Analyse de coût en Assurance Non-Vie: Modèles Linéaires Généralisés vs Méthodes d'Apprentissage Statistique

Camille PERRIN

Mémoire d'Actuariat
ISFA - Promotion 2012

Responsable entreprise
Christophe CROCHET

Responsable formation
Christian ROBERT

ISFA, Université Claude Bernard Lyon 1
50 Avenue Tony Garnier
69007 Lyon

PwC Luxembourg
400 route d'Esch
L-1471 Luxembourg

Remerciements

Je souhaite remercier Christophe Crochet, responsable de l'équipe actuarielle de PwC Luxembourg, pour son aide, ses conseils et son partage des connaissances.

Je remercie également Christian Robert pour son suivi du mémoire, sa réactivité et sa disponibilité.

Enfin, je remercie plus globalement toutes les personnes qui de près ou de loin ont su m'apporter leur soutien durant mes études.

Mots clés

Modèles Linéaires Généralisés, Apprentissage statistique, Classification And Regression Trees (CART), assurance non-vie, analyse de coût, Genmod, Rpart.

Résumé

L'objectif de ce projet est de modéliser le coût d'un contrat d'assurance non-vie, dont les caractéristiques seront décrites au début de ce travail, de deux manières différentes : la première par Modèle Linéaire Généralisé (MLG), la deuxième à l'aide de l'algorithme Classification And Regression Trees (CART), relatif à la théorie de l'Apprentissage Statistique.

Au terme de ces deux mises en application, nous serons en mesure de réaliser une comparaison des deux méthodes et d'en résumer les principaux intérêts et inconvénients.

Dans un premier temps, nous appliquerons plusieurs retraitements sur la base de données qui sera utilisée pour les deux modélisations. Nous effectuerons également une première analyse univariée afin d'observer l'évolution des coûts selon les variables caractérisant un contrat.

Ensuite, nous appliquerons un MLG à nos données, après avoir déterminé les variables explicatives entrant en jeu dans la tarification. Nous confronterons les résultats du MLG à l'analyse univariée, et validerons notre modèle par une étude des résidus. Enfin, nous présenterons une méthode de lissage des résultats.

Dans une troisième partie, nous aborderons les notions d'apprentissage statistique et d'arbre de décision binaire. La méthodologie sera ensuite appliquée aux données d'assurance. Nous mettrons notamment en pratique les techniques de validation croisée et d'élagage afin d'obtenir un arbre de décision optimal en termes de qualité et de complexité. Nous validerons également le modèle par une analyse des résidus.

Enfin, dans une dernière partie, nous confronterons les estimations aux observations afin d'évaluer la qualité de prédiction, et nous comparerons les résultats MLG et CART entre eux en vue de conclure quant à la stabilité des résultats d'une méthode à l'autre. Cela nous permettra d'évaluer les points forts et les points faibles de chaque méthode.

Le MLG étant largement la méthode privilégiée par les assureurs aujourd'hui, nous discuterons d'une éventuelle mise en pratique sur le marché d'une tarification par CART.

Keywords

Generalized Linear Models, Machine Learning, Classification And Regression Trees (CART), non-life insurance, cost analysis, Genmod, Rpart.

Abstract

The goal of this project is modeling the cost of a non-life insurance contract, whose characteristics will be described at the beginning of this work, in two different ways: the first using a Generalised Linear Model (GLM), the second thanks to the Classification And Regression Trees (CART) algorithm, being part of the Machine Learning theory.

At the end of these two modelings, we will be able to make a comparison of the two methods and to summarise their main interests and inconveniences.

First, we will apply several cleaning processes on the database which will be used for the two modelings. We will also do a first oneway analysis in order to observe the cost evolution in function of the variables defining a contract.

Then, we will apply a GLM to our data, once having determined the explanation variables involved in pricing. We will compare the GLM results with the oneway analysis, and validate our model with a residual analysis. Finally, we will present a smoothing method of the results.

In a third step, we will introduce to machine learning and binary decision trees concepts. The methodology will then be applied to the insurance data. We will particularly implement the cross validation and pruning techniques in order to get an optimal decision tree in terms of quality and complexity. We will also validate the model with a residual analysis.

Finally, in a last part, we will compare the estimates with the observations to evaluate the quality of forecasting, and the GLM and CART results to conclude on the stability of results from one method to another. This will enable us to evaluate strengths and weaknesses of each method.

The GLM being largly used by insurers nowadays, we will discuss about a possible application on the market of a CART pricing.

Table des matières

Introduction

En 1972, John Nelder [1] publie un article [7] contenant les premiers travaux réalisés sur les Modèles Linéaires Généralisés (MLG). Ceux-ci permettent alors d'étendre les régressions linéaires à des cas plus généraux, et ainsi de résoudre des problèmes de modélisation plus complexes.

Alors que la régression linéaire cantonne la distribution de la variable réponse à la loi normale, les MLG permettent à présent d'ouvrir le champ des possibilités quant à la loi sous-jacente de la variable réponse : principalement les lois Gamma, Poisson et Binomiale. La loi doit néanmoins toujours appartenir à la famille des lois exponentielles.

Des recherches ont peu à peu conduit à considérer une nouvelle théorie : l'apprentissage statistique, ou "machine learning", dont le principal atout est l'absence de contraintes sur la loi sous-jacente de la variable réponse (ce sont des modèles non paramétriques). Parmi ces méthodes, la plus répandue est l'algorithme Classification And Regression Trees (CART), développé par Leo Breiman [2] en 1984 [3], permettant de construire des arbres de décision binaires sans aucune hypothèse sur la structure des données.

Le but de ce travail est de comparer les performances d'une méthode classique et éprouvée qu'est le MLG et d'une méthode plus nouvelle et émergeante qu'est CART. Nous allons donc appliquer les deux modélisations à un même jeu de données issues d'un portefeuille d'assurance.

La première partie de ce mémoire concerne les travaux effectués sur la base de données mise à disposition par l'assureur, afin de la rendre exploitable dans le cadre de nos modélisations. Cela représente une part non négligeable dans l'ensemble du projet. En effet, il s'agit de comprendre la structure des données, et l'organisation de l'entreprise d'assurance dans la gestion de ces informations.

Il est ensuite nécessaire de créer des variables supplémentaires, de retraiter les données en fonction des besoins de la modélisation, de filtrer les observations inutiles (détermination du périmètre de travail), ...

Nous réalisons également une première analyse descriptive univariée afin d'observer l'évolution des coûts en fonction de chaque variable caractéristique du contrat.

La deuxième partie présente l'analyse de coût par MLG : nous rappelons quelques notions théoriques générales, puis nous mettons ces techniques en application sur la base de données. Pour cela, les variables explicatives qui entrent en jeu dans le tarif sont déterminées, et nous utilisons le logiciel SAS avec la procédure *Genmod* pour réaliser la régression. Les résultats obtenus sont comparés à l'analyse univariée précédemment réalisée.

Nous validons ensuite notre modèle par une étude des résidus, et présentons une méthode de lissage des résultats.

1. John Nelder, 1924-2010, statisticien Britannique
2. Leo Breiman, 1928-2005, statisticien Américain

Ensuite, nous nous intéressons à une analyse de coût par apprentissage statistique. Après une présentation théorique de la construction des arbres de décision binaires, nous appliquons l'algorithme CART à nos données à l'aide du logiciel R et du package *Rpart*.

Enfin, sur base d'une comparaison des résultats obtenus, nous concluons sur les intérêts et les points négatifs de chacune de ces deux méthodes.

Préambule : description du contrat d'assurance

Ce mémoire est en partie issu d'une mission de conseil en actuariat réalisée au sein de PwC Luxembourg, et la base de données utilisée dans ce travail est fournie par le client.

Pour des raisons de confidentialité, il m'est donc naturellement imposé de ne divulguer aucune information sur le client et son domaine d'activité. Toutefois, et ce afin de faciliter une meilleure compréhension du déroulement du travail, nous considérerons un contrat d'assurance fictif, dont les caractéristiques sont présentées ci-dessous.

Outre le gain de compréhension généré par la concrétisation du problème, cette analogie nous permettra également de pouvoir interpréter les résultats au mieux, même si ceux-ci n'auront pas toujours de sens compte-tenu du caractère imaginaire de ce contrat.

Couverture

Le produit d'assurance est un contrat d'entretien/réparation pour des suites d'hôtel. Le souscripteur est le propriétaire de la suite, ou le gérant, et le bien assuré est la suite en elle-même. Sont assurés tous les meubles et contenus de la suite (hors effets personnels des occupants), dans toutes les pièces qui la constituent (entrée, coin cuisine, chambre, salle de bains ...). Un contrat est souscrit pour une seule suite à la fois.

Les choses assurées sont couvertes pour des interventions d'entretien et/ou de réparation. Par exemple, si les murs ont besoin d'être repeints, il s'agit d'entretien. Si la porte d'un placard est cassée, il s'agit de réparation. Le souscripteur déclare le sinistre à l'assurance, et celle-ci prend en charge les frais d'interventions nécessaires à la remise en état du bien.

Durée

Le produit d'assurance est souscrit pour une durée limitée et fixée à la signature du contrat. La durée est exprimée en mois et peut varier de 6 mois à plus de 78 mois. Il n'y a pas de reconduction tacite du contrat lorsque celui-ci arrive à échéance.

Tarification

La prime d'assurance est déterminée pour toute la durée du contrat, puis divisée en mensualités. Par exemple, si l'on tarifie un contrat de 48 mois à 1500 €, la prime mensuelle payable par le souscripteur sera de $1500/48 = 31{,}25$ €.

Franchises et limites d'indemnisation

Il n'y a pas de franchises prévues au contrat, ni de limites d'indemnisation. Néanmoins, des experts interviennent à chaque sinistre pour juger si la déclaration ne présente pas d'abus. Certains remboursements peuvent être refusés au souscripteur si l'expert juge abusive la demande d'indemnisation. Ainsi, l'aléa moral [3] est limité.

3. L'aléa moral est la possibilité qu'un assuré augmente sa prise de risque, par rapport à la situation où il supporterait entièrement les conséquences négatives d'un sinistre.

Première partie

La base de données

1.1 Description

La base de données est composée de deux parties :

- une table **sinistres** contenant les informations relatives aux sinistres : date de l'événement, numéro d'identification de la suite d'hôtel (i.e. le numéro du contrat d'assurance), nature du sinistre (ce qui a fait l'objet d'un entretien ou d'une réparation), coût du sinistre ;
- une table **polices** regroupant les données relatives aux contrats : dates de début et de fin du contrat, type de produit d'assurance (gamme, avec différents niveaux de garantie), caractéristiques de la suite d'hôtel (étage, superficie, situation géographique, etc.).

Les bases regroupent des informations sur environ 130 000 contrats et 900 000 sinistres, sur une durée de 7 ans.

1.2 Retraitements nécessaires à la modélisation

Afin de pouvoir appliquer un modèle à nos données, il est nécessaire d'effectuer un certain nombre de retraitements au préalable. On dit souvent que les 75% du temps de travail sont passés sur la mise en forme des données, et que 25% sont dédiés à l'exploitation de celles-ci, à la modélisation proprement dite. Cela s'est vérifié dans ce projet.

Le propriétaire de la suite a la possibilité de souscrire un contrat d'entretien uniquement, de réparation uniquement, ou bien les deux. Le but de ce projet est donc de déterminer un prix d'assurance pour l'entretien et la réparation séparément. Ceci va constituer une des principales difficultés dans le retraitement des données, car celles-ci ne présentent pas de séparation des coûts entre entretien et réparation.

Les autres difficultés rencontrées dans l'exploitation de la base de données sont le retraitement structurel des informations (regroupements de plusieurs lignes correspondant à un même sinistre) et la création de variables additionnelles nécessaires à la modélisation (définition de la nature d'une intervention : entretien ou réparation ; création de variables d'exposition au risque).

Enfin, il est nécessaire d'agréger les coûts de tous les sinistres survenus sur un même contrat tout au long de la durée de celui-ci, puisque l'on souhaite modéliser la prime pure sur la durée totale d'un contrat (que l'on divisera ensuite par le nombre de mois pour obtenir une prime mensuelle).

Les paragraphes suivants détaillent les principales étapes effectuées sur les 2 bases décrites ci-dessus (polices et sinistres), et qui ont conduit à la base de données finale utilisée pour l'application des modèles : le Modèle Linéaire Généralisé (MLG) dans le cadre de la partie 2 et l'arbre de régression binaire (algorithme CART) dans le cadre des méthodes d'apprentissage statistique en partie 3.

1.2.1 Retraitement des polices

Certaines polices sont supprimées pour cause d'informations techniques mal renseignées (caractéristiques de la suite). Pour d'autres, il est possible de reconstituer les champs manquants sur base des informations fournies par d'autres variables. Ces ajustements sont faits manuellement, pour éliminer la majorité des cas où il manque des informations. Lorsque le travail de reconstitution devient trop fastidieux (peu de cas concernés), on retire simplement l'observation de la base.

Une manipulation essentielle pour la suite de l'étude est effectuée sur les polices. Chaque ligne correspond à une police, avec une date de début de contrat et une date de fin de contrat. Une police peut être présente sur plusieurs années, et les expositions au risque sont différentes d'une police à une autre. Chaque ligne est alors dédoublée autant de fois qu'il y a d'années de présence de la police. Par exemple, un contrat commençant le 1^{er} Janvier 2008 et se terminant le 30 Juin 2010 apparaitra 3 fois dans la base :

- une fois pour l'année 2008 et une fois pour l'année 2009, avec une exposition au risque de 1 sur chaque année (le contrat est présent sur la totalité de l'année) ;
- une fois pour l'année 2010 avec une exposition au risque de 0.5 (le contrat n'est présent dans le portefeuille que la moitié de l'année).

Ainsi, cette manipulation conduit naturellement à augmenter significativement le nombre de lignes dans la base, puisque chaque police apparaîtra plusieurs fois, selon sa durée.

Le but de ce retraitement est de pouvoir par la suite fusionner les polices avec les sinistres, sur une clé constituée du numéro d'identification de la police (numéro du contrat) et de la date du sinistre (sur notre exemple, un sinistre survenu en 2010 fusionnera avec la ligne 2010 de la police correspondante ; ainsi, on identifiera très clairement les années où la police n'a eu aucun sinistre, ici les années 2008 et 2009).

1.2.2 Retraitement des sinistres

Un sinistre correspond à une intervention sur un ou plusieurs des éléments de la suite (frigo, lit, penderie, mur, etc.). On distingue les éléments selon la pièce dans laquelle ils se trouvent (chambre, cuisine, salle de bain, ...) et chaque élément sinistré est classé en **Entretien** (ex : peinture des murs) ou **Réparation** (ex : volet cassé).

Les sinistres sont enregistrés de la manière suivante. Chaque ligne de la base de données correspond à un sinistre et on y trouve les informations suivantes :

- date du sinistre
- numéro d'identification de la suite d'hôtel (le numéro de contrat qui couvre la suite en question)
- pièces concernées : chambre, salle de bain, entrée, ...
- éléments concernés : évier, lit, mur, lampe, ...
- coût total du sinistre

Pour une meilleure compréhension de la structure des données, prenons l'exemple suivant (table 1.1 p. 20) :

N° contrat	Date	Entrée	Chambre	SDB	Cuisine	Salon	Coût total
123456	16/12/12	Mur(E)	Armoire(R)		Evier(R)		300.45

Table 1.1 – Structure des données de sinistres

Le 16 Décembre 2012, le souscripteur du contrat 123456 a déclaré le sinistre suivant sur sa suite d'hôtel : peinture des murs de l'entrée (entretien), réparation de l'armoire de la chambre, et réparation de l'évier de la cuisine, pour un total de 300.45€. Il n'y a pas eu d'intervention dans la salle de bain et le salon.

La méthode d'enregistrement des sinistres dans la base présente des faiblesses, il est donc indispensable d'effectuer quelques retraitements pour qu'elle soit exploitable. C'est l'objet des sections suivantes.

Lignes multiples

Si plusieurs interventions ont lieu sur des objets différents au sein d'une même pièce, les lignes sont dédoublées car on ne peut pas inscrire deux éléments dans la case "Chambre" par exemple. Le système va donc créer une deuxième ligne pour inscrire ce deuxième élément, mais le coût correspondant à cette ligne sera nul. Il s'agit d'un problème opérationnel lié à l'encodage des interventions. Si l'on reprend l'exemple précédent, et que l'on considère qu'il y a eu en plus le nettoyage de la moquette dans la chambre (c'est de l'entretien), le problème se présentera de la manière suivante (table 1.2 p. 20) :

N° contrat	Date	Entrée	Chambre	SDB	Cuisine	Salon	Coût total
123456	16/12/12	Mur(E)	Armoire(R)		Evier(R)		450.30
123456	16/12/12		Moquette (E)				0

Table 1.2 – Lignes multiples

Le but de la manipulation est alors d'arriver au résultat suivant (table 1.3 p. 20) :

N° contrat	Date	Entrée	Chambre	SDB	Cuisine	Salon	Coût total
123456	16/12/12	Mur(E)	Armoire(R) Moquette(E)		Evier(R)		450.30

Table 1.3 – Lignes multiples retraitées

Cela se corrige par un traitement itératif sur les lignes ayant le même numéro de contrat et la même date d'intervention (i.e. les lignes correspondant à un même sinistre) et en triant judicieusement la base.

Coûts annulés

Certaines lignes se présentent de la manière suivante (table 1.4 p. 21) :

N° contrat	Date	Entrée	Chambre	SDB	Cuisine	Salon	Coût total
123456	16/12/12	Mur(E)	Armoire(R)		Evier(R)		450.30
123456	16/12/12						-450.30

TABLE 1.4 – Coûts annulés

Les lignes de ce type doivent donc être retirées de la base, car elle n'engendrent pas de coût pour l'assureur. En effet, il s'agit de sinistres pour lesquels l'assureur a d'abord pris en charge les frais, puis finalement demandé remboursement au client pour diverses raisons (par exemple le sinistre ne rentrait pas dans les garanties du contrat). Le remboursement peut soit être total, comme c'est le cas dans l'exemple précédent, soit être partiel. Les remboursements totaux sont facilement repérables et éliminés systématiquement de la base. Par contre, pour les remboursement partiels, la tâche est plus compliquée. Parfois, la date du sinistre diffère de quelques jours entre les deux lignes, et il est alors techniquement difficile de regrouper les lignes relatives au même sinistre. Elles sont donc laissées dans la base, mais représentent une proportion mineure du coût total des sinistres.

1.3 Nature des interventions

1.3.1 Nature des interventions par pièce

La compagnie d'assurance souhaite tarifer son produit de manière distincte entre l'entretien et la réparation. En effet, elle souhaite offrir la possibilité à ses clients (les propriétaires d'hôtels) de souscrire un contrat d'entretien, un contrat de réparation, ou les deux. Il est donc indispensable de pouvoir proposer un tarif pour l'entretien et la réparation séparément.

Pour cela, nous avons vu précédemment que la base des sinistres fournissait l'information relative à la nature de l'intervention : "E" pour de l'entretien, et "R" pour de la réparation. Néanmoins, nous ne savons pas si, au sein d'une même pièce, il n'y a eu que de la réparation, que de l'entretien, ou les deux. Nous allons donc créer 5 variables "nature1" à "nature5" correspondant à la nature globale des interventions sur chaque pièce (entrée, chambre, salle de bain, cuisine, salon). Ainsi, ces variables pourront prendre les valeurs suivantes : "E" (que de l'entretien), "R" (que de la réparation), "ER" (à la fois entretien et réparation).

Par ailleurs, il est important de noter que les données présentent une sérieuse insuffisance au niveau du coût des sinistres, liée à la structure du système d'encodage. En effet, le coût concerne la totalité des interventions sur la suite d'hôtel, sur toutes les pièces et objets confondus. Il est donc impossible d'isoler les coûts. Par conséquent, si la suite d'hôtel a bénéficié d'interventions d'entretien et de réparation, il n'est pas possible de connaître la part du coût total correspondant aux interventions de type entretien et celle correspondant aux interventions de type réparation. Or, c'est là tout l'enjeu de la tarification séparée entre entretien et réparation. Les lignes "hybrides" qui concerneraient

à la fois des interventions d'entretien et de réparation ne seraient donc pas utilisables pour la tarification.

Ces cas sont en nombre non négligeable, c'est pourquoi nous allons mettre en place des techniques pour tenter de contourner cette difficulté, et utiliser quand même ces informations dans l'analyse de coût.

1.3.2 Retraitement des natures hybrides

Le but est donc de pouvoir tout de même utiliser les sinistres avec des interventions de nature hybride.

Reprenons l'exemple précédent (table 1.3 p. 20). La nature des interventions sur la chambre est hybride : "ER". Nous allons donc la ramener à une nature simple, "E" ou "R" de la manière suivante :

- Nous calculons les fréquences des natures des interventions sur la chambre : dans 51% des cas, il s'agit de réparation, dans 44% des cas il s'agit d'entretien, et dans 5% des cas il s'agit d'entretien et réparation (statistiques basées sur l'historique des données) ;
- Nous calculons le ratio suivant :

$$51/(51+44) = 53\%$$

- Nous introduisons une variable aléatoire uniformément distribuée entre 0 et 1, que l'on nomme "alea" ;
- Si $alea \leq 0.53$ alors "ER" sera remplacé par "R". Sinon, il sera remplacé par "E".

Cette méthode est appliquée sur chaque ligne de la base des sinistres, sur chaque pièce où plusieurs interventions de natures différentes ont eu lieu. De cette manière, on se ramène à des natures simples (uniquement "E" ou "R") sur chaque pièce de la suite d'hôtel. Cela permet de redistribuer les coûts aléatoirement, mais en tenant compte, dans cet exemple, de la probabilité plus importante que l'intervention soit de la réparation. Il est important de noter que l'intégralité des coûts est reconstituée (on ne perd aucun montant, on effectue simplement une répartition légèrement biaisée de ceux-ci).

A noter que le retraitement des natures hybrides a également été testé via une utilisation de proportions sur les coûts au lieu des fréquences, et les résultats ont été sensiblement les mêmes. Nous avons choisi d'utiliser finalement les proportions sur les fréquences de manière arbitraire.

1.3.3 Nature globale du sinistre, toutes pièces confondues

Même si nous avons retraité les natures par pièce de manière à n'avoir que des natures simples, la nature globale toutes pièces confondues peut toujours être hybride. Par exemple, si les interventions sur la chambre sont de nature "E" et les interventions sur la cuisine de nature "R", la nature globale toutes pièces confondues sera "ER". Il est donc indispensable de faire le même travail que nous avons fait sur chaque pièce séparément. Nous calculons les fréquences des natures globales toutes pièces confondues ("E", "R" ou "ER"), et nous introduisons une variable uniformément distribuée entre 0 et 1 pour redistribuer les coûts de manière à ne conserver que des natures simples.

1.4 Fusion des sinistres avec les polices

Une fois que les retraitements nécessaires ont été effectués, nous pouvons fusionner la base des sinistres avec celle des polices. Le but est de pouvoir dire, pour chaque police, si elle a eu des sinistres, combien, quand, les caractéristiques de ces sinistres, ... Ainsi, toutes les informations qui seront utilisées pour la modélisation sont réunies en une seule et même base.

La fusion se fait sous contrainte, sur une clé composée du numéro de contrat et de la date de survenance du sinistre. Nous fusionnons donc un sinistre avec sa police correspondante (via le numéro de contrat) et avec son année d'exposition correspondante (via la date du sinistre). Pour rappel, nous avons auparavant retraité les polices afin d'obtenir une ligne par année d'exposition (voir partie 1.2.1 p. 19).

1.5 Agrégation des coûts

Nous cherchons à évaluer le coût moyen pour l'entretien/réparation d'une suite d'hôtel **sur la durée totale du contrat**. Ainsi, à chaque nouvelle intervention, il faut aussi prendre en compte toutes les interventions passées, c'est à dire qu'il est nécessaire de cumuler les coûts au cours du temps. Pour illustrer cette notion, prenons l'exemple d'une suite d'hôtel qui a bénéficié des interventions suivantes (table 1.5 p. 23) :

Année	Date d'intervention	Nature	Coût	Coût E	Coût E cumulé
2007	.	.	0	0	0
2008	12Déc	E	36	36	36
2009	16Nov	E	172	172	208
2010	29Nov	E	39	39	247
2011	18Fév	R	199	0	247
2011	30Juin	R	29	0	247
2011	21Nov	E	353	353	600
2012	.	.	0	0	600

TABLE 1.5 – Aggrégation des coûts totaux en Entretien

Au lieu de considérer une seule suite d'hôtel avec 8 interventions indépendantes et des coûts correspondant à la colonne "Coût E", nous allons considérer 8 suites virtuelles différentes (i.e. correspondant à l'évolution au cours du temps de la même suite). Dans cette approche, les coûts sont cumulatifs au cours du temps, c'est pourquoi nous utiliserons la colonne "Coût E cumulé" pour modéliser le coût.

Nous cumulons les coûts aussi bien pour l'entretien que pour la réparation, selon le même processus.

1.6 Remise en classes des variables explicatives

Les variables potentiellement explicatives peuvent être continues, ou discrètes avec de nombreuses valeurs possibles. Par exemple, la durée du contrat d'assurance qui s'exprime en mois peut prendre plusieurs valeurs différentes. Il faut donc les réorganiser en classes de risques homogènes, de façon à disposer d'informations suffisamment nombreuses par

classe pour tarifer le produit. Par exemple, nous pouvons regrouper les durées de contrat par tranches de 6 mois.

Le choix des classes se fait sur jugement de l'utilisateur. Il faut essayer plusieurs solutions différentes de manière itérative jusqu'à obtenir des classes cohérentes avec suffisamment de données à l'intérieur. Si les volumes ne sont pas assez importants, le MLG ne sera pas en mesure de donner des estimations solides, et les intervalles de confiance seront trop grands pour conclure à un tarif avec un minimum de confort. En effet, SAS utilise un algorithme de type Newton-Raphson, et une trop grande quantité de modalités empêcherait l'algorithme de converger vers une solution.

Il est important de noter qu'une fois les variables remises en classes, elles sont toutes considérées comme alphanumériques du point de vue du MLG. Il cherche alors à mettre en évidence des coïncidences entre les classes de modalités que l'on a définies, sans regarder précisément ce que contiennent les variables. Ce qui importe, c'est de trouver des liens entre les classes constituées et d'en extraire une règle générale.

Cette étape n'est pas nécessaire pour la modélisation à l'aide de l'algorithme CART (que nous aborderons en partie III p. 57). Toutefois, pour comparer les deux méthodes équitablement, les données seront également déjà reclassées pour l'application de CART.

1.7 Introduction d'interactions

Il est possible de prendre en compte des éventuelles interactions entre les variables. Cela se produit lorsque l'effet d'une variable varie non linéairement en fonction du niveau d'une autre variable. Par exemple, considérons deux variables A et B, ayant respectivement deux et cinq niveaux ($a1$ et $a2$; $b1$, $b2$, $b3$, $b4$ et $b5$). Pour capturer l'interaction entre ces deux variables, on crée une troisième variable notée AB ayant pour niveaux les différentes combinaisons des niveaux de A et B, soit 10 niveaux possibles :

$a1\&b1$	$a2\&b1$
$a1\&b2$	$a2\&b2$
$a1\&b3$	$a2\&b3$
$a1\&b4$	$a2\&b4$
$a1\&b5$	$a2\&b5$

Dans notre analyse de coût, nous avons choisi d'utiliser une interaction entre deux variables : Catégorie de la suite (2 niveaux : Double, Simple) et Surface (6 niveaux). Nous avons donc créé une variable notée "Catsurf" qui prend les valeurs suivantes :

- D40 : double et superficie inférieure ou égale à 40 m^2
- D50 : double et superficie entre 40 et 50 m^2
- D60 : double et superficie entre 50 et 60 m^2
- D70 : double et superficie entre 60 et 70 m^2
- D80 : double et superficie entre 70 et 80 m^2
- D90 : double et superficie supérieure à 80 m^2
- S40 : simple et superficie inférieure ou égale à 40 m^2
- S50 : simple et superficie entre 40 et 50 m^2
- Sothers : regroupe les suites simples avec superficie supérieure à 50 m^2 (pas assez de données)

Cela permet de raffiner le modèle et d'augmenter en précision de prédiction.

1.8 Analyse descriptive univariée

Dans cette étape, nous réalisons plusieurs graphiques représentant l'évolution du coût moyen des interventions en fonction de chaque variable potentiellement explicative. De plus, sur ces mêmes graphiques, nous représentons le volume de données disponibles sur chaque niveau de la variable considérée (les niveaux sont ceux définis lors de la remise en classe), c'est-à-dire l'exposition au risque.

