4[1],.4))
plot(q_gam_bc,vy_gam_bc,lwd=3,col=clrpal4[2],xlim=c(0,1),xlab="Premium
quantile (GAM)",ylab="",ylim=c(0,.2),type="l")
lines(q_gam,vx,lwd=.4)
lines(q_gam,vy_gam,col=scales::alpha(clrpal4[2],.4))

plot(q_bst_bc,vy_bst_bc,lwd=3,col=clrpal4[3],xlim=c(0,1.1),xlab="Premium
quantile (GBM)",ylab="",ylim=c(0,.2),type="l")
lines(q_bst,vx,lwd=.4)
lines(q_bst,vy_bst,col=scales::alpha(clrpal4[3],.4))
```

```
...
```

Figure 5.6

```
```{r, tidy=TRUE}

par(mfrow=c(1,3))
plot(q_glm_bc,vy_glm_bc-vx,lwd=3,col=clrpal4[1],xlim=c(0,1),xlab="Premium
quantile (GLM)",ylab="",ylim=c(-.1,.1),type="l")
abline(h=0,lwd=.4)
lines(q_glm,vy_glm-vx,col=scales::alpha(clrpal4[1],.4))
plot(q_gam_bc,vy_gam_bc-vx,lwd=3,col=clrpal4[2],xlim=c(0,1),xlab="Premium
quantile (GAM)",ylab="",ylim=c(-.1,.1),type="l")
abline(h=0,lwd=.4)
lines(q_gam,vy_gam-vx,col=scales::alpha(clrpal4[2],.4))
```

```

plot(q_bst_bc,vy_bst_bc-vx,lwd=3,col=clrpal4[3],xlim=c(0,1.1),xlab="Premium
quantile (GBM)",ylab="",ylim=c(-.1,.1),type="l")
abline(h=0,lwd=.4)
lines(q_bst,vy_bst-vx,col=scales::alpha(clrpal4[1],.4))
...

```{r, tidy=TRUE}

par(mfrow=c(1,3))
plot(q_glm_bc,vy_glm_bc/vx,lwd=3,col=clrpal4[1],xlim=c(0,1),xlab="Premium
quantile (GLM)",ylab="",ylim=c(.8,1.6),type="l")
abline(h=1,lwd=.4)
lines(q_glm,vy_glm/vx,col=scales::alpha(clrpal4[1],.4))
plot(q_gam_bc,vy_gam_bc/vx,lwd=3,col=clrpal4[2],xlim=c(0,1),xlab="Premium
quantile (GAM)",ylab="",ylim=c(.8,1.6),type="l")
abline(h=1,lwd=.4)
lines(q_gam,vy_gam/vx,col=scales::alpha(clrpal4[2],.4))
plot(q_bst_bc,vy_bst_bc/vx,lwd=3,col=clrpal4[3],xlim=c(0,1),xlab="Premium
quantile (GBM)",ylab="",ylim=c(.0,2),type="l")
abline(h=1,lwd=.4)
lines(q_bst,vy_bst/vx,col=scales::alpha(clrpal4[3],.4))
...

## Correlation

### Nous pouvons également considérer la corrélation entre les modèles \pi
et les versions corrigées associées \pi_{BC}.

```{r, tidy=TRUE}
X = cbind(valid_prime_glm,valid_prime_gam,valid_prime_bst,
          valid_prime_glm_bc,valid_prime_gam_bc,
          valid_prime_bst_bc)
M1=stats::cor(X,method="pearson")
M2=stats::cor(X,method="spearman")
library(corrplot)
...

```{r, tidy=TRUE}

colnames(M2)[1]=rownames(M2)[1]="glm"
colnames(M2)[2]=rownames(M2)[2]="gam"
colnames(M2)[3]=rownames(M2)[3]="bst"

colnames(M2)[4]=rownames(M2)[4]="glm BC"
colnames(M2)[5]=rownames(M2)[5]="gam BC"
colnames(M2)[6]=rownames(M2)[6]="bst BC"

corrplot(M2, method="number")
...

