Institut des ACTUAIRES

ENSAE IP PARIS

Mémoire présenté devant l'ENSAE Paris
pour l'obtention du diplôme de la filière Actuariat
et l'admission à l'Institut des Actuaires
le 17/03/2021

Par : **Florian MONTMEAT**

Titre : **Fairness, interprétabilité et maîtrise des risques en en tarification non-vie**

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

Les signataires s'engagent à respecter la confidentialité indiquée ci-dessus

Membres présents du jury de la filière

Nom :

Membres présents du jury de l'Institut des Actuaires

Entreprise : Optimind optimind.

Signature :

Directeur du mémoire en entreprise :

Nom : André Grondin
Signature :

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

Signature du responsable entreprise

Secrétariat :

Signature du candidat

Bibliothèque :

Ecole Nationale de la Statistique et de l'Administration Economique (ENSAE)
5, avenue Henry Le Chatelier - 91120 PALAISEAU, FRANCE

Résumé

L'émergence de nouvelles méthodes d'apprentissage statistique a remis en cause la tarification classique en assurance. Dans un monde toujours plus concurrentiel, proposer le tarif le plus adapté à chaque assuré est devenu primordial. Aujourd'hui, les GLM ne répondent donc plus forcément à ces contraintes.

Ce mémoire s'intéressera à la tarification de la garantie dégât des eaux d'un contrat MRH par un modèle fréquence/coûts. La première partie de ce mémoire consistera à choisir le meilleur modèle possible pour la prédiction de la fréquence et des coûts. Les GLM seront conservés en point d'appui pour les comparer aux modèles de machine learning.

Dans un second temps, les modèles de machine learning étant jugés ininterprétables et "boites noires", il s'agira de les défendre en répondant aux critères d'explicabilité, de transparence, d'interprétabilité et de loyauté souhaités par les décideurs, les régulateurs et les assurés. Pour cela, des méthodes de transparence telles que les valeurs shapley et les ALE-plot seront utilisées. L'idée de ce mémoire est donc de rendre transparent les modèles « boites noires » par l'utilisation de sur-modèles aussi bien au niveau local que global. L'objectif est de mesurer les avantages et inconvénients de passer de l'ancien modèle au nouveau modèle.

Mots-clés : Assurance multirisque habitation (MRH), Tarification, Gradient boosting, Forêts aléatoires, Transparence, Interprétabilité, Explicabilité, Valeurs Shapley, ALE, GLM

Abstract

The emergence of new machine learning methods has challenged traditional insurance pricing. In an increasingly competitive world, it has become essential to offer the most appropriate premium for each policyholder. Today's GLM models no longer necessarily meet these constraints.

This paper will focus on the pricing of water damage coverage in an home insurance contract using a frequency/cost model. The first part of this paper will be to select the best possible model for frequency and cost prediction. The GLM will be kept as a basis for comparison with the machine learning models.

Secondly, since machine learning models are considered to be uninterpretable and "black boxes", it will be a matter of defending them by meeting the criteria of explicability, transparency, interpretability and loyalty desired by decision-makers, regulators and policyholders. For this, transparency methods such as shapley values and ALE-plot will be used. The idea of this paper is therefore to make "black box" models transparent through the use of over-models at both local and global levels. The objective is to measure the advantages and disadvantages of switching from the old model to the new one.

Keywords : Home insurance, Pricing, Gradient boosting, Random forest, Transparency, Interpretability, Fairness, Shap values, ALE, GLM

Remerciements

Je tiens à remercier en premier lieu mon tuteur au sein d'Optimind, André Grondin, qui a su m'accompagner tout au long de ce travail par son expertise éclairée et qui a veillé au bon déroulement de ce mémoire.

De plus, je tiens à remercier Optimind de m'avoir offert l'opportunité de faire mon stage de fin d'étude au sein de leur entreprise durant cette période spéciale.

Je tiens à remercier également l'ENSAE et l'ensemble des enseignants de l'ENSAE pour la qualité de l'enseignement qu'ils ont su m'apporter et particulièrement mon tuteur pédagogique Christian-Yann Robert. Une pensée particulière à Christophe Dutang qui a été mon professeur d'assurance Non-Vie à l'ENSAE et que j'ai pu côtoyer aussi lors de mes années à Dauphine.