Bien que le but final de ce projet soit de modéliser les coûts selon une méthode multivariée, il peut être intéressant de réaliser une analyse univariée pour plusieurs raisons. Premièrement, cela permet d'indiquer si une variable explicative contient assez d'information pour être incluse dans le modèle. Par exemple, si 99.5% du volume de données sont concentrés sur une seule valeur de la variable, cela suggère que la variable ne sera pas vraiment significative dans le modèle.

Par ailleurs, ces graphiques nous permettent de détecter une éventuelle insuffisance de données sur un niveau d'une variable, ce qui permet de revenir à l'étape précédente de la remise en classe (partie 1.6 p. 23), et de réorganiser les données de manière plus adaptée.

Enfin, une analyse univariée permet d'avoir un premier aperçu de l'effet sur le coût moyen de chaque facteur pris séparément (effet non décorrélé des autres variables).

Les graphiques obtenus (coût moyen en Entretien et Réparation, et expositions par niveaux pour chaque variable) sont les suivants (figures 1.1 p. 25 à 1.10 p. 30) :

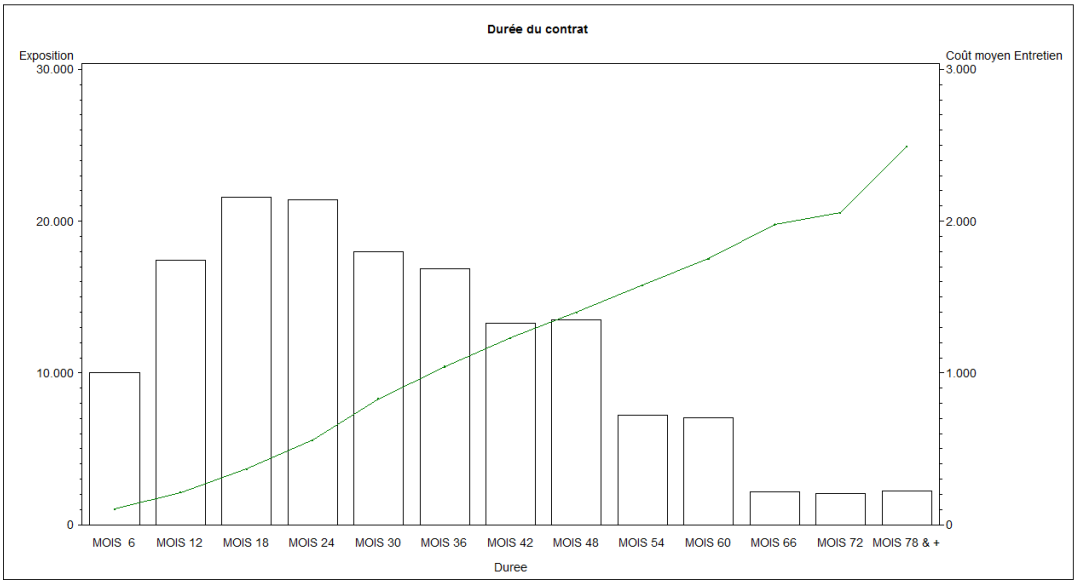

FIGURE 1.1 – Analyse univariée : Coût moyen Entretien et exposition - Durée du contrat d'assurance

On observe une croissance quasiment linéaire du coût moyen en Entretien en fonction de la durée du contrat.

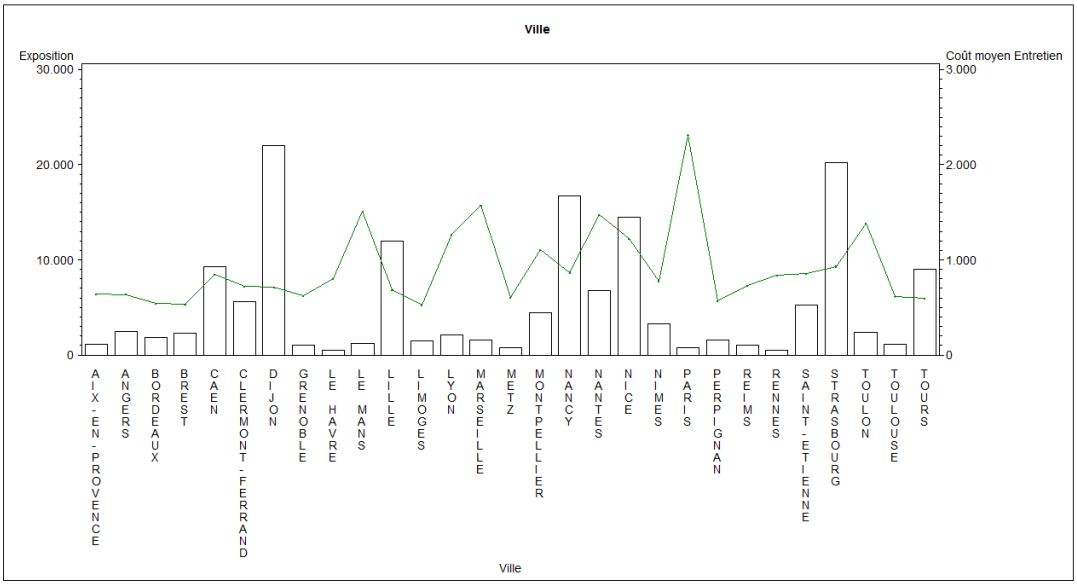

FIGURE 1.2 – Analyse univariée : Coût moyen Entretien et exposition - Localité de la suite d'hôtel

Il peut y avoir de fortes disparités entre les coûts moyens d'Entretien selon la ville dans laquelle se situe la suite, pouvant aller du simple au quadruple.

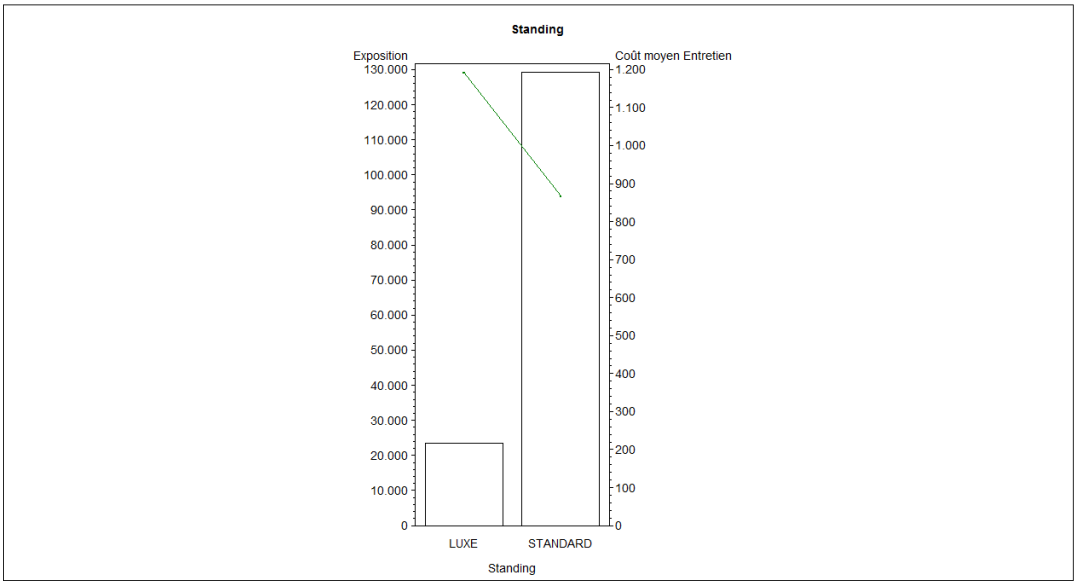

FIGURE 1.3 – Analyse univariée : Coût moyen Entretien et exposition - Standing de la suite d'hôtel

Le coût moyen en Entretien pour les suites de luxe est plus élevé que celui pour les suite standard, ce qui nous semble cohérent.

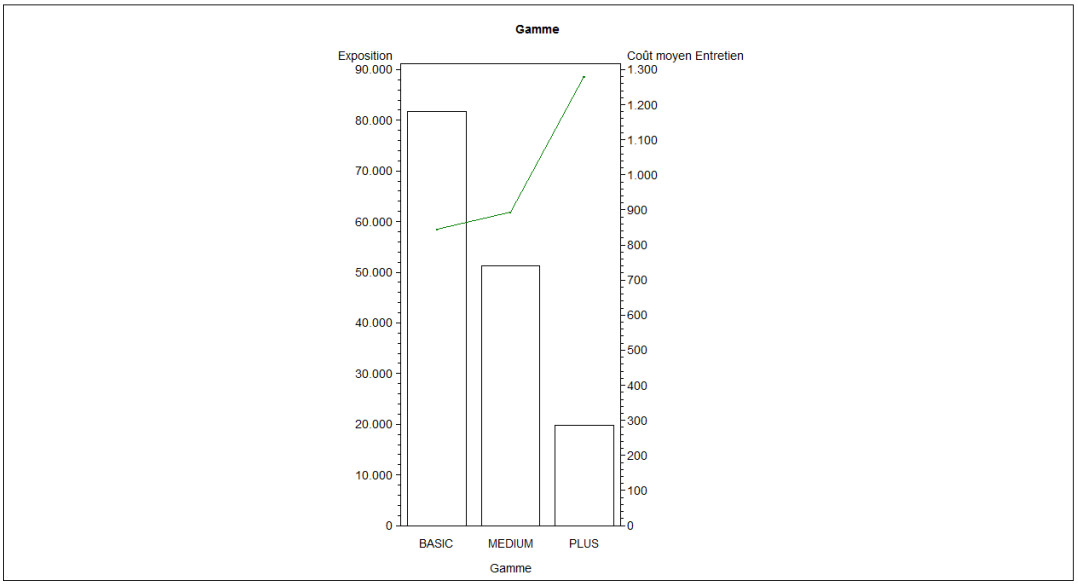

FIGURE 1.4 – Analyse univariée : Coût moyen Entretien et exposition - Gamme du produit d'assurance

Le coût moyen en Entretien est nettement plus élevé pour la garantie Plus, tandis que les garanties Basic et Medium sont plus ou moins au même niveau de prix.

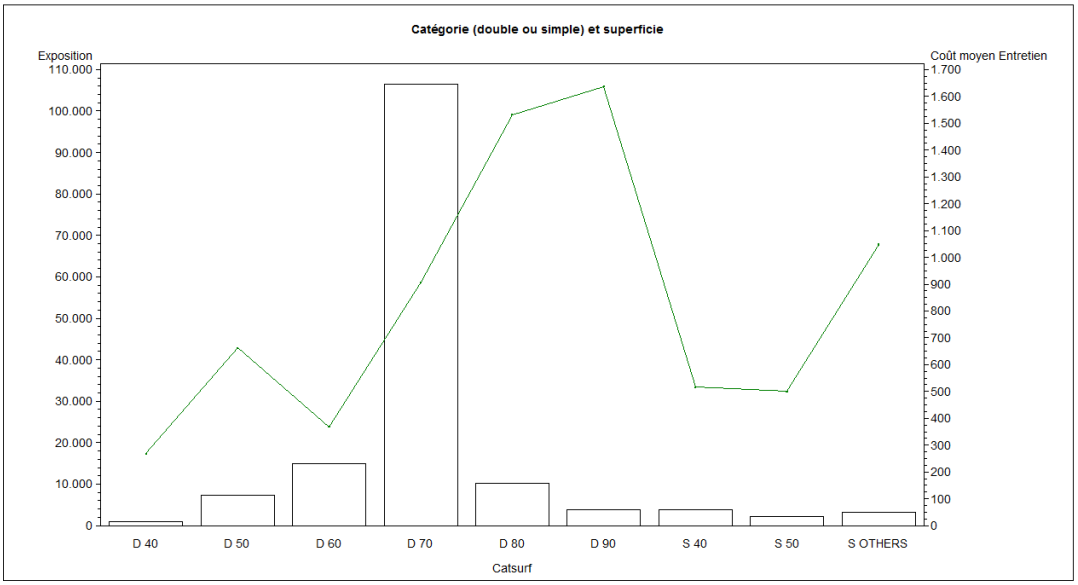

FIGURE 1.5 – Analyse univariée : Coût moyen Entretien et exposition - Catégorie et superficie de la suite d'hôtel

Le coût moyen en Entretien est globalement croissant avec la superficie de la suite.

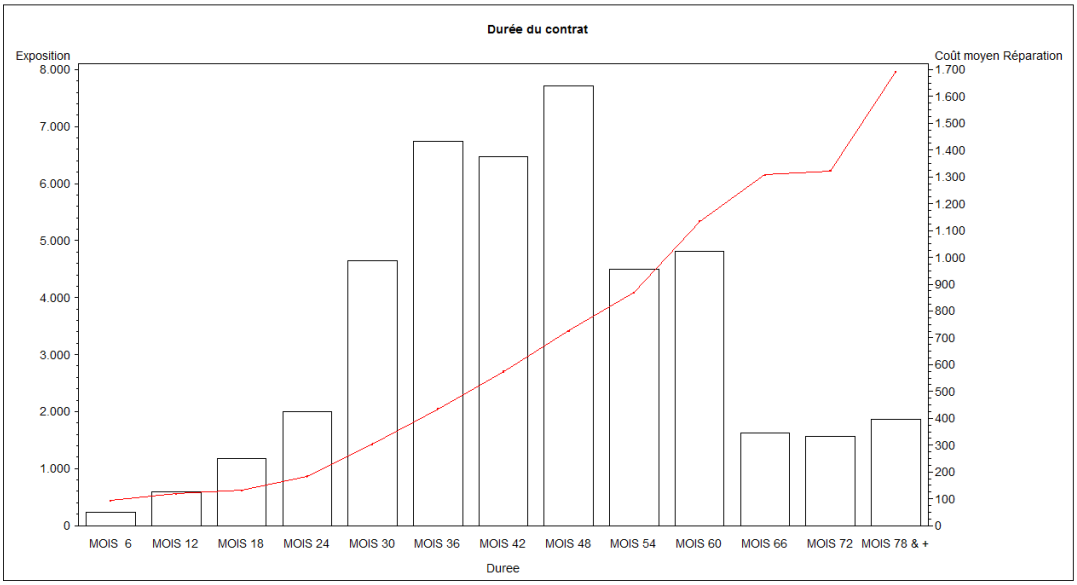

FIGURE 1.6 – Analyse univariée : Coût moyen Réparation et exposition - Durée du contrat d'assurance

Contrairement à l'Entretien, le coût moyen en Réparation a une croissance moins linéaire (faible croissance pour les courtes durées de contrat, puis forte augmentation à partir de 24 mois).

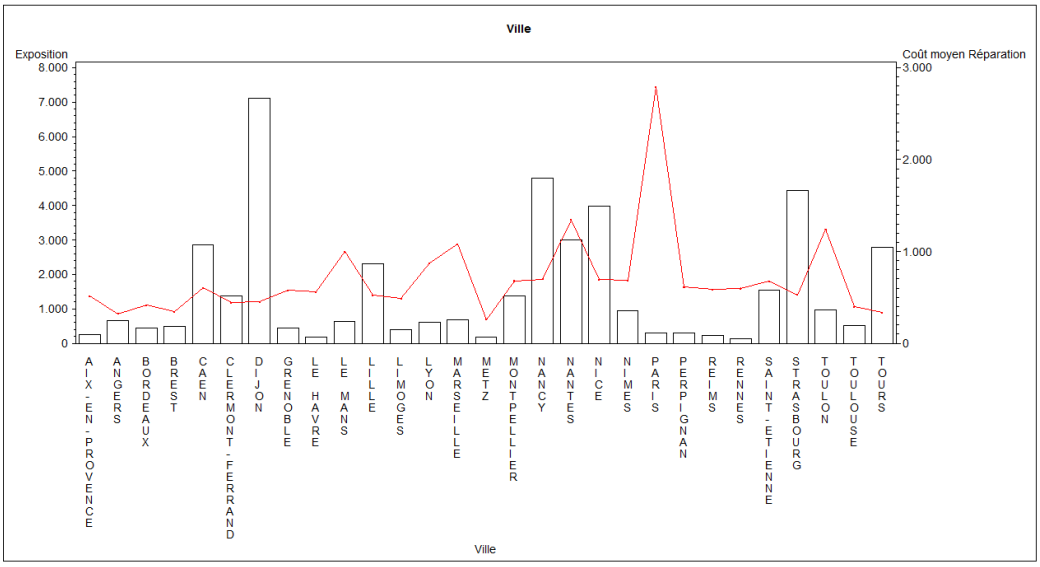

FIGURE 1.7 – Analyse univariée : Coût moyen Réparation et exposition - Localité de la suite d'hôtel

Les différences de coût selon la ville sont encore plus prononcées en Réparation qu'en Entretien. Par exemple, le coût moyen à Paris est plus de 6 fois plus cher qu'à Metz.

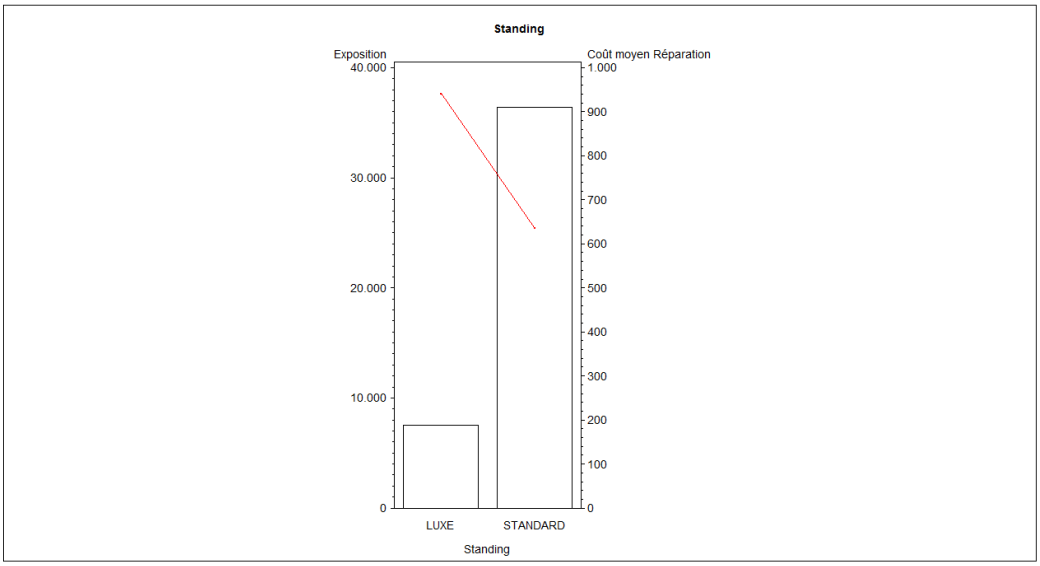

FIGURE 1.8 – Analyse univariée : Coût moyen Réparation et exposition - Standing de la suite d'hôtel

Nous pouvons faire le même constat que pour l'Entretien : le coût moyen en Réparation est plus élevé pour les suites de luxe que pour les suites standard, dans des proportions équivalentes à celles de l'Entretien (environ 1.5 fois). Cela est conforme au bon sens : le mobilier est plus cher s'il doit être remplacé, les matériaux sont plus nobles, etc.

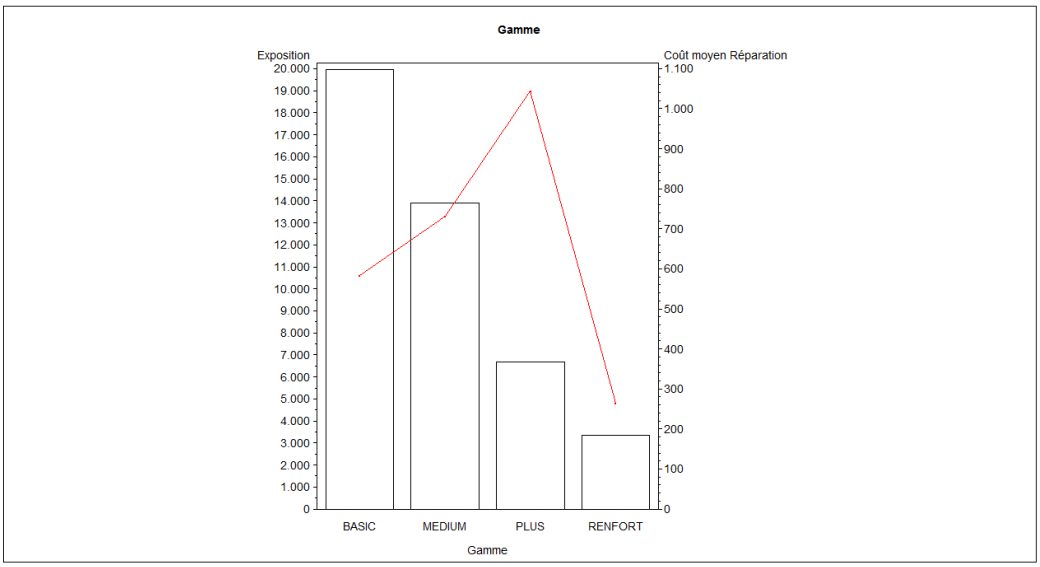

FIGURE 1.9 – Analyse univariée : Coût moyen Réparation et exposition - Gamme du produit d'assurance

Le coût moyen en Réparation est presque linéairement croissant sur les gammes Basic, Medium et Plus, ce qui nous semble cohérent puisque les garanties sont également

croissantes. Pour la gamme Renfort, uniquement disponible pour les contrats d'assurance réparation, le coût est bien plus bas. Cela n'est pas absurde puisqu'il s'agit d'un renfort généralement souscrit en complément d'une autre gamme.

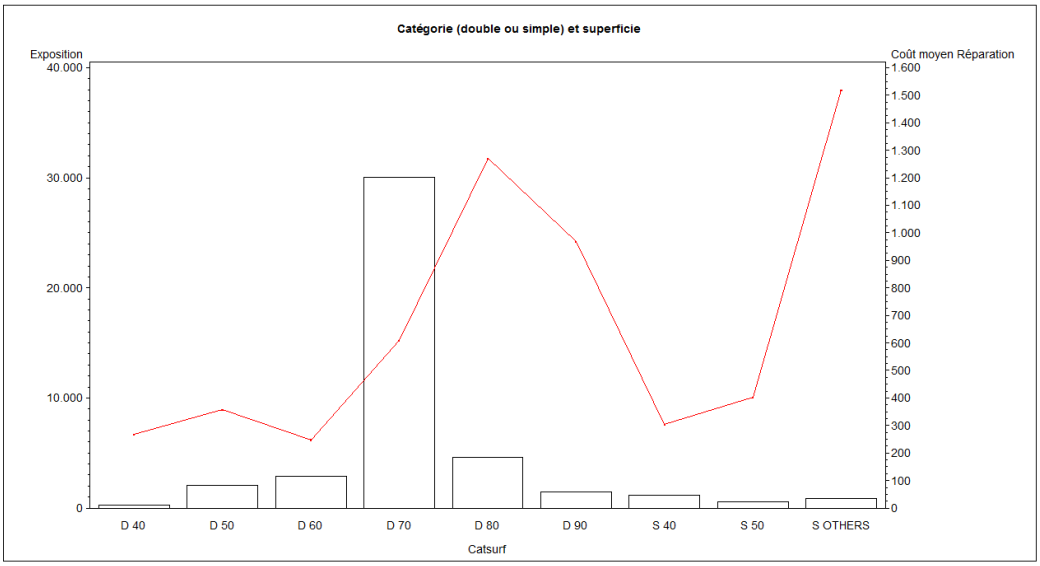

FIGURE 1.10 – Analyse univariée : Coût moyen Réparation et exposition - Catégorie et superficie de la suite d'hôtel

Le coût moyen en Réparation est globalement croissant avec la superficie de la suite.

Que ce soit en Entretien ou en Réparation, les expositions au risque nous semblent suffisamment élevées pour chaque niveau de chaque variable. Toutefois, il convient d'apporter plus d'attention aux niveaux bénéficiant de gros volumes de données.

Nous pouvons également noter que les expositions au risque sont bien plus élevées en Entretien, ce qui signifie que les sinistres comptabilisés concernent en majorité de l'Entretien, et peu de Réparation.

1.9 Partitionnement de la base de données

La base de données finale est divisée en deux parties, de manière aléatoire :

- une base d'**apprentissage**, représentant 75% des données
- une base de **test**, représentant 25% des données

Les modèles (MLG et CART) vont être construits sur la base d'apprentissage uniquement. Etant donné que nous avons la chance de disposer d'un grand nombre d'observations, la base d'apprentissage reste un nombre suffisant d'informations pour la modélisation.

Ensuite, la base de test servira de base impartiale pour comparer les performances des deux modèles (voir partie IV p. 79). En effet, cette base n'ayant pas servi à la construction des modèles, elle permet de simuler l'ajout de nouvelles données.

Deuxième partie

Analyse de coût par Modèle Linéaire Généralisé

2.1 Notions théoriques

Le Modèle Linéaire Généralisé (MLG) est un outil statistique composé de 3 éléments :

- une composante aléatoire : une variable à expliquer, dont la densité appartient à la famille exponentielle
- une composante déterministe : on dispose des valeurs d'une ou plusieurs variables explicatives décrivant la variable à expliquer
- une fonction lien permettant de relier la moyenne de la variable à expliquer aux variables explicatives.

Notons Y la variable à expliquer, X_1 à X_p les p variables explicatives, g la fonction lien, et β_1 à β_p les coefficients de la régression. Le MLG s'exprime alors de la manière suivante :

$$g(E[Y]) = \beta_0 + \beta_1 X_1 + \ldots + \beta_p X_p$$

La densité f de Y doit appartenir à la famille exponentielle, i.e. elle peut s'écrire sous la forme :

$$f(y, \Theta, \Phi) = \exp(\frac{y\Theta - b(\Theta)}{a(\Phi)} + c(y, \Phi)) \ \forall y \in S$$

où :
- le support S est un sous-ensemble de $\mathbb{N}$ ou de $\mathbb{R}$
- $\Theta \in \mathbb{R}$ est le paramètre canonique (ou paramètre de la moyenne)
- $\Phi \in \mathbb{R}$ est le paramètre de dispersion
- a est une fonction définie sur $\mathbb{R}$ et non nulle
- b est une fonction définie sur $\mathbb{R}$ et 2 fois dérivable
- c est une fonction définie sur $\mathbb{R}^2$.

2.2 Choix de modélisation

Un Modèle Linéaire Généralisé va être appliqué à chacune des deux natures d'intervention : Entretien et Réparation. Pour cela, on utilise la procédure *Genmod* sous SAS, dont la syntaxe minimale est la suivante :

```
Proc Genmod data = [base de données finale];
Class [variables explicatives séparées d'un espace];
Model [variable à expliquer] = [variables explicatives]
/ link = [fonction lien] dist = [distribution de la variable à expliquer];
Run;
```

La base de données finale est la base d'apprentissage (cf. partie 1.9 p. 30) qui contient à la fois les informations sur les sinistres et sur les polices. Les variables explicatives sont à choisir parmi les variables disponibles, selon des critères que nous définirons dans la section suivante (partie 2.3 p. 35). La variable à expliquer est le coût cumulé résultant du processus d'agrégation des coûts abordé en partie 1.5 p. 23. On utilisera le coût cumulé pour l'Entretien ou la Réparation selon le modèle. La densité du coût moyen est

supposée de loi *Gamma*, généralement utilisée en tarification non-vie (au même titre que la log-normale). La fonction lien choisie est la fonction ln, car elle permet de passer d'un modèle additif à un modèle multiplicatif.

2.3 Choix des variables explicatives

Les différents critères de choix énumérés ci-dessous sont à considérer de manière globale, en ce sens où un critère non satisfait ne doit pas mener à l'élimination systématique de la variable qui fait défaut. Les variables ne doivent pas forcément être toutes en accord avec les différents critères considérés séparément.

Dans les sections qui suivent (parties 2.3.1 p. 35 à 2.3.2 p. 41), les résultats graphiques ne seront présentés que pour les variables finalement retenues (bien que les études aient été réalisées sur beaucoup plus de variables potentiellement explicatives, de manière à choisir justement lesquelles seraient les plus pertinentes). Les variables explicatives retenues sont les suivantes :

- La durée du contrat d'assurance (par tranches de 6 mois, centrées en 6, 12, 18, 24 etc.)
- La localité de la suite d'hôtel (Paris, Lyon, Marseille, etc.)
- Le standing de la suite d'hôtel (Standard ou Luxe)
- La gamme du produit d'assurance (Basic, Medium ou Plus, et la garantie Renfort disponible uniquement sur les contrats de réparation)
- Une variable d'interaction (voir partie 1.7 p. 24) entre la catégorie de la suite (Double ou Simple) et sa superficie (entre 0 et 40 m^2, entre 40 et 50 m^2, etc.)