### Courbe de Concentration

```{r, tidy=TRUE}
CC = function(alpha, pi, y){
  q = quantile(pi,alpha)

```

```

    sum(y * (pi<=q))/sum(y)
}

valpha= seq(0,1,by=.001)

v_prime_glm = Vectorize(function(a)
CC(a,pi=valid_prime_glm,y=valid_LM$ClaimNb))(valpha)
v_prime_gam = Vectorize(function(a)
CC(a,pi=valid_prime_gam,y=valid_LM$ClaimNb))(valpha)
v_prime_bst = Vectorize(function(a)
CC(a,pi=valid_prime_bst,y=valid_LM$ClaimNb))(valpha)

v_prime_glm_bc = Vectorize(function(a)
CC(a,pi=valid_prime_glm_bc,y=valid_LM$ClaimNb))(valpha)
v_prime_gam_bc = Vectorize(function(a)
CC(a,pi=valid_prime_gam_bc,y=valid_LM$ClaimNb))(valpha)
v_prime_bst_bc = Vectorize(function(a)
CC(a,pi=valid_prime_bst_bc,y=valid_LM$ClaimNb))(valpha)

...

##### courbe de concentration GLM et GLM_BC

```{r, tidy=TRUE}
par(mfrow=c(1,2))
plot(valpha,v_prime_glm,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^glm ~" et "~ pi[bc]^glm))
abline(a=0,b=1,lwd=.4)
lines(valpha,v_prime_glm_bc,type="l",lwd=3,col=clrpal4[4])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[4]), lty = 1,
  legend = c(expression(pi^glm), expression(pi[bc]^glm))
)

plot(valpha,v_prime_bst-
v_prime_bst_bc,type="h",lwd=3,col=clrpal4[1],xlim=c(0,1), ylim=c(-.2,
.08),
      xlab=expression(alpha),ylab=expression(pi^glm - pi[bc]^glm))

...

##### courbe de concentration GAM et GAM_BC

```{r, tidy=TRUE}
par(mfrow=c(1,2))
plot(valpha,v_prime_gam,type="l",lwd=3,col=clrpal4[2],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^gam ~" et "~ pi[bc]^gam))
abline(a=0,b=1,lwd=.4)
lines(valpha,v_prime_gam_bc,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[2], clrpal4[3]), lty = 1,
  legend = c(expression(pi^gam), expression(pi[bc]^gam))
)

plot(valpha,v_prime_gam-
v_prime_gam_bc,type="h",lwd=3,col=clrpal4[2],xlim=c(0,1), ylim=c(-.2,
.08),

```

```

        xlab=expression(alpha),ylab=expression(pi^gam - pi[bc]^gam))
    ...
##### courbe de concentration GBM et GBM_BC

```{r, tidy=TRUE}
par(mfrow=c(1,2))
plot(valpha,v_prime_bst,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^gbm ~" et "~ pi[bc]^gbm))
abline(a=0,b=1,lwd=.4)
lines(valpha,v_prime_bst_bc,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[3]), lty = 1,
  legend = c(expression(pi^gbm), expression(pi[bc]^gbm))
)

plot(valpha,v_prime_bst-
v_prime_bst_bc,type="h",lwd=3,col=clrpal4[3],xlim=c(0,1), ylim=c(-.2,
.08),
      xlab=expression(alpha),ylab=expression(pi^gbm - pi[bc]^gbm))
    ...
##### Autre representation courbe de concentration

```{r, tidy=TRUE}

idx = which(valid_LM$Exposure == 1)
idx1 = which(valid_LM$Exposure == 1)
v_prime_glm_i = Vectorize(function(a)
CC(a,pi=valid_prime_glm[idx],y=valid_LM$ClaimNb[idx]))(valpha)
v_prime_gam_i = Vectorize(function(a)
CC(a,pi=valid_prime_gam[idx],y=valid_LM$ClaimNb[idx]))(valpha)
v_prime_bst_i = Vectorize(function(a)
CC(a,pi=valid_prime_bst[idx1],y=valid_LM$ClaimNb[idx1]))(valpha)
v_prime_glm_bc_i = Vectorize(function(a)
CC(a,pi=valid_prime_glm_bc[idx],y=valid_LM$ClaimNb[idx]))(valpha)
v_prime_gam_bc_i = Vectorize(function(a)
CC(a,pi=valid_prime_gam_bc[idx],y=valid_LM$ClaimNb[idx]))(valpha)
v_prime_bst_bc_i = Vectorize(function(a)
CC(a,pi=valid_prime_bst_bc[idx1],y=valid_LM$ClaimNb[idx1]))(valpha)
    ...

##### comparaison des primes glm gam et bst avec la prime_bc_bst