Merci également à l'ensemble des membres du practice Actuarial Financial Services d'Optimind et notamment aux stagiaires qui ont su apporter leur bonne humeur au quotidien.

Enfin, j'aimerais remercier l'ensemble des personnes qui ont pu m'apporter leur soutien tout au long de mes études, mes parents ainsi qu'Estelle et mes amis rencontrés à Dauphine et à l'ENSAE.

Synthèse

Ce mémoire s'intéresse à l'apport des méthodes de machine learning pour la tarification de la garantie dégât des eaux de l'assurance habitation. L'objectif est de montrer qu'à partir de variables explicatives telles que la nature d'habitation, le nombre de pièces, le type d'occupant et montant de valeur assurée, une prime pure peut être fixée de façon plus performante qu'avec les GLM, tout en s'assurant de la transparence du modèle de tarification. Pour ce faire, un portefeuille d'un peu plus d'un million d'assurés répartis sur huit années a été utilisé.

Mise en place d'un modèle de tarification performant :
Une approche classique a été mise en œuvre pour la modélisation de la prime pure en utilisant une approche un modèle de fréquence (nombre de sinistres par an) et de coûts (coûts moyens d'un sinistre) calibré indépendamment. Un pré-traitement des données consistant à enlever les sinistres forfaitaires et à écrêter les sinistres grâce à des méthodes de théorie des valeurs extrêmes a été mise en œuvre avant de réaliser la modélisation. Enfin, les coûts ont été ajustés pour pallier la décroissance de ceux-ci dans le temps et une méthode d'oversampling a été appliquée aux fortes fréquences pour mieux les capter. Les hyperparamètres des modèles de machine learning ont ensuite pu être optimisés par un processus de validation par bloc définis de manière temporelle. Les scores obtenus sur les sorties issues des blocs de la validation sont mis en avant ci-dessous. On observe notamment que pour les modèles de fréquence, les forêts aléatoires et les gradient boostings montrant en gain en termes de gini de 1 point environ. Et que pour les modèles de coûts, l'apport des méthodes de machine learning est plus remarquable, avec un bénéfice de 6 points (soit un score deux fois meilleur) pour les méthodes de machine learning par rapport au GLM classique.

	Modèles de fréquence			Modèles de coûts		
Métrique	RF	GB	GLM	RF	GB	GLM
Gini	0.3545	0.3572	0.3435	0.1297	0.1267	0.0633
True/Pred	1.0013	1.0003	0.9978	0.9850	0.9862	1.0083
RMSE	0.1205	0.1203	0.1210	1540	1544	1549
MAE	0.0556	0.0550	0.0562	826	829	819

TABLE 1 – Modèle de fréquence et coûts : comparaison des modèles sur les sorties de la validation par bloc

L'étude de la comparaison des modèles a été poussée jusqu'à s'attarder sur l'étude des scores sur des segments plus précis. Pour cela, les modalités des trois variables les plus importantes pour les modèles de coûts et de fréquence ont été fixées ce qui a permis d'observer les métriques sur ces segments. Grâce à ce processus, il est possible de se rendre compte si un modèle prédit réellement bien sur un segment ou au contraire ne capte pas du tout les effets d'un segment. Dans le tableau de sortie ci-dessous, seront affichés la métrique de gini pour le modèle de fréquence et le ratio True/Pred pour le modèle de coûts.

NATURE HABITATION	Hist_Dom	TYPE OCCUPANT	% expo	% sinistre	Gini RF	Gini GB	Gini GLM
Appartement	0	Locataire	0.282	0.103	0.289	0.292	0.286
Appartement	0	Proprietaire	0.161	0.160	0.268	0.270	0.259
Appartement	1	Locataire	0.032	0.082	0.237	0.231	0.228
Appartement	1	Proprietaire	0.043	0.181	0.218	0.224	0.212
Maison	0	Locataire	0.080	0.055	0.335	0.332	0.311
Maison	0	Proprietaire	0.309	0.201	0.257	0.262	0.231
Maison	1	Locataire	0.007	0.023	0.239	0.240	0.236
Maison	1	Proprietaire	0.087	0.195	0.181	0.187	0.175

TABLE 2 – Modèle de fréquence : comparaison par couple de modalités

Pour la fréquence le modèle de gradient boosting capte mieux que le GLM sur l'ensemble des segments et est légèrement meilleur que le modèle de forêts aléatoires.