2.3.1 Analyse des sorties du MLG

La procédure *Genmod* fournit plusieurs sorties. Un tableau récapitule, pour chaque niveau de chaque variable, le coefficient estimé associé, ses intervalles de confiance, et la p-valeur. Cette dernière permet de conclure si l'influence du niveau de la variable est significativement non nulle. On considère qu'une p-valeur inférieure à 5% est synonyme de significativité du niveau de la variable. A noter que pour certains niveaux, la p-valeur est légèrement supérieure à 5%, sans pour autant que la variable ne soit pas significative. En effet, dans ces cas-là, l'estimation est souvent très proche de 0, ce qui signifie que le niveau de la variable n'est pas significativement différent de la référence, mais cela ne veut pas dire pour autant qu'il n'est pas significatif du tout pour le modèle.

Les sorties des deux procédures *Genmod* (Entretien et Réparation) se trouvent en annexe.

En parallèle des informations fournies par la procédure *Genmod*, il peut être intéressant de considérer une approche graphique complémentaire. Nous traçons donc un graphique pour chaque variable explicative, représentant l'évolution des multiplicateurs en fonction des niveaux de la variable. Le multiplicateur est obtenu en appliquant l'inverse de la fonction lien à l'estimateur. Dans notre cas, le multiplicateur est donc l'exponentielle de l'estimateur fourni par le MLG. Ces graphiques nous permettent d'apprécier la cohérence et la pertinence des potentielles variables explicatives.

Les graphiques des variables retenues sont les suivants (figures 2.11 p. 36 à 2.20 p. 41).

La courbe rouge représente les valeurs prises par les multiplicateurs, et les courbes bleues sont les intervalles de confiance à 95% pour ces valeurs.

Nous avons également tracé les expositions déjà observées lors des analyses univariées. Néanmoins, les expositions sur les graphiques des MLG sont plus basses, car le modèle a été appliqué sur la base d'apprentissage uniquement, alors que l'analyse univariée a été effectuée sur la base entière, avant partition.

Sur chaque graphique, il est important de vérifier 3 critères :
- on observe une certaine tendance sur le graphique, pour les variables ordonnées telles que la durée du contrat
- l'intervalle de confiance est suffisamment étroit
- l'amplitude du graphique doit être assez grande, et deux niveaux doivent avoir des multiplicateurs suffisamment différents

Cette approche graphique nous permet également d'avoir une représentation plus parlante des résultats (visualisation d'une évolution pour les variables ordonnées comme la durée du contrat).

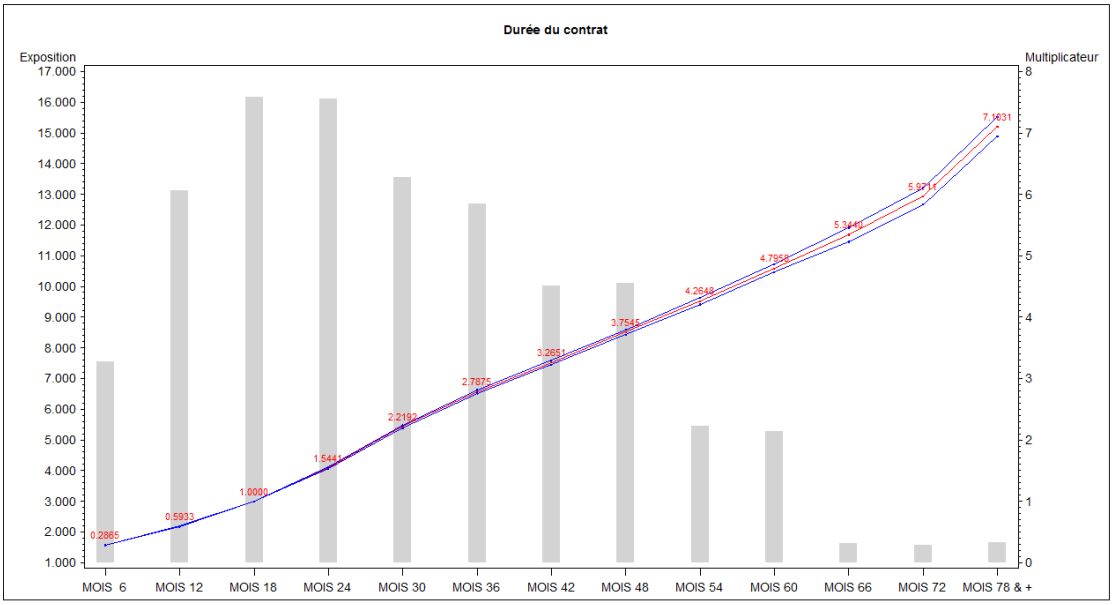

FIGURE 2.11 – MLG Entretien : Multiplicateurs en fonction de la durée du contrat d'assurance

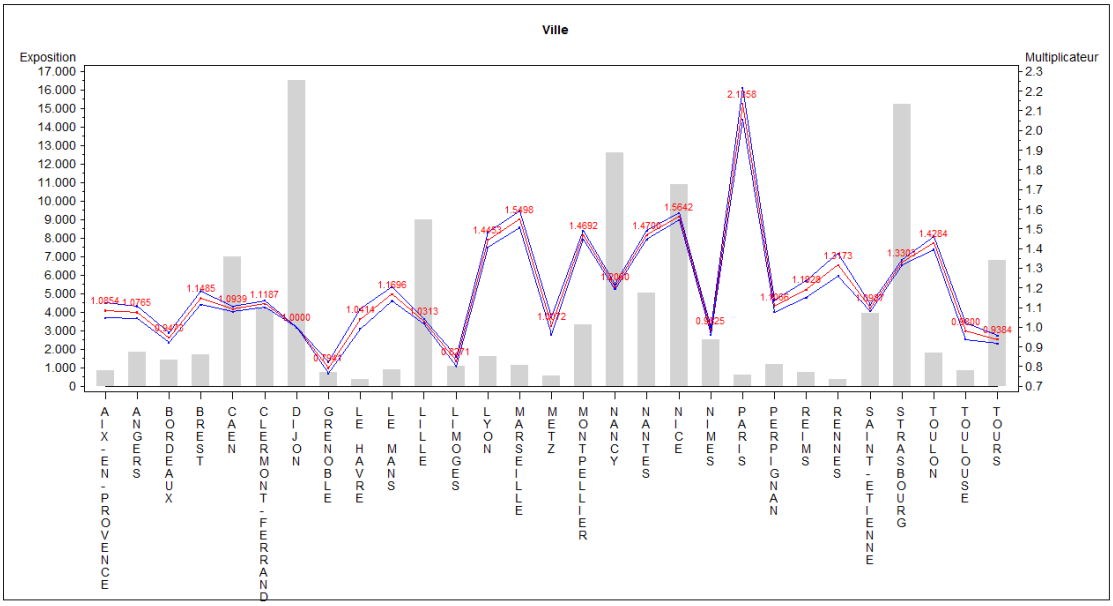

FIGURE 2.12 – MLG Entretien : Multiplicateurs en fonction de la localité de la suite d'hôtel

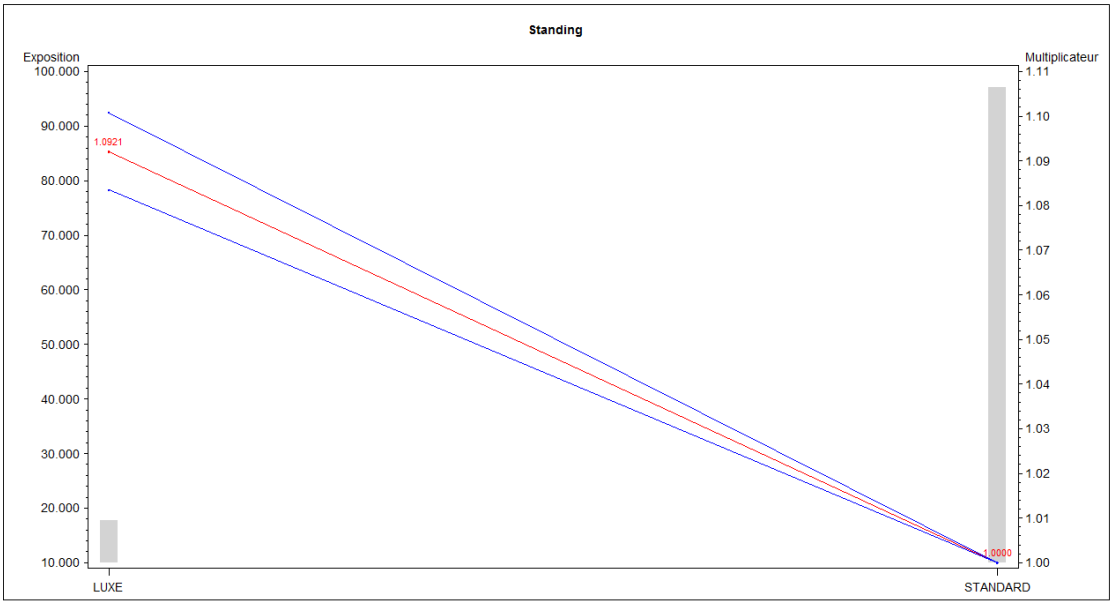

FIGURE 2.13 – MLG Entretien : Multiplicateurs en fonction du standing de la suite d'hôtel

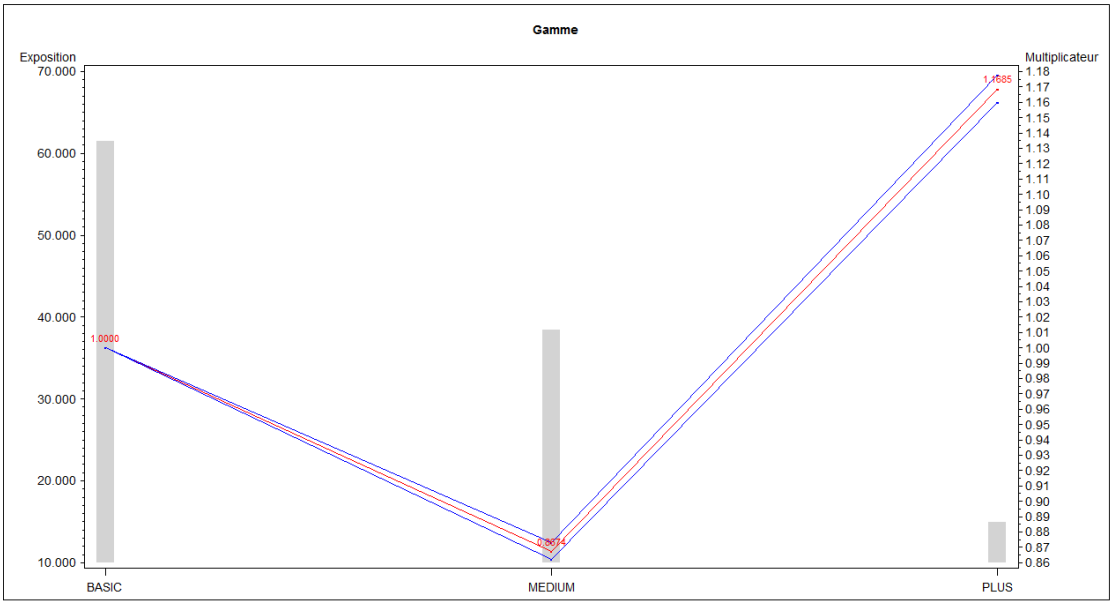

FIGURE 2.14 – MLG Entretien : Multiplicateurs en fonction de la gamme du produit d'assurance

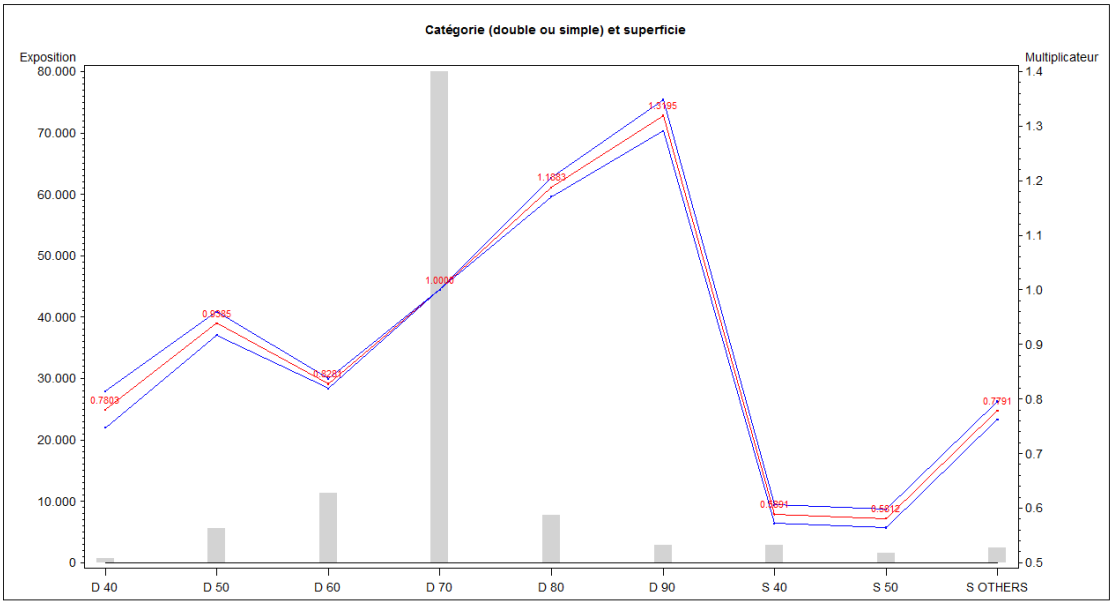

FIGURE 2.15 – MLG Entretien : Multiplicateurs en fonction de la catégorie et de la superficie de la suite d'hôtel

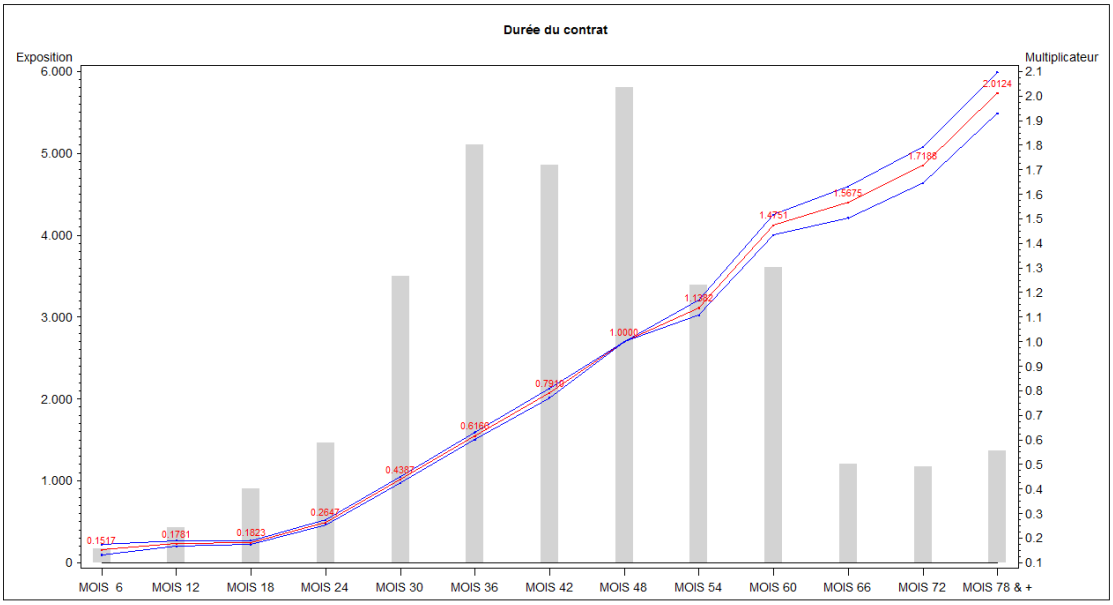

FIGURE 2.16 – MLG Réparation : Multiplicateurs en fonction de la durée du contrat d'assurance

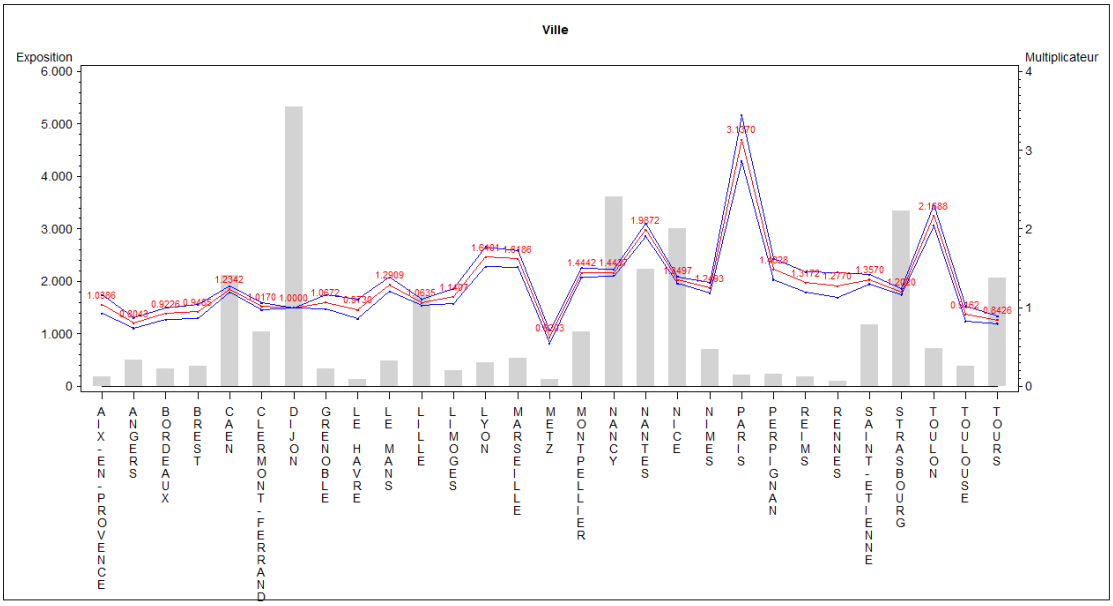

FIGURE 2.17 – MLG Réparation : Multiplicateurs en fonction de la localité de la suite d'hôtel

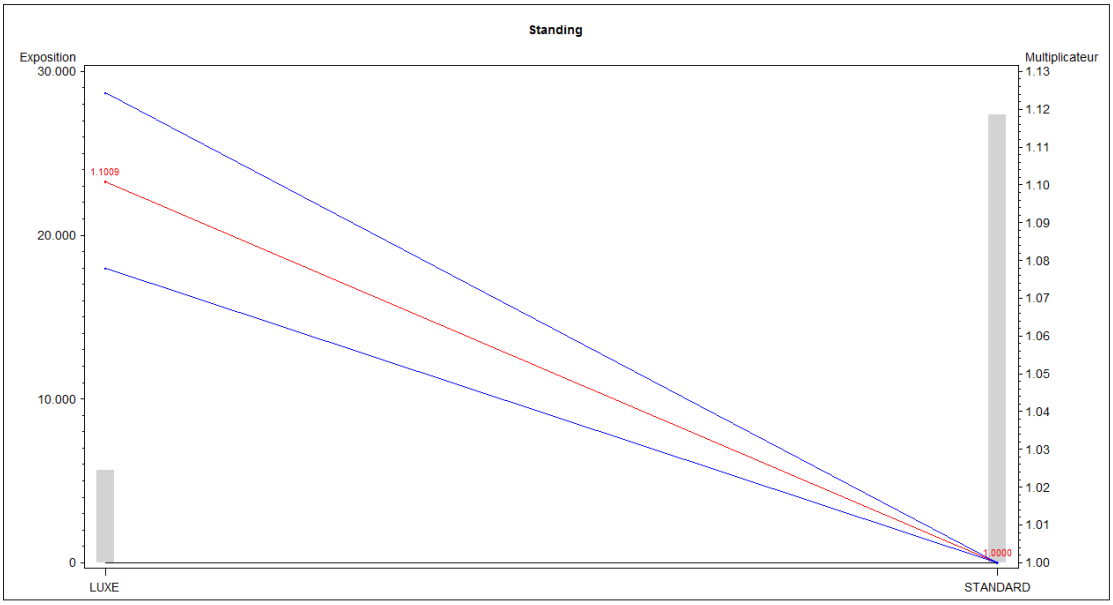

FIGURE 2.18 – MLG Réparation : Multiplicateurs en fonction du standing de la suite d'hôtel

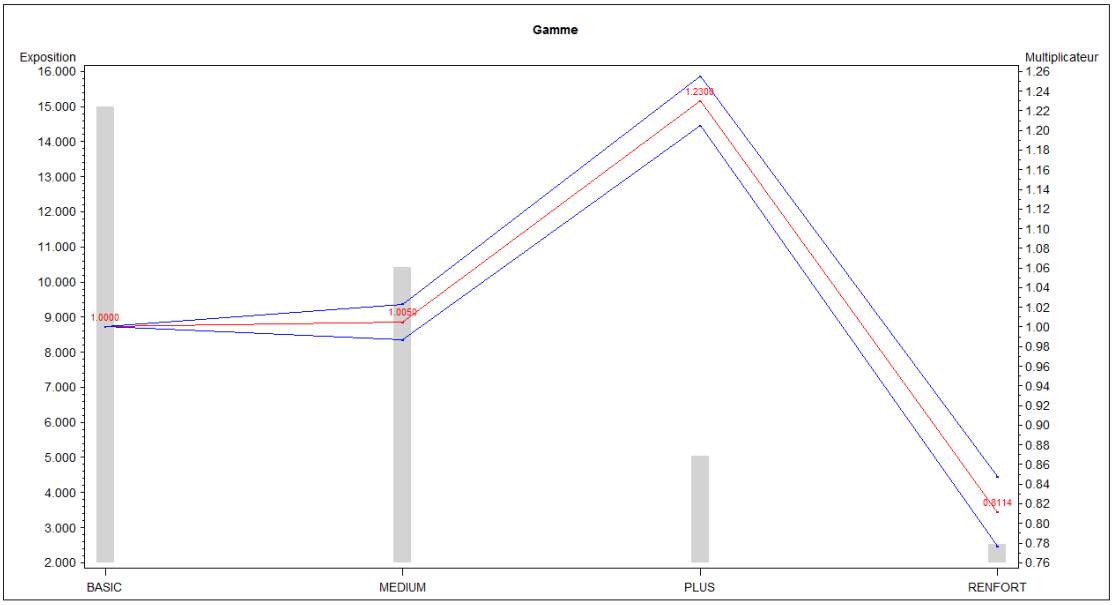

FIGURE 2.19 – MLG Réparation : Multiplicateurs en fonction de la gamme du produit d'assurance

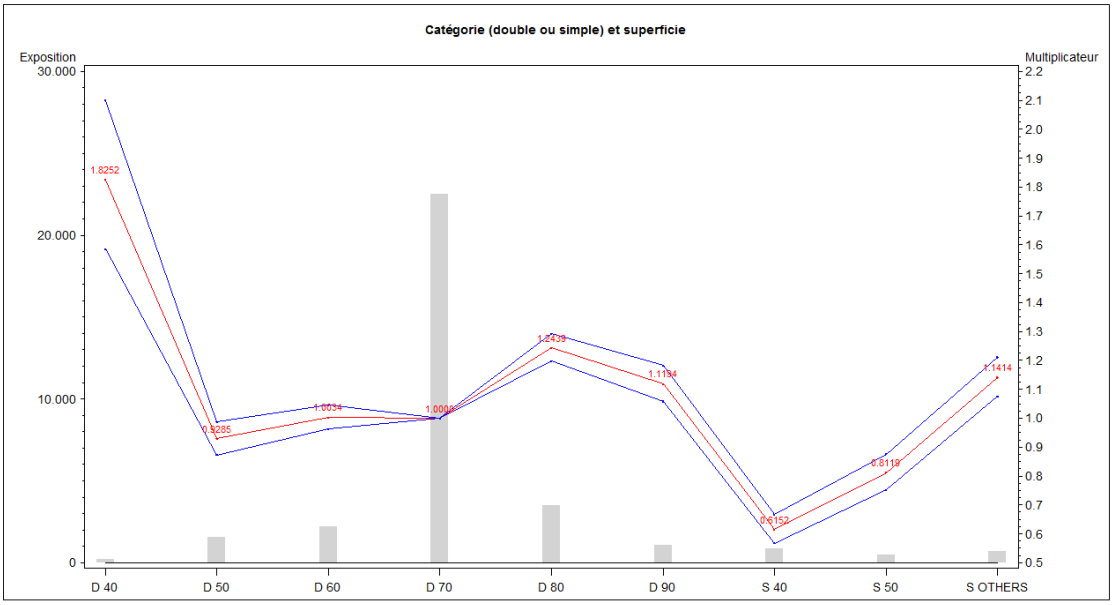

FIGURE 2.20 – MLG Réparation : Multiplicateurs en fonction de la catégorie et de la superficie de la suite d'hôtel

2.3.2 Bon sens

Enfin, le choix des variables explicatives s'effectue aussi par bon sens, de manière tout à fait subjective. L'utilisateur sait quelles variables seraient a priori pertinentes pour sa tarification, et quelles variables n'auraient aucun sens. Par exemple, il est évident que la couleur des murs dans la suite d'hôtel n'entrera pas en compte dans l'analyse des coûts.

2.4 Confrontation des résultats univariés et multivariés

Il peut être intéressant de comparer les résultats selon que l'on considère une approche univariée ou multivariée.

Prenons l'exemple d'une assurance tous risques automobile, où l'on identifie deux facteurs de risque : l'âge du conducteur et celui du véhicule.

En analyse univariée, nous obtiendrions probablement des tarifs plus élevés pour un jeune conducteur par rapport à un conducteur expérimenté. De la même manière, le tarif serait plus élevé pour un véhicule ancien que pour un véhicule neuf (car plus de risques de défaillances techniques).

Ainsi, le risque serait pris en compte deux fois, sans tenir compte d'un effet de diversification dû à la corrélation existant entre ces deux facteurs de risque. Le tarif serait doublement pénalisant pour un conducteur de 18 ans possédant une voiture roulant depuis 20 ans. En effet, si ce dernier a beaucoup d'accidents, c'est peut-être simplement dû à sa voiture qui est en mauvais état, et pas à son manque d'expérience (ou alors dans une moindre mesure), ou bien l'inverse.

Les méthodes multivariées sont justement intéressantes dans ce contexte, car elles permettent de décorréler les facteurs de risque, et d'isoler l'effet produit par chaque facteur pris séparément.

Nous allons donc représenter sur un même graphique les multiplicateurs en univarié et multivarié, pour chacune des 5 variables explicatives : durée du contrat, ville, standing, gamme, catégorie-superficie.

Les graphiques générés lors des analyses univariées (voir partie 1.8 p. 25) représentaient le coût moyen observé sur la base toute entière. Pour comparer des quantités comparables, nous avons refait l'analyse univariée sur la base d'apprentissage uniquement (les résultats ont la même structure, mais les expositions sont plus faibles, et concordantes avec celles des graphiques des MLG, eux aussi effectués sur la base d'apprentissage).

Nous avons également divisé les coûts obtenus par le coût de la référence, pour chaque variable, afin d'obtenir des multiplicateurs (la référence porte ainsi le multiplicateur 1 comme dans les graphiques des MLG).

Nous obtenons les graphiques de comparaison suivants, pour l'Entretien (figures 2.21 p. 42 à 2.25 p. 44) et la Réparation (figures 2.26 p. 45 à 2.30 p. 47). La courbe rouge représente le coût moyen en analyse multivariée, avec les intervalles de confiance issus du MLG en bleu, et la courbe verte représente le coût moyen en analyse univariée. Nous avons également représenté sous forme de barres verticales les expositions au risque pour chaque modalité.