```{r, tidy=TRUE}
par(mfrow=c(1,3))
plot(valpha,v_prime_glm_i,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^glm ~" and "~ pi[bc]^bst))
lines(valpha,v_prime_bst_bc_i,type="l",lwd=3,col=clrpal4[4])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[4]), lty = 1,
  legend = c(expression(pi^glm), expression(pi[bc]^bst))
)

plot(valpha,v_prime_gam_i,type="l",lwd=3,col=clrpal4[2],xlim=c(0,1),

```

```

        xlab=expression(alpha),ylab=expression(pi^gam ~" and "~ pi[bc]^bst))
lines(valpha,v_prime_bst_bc_i,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[2], clrpal4[3]), lty = 1,
  legend = c(expression(pi^gam), expression(pi[bc]^bst))
)

plot(valpha,v_prime_bst_i,type="l",lwd=3,col=clrpal4[3],xlim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^bst ~" and "~ pi[bc]^bst))
lines(valpha,v_prime_bst_bc_i,type="l",lwd=3,col=clrpal4[4])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[3], clrpal4[4]), lty = 1,
  legend = c(expression(pi^bst), expression(pi[bc]^bst))
)

...

```{r, tidy=TRUE}
par(mfrow=c(1,3))
plot(valpha,v_prime_glm_i-
v_prime_bst_bc_i,type="h",lwd=3,col=clrpal4[4],xlim=c(0,1),ylim=c(-.1,.07),
      xlab=expression(alpha),ylab=expression(pi^glm - pi[bc]^bst))

plot(valpha,v_prime_gam_i-
v_prime_bst_bc_i,type="h",lwd=3,col=clrpal4[4],xlim=c(0,1),ylim=c(-.1,.07),
      xlab=expression(alpha),ylab=expression(pi^gam - pi[bc]^bst))

plot(valpha,v_prime_bst_i-
v_prime_bst_bc_i,type="h",lwd=3,col=clrpal4[4],xlim=c(0,1),ylim=c(-.
07,.07),
      xlab=expression(alpha),ylab=expression(pi^bst - pi[bc]^bst))

...

##### Courbe de Lorentz

```{r, tidy=TRUE}
LC = function(alpha, pi, y){
  q = quantile(pi,alpha)
  sum(y * (pi<=q))/sum(y)
}

valpha= seq(0,1,by=.001)
L_prime_glm = Vectorize(function(a)
LC(a,pi=valid_prime_glm,y=valid_prime_glm))(valpha)
L_prime_gam = Vectorize(function(a)
LC(a,pi=valid_prime_gam,y=valid_prime_gam))(valpha)
L_prime_bst = Vectorize(function(a)
LC(a,pi=valid_prime_bst,y=valid_prime_bst))(valpha)

L_prime_glm_bc = Vectorize(function(a)
LC(a,pi=valid_prime_glm_bc,y=valid_prime_glm_bc))(valpha)

```

```

L_prime_gam_bc = Vectorize(function(a)
LC(a,pi=valid_prime_gam_bc,y=valid_prime_gam_bc))(valpha)
L_prime_bst_bc = Vectorize(function(a)
LC(a,pi=valid_prime_bst_bc,y=valid_prime_bst_bc))(valpha)
...

##### Lorentz GLM
```{r, tidy=TRUE}

par(mfrow=c(1,2))
plot(valpha,L_prime_glm,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^glm ~" et "~ pi[bc]^glm))
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_glm_bc,type="l",lwd=3,col=clrpal4[4])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[4]), lty = 1,
  legend = c(expression(pi^glm), expression(pi[bc]^glm))
)

plot(valpha,L_prime_bst-
L_prime_bst_bc,type="h",lwd=3,col=clrpal4[1],xlim=c(0,1), ylim=c(-.1,
.08),
      xlab=expression(alpha),ylab=expression(pi^glm - pi[bc]^glm))

...

##### Lorentz GAM
```{r, tidy=TRUE}
par(mfrow=c(1,2))
plot(valpha,L_prime_gam,type="l",lwd=3,col=clrpal4[2],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^gam ~" et "~ pi[bc]^gam))
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_gam_bc,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[2], clrpal4[3]), lty = 1,
  legend = c(expression(pi^gam), expression(pi[bc]^gam))
)

plot(valpha,L_prime_gam-
L_prime_gam_bc,type="h",lwd=3,col=clrpal4[2],xlim=c(0,1), ylim=c(-.1,
.08),
      xlab=expression(alpha),ylab=expression(pi^gam - pi[bc]^gam))

...

##### Lorentz GBM
```{r, tidy=TRUE}
par(mfrow=c(1,2))
plot(valpha,L_prime_bst,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^gbm ~" et "~ pi[bc]^gbm))
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_bst_bc,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",

```

```

col = c(clrpal4[1], clrpal4[3]), lty = 1,
legend = c(expression(pi^gbm), expression(pi[bc]^gbm))
)

plot(valpha,L_prime_bst-
L_prime_bst_bc,type="h",lwd=3,col=clrpal4[4],xlim=c(0,1), ylim=c(-.1,
.08),
      xlab=expression(alpha),ylab=expression(pi^gbm - pi[bc]^gbm))

...

#####----- La courbe ABC -----

##### cas GLM et GLM_bc
```{r, tidy=TRUE}

par(mfrow=c(1,2))
plot(valpha,v_prime_glm,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^glm_cc ~" et "~ pi^glm_lc))
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_glm,type="l",lwd=3,col=clrpal4[4])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[4]), lty = 1,
  legend = c(expression(pi^glm_cc), expression(pi^glm_lc))
)

plot(valpha,v_prime_glm-
L_prime_glm,type="h",lwd=3,col=clrpal4[4],xlim=c(0,1), ylim=c(-.1, .1),
      xlab=expression(alpha),ylab=expression(pi^glm_cc - pi[bc]^glm_lc))

...

```{r, tidy=TRUE}

par(mfrow=c(1,2))
plot(valpha,v_prime_glm,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^glm_cc ~" et "~ pi^glm_lc))
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_glm,type="l",lwd=3,col=clrpal4[4])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[4]), lty = 1,
  legend = c(expression(pi^glm_cc), expression(pi^glm_lc))
)

plot(valpha,v_prime_glm-
L_prime_glm,type="h",lwd=3,col=clrpal4[4],xlim=c(0,1), ylim=c(-.1, .1),
      xlab=expression(alpha),ylab=expression(pi^glm_cc - pi[bc]^glm_lc))

...

##### cas GAM et GAM_bc

```

```

```{r, tidy=TRUE}

par(mfrow=c(1,2))
plot(valpha,v_prime_gam,type="l",lwd=3,col=clrpal4[2],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^gam_cc ~" et "~ pi^gam_lc))
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_gam,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[2], clrpal4[3]), lty = 1,
  legend = c(expression(pi^gam_cc), expression(pi^gam_lc))
)

plot(valpha,v_prime_gam-
L_prime_gam,type="h",lwd=3,col=clrpal4[2],xlim=c(0,1), ylim=c(-.1, .1),
      xlab=expression(alpha),ylab=expression(pi^gam_cc - pi^gam_lc))

...

```{r, tidy=TRUE}
par(mfrow=c(1,2))
plot(valpha,v_prime_gam_bc,type="l",lwd=3,col=clrpal4[2],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi[bc]^gam_cc ~" et "~
pi[bc]^gam_lc))
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_gam_bc,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[2], clrpal4[3]), lty = 1,
  legend = c(expression(pi[bc]^gam_cc), expression(pi[bc]^gam_lc))
)

plot(valpha,v_prime_gam_bc-
L_prime_gam_bc,type="h",lwd=3,col=clrpal4[2],xlim=c(0,1), ylim=c(-.1, .1),
      xlab=expression(alpha),ylab=expression(pi[bc]^gam_cc - pi[bc]^gam_lc))

...

##### cas GBM et GBM_bc