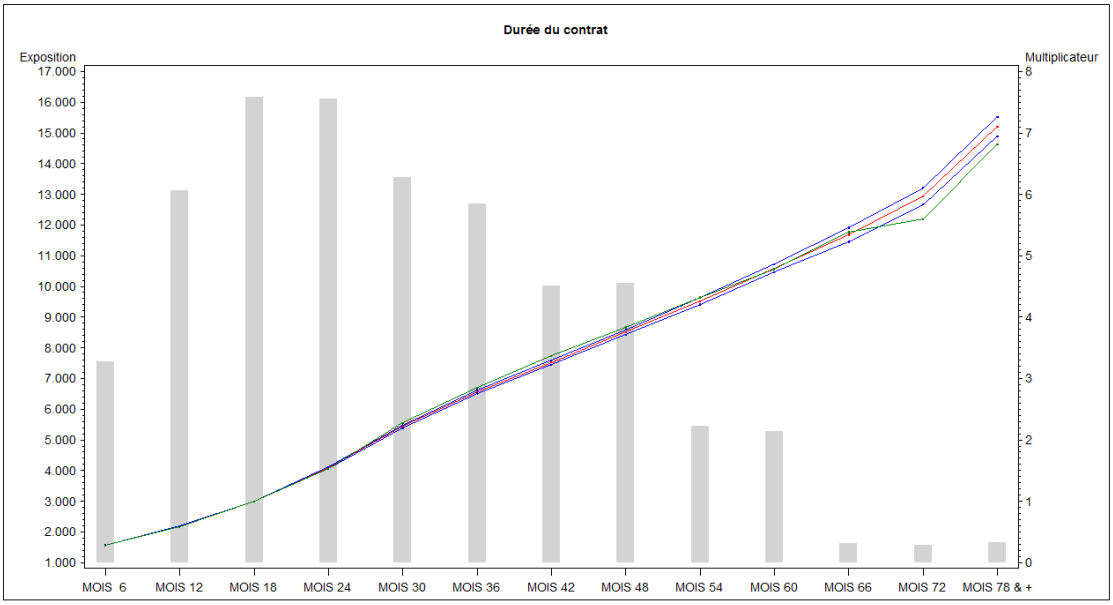

FIGURE 2.21 – Comparaison univarié/multivarié : Coût moyen Entretien en fonction de la durée du contrat d'assurance

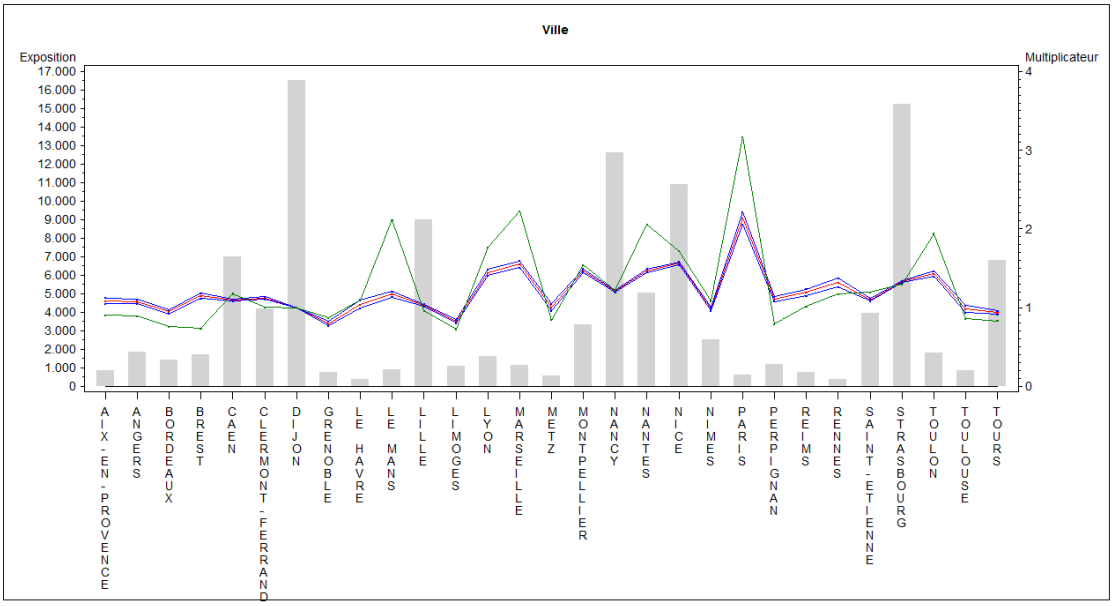

FIGURE 2.22 – Comparaison univarié/multivarié : Coût moyen Entretien en fonction de la localité de la suite d'hôtel

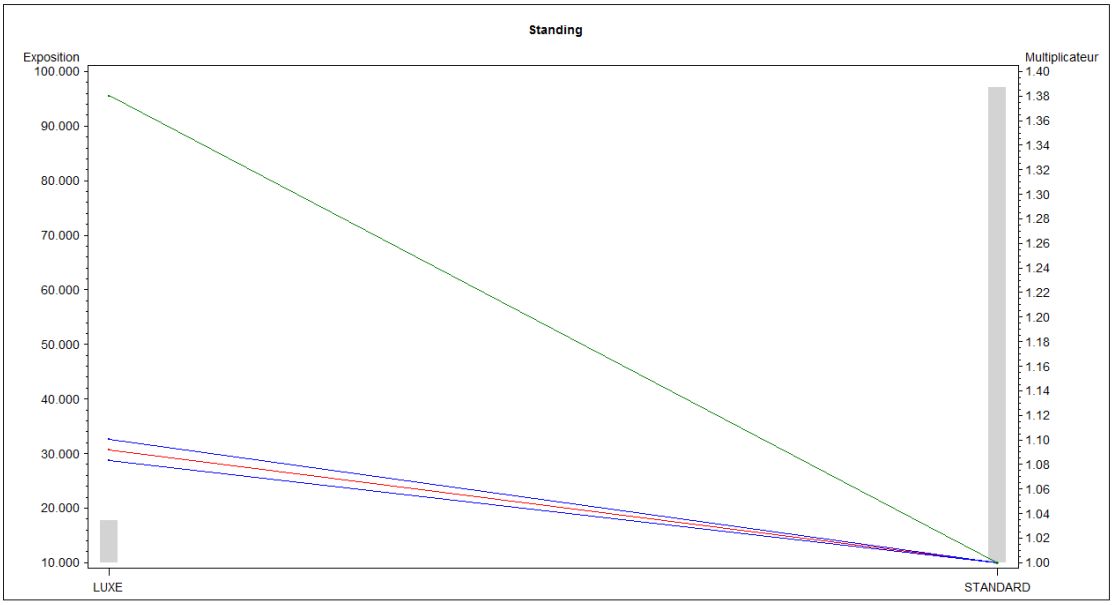

FIGURE 2.23 – Comparaison univarié/multivarié : Coût moyen Entretien en fonction du standing de la suite d'hôtel

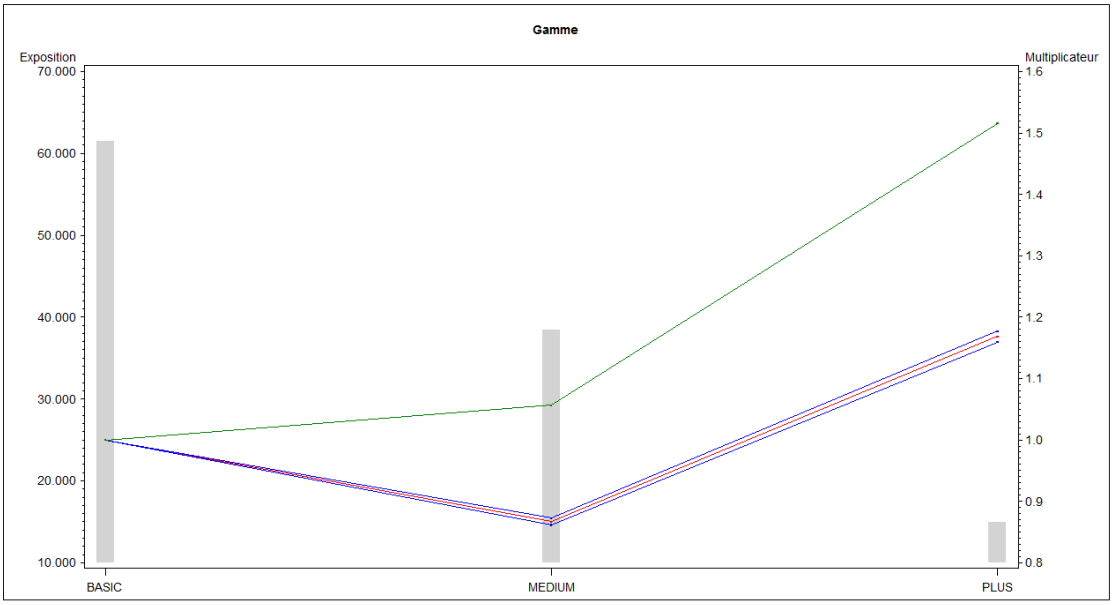

FIGURE 2.24 – Comparaison univarié/multivarié : Coût moyen Entretien en fonction de la gamme du produit d'assurance

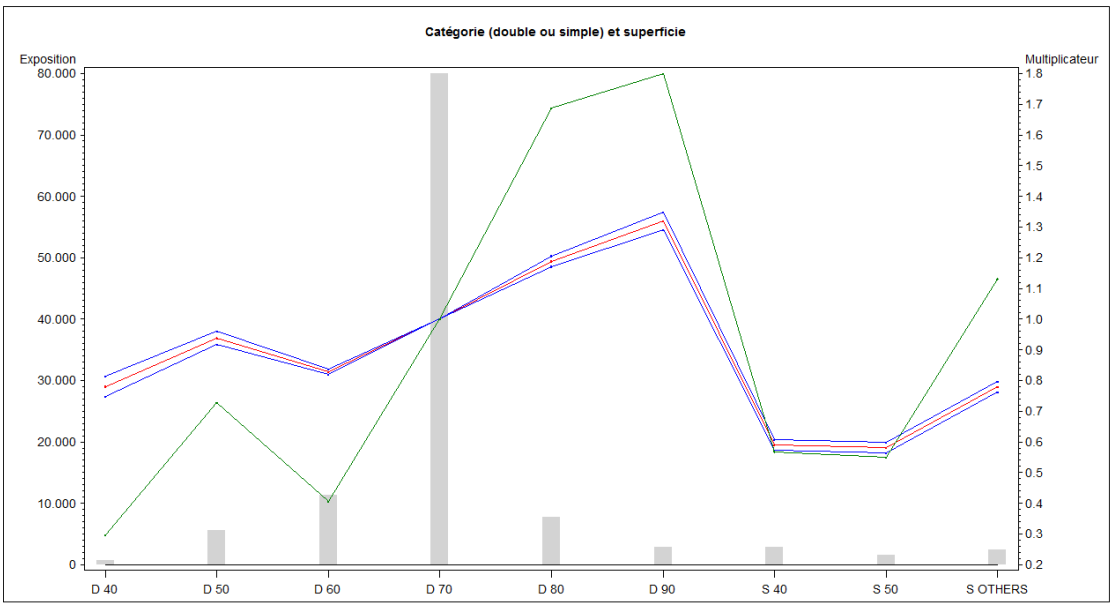

FIGURE 2.25 – Comparaison univarié/multivarié : Coût moyen Entretien en fonction de la catégorie et de la superficie de la suite d'hôtel

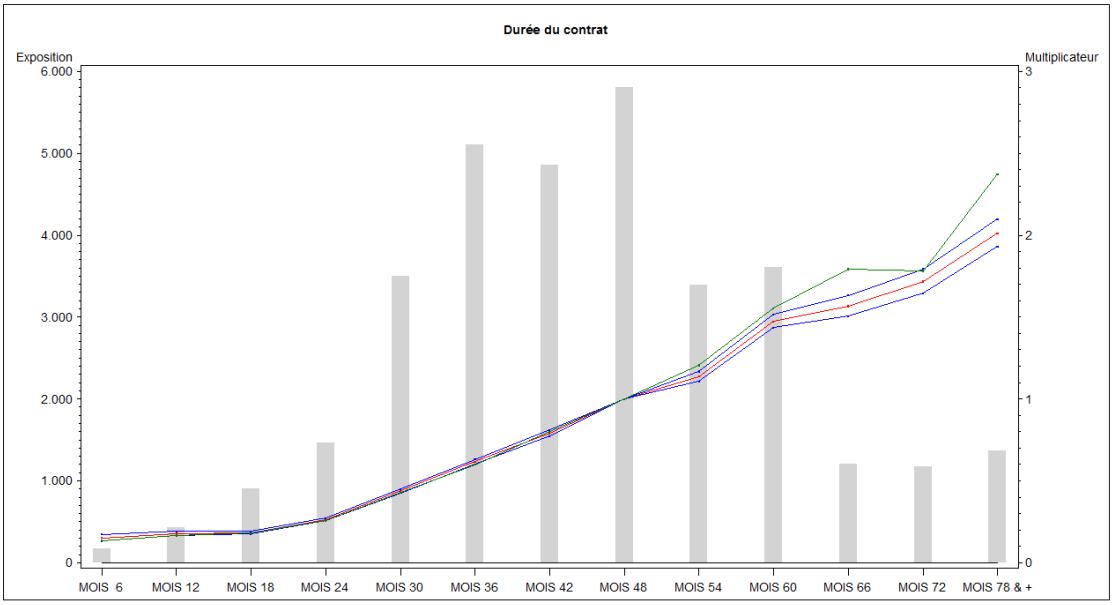

FIGURE 2.26 – Comparaison univarié/multivarié : Coût moyen Réparation en fonction de la durée du contrat d'assurance

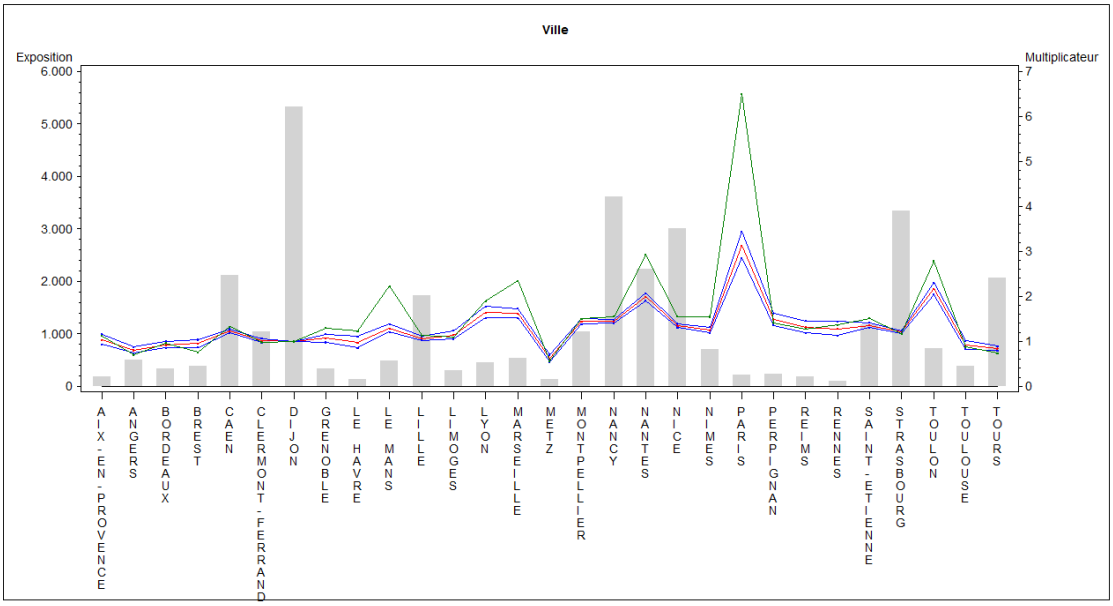

FIGURE 2.27 – Comparaison univarié/multivarié : Coût moyen Réparation en fonction de la localité de la suite d'hôtel

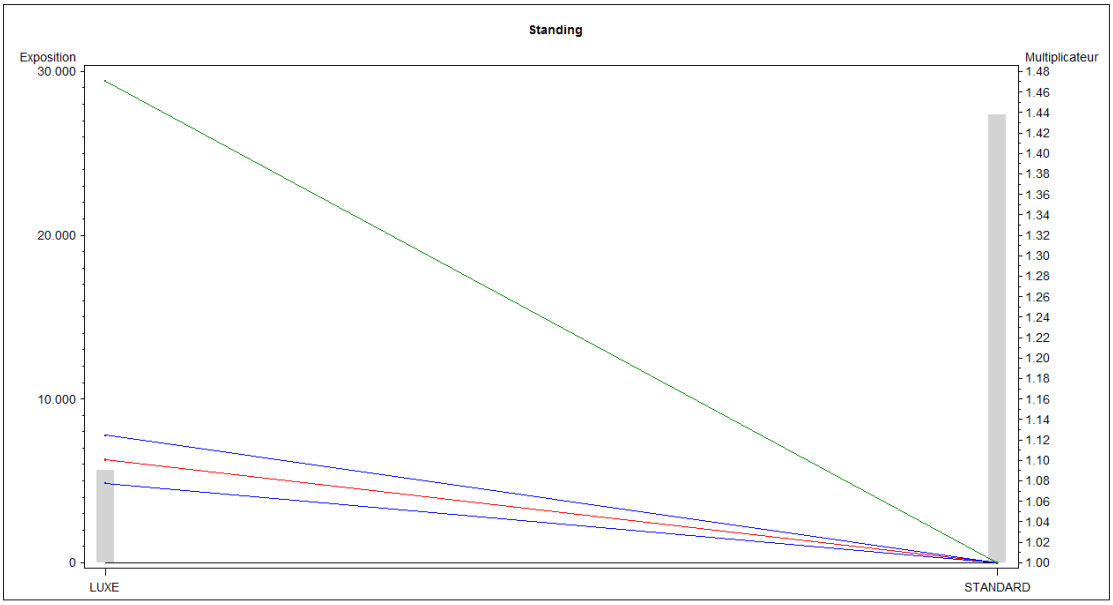

FIGURE 2.28 – Comparaison univarié/multivarié : Coût moyen Réparation en fonction du standing de la suite d'hôtel

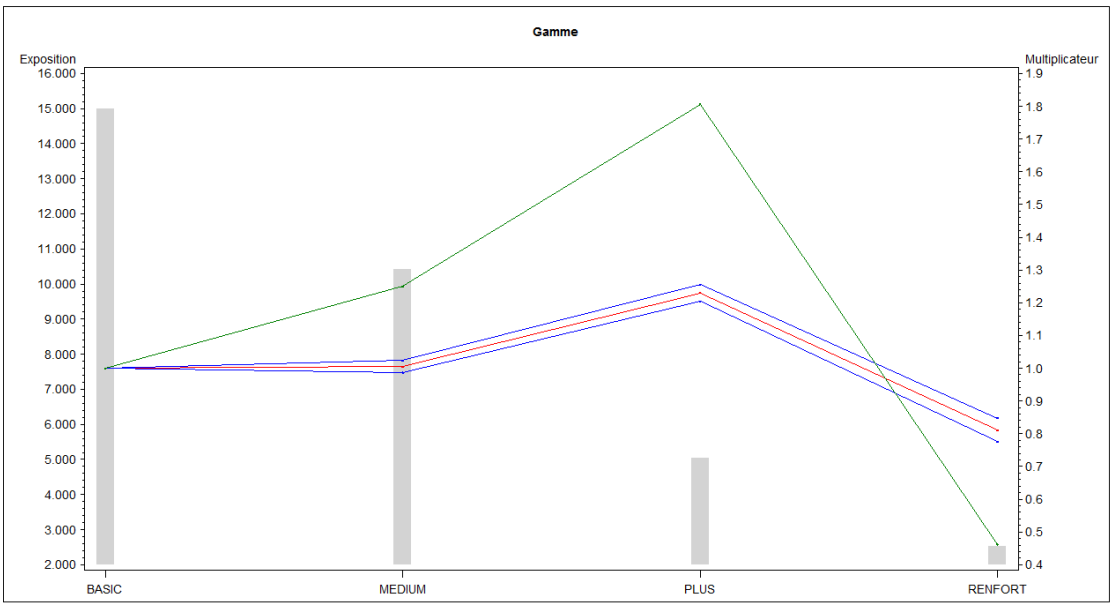

FIGURE 2.29 – Comparaison univarié/multivarié : Coût moyen Réparation en fonction de la gamme du produit d'assurance

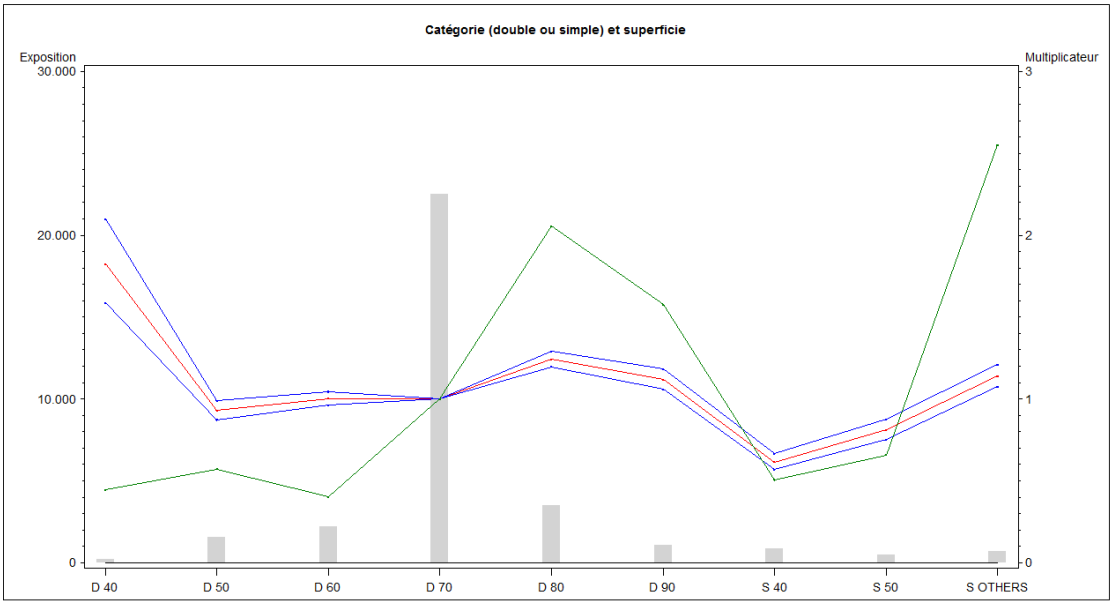

FIGURE 2.30 – Comparaison univarié/multivarié : Coût moyen Réparation en fonction de la catégorie et de la superficie de la suite d'hôtel

En Entretien, il n'y a quasiment pas de différence entre les multiplicateurs univariés et multivariés pour la variable Durée du contrat.

En effet, il s'agit de la variable la plus discriminante du coût moyen, c'est elle qui contient le plus de pouvoir explicatif. Ainsi, en analyse univariée, si l'on ne regarde la variation du coût moyen que par rapport à la durée du contrat, on a déjà un effet presque isolé (décorrélé) de la variable, puisque les autres variables n'interviennent pas beaucoup sur l'explication du coût moyen.

A l'inverse, les différences entre les multiplicateurs univariés et multivariés sont bien plus visibles sur les autres variables : en analyse univariée sur les 4 autres variables, le coût moyen observé est en fait en grande partie dû à la variable Durée du contrat. D'où les différences notables avec l'analyse multivariée, où là on observe les effets isolés, ne tenant pas compte de l'effet de la variable Durée du contrat.

Les observations que nous pouvons faire pour l'assurance Réparation sont sensiblement les mêmes.

2.5 Analyse des résidus

Plusieurs types de résidus peuvent être analysés pour savoir si la valeur modélisée est plus ou moins proche de la valeur réelle. Dans notre cas, nous avons choisi d'analyser les résidus de déviance, d'une part parce que ce sont les résidus le plus fréquemment utilisés, et d'autre part parce qu'ils sont automatiquement calculés par SAS lors de la procédure *Genmod* (option à préciser). Notons qu'il existe d'autres résidus fréquemment utilisés, tels que les résidus de Pearson, qui ne feront pas l'objet d'une étude ici.

Les résidus de déviance sont donnés par la formule suivante, pour chaque observation i :

$$r_i^D = signe(Y_i - \mu_i) \times \sqrt{2\omega_i \int_{\mu_i}^{Y_i} \frac{(Y_i - \xi)}{V(\xi)} d\xi}$$

Il s'agit de la racine carrée de la contribution de l'observation à la déviance totale, multipliée par + ou - 1 selon que la valeur réelle (Y_i) est plus ou moins grande que la valeur modélisée (μ_i). En d'autres termes, les résidus de déviance sont une mesure de la distance entre les observations et les estimations.

Sur base de ces résidus, nous traçons plusieurs grahiques qui vont nous permettre d'apprécier la qualité de la modélisation :

- Histogramme des résidus de déviance
- Graphique Quantile-Quantile des résidus de déviance
- Graphique de l'effet levier, ou "leverage" (que nous définirons ci-dessous) en fonction des valeurs modélisées

2.5.1 Histogramme des résidus de déviance

Si le modèle est correctement évalué, il doit y avoir un certain équilibre entre la quantité de résidus positifs et de résidus négatifs, et la plupart d'entre eux doivent être proches de 0. Ainsi, l'histogramme des résidus de déviance doit approximer la courbe de la densité d'une loi normale centrée.

Le graphique suivant représente l'histogramme des résidus de déviance pour le modèle Entretien (figure 2.31 p. 48). Celui du modèle Réparation est présenté en annexe.

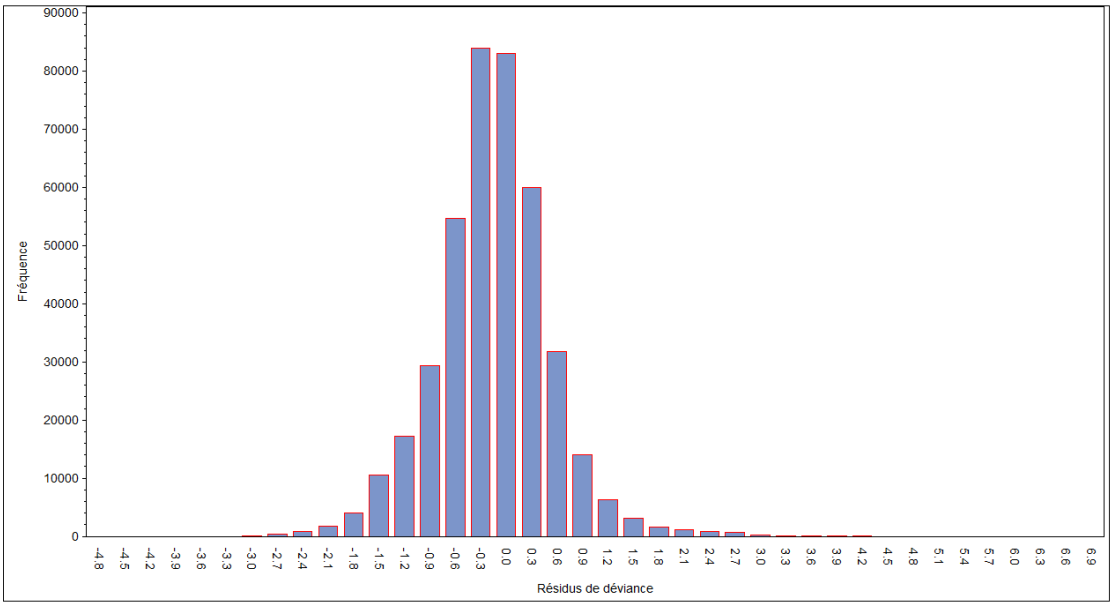

FIGURE 2.31 – MLG Entretien : Histogramme des résidus de déviance

2.5.2 Graphique Quantile-Quantile des résidus de déviance

Ce graphique représente les résidus de déviance en fonction des quantiles d'une loi normale centrée. Ainsi, les points doivent approximativement être localisés sur une droite (droite de Henry) pour valider que le modèle est correctement estimé.

Le graphique suivant représente le QQ-plot des résidus de déviance pour le modèle Entretien (figure 2.32 p. 49). Celui du modèle Réparation est présenté en annexe.