```{r, tidy=TRUE}
par(mfrow=c(1,2))
plot(valpha,v_prime_bst,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^gbm_cc ~" et "~ pi^gbm_lc))
abline(a=0,b=1,lwd=.4)
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_bst,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[3]), lty = 1,
  legend = c(expression(pi^gbm_cc), expression(pi^gbm_lc))
)

```

```

plot(valpha,v_prime_bst-
L_prime_bst,type="h",lwd=3,col=clrpal4[2],xlim=c(0,1), ylim=c(-.1, .1),
      xlab=expression(alpha),ylab=expression(pi^bst_cc - pi^bst_lc))

...

```{r, tidy=TRUE}
par(mfrow=c(1,2))

plot(valpha,v_prime_bst_bc,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi[bc]^gbm_cc ~" et "~
pi[bc]^gbm_lc))
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_bst_bc,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[3]), lty = 1,
  legend = c(expression(pi[bc]^gbm_cc), expression(pi[bc]^gbm_lc))
)

plot(valpha,v_prime_bst_bc-
L_prime_bst_bc,type="h",lwd=3,col=clrpal4[4],xlim=c(0,1), ylim=c(-.1, .1),
      xlab=expression(alpha),ylab=expression(pi[bc]^gbm_cc - pi[bc]^gbm_lc))

...

### cas GBM et GBM_bc

```{r, tidy=TRUE}
par(mfrow=c(1,2))
plot(valpha,v_prime_bst,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi^gbm_cc ~" et "~ pi^gbm_lc))
abline(a=0,b=1,lwd=.4)
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_bst,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[3]), lty = 1,
  legend = c(expression(pi^gbm_cc), expression(pi^gbm_lc))
)

plot(valpha,v_prime_bst-
L_prime_bst,type="h",lwd=3,col=clrpal4[2],xlim=c(0,1), ylim=c(-.1, .1),
      xlab=expression(alpha),ylab=expression(pi^bst_cc - pi^bst_lc))

...

```{r, tidy=TRUE}
par(mfrow=c(1,2))

```

```

plot(valpha,v_prime_bst_bc,type="l",lwd=3,col=clrpal4[1],xlim=c(0,1),ylim=c(0,1),
      xlab=expression(alpha),ylab=expression(pi[bc]^gbm_cc ~" et "~
pi[bc]^gbm_lc))
abline(a=0,b=1,lwd=.4)
lines(valpha,L_prime_bst_bc,type="l",lwd=3,col=clrpal4[3])
legend(
  x="topleft", bty= "n",
  col = c(clrpal4[1], clrpal4[3]), lty = 1,
  legend = c(expression(pi[bc]^gbm_cc), expression(pi[bc]^gbm_lc))
)

```

```

plot(valpha,v_prime_bst_bc-
L_prime_bst_bc,type="h",lwd=3,col=clrpal4[4],xlim=c(0,1), ylim=c(-.1, .1),
      xlab=expression(alpha),ylab=expression(pi[bc]^gbm_cc - pi[bc]^gbm_lc))

```

...

```
#####----- valeur de l'aire ABC
```

```
##### GLM
```

```
```{r, tidy=TRUE}
library(tis)

```

...

```
```{r, tidy=TRUE}
ABC_glm = sum(lintegrate(valpha, v_prime_glm - L_prime_glm, xint = valpha))

```

```
ABC_glm_bc = sum(lintegrate(valpha, v_prime_glm_bc - L_prime_glm_bc, xint =
valpha))

```

```
ABC_glm
```

```
ABC_glm_bc
```

...

```
## cas GAM
```

```
```{r, tidy=TRUE}
ABC_gam = sum(lintegrate(valpha, v_prime_gam - L_prime_gam, xint = valpha))

```

```
ABC_gam_bc = sum(lintegrate(valpha, v_prime_gam_bc - L_prime_gam_bc, xint =
valpha))

```

```
ABC_gam
```

```
ABC_gam_bc
```

...