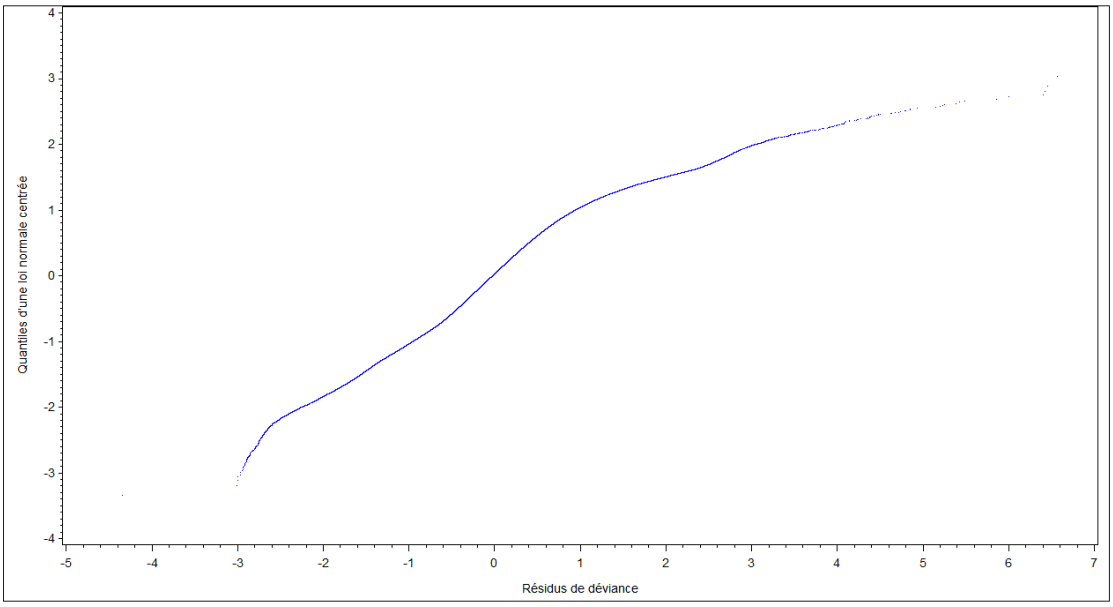

FIGURE 2.32 – MLG Entretien : QQ-plot des résidus de déviance

2.5.3 Graphique de l'effet levier en fonction des valeurs modélisées

L'effet levier est une mesure de l'influence qu'a une observation sur sa valeur modélisée. Cela permet de rendre compte du changement qu'implique une variation de l'observation sur sa valeur modélisée.

Ainsi, en traçant cet indicateur en fonction des valeurs modélisées, nous sommes capables d'identifier les observations qui auraient une influence disproportionnée sur le modèle. En effet, il s'agit de repérer les points isolés des autres, avec un effet levier plus grand que l'ensemble des autres points. Dans ce cas, il est nécessaire de regarder plus en détail les caractéristiques de ces observations, afin d'identifier si oui ou non elles faussent les estimations du modèle. Enfin, il faut être vigilent aux échelles du graphique : on pourrait croire que certains points sont parfois isolés des autres, mais l'échelle est parfois très petite et peut être trompeuse. Il faut donc prêter attention à l'écart relatif entre les points.

Le graphique suivant représente le "leverage" des résidus de déviance pour le modèle Entretien (figure 2.33 p. 50). Celui du modèle Réparation est présenté en annexe.

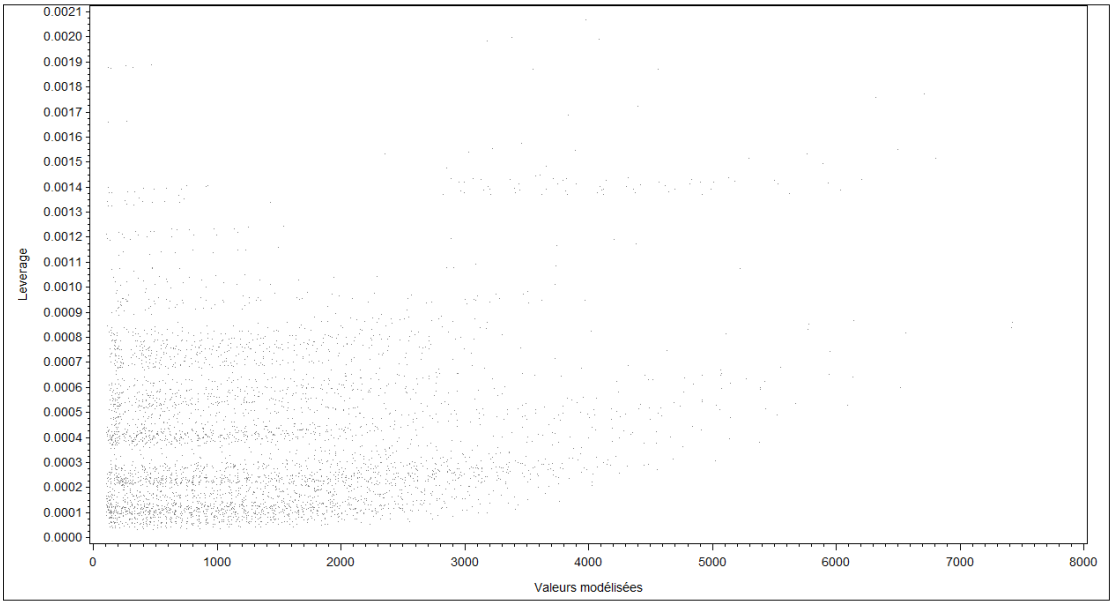

FIGURE 2.33 – MLG Entretien : Leverage des résidus de déviance

2.6 Lissage des résultats

2.6.1 Utilité

Le lissage des résultats permet d'incorporer au modèle une part de connaissance de l'utilisateur. En effet, son bon sens lui indique que certaines variables explicatives doivent avoir un ordre naturel. Par exemple, le coût moyen d'entretien d'une suite d'hôtel doit être croissant avec les niveaux de garantie choisis : plus les garanties sont élevées, plus le contrat est cher. Il peut s'avérer que les graphiques issus du modèle ne reflètent pas toujours cette évolution intuitive. Ainsi, il est possible de rectifier les courbes des estimateurs en fonction des niveaux de variables explicatives, simplement de manière visuelle, en ajustant les points qui ne sont pas cohérents.

Une fois que le lissage est effectué sur une des variables explicatives, il faut réexécuter le MLG. En effet, le fait de lisser une variable revient à la restreindre à certaines valeurs. En réexécutant le MLG, on permet au modèle de compenser sur d'autres variables les biais que l'on a volontairement créés (ce sont les variables fortement corrélées qui vont compenser).

Les lissages sont surtout pratiqués pour des raisons commerciales et de crédibilité vis-à-vis des clients. En effet, il apparaît difficile de justifier au client que son contrat est plus cher que celui de son voisin, alors que sa durée de couverture est inférieure. Cela permet aussi d'éviter les sauts de tarification.

2.6.2 Méthode

En pratique, on sait que notre modèle est multiplicatif et se compose de la façon suivante :

$$y = constante \times multiplicateur_1 \times multiplicateur_2 \times multiplicateur_3 \times \dots$$

où y est la variable à expliquer, ici le coût moyen (entretien ou réparation), et les multiplicateurs sont les estimations fournies par le MLG pour chaque variable explicative. Enfin, la constante est l'*intercept* fourni par le MLG, c'est à dire le coût moyen pour le profil de référence (i.e. le plus représenté dans la base de données). Les estimations pour chaque profil se font donc à partir de la valeur pour le profil de référence, considéré comme un étalon, que l'on multiplie par les différents ratios relatifs aux niveaux de chaque variable explicative.

Lorsque l'on réalise le lissage, par exemple sur la variable explicative numéro 1, on force les valeurs prises par cette variable. Ainsi, le terme $multiplicateur_1$ est connu. En réexécutant le MLG pour compenser ces restrictions, on considère alors le nouveau modèle suivant :

$$\frac{y}{multiplicateur_1} = constante \times multiplicateur_2 \times multiplicateur_3 \times \dots$$

La variable numéro 1 ne fait plus partie des variables explicatives, et la variable à expliquer est modifiée en tenant compte des restrictions faites sur la variable numéro 1.

2.6.3 Résultats du lissage

Les graphiques suivants (figures 2.34 p. 52 à 2.38 p. 54) représentent les compensations effectuées par le MLG sur les autres variables suite au lissage de la variable d'interaction $Catsurf$ (catégorie : double ou simple, et superficie). La courbe en rouge représente les résultats avant lissage, et la courbe en vert après lissage. Pour plus de lisibilité, nous n'avons pas tracé les intervalles de confiance. Les autres variables n'ont pas fait l'objet de lissage.

Les résultats pour la réparation sont présentés en annexes.

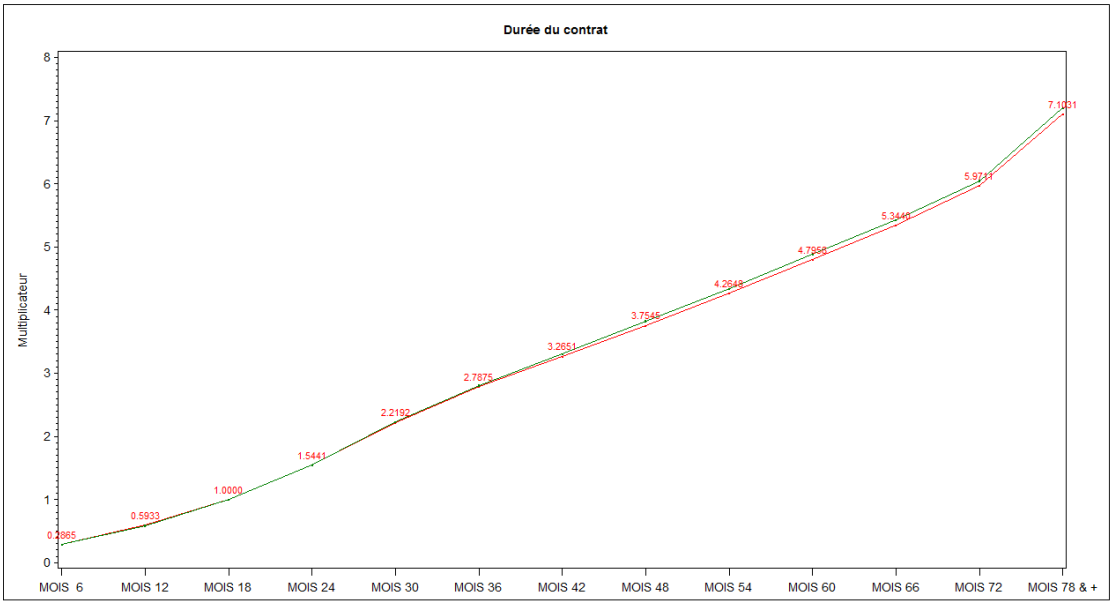

FIGURE 2.34 – MLG Entretien : Effet du lissage - Durée du contrat d'assurance

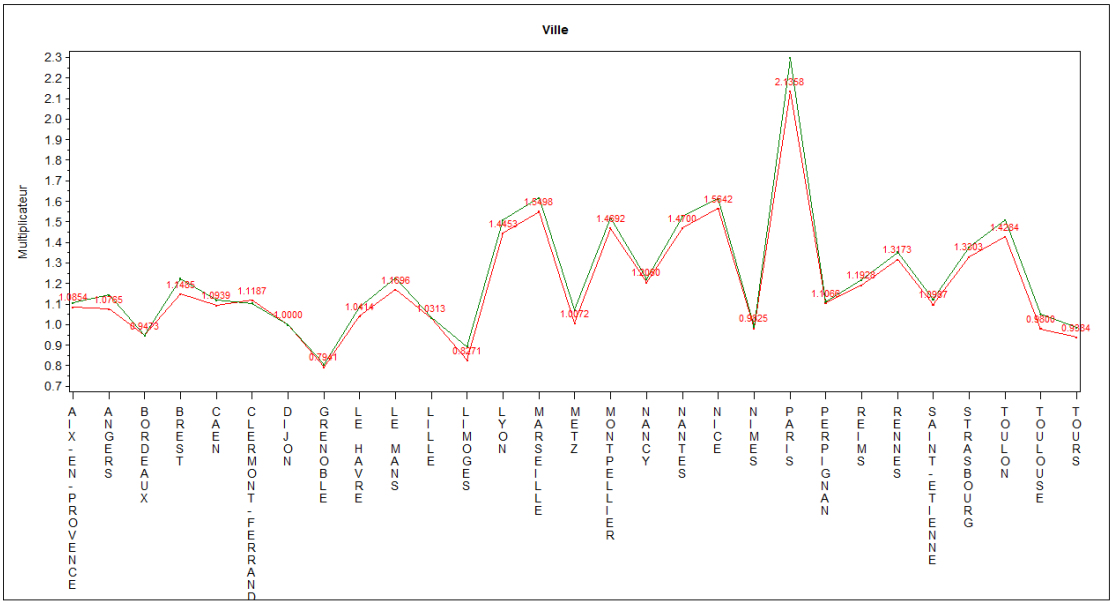

FIGURE 2.35 – MLG Entretien : Effet du lissage - Localité de la suite d'hôtel

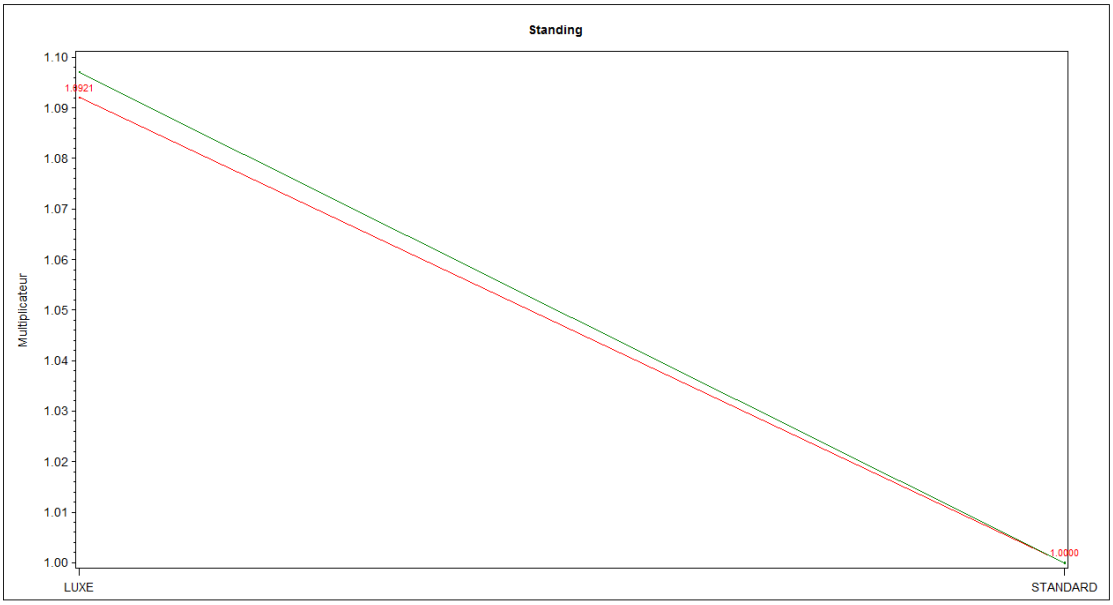

FIGURE 2.36 – MLG Entretien : Effet du lissage - Standing de la suite d'hôtel

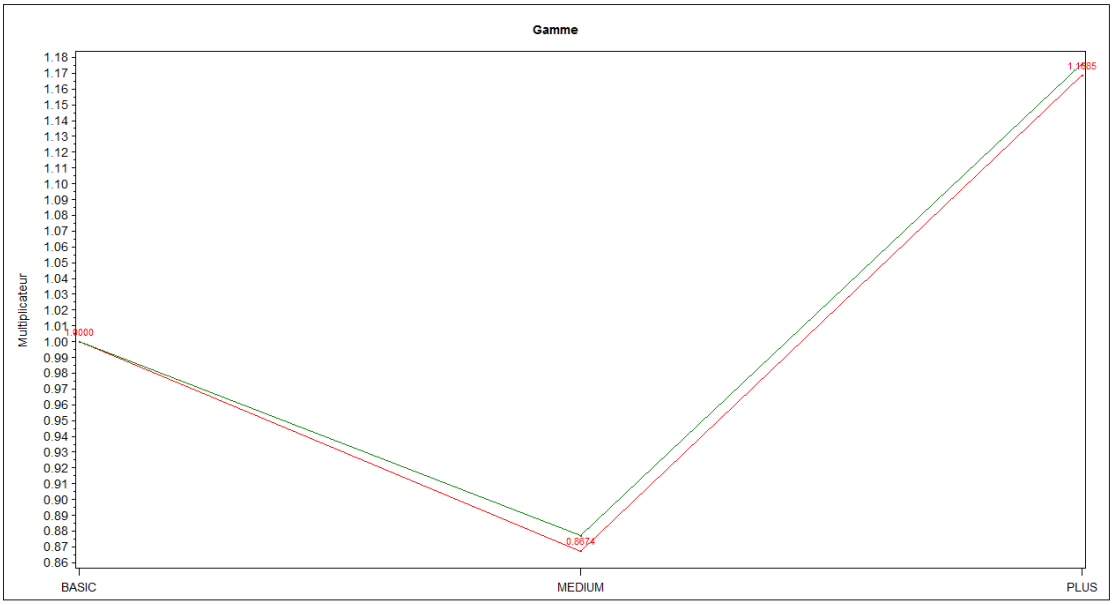

FIGURE 2.37 – MLG Entretien : Effet du lissage - Gamme du produit d'assurance

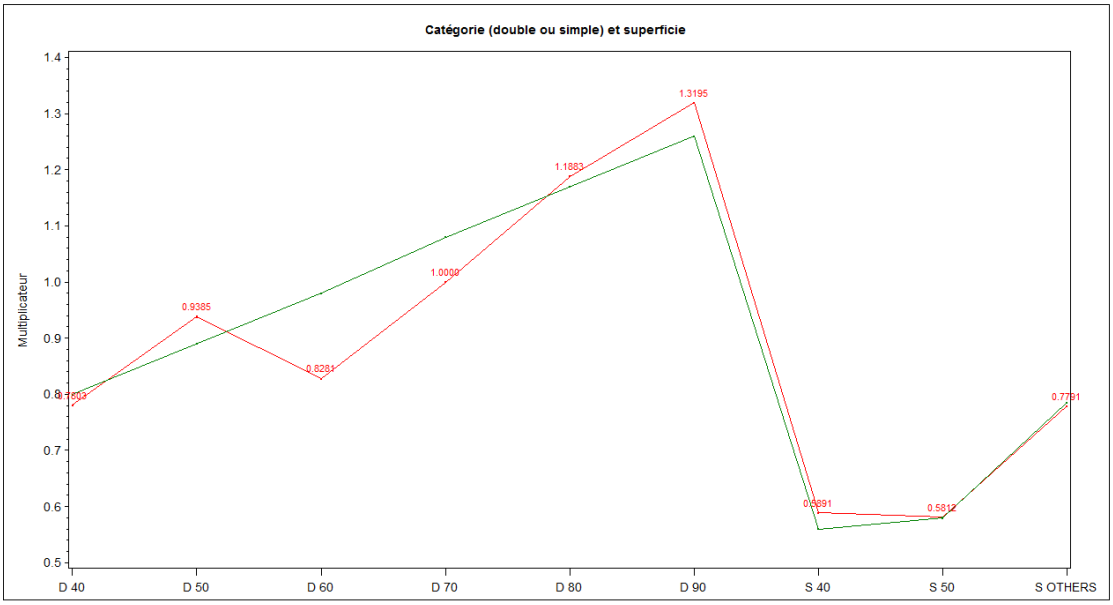

FIGURE 2.38 – MLG Entretien : Lissage effectué sur la catégorie et la superficie de la suite d'hôtel

On remarque que ce sont surtout les variables de localité et standing de la suite qui ont compensé le lissage effectué sur la variable catégorie-superficie.

Troisième partie

Analyse de coût par Apprentissage Statistique

3.1 Limites des Modèles Linéaires Généralisés

Le principal atout de l'apprentissage statistique est qu'il ne nécessite aucune hypothèse sur la strucure des données. En effet, nous avons vu en partie 2.1 p. 34 que les Modèles Linéaires Généralisés supposaient que la loi de la variable à expliquer appartienne à la famille exponentielle. De plus, il était indispensable d'attribuer une fonction lien permettant de relier la moyenne de la variable à expliquer aux variables explicatives. Le fait de devoir imposer ces conditions aux données constitue une réelle limite aux MLG, puisque les libertés de modélisation sont restreintes, et que l'on ne peut pas de ce fait effectuer une modélisation sur n'importe quel type de données.

Or, dans la réalité, les données sont bien souvent non conformes aux standards imposés par la modélisation par MLG.

3.2 L'apprentissage statistique

Aussi appelée "Machine Learning" en anglais, cette technique se veut la plus autonome possible. En effet, comme son nom l'indique, le but est de faire apprendre à l'ordinateur tout seul comment étudier les données qu'on lui soumet, et comment en extraire des informations pertinentes. En observant les données, l'ordinateur apprend au fur et à mesure, par le biais d'algorithmes que nous aborderons par la suite, comment les données interagissent entre elles et quelle est leur influence sur la variable à prédire.

Les algorithmes d'apprentissage statistique se répartissent en deux grandes familles :

- **Apprentissage supervisé :** Il consiste à établir des règles de comportement à partir d'une base de données contenant des exemples de cas déjà étiquetés. L'objectif est de prédire la valeur de sortie (variable à expliquer) pour une nouvelle donnée d'entrée (caractérisée par ses variables explicatives).
- **Apprentissage non supervisé :** Lorsqu'on ne dispose que d'exemples, mais non d'étiquettes, on parle d'apprentissage non supervisé. L'algorithme doit alors cibler les données selon leurs attributs disponibles, pour les classer en groupe homogènes d'exemples. La similarité est généralement calculée selon une fonction de distance entre paires d'exemples.

Il existe plusieurs algorithmes d'apprentissage statistique. Dans notre étude, nous avons choisi de privilégier l'algorithme CART [4] (apprentissage supervisé), pour créer des arbres de décision, qui ont l'avantage d'être faciles d'utilisation et d'interprétation. De plus, CART est l'un des algorithmes les plus répandus, et donc le plus développé sur des logiciels (notamment SAS et R). Nous faisons le choix d'utiliser le logiciel R, pour lequel de nombreux packages ont été développés (dont Rpart [12] que nous utiliserons) et dont la documentation d'aide est largement disponible sur Internet notamment.

4. Classification And Regression Trees

3.3 L'algorithme CART : Classification And Regression Tree

L'algorithme CART a été développé et publié par Leo Breiman en 1984 [3]. Il permet de construire des arbres de décision binaires à partir des données fournies. On parlera de **régression** lorsque la variable à prédire possède un nombre infini de valeurs (comme dans notre cas, le coût moyen), et de **classification** lorsqu'elle possède un nombre fini de valeurs.

3.3.1 Arbre de décision binaire

Un arbre de décision est un outil d'aide à la décision qui représente une situation sous la forme graphique d'un arbre de façon à faire apparaître à l'extrémité de chaque branche les différents résultats possibles en fonction des décisions prises à chaque étape.

L'arbre de décision est un outil très apprécié pour sa lisibilité. Il est utilisé pour répartir une population d'individus en groupes homogènes selon un ensemble de variables discriminantes. L'arbre de décision se place parmi les méthodes d'apprentissage supervisé. Il s'agit de prédire avec le plus de précision possible les valeurs prises par la variable à expliquer à partir d'un ensemble de descripteurs. On rejoint là complètement l'objectif des Modèles Linéaires Généralisés.

En partant de la racine de l'arbre, qui constitue l'ensemble des données, l'algorithme sépare successivement les données en deux groupes, appelés des noeuds. C'est pourquoi l'on parle d'arbre binaire.

Les séparations sont effectuées selon un critère de segmentation, qui peut varier selon le type de modélisation. L'algorithme cesse de réitérer les séparations lorsqu'un critère d'arrêt (défini au préalable par l'utilisateur) est atteint. Le critère d'arrêt peut être par exemple un nombre minimum d'individus au sein d'un groupe. Ainsi, l'algorithme s'arrête avant de créer des groupes trop petits qui ne seraient pas vraiment significatifs. Un noeud final où aucune séparation n'est effectuée (du fait qu'un critère d'arrêt est atteint) s'appelle une feuille.

3.3.2 Critère de segmentation

Pour choisir la variable de segmentation sur un noeud, ainsi que la valeur de cette variable, l'algorithme teste toutes les variables potentielles et choisit celle qui maximise la réduction de déviance R :

$$R = D_{parent} - (D_{filsgauche} + D_{filsdroit})$$

Ainsi, il maximise le gain en pureté lors du passage du noeud parent aux noeuds fils. Plusieurs fonctions de déviance D peuvent être définies. Pour les problèmes de classification, on utilise généralement l'indice de Gini, tandis que la somme des carrés résiduelle est préconisée pour les problèmes de régression.

L'indice de Gini

Pour une classification, i.e. pour prédire la valeur d'une variable discrète, CART utilise par défaut un critère de segmentation basé sur l'indice de Gini, dont la formule est la suivante :

$$I = 1 - \sum_{i=1}^{n} f_i^2$$

où n est le nombre de classes à prédire et f_i la fréquence de la classe i dans le noeud. Plus l'indice de Gini est bas, plus le noeud est pur.

Somme des carrés résiduelle

Pour une régression, i.e. pour prédire la valeur d'une variable continue, CART utilise par défaut un critère de segmentation basé sur la somme des carrés résiduelle (Residual Sum of Squares en anglais, ou RSS). La déviance pour le noeud j s'exprime alors de la façon suivante :

$$D_j = \sum_{i \in noeud\ j} (y_i - \bar{y_j})^2$$

où y_i est la valeur de la variable à expliquer pour l'observation i, et $\bar{y_j}$ la moyenne empirique de la variable à expliquer calculée sur les i observations contenues dans le noeud j. Plus la somme des carrés résiduelle est basse, plus le noeud est pur.

3.4 Exemple introductif

Pour aborder la notion d'arbre de décision, reprenons un exemple simple couramment utilisé, dont les données sont les suivantes (table 3.6 p. 60) :

N°	Ensoleillement	Température (°F)	Humidité (%)	Vent	Jouer
1	Soleil	75	70	Oui	Oui
2	Soleil	80	90	Oui	Non
3	Soleil	85	85	Non	Non
4	Soleil	72	95	Non	Non
5	Soleil	69	70	Non	Oui
6	Couvert	72	90	Oui	Oui
7	Couvert	83	78	Non	Oui
8	Couvert	64	65	Oui	Oui
9	Couvert	81	75	Non	Oui
10	Pluie	71	80	Oui	Non
10	Pluie	65	70	Oui	Non
12	Pluie	75	80	Non	Oui
13	Pluie	68	80	Non	Oui
14	Pluie	70	96	Non	Oui

TABLE 3.6 – Exemple introductif : données

Le but est de prédire si un joueur de tennis va s'entraîner ou non, compte tenu de variables météorologiques : *Ensoleillement* (variable discrète : Soleil, Couvert, Pluie), $Humidite$ (variable continue), $Temperature$ (variable continue), $Vent$ (variable discrète : oui, non).

Sur base de cet exemple, nous allons tracer avec le package *Rpart* [12] de R l'arbre de décision correspondant. La variable à expliquer est *Jouer* et les variables explicatives sont *Ensoleillement*, *Temperature*, *Humidite* et *Vent*.

Les lignes de commande sont les suivantes :

```
> library(rpart)
> library(rpart.plot)
# Chargement des packages
```

Le package *rpart* contient l'algorithme CART pour construire les arbres, tandis que le package *rpart.plot* [6] est très utile pour avoir une bonne représentation graphique de ceux-ci, avec beaucoup d'options de personnalisation.

```
> tennis<-read.table("C:/Users/Camille/Desktop/Memoire_data/tennis.txt",
sep="",header=TRUE)
# Chargement des données

> ad.tennis.cnt <- rpart.control (minsplit = 1)
# Définition des paramètres de contrôle
```

minsplit est le nombre minimum d'observations qui doivent exister dans un noeud pour pouvoir effectuer une séparation sur ce noeud. La valeur par défaut étant 20, nous devons changer ce paramètre et le fixer à 1 pour obtenir un arbre, puisque la base ne contient que 14 observations.

```
> ad.tennis <- rpart (Jouer ~ Ensoleillement + Temperature + Humidite +
Vent, tennis, control = ad.tennis.cnt, method = "class")
# Construction de l'arbre
```

Une méthode de classification est utilisée car la variable à expliquer *Jouer* est discrète (deux valeurs possibles : oui ou non). Les paramètres de contrôle utilisés sont ceux définis ci-dessus.

```
> ad.tennis
# Affichage de l'arbre sous forme textuelle

> prp(ad.tennis,type=3,extra=1,compress=F,ycompress=F)
# Affichage de l'arbre sous forme graphique (fonction prp disponible grâce
au package rpart.plot)
```

Nous obtenons l'arbre de décision sous forme textuelle (figure 3.39 p. 62) et graphique (figure 3.40 p. 62).

```
n= 14

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 14 5 oui (0.3571429 0.6428571)
   2) Ensoleillement=pluie,soleil 10 5 non (0.5000000 0.5000000)
     4) Temperature>=77.5 2 0 non (1.0000000 0.0000000) *
     5) Temperature< 77.5 8 3 oui (0.3750000 0.6250000)
      10) Temperature< 66.5 1 0 non (1.0000000 0.0000000) *
      11) Temperature>=66.5 7 2 oui (0.2857143 0.7142857)
        22) Temperature>=70.5 4 2 non (0.5000000 0.5000000)
          44) Temperature< 73.5 2 0 non (1.0000000 0.0000000) *
          45) Temperature>=73.5 2 0 oui (0.0000000 1.0000000) *
        23) Temperature< 70.5 3 0 oui (0.0000000 1.0000000) *
   3) Ensoleillement=couvert 4 0 oui (0.0000000 1.0000000) *
```

FIGURE 3.39 – Forme textuelle de l'arbre de décision

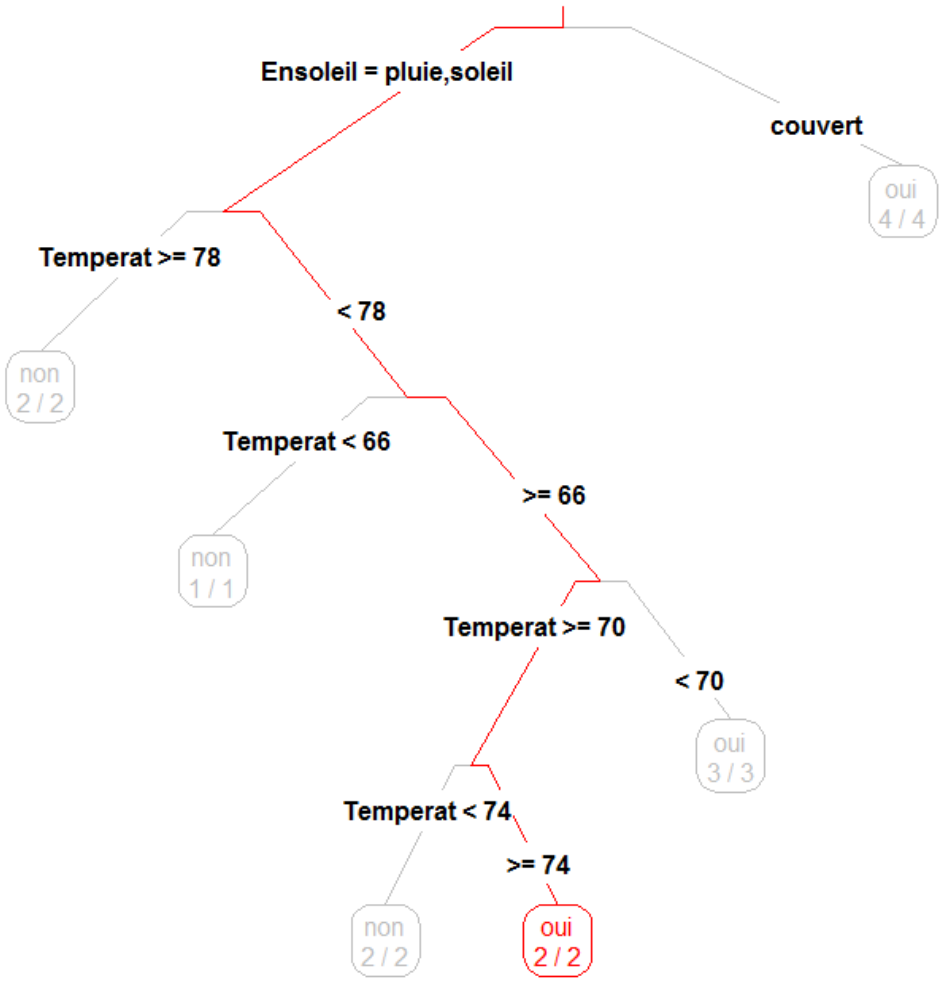

FIGURE 3.40 – Forme graphique de l'arbre de décision

La forme textuelle nous donne les informations suivantes pour chaque noeud :

- *split* le critère de séparation appliqué. Par exemple $Ensoleillement = Soleil$ pour signifier que la séparation est effectuée sur le fait qu'il y ait du soleil ou non.
- n le nombre d'observations contenues dans le noeud.
- *loss* le nombre d'observations mal classées dans le noeud. Ainsi, $loss/n$ est le taux de mauvaise classification du noeud. On remarque que pour tous les noeuds terminaux, *loss* est égal à 0, ce qui signifie que l'arbre classifie parfaitement toutes les données.
- *yval* la valeur prise par la variable à prédire.
- le vecteur (*yprob*) n'apporte pas d'information nouvelle mais récapitule le taux de mauvaise classification et le taux de bonne classification.

Remarquons que les variables $Humidite$ et $Vent$ n'ont pas été considérées comme discriminantes par l'algorithme.

La forme graphique est plus parlante visuellement, et vient en complément de la forme textuelle.

L'ensoleillement est la variable la plus discriminante, car c'est celle qui intervient en premier tout en haut de l'arbre. Puis c'est ensuite sur la température que se font les séparations suivantes. La forme graphique permet de bien visualiser l'importance hiérarchique des descripteurs dans l'explication de la variable $Jouer$.

A chaque noeud terminal (ou feuille), l'arbre indique dans une bulle la valeur de $Jouer$ ainsi que le taux de bonne classification. Les différents éléments qui apparaissent sur le graphique de l'arbre sont personnalisables avec les options de la fonction *prp*.

Nous avons tracé à titre d'exemple un chemin en rouge, pour lequel le résultat se lit de la manière suivante : s'il y a de la pluie ou du soleil, et que la température est comprise entre 74°F et 77°F, alors le sportif jouera au tennis. Sur les données fournies, l'algorithme ne fait pas d'erreur de prédiction sur la valeur de $Jouer$ à l'issue de ce chemin.

Que ce soit sous forme textuelle ou graphique, les informations affichées sont les mêmes. Seule la manière de présenter les résultats est différente.

Dans cet exemple très simple, nous n'avons indiqué à R aucune option, et l'arbre a été construit avec les options par défaut (mis à part le changement de paramètre *minsplit*). Pour un échantillon si petit, l'arbre fourni avec les options par défaut est déjà optimal et ne nécessite aucun travail supplémentaire. Par contre, pour notre base de données d'assurance qui contient plusieurs centaines de milliers de lignes, il convient de raffiner l'arbre afin d'obtenir un arbre optimal en termes de taille et de prédiction. C'est ce que nous abordons dans la suite de ce travail.

Par ailleurs, l'exemple introductif était un problème de classification. Pour notre modélisation en assurance, nous nous intéressons à une variable de coût, continue. Nous allons donc utiliser une méthode de régression. Les instructions resteront néanmoins sensiblement les mêmes.

3.5 Application aux données d'assurance

Plus le modèle est complexe, plus l'arbre le sera aussi. Il n'est pas souhaitable d'avoir un arbre trop grand et qui perdrait en simplicité d'interprétation. De plus, si le modèle s'ajuste quasiment parfaitement sur les données, on peut craindre qu'il soit inapte à prendre en compte de nouvelles données dans le futur, et à en fournir des estimations fiables.

Tout l'enjeu réside donc dans la taille de l'arbre que l'on doit envisager : ni trop petit (pour avoir suffisamment de précision dans les prédictions) ni trop grand (pour éviter un surajustement). Il faut trouver l'arbre le plus petit possible, mais ayant la performance la plus grande possible. Pour cela, nous devons réaliser des arbitrages entre la performance et la complexité de l'arbre.

Les sections suivantes présentent un rappel sur le partitionnement de la base et les différentes étapes de la modélisation : la construction d'un arbre de taille maximale, puis l'élagage de celui-ci grâce à la validation croisée.

3.5.1 Partitionnement de la base

En première partie de ce travail (voir partie 1.9 p. 30), nous avions divisé aléatoirement la base de données finale en deux parties : 75% pour la base d'apprentissage et 25% pour la base de test.

Pour la modélisation par arbre de régression, nous avons en plus besoin de diviser la base d'apprentissage de la manière suivante :

- 2/3 des données serviront à la **construction** d'un arbre de taille maximale
- 1/3 des données sera consacré à l'**élagage** de celui-ci via le principe de la **validation croisée**, que nous aborderons dans la suite de ce travail

Le tableau suivant (tableau 3.7 p. 64) récapitule le partitionnement de la base :

Base d'Apprentissage		Base de Test
Construction	Validation croisée	
50%	25%	25%

Table 3.7 – Partition de la base de données

3.5.2 Construction d'un arbre de taille maximale

La base d'apprentissage va nous servir à déterminer un arbre de décision de taille maximale, c'est à dire prédisant presque parfaitement la variable explicative sur ce jeu de données. Les instructions en R sont les suivantes :

```
> library(rpart)
# Chargement du package rpart

> apprentissage <- read.table("C:/Users/Camille/Desktop/Memoire_data/
apprentissage_ent.txt",sep="",header=TRUE)
# Chargement de la base d'apprentissage
```

```
> ad.apprentissage.cnt <- rpart.control (minbucket=1000,cp=0,xval=3)
# On définit les options
```

cp désigne le "Complexity Parameter" (ou coût de complexité), que nous aborderons dans la section suivante. Fixer sa valeur à 0 permet d'obtenir un arbre de taille maximale sans élagage. On souhaite toutefois limiter la taille de l'arbre en imposant un nombre minimum de 1000 observations dans un noeud final via l'option $minbucket = 1000$. L'option $xval = 3$ est relative à la validation croisée, que nous aborderons dans la section suivante.

```
> ad.apprentissage <- rpart (amount_ent_cumul ~ Duree + Ville +
Standing + Gamme + Catsurf, apprentissage, method="Anova",
control = ad.apprentissage.cnt)
```

On stocke l'arbre dans la variable *ad.apprentissage*, et on indique au logiciel que l'on souhaite modéliser *amount_ent_cumul* (la variable des coûts cumulés en Entretien, résultant du processus d'agrégation des coûts en partie 1.5 p. 23) en fonction des variables $Duree$, $Ville$, $Standing$, $Gamme$, et $Catsurf$. On précise également les options de contrôle à prendre en compte que nous avons définies au préalable. Nous choisissons la méthode Anova, qui correspond aux problèmes de régression (variable à expliquer continue).

```
> ad.apprentissage
# On affiche l'arbre sous forme textuelle
```

L'arbre obtenu est extrêmement volumineux sous forme textuelle : il possède 140 noeuds. Il ne sert à rien de le représenter sous forme graphique.

3.5.3 Validation croisée et élagage

Principe de la validation croisée

Par défaut, la fonction *rpart* partitionne en interne les données en entrée en $xval = 10$ portions (ce nombre est paramétrable dans les options). Elle ajuste un arbre sur les $9/10^{emes}$ de la base en entrée, et utilise la fraction restante de $1/10^{eme}$ pour estimer l'erreur en validation croisée. Pour ce faire, elle prédit le $1/10^{eme}$ restant à l'aide de l'arbre construit sur les $9/10^{emes}$, et calcule l'erreur de prédiction. Le taux d'erreur en validation croisée est alors la moyenne des taux d'erreur ainsi collectés sur les 10 arbres (chacun d'entre eux étant construit sur $9/10^{emes}$ de la base en entrée et testés sur le $1/10^{eme}$ restant).

La méthode par défaut de validation croisée à 10 plis est justifiée lorsque l'on ne possède pas beaucoup de données. Dans notre cas, étant donné que l'on dispose d'un nombre important d'observations, il est plus adéquat de changer les proportions de données dédiées à la construction et à la validation croisée.

Nous avons émis précédemment le souhait de partitionner la base de données en plusieurs parties (voir tableau 3.7 p. 64). Ainsi, au sein de la base d'apprentissage,

nous souhaitons dédier 2/3 des informations à la construction de l'arbre, et 1/3 à la validation croisée. Pour cela, nous allons donner en entrée à la fonction *rpart* la base d'apprentissage (soit 75% des données totales), et changer la valeur par défaut de *xval* en la fixant à 3. Ainsi, *rpart* ajustera un arbre sur 2/3 de la base d'apprentissage, et effectuera une validation croisée sur le tiers restant. C'est bien ce qui a été précisé plus haut dans le vecteur des paramètres ($xval = 3$).

Elagage de l'arbre : utilisation du "Complexity Parameter"

La fonction *rpart* va réitérer ce processus de validation croisée pour des "Complexity Parameters" (*cp*) différents.

Ce paramètre est fonction du nombre de segmentations (et donc de la taille de l'arbre), et *rpart* va considérer plusieurs valeurs de *cp*. Plus le paramètre *cp* sera petit, plus l'arbre sera grand. Il faut noter que la valeur par défaut du *cp* minimum est 0.01, ce qui constitue un critère d'arrêt sur la taille de l'arbre. Il est possible de modifier ce minimum dans les paramètres et de le fixer à 0 de manière à neutraliser ce critère d'arrêt. C'est bien ce qui a été précisé plus haut dans le vecteur des paramètres ($cp = 0$).

rpart construit alors une *cptable* regroupant les taux d'erreur en validation croisée correspondant aux différents *cp* considérés.

La commande suivante permet d'afficher la *cptable* :

```
> printcp(ad.apprentissage)
```

En voici un extrait des premières lignes (figure 3.41 p. 66) et dernières lignes (figure 3.42 p. 67) :

	CP	nsplit	rel error	xerror	xstd
1	2.2800e-01	0	1.00000	1.00001	0.0120293
2	6.0701e-02	1	0.77200	0.77201	0.0111001
3	2.5114e-02	2	0.71130	0.71131	0.0103844
4	2.2455e-02	3	0.68618	0.67975	0.0102562
5	1.8084e-02	4	0.66373	0.66381	0.0098918
6	8.2230e-03	5	0.64565	0.64575	0.0098217
7	7.1713e-03	6	0.63742	0.63610	0.0097508
8	5.6438e-03	7	0.63025	0.63396	0.0096697
9	5.1121e-03	8	0.62461	0.62544	0.0094599
10	3.3456e-03	9	0.61950	0.62034	0.0094389
11	3.3301e-03	10	0.61615	0.61761	0.0094241
12	2.4778e-03	11	0.61282	0.61320	0.0094208
13	2.3953e-03	12	0.61034	0.61254	0.0094136
14	2.1960e-03	13	0.60795	0.60935	0.0093834
15	2.1531e-03	14	0.60575	0.60820	0.0093492
16	1.9805e-03	15	0.60360	0.60537	0.0093447
17	1.8888e-03	16	0.60162	0.60339	0.0093379
18	1.7821e-03	17	0.59973	0.60238	0.0093390
19	1.7550e-03	18	0.59795	0.60073	0.0093387
20	1.7139e-03	19	0.59619	0.60012	0.0093385
21	1.6326e-03	20	0.59448	0.59819	0.0093267
22	1.4988e-03	21	0.59284	0.59595	0.0093145
23	1.0827e-03	22	0.59135	0.59441	0.0093095
24	9.6042e-04	23	0.59026	0.59277	0.0092976

FIGURE 3.41 – Premières lignes de la cptable - Entretien

```
114 7.6110e-06    116   0.57061 0.57860 0.0092417
115 7.1510e-06    117   0.57060 0.57859 0.0092417
116 6.9585e-06    118   0.57059 0.57859 0.0092417
117 5.6794e-06    119   0.57059 0.57859 0.0092417
118 5.6421e-06    120   0.57058 0.57859 0.0092417
119 5.5238e-06    121   0.57058 0.57859 0.0092417
120 5.2314e-06    122   0.57057 0.57859 0.0092417
121 5.1071e-06    123   0.57057 0.57858 0.0092417
122 4.8809e-06    124   0.57056 0.57858 0.0092417
123 3.0564e-06    125   0.57056 0.57858 0.0092417
124 2.8971e-06    126   0.57055 0.57858 0.0092417
125 2.4838e-06    127   0.57055 0.57857 0.0092417
126 1.3379e-06    128   0.57055 0.57857 0.0092417
127 1.1793e-06    129   0.57055 0.57857 0.0092417
128 9.3922e-07    130   0.57054 0.57857 0.0092417
129 7.0134e-07    131   0.57054 0.57857 0.0092417
130 5.8073e-07    132   0.57054 0.57857 0.0092417
131 4.0249e-07    133   0.57054 0.57857 0.0092417
132 3.4410e-07    134   0.57054 0.57857 0.0092417
133 2.8074e-07    135   0.57054 0.57857 0.0092417
134 1.7317e-07    136   0.57054 0.57857 0.0092417
135 9.0473e-08    137   0.57054 0.57857 0.0092417
136 2.3186e-08    138   0.57054 0.57857 0.0092417
137 0.0000e+00    139   0.57054 0.57857 0.0092417
```

FIGURE 3.42 – Dernières lignes de la cptable - Entretien

La *cptable* nous fournit un certain nombre d'informations :

- *nsplit* est le nombre de segmentations réalisées sur l'arbre, aussi égal au nombre de noeuds - 1.
- *rel error* mesure l'erreur apparente (erreur calculée lors de la construction de l'arbre), normalisée de manière à ce que l'erreur sur la racine soit égale à 1.
- *xerror* mesure le taux d'erreur (normalisé également) dans la validation croisée que l'on considère comme un estimateur correct de l'erreur réelle.
- *xstd* est l'écart-type de l'erreur de validation croisée.

On peut noter que l'erreur apparente est en règle générale toujours plus faible (même légèrement) que l'erreur en validation croisée. En effet, il est logique que l'erreur soit plus faible sur la base qui a servi à construire l'arbre plutôt que sur les bases de validation croisée.

La commande *rpart* affiche toujours l'arbre de taille maximale, qui correspond à la dernière ligne de la *cptable*. Pour obtenir l'un des arbres intermédiaires, il suffit d'exécuter la commande *prune* ("élaguer" en anglais) qui prend en paramètre une valeur de *cp*. Par exemple, pour obtenir l'arbre avec 5 séparations (ligne 6 de la *cptable*) il faudrait choisir un *cp* dans l'intervalle]0.0082230 ; 0.018084].

Pour savoir quel est le paramètre *cp* optimal pour l'élagage, nous allons nous servir des informations contenues dans la *cptable*, et utiliser la règle de l'écart-type [5]. Nous allons calculer α le minimum de $xerror + xstd$ (l'erreur moyenne estimée + 1 écart-type), et choisir le plus petit arbre dont *xerror* est inférieure à ce seuil α.

5. la règle de l'écart-type (ou "1-SE rule") a été développée par Breiman en 1984

Les instructions suivantes nous permettent d'obtenir la ligne de la *cptable* qui minimise $xerror + xstd$, et la valeur du seuil α :

```
> which.min(ad.apprentissage$cptable[,"xerror"]
+ ad.apprentissage$cptable[,"xstd"])
[1] 135
> alpha <- ad.apprentissage$cptable[135,"xerror"]
+ ad.apprentissage$cptable[135,"xstd"]
> alpha
[1] 0.5878106
```

En étudiant la *cptable*, on observe que le plus petit arbre dont l'erreur en validation croisée est inférieure à α est celui possédant 32 séparations (donc 33 noeuds), avec $xstd = 0,58706$. Nous devons donc choisir une valeur de *cp* comprise dans l'intervalle $]0.00056282\,;\ 0.00068814]$. Prenons par exemple $cp = 0.0006$. Peu importe la valeur choisie dans cet intervalle, les résultats resteront inchangés.

Une autre manière de déterminer le paramètre *cp* optimal (et donc la taille de l'arbre optimale) est d'exploiter une représentation graphique fournie par la commande suivante (figure 3.43 p. 68) :

```
> plotcp(ad.apprentissage)
```

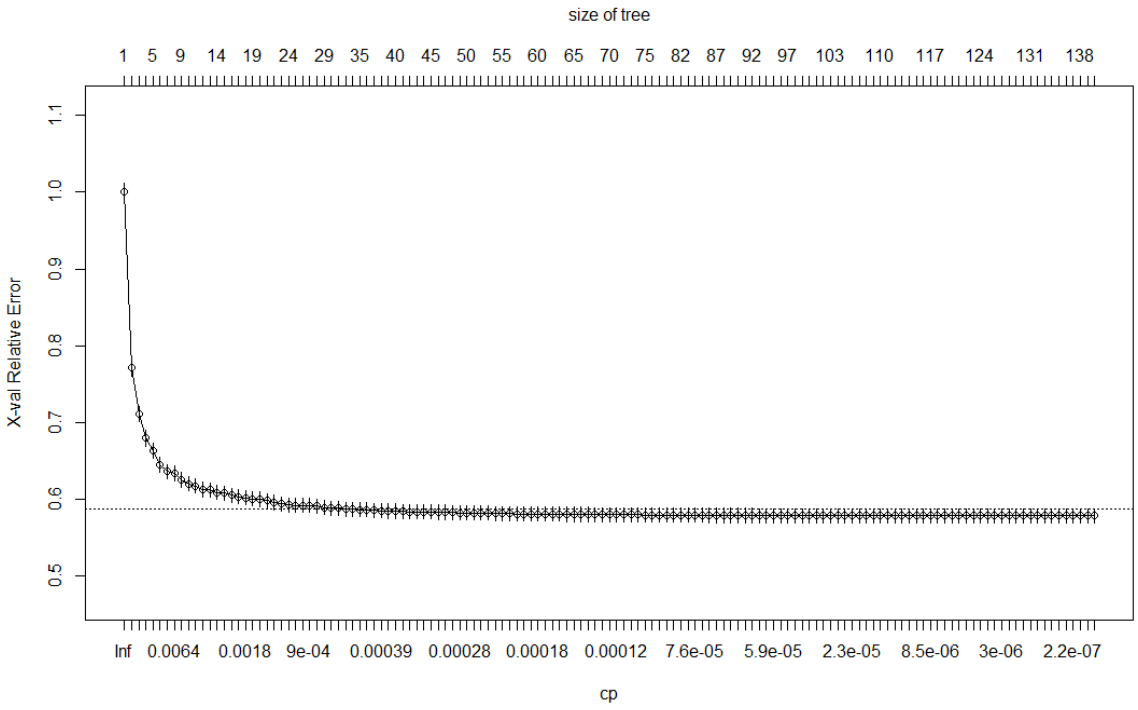

FIGURE 3.43 – Erreur de validation croisée en fonction de cp (ou nombre de noeuds de l'arbre) - Entretien

La ligne horizontale en pointillés correpond au seuil α. On peut donc lire approximativement que la taille du plus petit arbre dont l'erreur de validation passe en-dessous de cette ligne se situe vers 35 noeuds. La lecture est moins précise de cette manière, mais permet de mieux visualiser l'évolution caractéristique de l'erreur en validation croisée : une forte décroissance au début, puis rapidement une stagnation à une certaine valeur. Dans certains cas, on peut même avoir une erreur qui réaugmente à partir d'une certaine taille de l'arbre, et cela signifie alors que l'algorithme est en surapprentissage.

Dans notre cas, on observe simplement qu'à partir d'un certain stade, l'augmentation de la taille de l'arbre ne contribue plus de manière significative à l'apprentissage. Il est donc inutile d'aller au-delà, car on augmenterait en complexité sans augmenter en pouvoir de prédiction.

On obtient enfin l'arbre élagué grâce à la commande suivante :

```
> elague <- prune(ad.apprentissage,0.0006)
```

Il est composé de 33 noeuds. L'arbre de taille maximale possédait 140 noeuds, nous avons donc significativement réduit la complexité de celui-ci, tout en conservant un maximum de ses qualités de prédiction.

3.5.4 Résultats

Pour une meilleure lecture, l'arbre élagué (figure 3.44 p. 70) est découpé en 3 parties puis zoomé dans les figures 3.45 p. 71 à 3.47 p. 73.

Pour la Réparation, l'étude est analogue. On choisit par contre de fixer le paramètre *minbucket* à 400 car la base possède moins de données en réparation. L'arbre de taille maximale possède 133 noeuds. Après élagage, on se ramène à un arbre de 14 noeuds.

Le graphique de l'arbre élagué pour la réparation est présenté dans la figure 3.48 p. 74. Les graphiques intermédiaires de l'étude (relatifs à la *cptable*) sont présentés en annexe.