```
## Cas GBM
```

```
```{r, tidy=TRUE}
ABC_bst = sum(lintegrate(valpha, v_prime_bst - L_prime_bst, xint = valpha))

```

```
ABC_bst_bc = sum(lintegrate(valpha, v_prime_bst_bc - L_prime_bst_bc, xint =
valpha))

```

ABC\_bst

ABC\_bst\_bc

...

#####----- Deviance de poisson

##### Cas GLM

```
```{r, tidy=TRUE}
```

```
dev_glm = 2*sum((ifelse(valid_LM$ClaimNb==0, 0,  
valid_LM$ClaimNb*(log(valid_LM$ClaimNb/valid_prime_glm))) -  
(valid_LM$ClaimNb - valid_prime_glm)))
```

```
dev_glm_bc = 2*sum((ifelse(valid_LM$ClaimNb==0, 0,  
valid_LM$ClaimNb*(log(valid_LM$ClaimNb/valid_prime_glm_bc))) -  
(valid_LM$ClaimNb - valid_prime_glm_bc)))
```

dev\_glm

dev\_glm\_bc

...

##### Cas GAM

```
```{r, tidy=TRUE}
```

```
dev_gam = 2*sum((ifelse(valid_LM$ClaimNb==0, 0,  
valid_LM$ClaimNb*(log(valid_LM$ClaimNb/valid_prime_gam))) -  
(valid_LM$ClaimNb - valid_prime_gam)))
```

```
dev_gam_bc = 2*sum((ifelse(valid_LM$ClaimNb==0, 0,  
valid_LM$ClaimNb*(log(valid_LM$ClaimNb/valid_prime_gam_bc))) -  
(valid_LM$ClaimNb - valid_prime_gam_bc)))
```

dev\_gam

dev\_gam\_bc

...

##### Cas GBM

```
```{r, tidy=TRUE}
```

```
dev_gbm = 2*sum((ifelse(valid_LM$ClaimNb==0, 0,  
valid_LM$ClaimNb*(log(valid_LM$ClaimNb/valid_prime_bst))) -  
(valid_LM$ClaimNb - valid_prime_bst)))
```

```
dev_gbm_bc = 2*sum((ifelse(valid_LM$ClaimNb==0, 0,  
valid_LM$ClaimNb*(log(valid_LM$ClaimNb/valid_prime_bst_bc))) -  
(valid_LM$ClaimNb - valid_prime_bst_bc)))
```

```

dev_gbm

dev_gbm_bc
...

#### Déviance de tweedie

```{r, tidy=TRUE}
  devTweed = function(u,v,e){
    weighted.mean(u-(v/e)*log(u),e)
  }
...

```{r, tidy=TRUE}
  dvt_glm =
devTweed(valid_prime_glm,valid_LM_1$ClaimNb,valid_LM_1$Exposure)
  dvt_gam =
devTweed(valid_prime_gam,valid_LM_1$ClaimNb,valid_LM_1$Exposure)
  dvt_gbm =
devTweed(valid_prime_bst,valid_LM_1$ClaimNb,valid_LM_1$Exposure)
  dvt_glm_bc =
devTweed(valid_prime_glm_bc,valid_LM_1$ClaimNb,valid_LM_1$Exposure)
  dvt_gam_bc =
devTweed(valid_prime_gam_bc,valid_LM_1$ClaimNb,valid_LM_1$Exposure)
  dvt_gbm_bc =
devTweed(valid_prime_bst_bc,valid_LM_1$ClaimNb,valid_LM_1$Exposure)
...

```{r, tidy=TRUE}
dvt=c(dvt_glm,dvt_gam,dvt_gbm,dvt_gbm_bc,dvt_gam_bc,dvt_gbm_bc)

names(dvt)=c("glm","gam","gbm","glm_bc","gam_bc","gbm_bc")

dvt

...

```