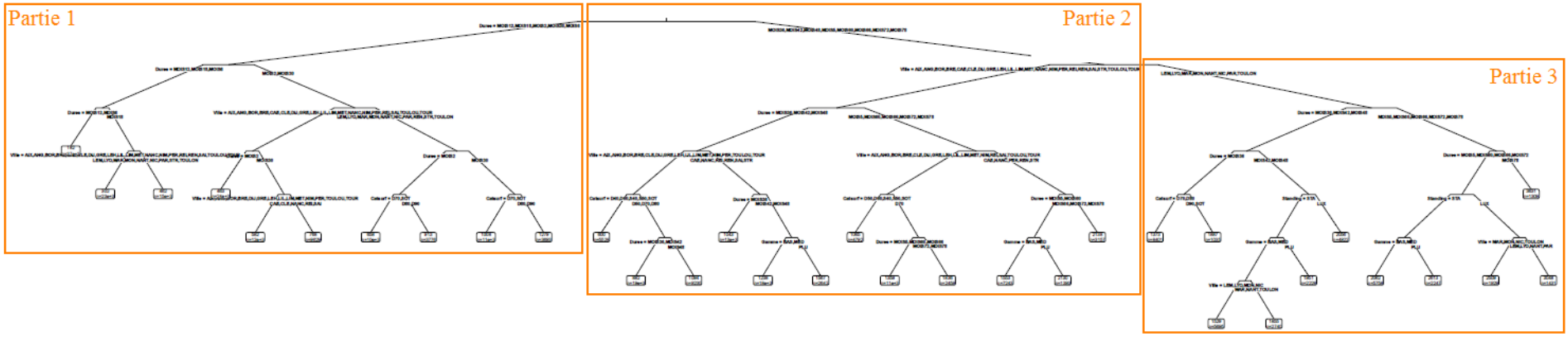

FIGURE 3.44 – Arbre élagué pour l'Entretien

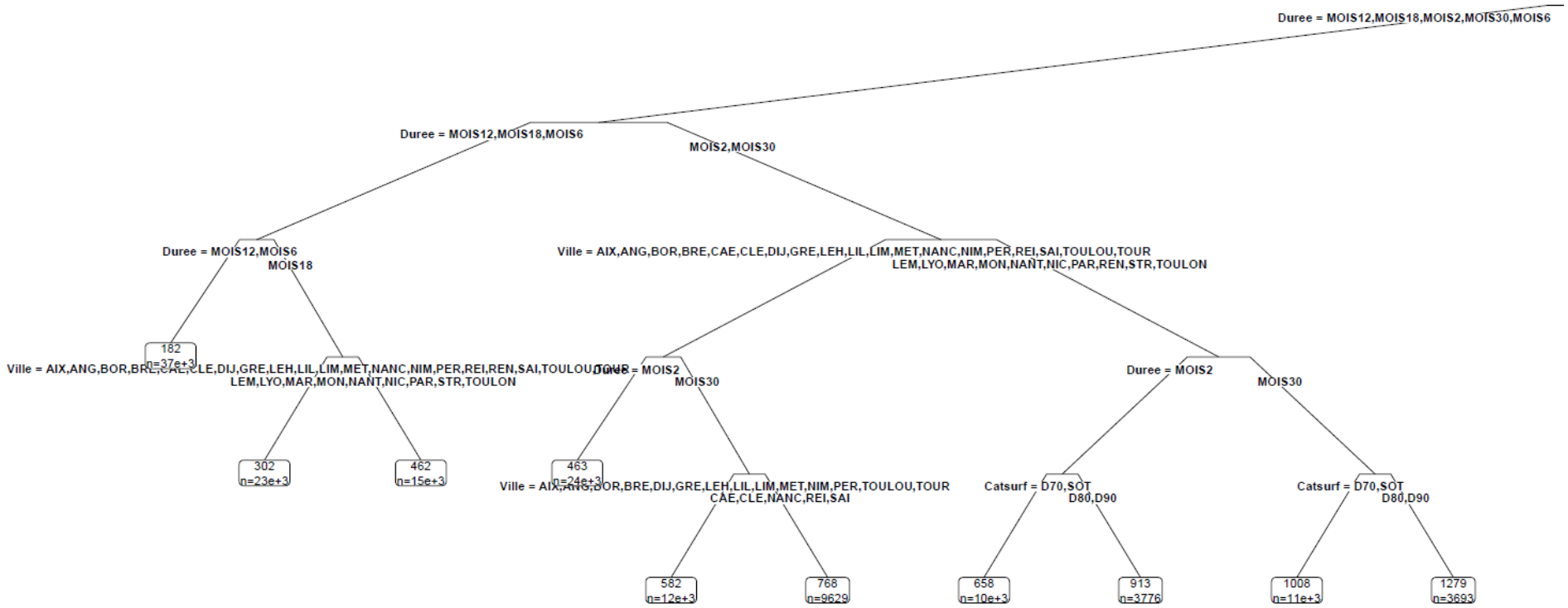

FIGURE 3.45 – Arbre élagué pour l'Entretien - Partie 1

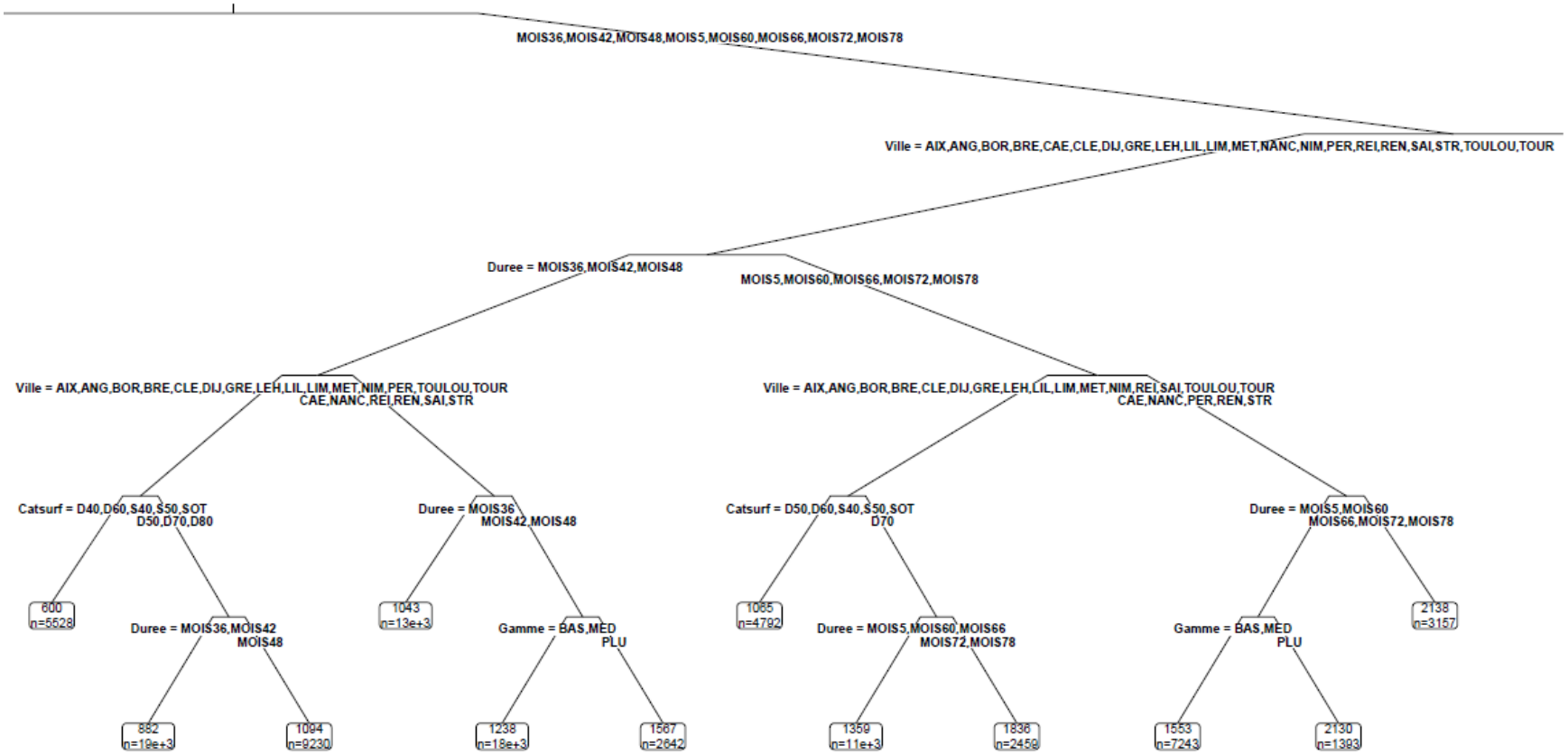

FIGURE 3.46 – Arbre élagué pour l'Entretien - Partie 2

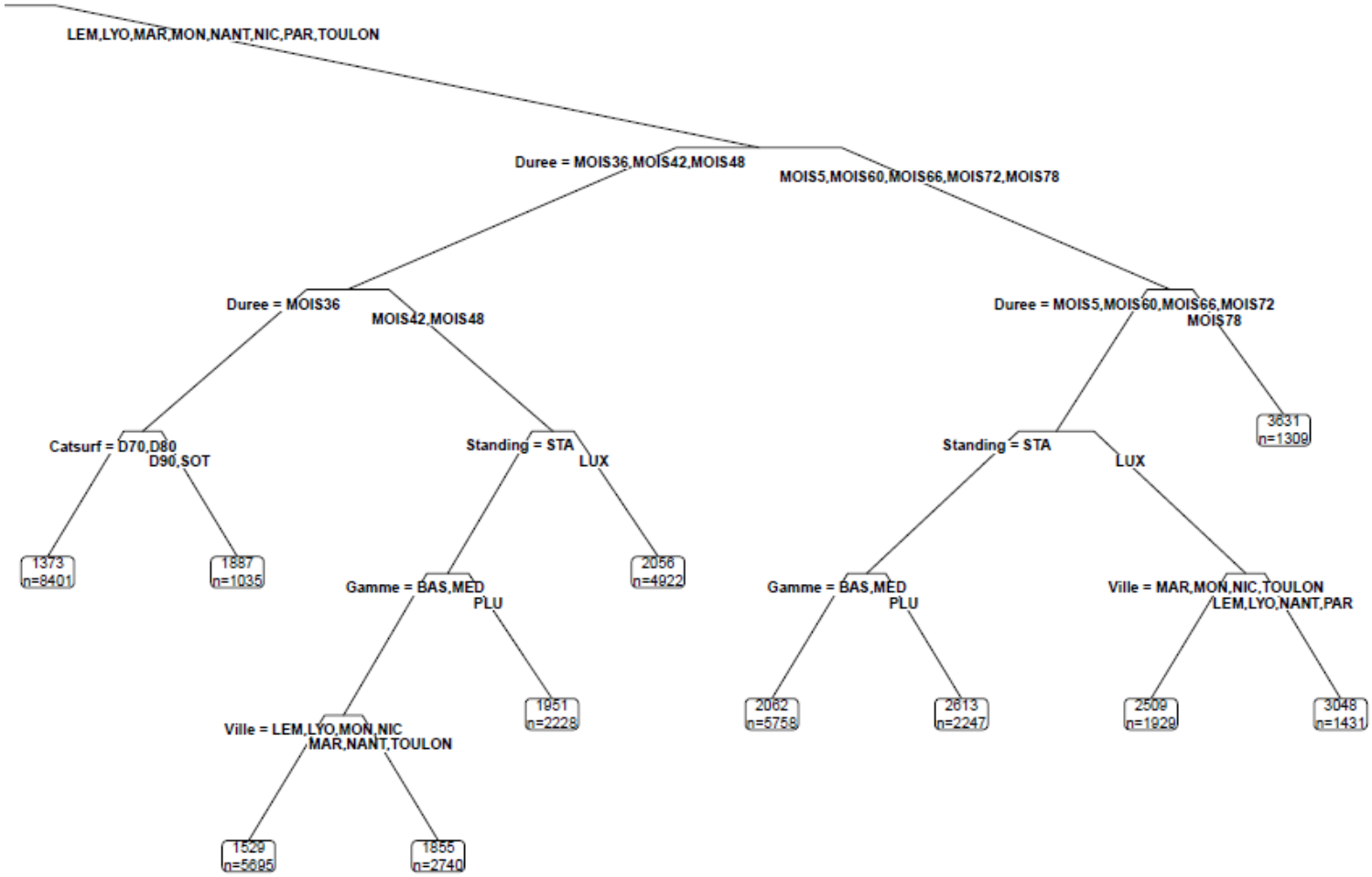

FIGURE 3.47 – Arbre élagué pour l'Entretien - Partie 3

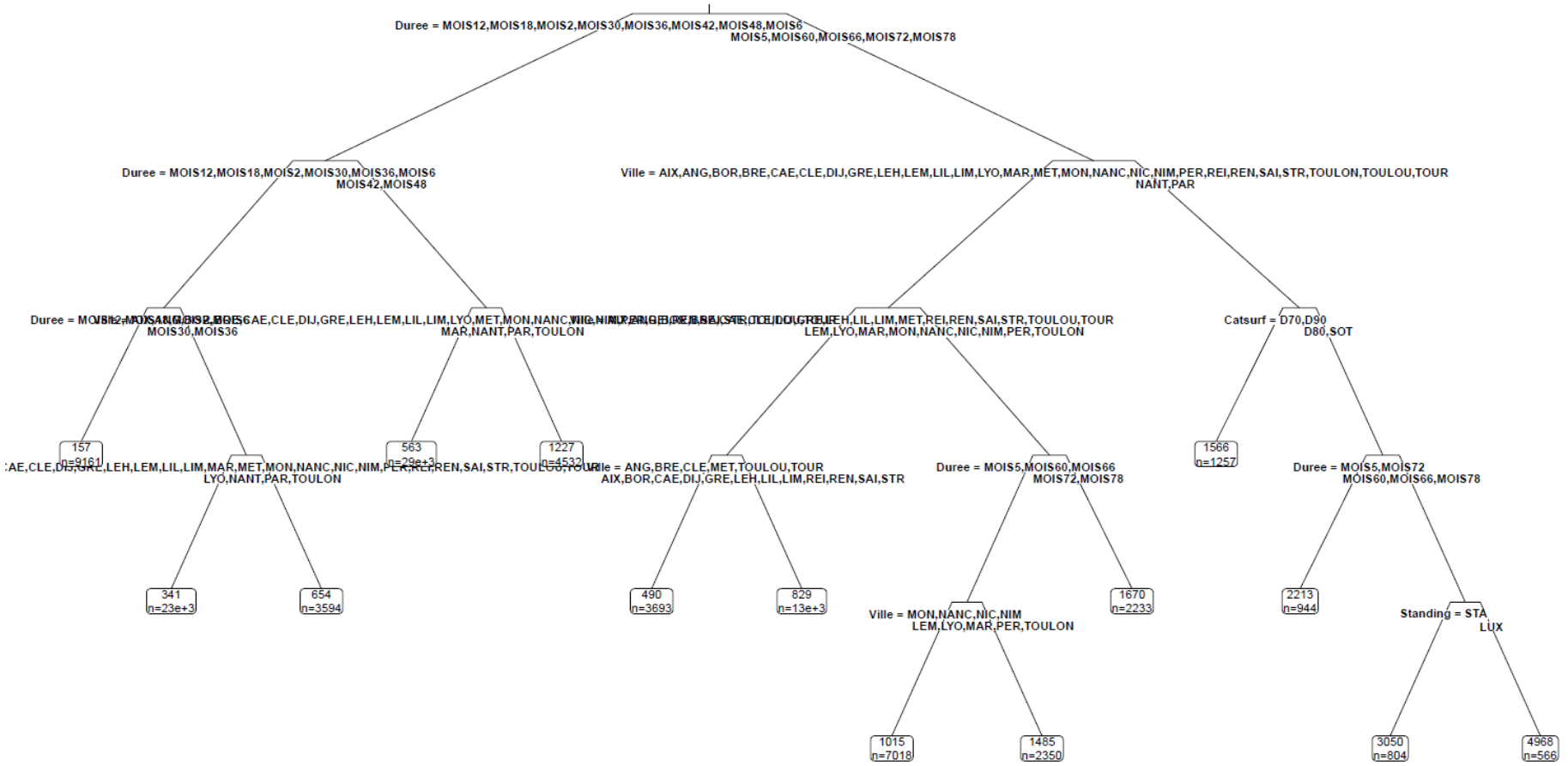

FIGURE 3.48 – Arbre élagué pour la Réparation

La première remarque que l'on peut faire porte sur l'importance hiérarchique de chaque variable explicative. Il est clair que la durée du contrat d'assurance apparaît comme étant la variable expliquant au plus le coût du contrat d'assurance. En effet, celle-ci se retrouve dans la plupart des séparations le plus en amont de l'arbre. La ville semble être la deuxième variable la plus discriminante.

A l'inverse, des variables comme le standing de la suite ou la variable d'interaction catégorie-superficie interviennent beaucoup plus bas dans l'arbre, et n'apparaissent même pas du tout sur certains chemins.

D'autre part, la lecture de l'arbre nous permet de mettre en évidence des variables discriminantes différentes selon la sous-population considérée. Par exemple, la seconde variable d'influence chez les contrats longs terme (partie droite de la première séparation) est la ville, tandis que pour les contrats court terme (partie gauche), la séparation se fait de nouveau par rapport à la durée du contrat.

En ce qui concerne l'arbre de la Réparation, les variables les plus discriminantes sont également la durée du contrat d'assurance, et la localité de la suite d'hôtel. Par contre, la variable *Gamme* n'apparaît sur aucun chemin de l'arbre : elle n'a pas été considérée comme significativement explicative par l'algorithme.

3.5.5 Analyse des résidus

Pour les Modèles Linéaires Généralisés, certains graphiques (comme le QQ-plot) avaient pour but de vérifier que les résidus étaient approximativement de loi normale. Puisqu'aucune loi n'a été spécifiée a priori pour l'analyse par CART, il n'y a pas lieu de vérifier la normalité des résidus. Par contre, il peut être intéressant d'observer la répartition de ceux-ci, et de vérifier qu'ils sont quand même en majorité assez proches de 0.

Pour cela, nous calculons quelques statistiques simples (table 3.8 p. 75) et traçons une boîte à moustaches (figure 3.49 p. 76) ainsi qu'un histogramme (table 3.50 p. 76) :

```
res<-residuals(elague,type="deviance")
summary(res)
boxplot(res,range=0,horizontal=TRUE)
hist(res,xlab="Résidus de Déviance",ylab="Fréquence",main="")
```

Minimum	Premier quartile	Médiane	Moyenne	Troisième quartile	Maximum
-3540.00	-329.30	-96.02	0.00	207.90	24100.00

TABLE 3.8 – Principales statistiques des résidus de Déviance pour l'algorithme CART - Entretien

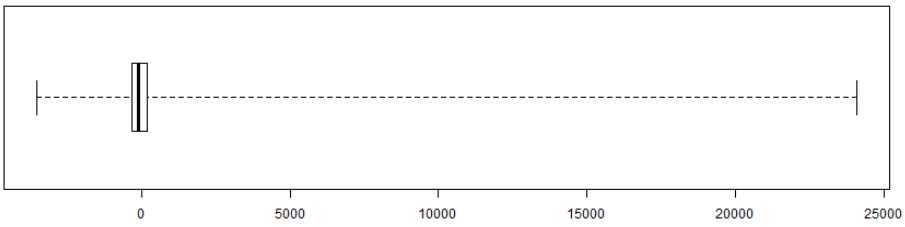

FIGURE 3.49 – Boîte à moustache des résidus de Déviance pour l'algorithme CART - Entretien

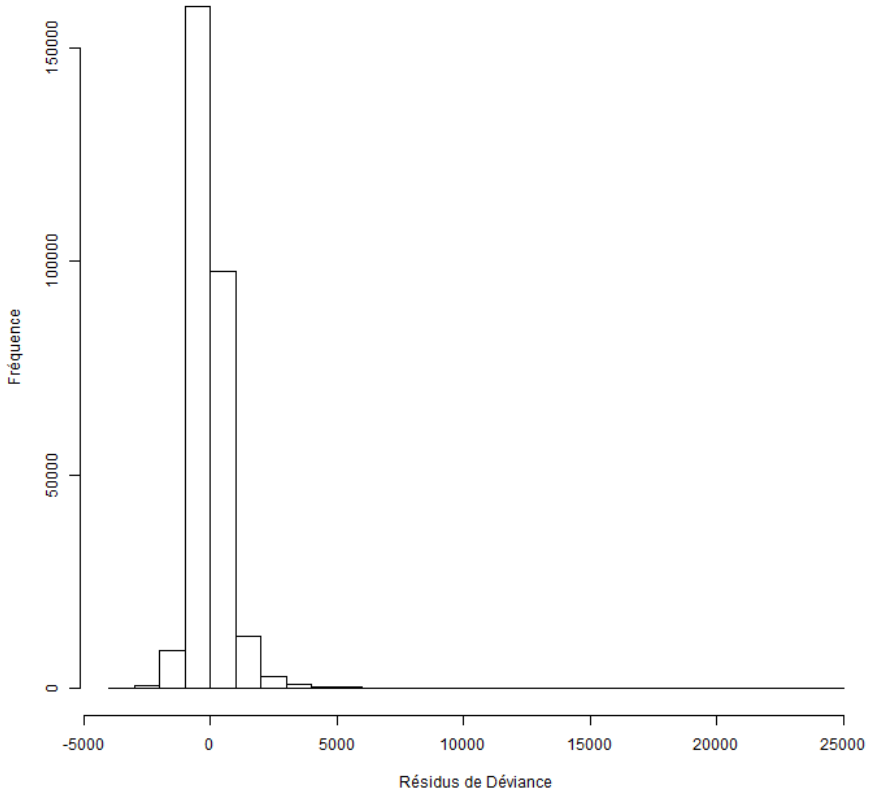

FIGURE 3.50 – Répartition des résidus de Déviance pour l'algorithme CART - Entretien

La boîte à moustaches nous indique que 50% des résidus se trouvent dans le rectangle (tracé entre le premier et le troisième quartile). De plus, la médiane est très proche de 0.

L'histogramme met également en évidence une forte concentration en 0.

Quatrième partie

Comparatif des méthodes employées

4.1 Comparaison des résultats obtenus

4.1.1 L'Erreur Quadratique Moyenne

Notions théoriques

Pour comparer la qualité de deux estimateurs, on se base en principe sur l'utilisation de l'Erreur Quadratique Moyenne (appelée Mean Squared Error en anglais, ou MSE), définie mathématiquement de la manière suivante :

$$MSE(\hat{\theta}|\theta) = \mathbb{E}[(\hat{\theta} - \theta)^2]$$

où $\hat{\theta}$ est l'estimateur de la variable θ.

On démontre facilement [6] que l'Erreur Quadratique Moyenne d'un estimateur peut s'exprimer de façon très simple en fonction de sa variance et de son biais :

$$MSE(\hat{\theta}|\theta) = Biais(\hat{\theta})^2 + Var(\hat{\theta})$$

avec $Biais(\hat{\theta}) = \mathbb{E}[\hat{\theta} - \theta]$ et $Var(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - \mathbb{E}[\hat{\theta}])^2]$

En faisant intervenir le biais et la variance, l'Erreur Quadratique Moyenne permet donc de trancher dans une situation où il existe un estimateur sans biais et un autre biaisé mais de variance plus petite.

En pratique, pour calculer l'Erreur Quadratique Moyenne, les erreurs individuelles sont tout d'abord élevées au carré, puis additionnées les unes aux autres. On divise ensuite le résultat obtenu par le nombre total d'erreurs individuelles. On obtient ainsi l'erreur empirique.

Pour chaque observation i, notons y_i la valeur observée du coût, et Y_i la valeur prédite par le modèle. L'erreur empirique est alors calculée de la manière suivante :

$$MSE_{empirique} = \frac{1}{n} \times \sum_{i=1}^{n} (Y_i - y_i)^2$$

D'après la loi des grands nombres, l'erreur empirique tend vers l'erreur théorique lorsque le nombre d'observations n tend vers l'infini.

Application

Rappelons que jusqu'à présent, nous n'avons utilisé que la base d'apprentissage pour déterminer les modèles, que ce soit le MLG ou l'algorithme CART. Nous avons volontairement mis de côté une base de test, composée de 25% des données. Ces données n'ayant en aucune manière participé à l'élaboration des modèles, elles constituent une base totalement impartiale pour tester les performances des modèles en terme de MSE.

Pour l'algorithme CART, nous utilisons la fonction *pred* pour appliquer l'arbre de décision sur la base de test, afin d'obtenir le vecteur des estimations du coût. Les commandes sont les suivantes :

6. Une démonstration est notamment disponible sur le site internet de Wikipédia : *http ://fr.wikipedia.org/wiki/Erreur_quadratique_moyenne*

```
test<-read.table("C:/Users/Camille/Desktop/Memoire_data/test_ent.txt",
sep="",header=TRUE)

pred<-predict(elague,test)
# elague est l'arbre de décision construit, test est la nouvelle base

mse<-sum((pred-test$amount_ent_cumul)*(pred-test$amount_ent_cumul))/
length(pred)

mse
[1] 547588.2
```

Pour le MLG, nous reprenons la base de test en SAS, à laquelle nous ajoutons les coefficients et l'intercept estimés par le modèle. Ainsi, pour chaque observation de la base de test, nous lui associons son estimation du coût par MLG. Nous appliquons ensuite la formule de calcul et obtenons un MSE de 534173,3.

Nous avons aussi effectué les mêmes calculs sur la base d'apprentissage, en nous attendant logiquement à avoir des MSE plus faibles que sur la base de test, puisque les modèles ont été construits par rapport à la base d'apprentissage.

Les résultats pour l'Entretien et la Réparation sont regroupés dans les tableaux suivants (tableaux 4.9 p. 81 et 4.10 p. 81) :

Modèle	MSE Apprentissage	MSE Test
MLG	521 634.8	534 173.3
CART	533 855.6	547 588.2

TABLE 4.9 – MSE sur les bases de test et d'apprentissage - Entretien

Modèle	MSE Apprentissage	MSE Test
MLG	1 257 418.2	1 288 093.7
CART	1 266 727.0	1 290 365.0

TABLE 4.10 – MSE sur les bases de test et d'apprentissage - Réparation

D'après ces résultats, le MLG semble plus performant en terme de MSE pour prédire le coût sur des nouvelles données. Toutefois, les différences restent relativement minimes entre les deux modèles : sur la base de test, le MSE du MLG est inférieur d'environ 2.4% (Entretien) et 0.2% (Réparation) à celui de CART.

4.1.2 Comparaison des estimations

Comparaison sur les profils représentatifs

Nous avons mis en oeuvre deux méthodes pour modéliser de deux manières différentes une variable de coût. Pour comparer les deux résultats fournis, nous allons sélectionner un certain nombre de profils de risques parmi ceux les plus représentés dans la base d'apprentissage.

Nous avons déterminé à l'aide d'une procédure *Freq* sous SAS les 10 profils de risque les plus représentés dans la base d'apprentissage, sur les interventions d'entretien. Le tableau 4.11 p. 82 résume les résultats obtenus sur ces profils. Le tableau analogue pour les interventions de réparation se trouve en annexes.

Profil de risque					Nb. obs.	CART	MLG	Ecart
Durée	Ville	Standing	Gamme	Catsurf				
18 mois	Strasb.	Standard	Basic	D70	3 389	462 €	399 €	-14%
30 mois	Strasb.	Standard	Basic	D70	3 336	1008 €	886 €	-12%
24 mois	Strasb.	Standard	Basic	D70	2 988	658 €	617 €	-6%
36 mois	Strasb.	Standard	Basic	D70	2 915	1043 €	1113 €	7%
36 mois	Nancy	Standard	Basic	D70	2 733	1043 €	1009 €	-3%
30 mois	Nancy	Standard	Basic	D70	2 710	768 €	803 €	5%
42 mois	Strasb.	Standard	Basic	D70	2 492	1238 €	1304 €	5%
24 mois	Nancy	Standard	Basic	D70	2 355	463 €	559 €	21%
48 mois	Strasb.	Standard	Basic	D70	2 286	1238 €	1499 €	21%
30 mois	Dijon	Standard	Basic	D70	2 255	582 €	666 €	14%

TABLE 4.11 – Comparaison des résultats MLG et CART sur 10 profils - Entretien

Sur ces profils de risque, les résultats sont acceptables : les écarts sont équilibrés (pas tous positifs ni tous négatifs) et assez faibles sur 5 de ces profils (entre 3 et 7 % en valeur absolue). Ces écarts doivent par ailleurs être relativisés compte tenu du nombre d'observations sur lesquelles les calculs sont effectués (rappelons que la base d'apprentissage possède un total de 283 653 observations).

Comparaison par variable

Dans un deuxième temps, nous nous sommes intéressés aux résultats variable par variable. Le tableau 4.12 p. 83 contient les résultats selon la variable *Duree*, et les écarts sont calculés sur chaque niveau de la variable. Par exemple, sur les observations ayant une durée de 18 mois (quelles que soient les valeurs prises par les autres variables), nous avons résumé le coût moyen renvoyé par CART et celui renvoyé par le MLG. L'écart est alors de 1€, soit 0%.

		Nombre d'observations	Moyenne CART (€)	Moyenne MLG (€)	Ecart
Par durée	6 mois	10 632	182	104	-43 %
	12 mois	26 212	182	214	18 %
	18 mois	37 277	365	364	0 %
	24 mois	37 953	560	557	-1 %
	30 mois	36 538	831	825	-1 %
	36 mois	35 629	1 062	1 041	-2 %
	42 mois	29 142	1 272	1 230	-3 %
	48 mois	27 663	1 336	1 407	5 %
	54 mois	15 841	1 706	1 587	-7 %
	60 mois	14 171	1 705	1 777	4 %
	66 mois	4 449	1 864	1 989	7 %
	72 mois	3 904	1 957	2 112	8 %
	78 mois et plus	4 242	2 348	2 532	8 %
Au global		283 653	920	922	0 %

TABLE 4.12 – Comparaison des résultats selon la durée du contrat d'assurance - Entretien

Les résultats doivent de nouveau être relativisés par rapport au nombre des observations concernées par le calcul. Dans la majorité, les résultats sont satisfaisants compte tenu du nombre d'observations.

Nous pouvons noter qu'au global, c'est-à-dire sur toutes les observations de la base d'apprentissage, les résultats des deux méthodes coïncident parfaitement, avec un écart de 2€.

Les tableaux 4.13 p. 84 à 4.16 p. 85 représentent les mêmes résultats pour les 4 autres variables : $Ville$, $Standing$, $Gamme$ et $Catsurf$.

Les résultats analogues pour la réparation sont présentés en annexes.

		Nombre d'observations	Moyenne CART (€)	Moyenne MLG (€)	Ecart
Par ville	Aix-en-Provence	1 620	678	693	2 %
	Angers	3 592	659	644	-2 %
	Bordeaux	2 630	614	570	-7 %
	Brest	3 108	553	554	0 %
	Caen	18 402	906	830	-8 %
	Clermont-Ferrand	9 797	704	775	10 %
	Dijon	38 911	703	710	1 %
	Grenoble	1 405	836	619	-26 %
	Le Havre	801	781	767	-2 %
	Le Mans	2 788	1 708	1 546	-9 %
	Lille	18 830	671	685	2 %
	Limoges	2 320	645	513	-20 %
	Lyon	3 907	1 286	1 229	-4 %
	Marseille	3 491	1 565	1 655	6 %
	Metz	1 122	675	631	-7 %
	Montpellier	10 438	1 140	1 085	-5 %
	Nancy	35 294	854	864	1 %
	Nantes	16 407	1 483	1 503	1 %
	Nice	29 378	1 181	1 212	3 %
	Nimes	5 899	762	771	1 %
	Paris	1 695	1 580	2 264	43 %
	Perpignan	2 705	551	573	4 %
	Reims	1 553	745	743	0 %
	Rennes	977	820	826	1 %
	Saint-Etienne	9 145	895	905	1 %
	Strasbourg	36 780	905	925	2 %
	Toulon	5 695	1 353	1 409	4 %
	Toulouse	1 494	742	609	-18 %
	Tours	13 469	685	598	-13 %
Au global		283 653	920	922	0 %

TABLE 4.13 – Comparaison des résultats selon la localité de la suite d'hôtel - Entretien

		Nombre d'observations	Moyenne CART (€)	Moyenne MLG (€)	Ecart
Par standing	Luxe	45 875	1 133	1 194	5 %
	Standard	237 778	879	870	-1 %
Au global		283 653	920	922	0 %

TABLE 4.14 – Comparaison des résultats selon le standing de la suite d'hôtel - Entretien

		Nombre d'observations	Moyenne CART (€)	Moyenne MLG (€)	Ecart
Par gamme	Basic	159 093	835	853	2 %
	Medium	84 674	965	874	-9 %
	Plus	39 886	1 161	1 304	12 %
Au global		283 653	920	922	0 %

TABLE 4.15 – Comparaison des résultats selon la gamme du produit d'assurance - Entretien

		Nombre d'observations	Moyenne CART (€)	Moyenne MLG (€)	Ecart
Par catégorie et superficie	D40	1 249	360	276	-23 %
	D50	11 959	706	685	-3 %
	D60	21 420	408	375	-8 %
	D70	205 018	904	905	0 %
	D80	24 542	1 526	1 583	4 %
	D90	7 536	1 470	1 703	16 %
	S40	4 635	669	501	-25 %
	S50	2 638	653	465	-29 %
	S Others	4 656	984	900	-9 %
Au global		283 653	920	922	0 %

TABLE 4.16 – Comparaison des résultats selon la catégorie-superficie de la suite d'hôtel - Entretien

Comparaison globale

Enfin, nous pouvons également tracer un graphique représentant les estimations MLG en fonction des estimations CART, pour toutes les observations de la base d'apprentissage (figure 4.51 p. 86). Nous pourrons ainsi visualiser plus facilement si les deux méthodes donnent des résultats similaires, i.e. si les points sont situés en majorité autour de la droite d'équation $y = x$ (tracée en rouge sur le graphique).

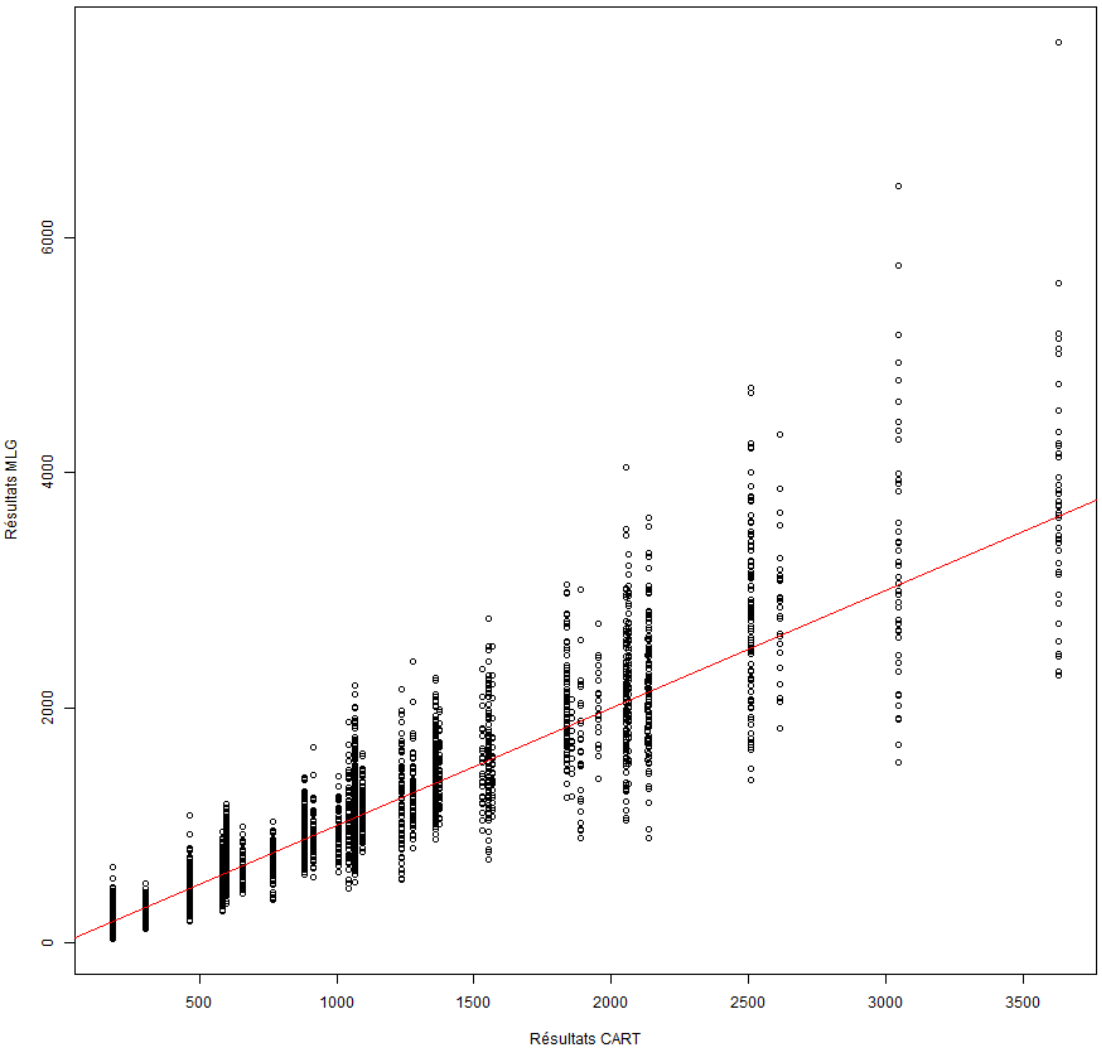

FIGURE 4.51 – Comparaison des résultats MLG et CART - Entretien

Les résultats sont plutôt satisfaisants, même si l'on observe un cône de croissance : plus les coûts sont élevés, plus les résultats divergent. On constate tout de même que la majorité des points se situent autour de la première bissectrice.

Concernant les résultats CART, nous pouvons noter que les différents paliers apparaissant sur le graphique correspondent aux estimations contenues dans les feuilles de

l'arbre. Nous pouvons en effet vérifier qu'il y a bien 33 paliers et 33 noeuds terminaux dans l'arbre (figure 3.44 p. 70).

4.1.3 Qualité des estimations

Après avoir comparé les deux méthodes entre elles, nous pouvons aussi nous pencher sur la qualité d'estimation de chacune d'elles prise séparément. Pour cela, nous allons tracer un graphique représentant les estimations en fonction des observations de la base de test, pour chaque méthode (figures 4.52 p. 87 et 4.53 p. 88). Nous nous attendons à ce que les points soient en majorité situés autour de la droite d'équation $y = x$, tracée en rouge sur chaque graphique. Notons qu'en appliquant la même étude sur la base d'apprentissage, nous obtenons sensiblement les mêmes graphiques, qui ne seront donc pas représentés ici.

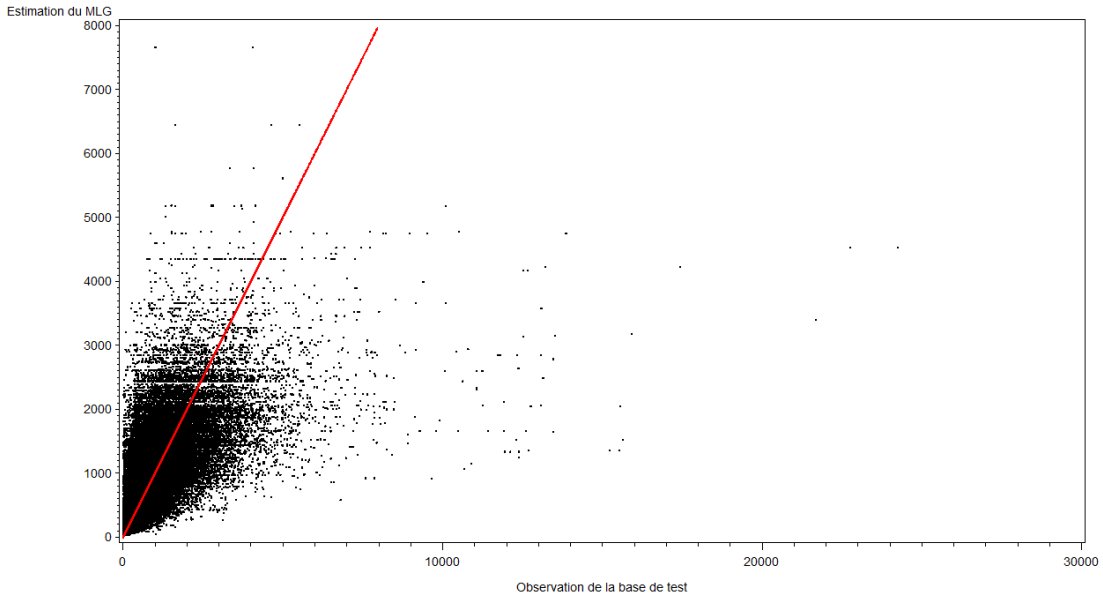

FIGURE 4.52 – Estimations fournies par le MLG pour les données de la base de test - Entretien

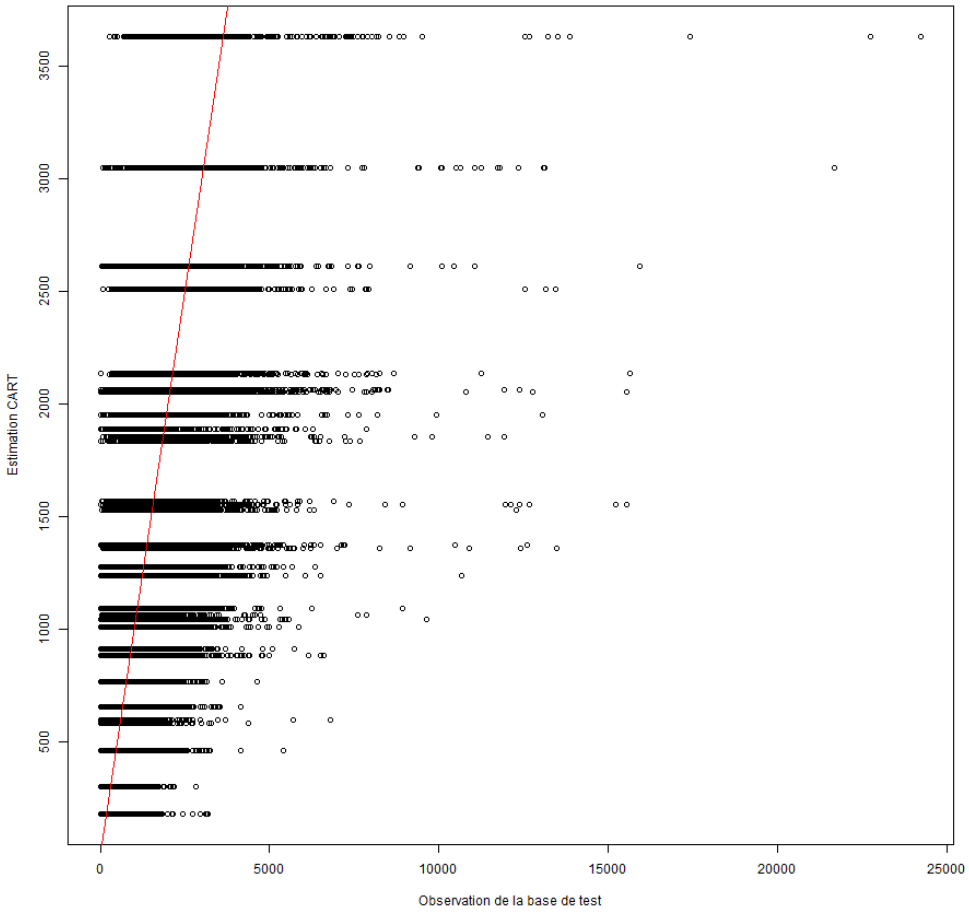

FIGURE 4.53 – Estimations fournies par l'algorithme CART pour les données de la base de test - Entretien

On observe de nouveau un cône de croissance pour les deux méthodes, même si une majorité des points se situe bien autour de la première bissectrice. Les écarts entre les observations et les estimations semblent du même ordre de grandeur pour le MLG et l'algorithme CART, ce qui ne nous permet pas de trancher quant à une meilleure qualité de prédiction de l'une ou l'autre méthode.

On peut également remarquer que pour CART, la valeur maximale des prédictions se situe vers 3700 € (on peut d'ailleurs lire la valeur exacte sur la figure 3.47 p. 73, dans la feuille la plus à droite : 3631 €), alors que pour le MLG, la valeur maximale est aux alentours de 7700 €.

Bien qu'on lui fournisse des données déjà remises en classes (par exemple les durées de contrat regorupées par tranches de 6 mois), l'algorithme CART effectue une sorte de deuxième remise en classes. En effet, deux profils différents du point de vue de la remise en classe initiale peuvent tout de même suivre le même chemin sur l'arbre et alors aboutir à la même estimation. Cela conduit donc CART à être moins précis que le MLG, et à plus regrouper les variables. Cela intervient notamment sur les valeurs les moins

représentées, i.e. les valeurs extrêmes, d'où une estimation maximale bien inférieure à celle du MLG.

4.2 Avantages et inconvénients de chaque méthode

4.2.1 Finesse des estimations

La méthode CART ne permet pas une granularité aussi fine que les MLG. En effet, plusieurs profils de risques différents peuvent aboutir au même résultat s'ils suivent le même chemin de l'arbre. L'arbre de décision n'est donc pas aussi précis. Cela est dû à l'élagage, où nous avons supprimé les branches superflues qui n'apportaient pas d'intérêt significatif.

A l'inverse, le MLG fournit un coefficient différent pour chaque modalité de chaque variable. Ainsi, deux profils de risques différents auront toujours un résultat différent. Cela peut être considéré comme un avantage des MLG.

4.2.2 Facilité de mise en oeuvre

Vitesse d'exécution

Les deux méthodes ont été exécutées sur une même machine dont les caractéristiques sont les suivantes :

- Processeur : Intel® Core 2 Duo T6500 2.10 GHz
- Mémoire vive : 4 Go 800 MHz DDR2
- Disque dur : SATA 5400 tours minute
- Système d'exploitation : Windows Vista Edition Familiale Premium 64 bits

Pour les logiciels utilisés, nous avons bénéficié de SAS version 9.2 et R version 3.0.1.

La première partie concernant tous les retraitements sur la base de données a été exécutée sous SAS, et nécessite environ 1h. Pour l'application des MGL, la procédure *Genmod* s'effectue assez rapidement en environ 20s (tableaux de sortie et tracé des graphiques des mutliplicateurs). L'algortithme CART s'exécute également très rapidement, environ 15s pour l'application de la fonction *rpart*, la validation croisée, l'élagage et le tracé de l'arbre élagué final.

De ce point de vue, les temps de calcul sont tout à fait raisonnables et similaires. L'étape la plus longue est la mise en forme et les retraitements de la base de donnée, mais ceci représente un prérequis commun pour les deux méthodes.

Documentation disponible

De nombreux tutoriels et documentations pratiques sont disponibles sur Internet, que ce soit pour les MLG ou l'algorithme CART. J'ai surtout fait appel à cette aide en ligne pour l'algorithme CART que je ne connaissais pas, et les exemples d'application avec *Rpart* ne manquent pas. La documentation théorique est également très développée pour les deux méthodes.

Gestion des valeurs manquantes

Lorsqu'une donnée est manquante, la plupart des méthodes statistiques ignorent l'observation relative. C'est le cas notamment avec la procédure *Genmod* de SAS qui indique dans la sortie le nombre d'observations qui n'ont pas été prises en compte dans la modélisation à cause de valeurs manquantes.

L'algorithme CART propose une alternative pour palier au manque d'information sur certaines observations. Quand, pour une observation, une donnée est manquante pour une variable divisant un segment, l'algorithme cherche la variable la remplaçant au mieux. Le nombre maximum de variables de substitution peut être modifié dans les paramètres de CART (paramètre *maxsurrogate*).

Dans notre cas, nous avons retraité et nettoyé les données au préalable dans SAS, de manière à ce qu'il n'y ait plus de valeurs manquantes ou incohérentes. Cette solution proposée par CART est donc inutile pour cette étude, mais reste un atout par rapport à d'autres méthodes statistiques classiques.

Gestion des variables continues

Lors de l'étape de la remise en classes, nous avons volontairement transformé les variables continues en variables catégorielles (par exemple, la durée du contrat d'assurance). En effet, cette étape est indispensable pour le MLG, qui n'a pas la capacité de traiter les variables explicatives continues. En revanche, CART peut s'abstenir de cette étape. Nous l'avons vu dans l'exemple introductif du joueur de tennis, l'algorithme a effectué tout seul les séparations selon les variables qu'il jugeait les plus discriminantes, et lorsqu'une de ces variables était continue (c'était le cas de la température), il a automatiquement trouvé le point de coupure pour la séparation.

Pour pouvoir comparer correctement les deux méthodes, nous avons conservé la remise en classes qui avait été effectuée pour le MLG, mais cela n'était pas obligatoire pour le bon fonctionnement de CART.

4.3 Quid d'une mise en application réelle d'une tarification CART par un assureur ?

Etant largement reconnus et éprouvés, la grande majorité des assureurs non-vie utilisent les Modèles Linéaires Généralisés pour tarifer leurs contrats.

Faisons l'hypothèse que deux assureurs proposent le même produit sur le marché, mais que l'un tarifie de manière classique avec un MLG, et l'autre utilise une méthode par arbre de décision (CART). Vers quel assureur se porterait le choix du client ? Quel est le risque pour l'assureur qui tarifie avec CART ?

4.3.1 Antisélection

A garanties et services équivalents, nous pouvons faire l'hypothèse que le client est rationnel et choisit l'assureur offrant le prix le plus bas. Comme nous l'avons vu dans la comparaison des résultats entre les méthodes MLG et CART (partie 4.1.2 p. 82), il n'y a pas de différence nette de l'une par rapport à l'autre : nous n'avons pas les résultats MLG qui sont systématiquement plus hauts que les résultats CART, ni l'inverse. Ainsi,

un assureur utilisant CART ne prend pas forcément le risque de n'attirer aucun client, ce qui pourrait être le cas s'il choisissait une méthode de tarification qui lui donne des prix systématiquement plus hauts que les pratiques du marché (le marché étant largement guidé par les MLG).

Cependant, les différences de prix peuvent donner lieu à de l'antisélection. Reprenons le graphique de comparaison des résultats MLG et CART pour l'entretien, et intéressons-nous par exemple à une des classes de tarif proposées par CART (figure 4.54 p. 91, classe de tarif entourée en rouge).

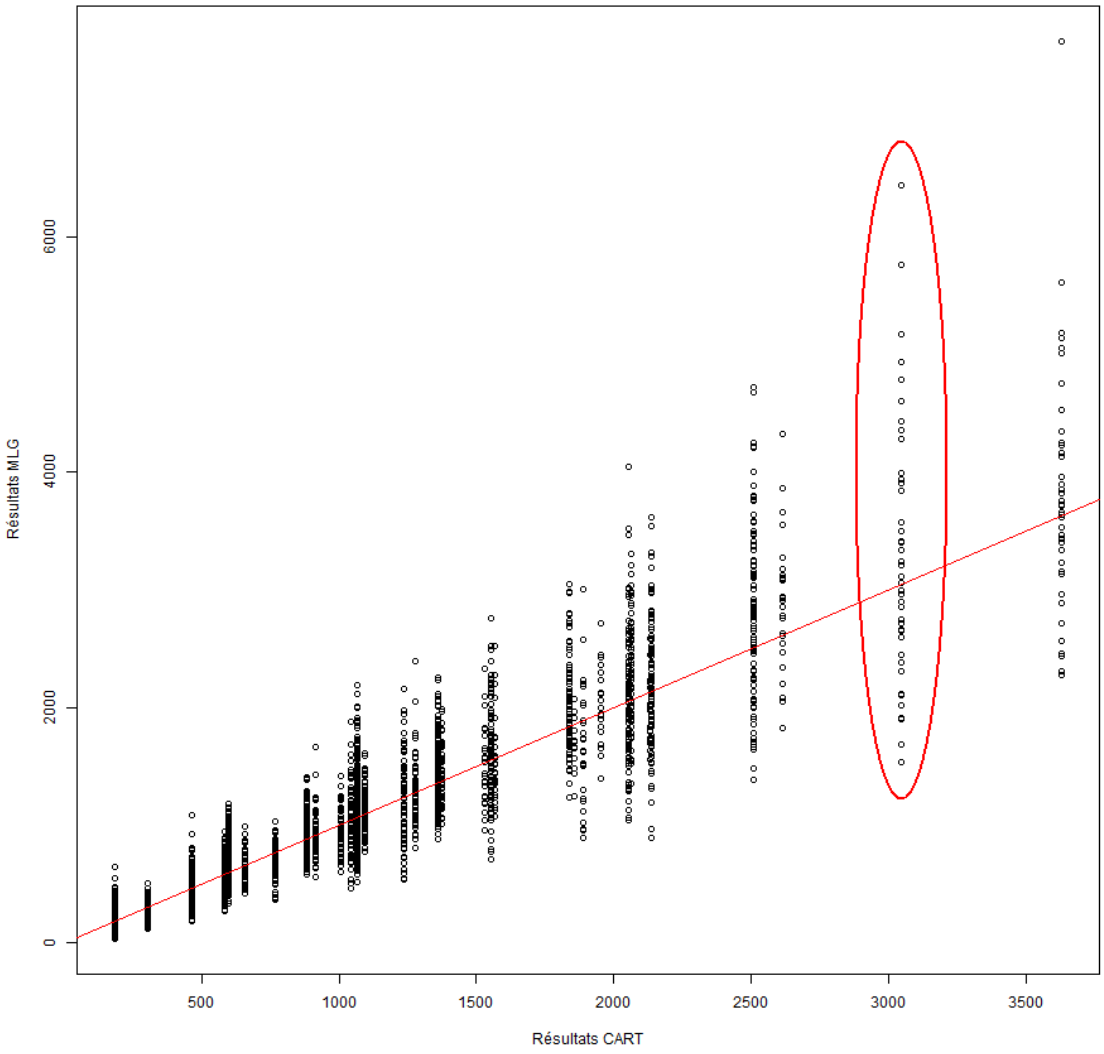

FIGURE 4.54 – Mise en évidence de l'antisélection - Entretien

D'après les informations données par l'arbre de régression pour l'entretien (figure 3.47 p. 73), nous savons que le tarif entouré en rouge est plus exactement de 3048€, et qu'il correspond aux profils de risque suivants :

– La suite d'hôtel se situe à Le Mans, Lyon, Nantes ou Paris
– Il s'agit d'un standing de Luxe
– La durée du contrat est de 54, 60, 66 ou 72 mois

En particulier, le tarif ne dépend pas de la gamme du produit d'assurance, ni de la catégorie-superficie de la suite d'hôtel. Il représente donc une moyenne des tarifs de tous les "sous-profils de risque" possibles.

A l'inverse, le MLG fournit un tarif distinct pour tous ces sous-profils : on compte 43 points sur la même verticale du graphique. Le MLG propose donc 43 tarifs différents quand CART n'en propose qu'un seul.

Ainsi, les profils de risque pour lesquels le MLG offre un tarif plus faible (i.e. les points de la verticale situés en dessous de la droite d'équation $y = x$) iront donc vers l'assureur qui tarifie par MLG. L'assureur utilisant CART se retrouvera alors avec les profils de risque les plus élevés (au sens du MLG : ce sont les tarifs les plus hauts parmi les 43 points). Le nouveau tarif moyen de la classe considérée est alors supérieur à 3048€, et ces 3048€ ne sont plus suffisants pour couvrir les risques. L'assureur utilisant CART est en sous-tarification.

C'est le phénomène d'antisélection, qui se répète selon le même processus pour toutes les classes de tarification de CART.

4.3.2 Choix des variables et remise en classes

Nous avons vu au travers de ce projet que les MLG offraient une plus grande précision, avec des estimations différentes pour des profils de risque distincts. Cela est bien sûr fonction de la remise en classes, que l'on effectue plus ou moins fine.

Néanmoins, la même remise en classes a été appliquée aux deux méthodes, mais CART a effectué automatiquement une deuxième remise en classes, estimant que le degré de finesse était trop important par rapport à la qualité de prédiction exigée.

Nous avons vu également qu'il était possible de rendre CART plus fin en construisant un arbre plus grand (via l'utilisation du complexity parameter *cp*).

L'assureur a donc la possibilité, que ce soit avec les MLG ou avec CART, de choisir le degré de finesse qu'il souhaite donner à son modèle. Toutefois, le fait que CART s'occupe lui-même du choix des variables explicatives et de leur remise en classes peut apparaître comme un sérieux inconvénient technique pour l'assureur.

Nous avons vu par exemple que pour l'arbre en Réparation (figure 3.48 p. 74), CART n'avait pas considéré la variable *Gamme* comme significative pour le modèle. Si l'assureur a choisi d'inclure cette variable par exemple pour des raisons commerciales, il est dans l'incapacité de fournir un tarif avec CART dépendant de cette variable.

Ainsi, l'avantage de l'apprentissage statistique, qui comme son nom l'indique apprend seul, peut se révéler contraignant pour l'assureur, qui ne peut alors plus introduire son jugement d'expert dans le modèle.

Conclusion

Nous avons abordé à travers ce mémoire deux méthodes d'analyse de coût en assurance non-vie : d'une part les Modèles Linéaires Généralisés, déjà bien installés dans le domaine, et d'autre part l'algorithme Classification And Regression Trees, appartenant à la branche de l'apprentissage statistique, et encore méconnu de la plupart des acteurs de l'assurance.

Nous avons mis en pratique ces deux approches, afin de les confronter en termes de qualité d'estimation, et d'apprécier les points forts et les points faibles de chacune.

D'après nos résultats, le MLG s'est avéré meilleur en prédiction. Cela peut être dû à l'élagage de l'arbre de décision, qui a été trop brutal. Une solution envisageable est de revenir à l'étape d'élagage, et de choisir un *cp* plus élevé de manière à obtenir un arbre un peu plus complexe et précis.

Par ailleurs, même si le MLG a été construit de manière plus fine, nous avons vu qu'il n'a pas perdu pour autant sa qualité neutre vis à vis de nouvelles données, puisqu'il obtient un meilleur MSE sur la base de test par rapport à l'algorithme CART.

Encore une fois, toute la difficulté de l'application de CART réside dans l'arbitrage à faire entre la complexité et la précision.

Nos travaux ont également mis en évidence certains résultats inattendus, tels que les cônes de croissance sur les graphiques des estimations en fonction des observations. Deux possibilités peuvent expliquer ce phénomène.

D'une part, le fait de ne pas isoler dès le départ les éventuelles valeurs extrêmes pourrait être la cause de ces dérives. En effet, elles sont restées intégrées à l'étude générale, et ont donc certainement influencé les résultats, d'où des graphiques s'éloignant parfois des structures attendues.

D'autre part, la construction de la variable d'exposition, et donc du comptage des individus, peut être responsable d'un effet taille. Nous avons en effet comptabilisé plusieurs contrats fictifs découlant de l'évolution au cours du temps d'un seul et même contrat. Ainsi, nous avons obtenu un grand nombre d'informations pour les durées de contrat les plus basses, et peu pour les durées élevées. Le fait de créer autant d'individus que de durées a pu effacer d'autres effets rendus alors plus mineurs. Une amélioration possible de l'étude serait de considérer les coûts moyens par police, tout en incluant l'information sur la durée dans les variables explicatives.

Le Modèle Linéaire Généralisé reste en grande majorité la méthode de tarification privilégiée en assurance non-vie. L'algorithme CART, et plus généralement la théorie des arbres de décision, est plus développé et mieux adapté dans d'autres domaines tels que la médecine, pour l'établissement de diagnostics par exemple.

Bibliographie

[1] Duncan Anderson, Sholom Feldblum, Claudine Modlin, Doris Schirmacher, Ernesto Schirmacher, and Neeza Thandi. *A practitioner's Guide to Generalized Linear Models*. February 2007.

[2] Paul Beinat. Machine learning vs. traditional methods. *EagleEye Analytics*, March 2009.

[3] L. Breiman, J. Friedman, R. Olshen, and C. Stone. Cart : Classification and regression trees. *Wadsworth International*, 1984.

[4] Pierre Connault. *Présentation du domaine de recherche ; Algorithme CART*. February 2008.

[5] Esterina Masiello. Cours isfa troisième année : Modèles linéaires généralisés.

[6] Stephen Milborrow. Plotting rpart trees with prp. `http://www.milbo.org/rpart-plot/prp.pdf`, October 2012.

[7] John Nelder and Robert Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society*, 1972.

[8] Antoine Paglia. *Tarification des risques en assurance non-vie, une approche par modèle d'apprentissage statistique*. 2010.

[9] Ricco Rakotomalala. Comparer les implémentations de tanagra et r (package rpart) de la méthode cart. `http://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/fr_Tanagra_R_CART_algorithm.pdf`.

[10] Ricco Rakotomalala. Utiliser la validation croisée pour l'évaluation des arbres de décision avec r, knime et rapidminer. `http://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/fr_Tanagra_Validation_Croisee_Suite.pdf`.

[11] Terry Therneau and Elizabeth Atkinson. *An Introduction to Recursive Partitioning Using the RPART Routines*. September 1997.

[12] Terry Therneau, Elizabeth Atkinson, and Brian Ripley. *Package rpart*. March 2013.

[13] Roman Timofeev. *Classification and Regression Trees (CART) Theory and Applications*. 2004.

Table des figures

Liste des tableaux

Annexes : reprise des principaux résultats