Nature habitation	Type d'occupant	Valeur assurée	% nb	% charge	T/P RF	T/P GB	T/P GLM
Maison	Proprietaire	]13700 ;20500]	0.061	0.079	0.949	0.967	1.169
Maison	Proprietaire	]27300 ;41000]	0.058	0.076	0.920	0.935	1.048
Maison	Proprietaire	>54600	0.049	0.074	0.977	0.971	1.066
Appartement	Proprietaire	]6800 ;13700]	0.084	0.074	1.046	1.037	1.001
Maison	Proprietaire	]6800 ;13700]	0.053	0.072	1.018	1.019	1.303
Maison	Proprietaire	]20500 ;27300]	0.052	0.071	0.977	0.976	1.152
Maison	Proprietaire	]41000 ;54600]	0.047	0.068	0.986	0.989	1.088
Appartement	Locataire	]3500 ;6800]	0.090	0.056	1.000	1.021	0.843
Appartement	Locataire	]6800 ;13700]	0.080	0.056	1.029	1.035	0.877
Appartement	Proprietaire	]13700 ;20500]	0.060	0.052	0.987	0.954	0.919

TABLE 3 – Modèle de coûts : comparaison par couple de modalités

Pour le modèle de coûts, les deux modèles de machine learning s'en sortent nettement mieux que le GLM sur l'ensemble des segments.

Enfin, pour finir l'analyse des modèles, les scores sur les primes pures ont été étudiés. Pour cela, plusieurs combinaisons du produit d'un modèle de fréquence et d'un modèle de coût ont été analysées. Le ratio qui nous a particulièrement intéressés est le S/P (Charge globale de sinistralité sur la somme des primes prédites). De la même manière que pour les modèles de fréquence et de coûts, une analyse par segment a été effectuée. Une pour les variables les plus importantes du modèle de fréquence et une pour celles les plus importantes pour le modèle de coûts.

Nature habitation	Hist_Dom	Type d'occupant	% exposition	S/P RF	S/P GB	S/P GLM
Appartement	0	Locataire	0.282	0.979	1.044	0.832
Appartement	0	Proprietaire	0.161	1.044	0.976	0.952
Appartement	1	Locataire	0.032	1.003	1.004	0.954
Appartement	1	Proprietaire	0.043	1.073	0.988	0.838
Maison	0	Locataire	0.080	0.869	0.966	0.969
Maison	0	Proprietaire	0.309	0.931	0.962	1.143
Maison	1	Locataire	0.007	0.905	0.964	1.095
Maison	1	Proprietaire	0.087	0.979	0.957	1.029

TABLE 4 – Modèle de coûts × fréquence : comparaison des S/P par couple de modalités

Nature habitation	Type d'occupant	Valeur assurée	% exposition	S/P RF	S/P GB	S/P GLM
Appartement	Locataire	]3500 ;6800]	0.114	0.971	1.043	0.803
Appartement	Locataire	]6800 ;13700]	0.076	1.018	1.056	0.932
Maison	Proprietaire	]6800 ;13700]	0.073	0.961	1.008	1.276
Maison	Proprietaire	]13700 ;20500]	0.068	0.952	0.966	1.208
Appartement	Locataire	]0 ;3500]	0.062	0.925	1.064	0.715
Maison	Proprietaire	]20500 ;27300]	0.058	0.937	0.940	1.095
Maison	Proprietaire	]27300 ;41000]	0.056	0.920	0.915	1.008
Appartement	Proprietaire	]6800 ;13700]	0.053	1.084	1.031	1.026
Appartement	Proprietaire	]3500 ;6800]	0.041	1.026	1.003	1.019
Maison	Proprietaire	>54600	0.041	0.949	0.920	0.953

TABLE 5 – Modèle de coûts × fréquence : comparaison des S/P par couple de modalités

Les modèles de machine learning donnent de meilleurs résultats que le GLM sur les segments propres à la fréquence et aux coûts. Par ailleurs, pour le modèle final, le choix s'est porté sur la combinaison des modèles de gradient boosting (fréquence x coûts), étant donné qu'il donne de meilleurs résultats sur les segments propres à la fréquence tout en donnant des résultats similaires sur les segments propres aux coûts.
Pour conforter notre analyse et notre choix de modèle, nous avons fait une analyse de prime pure par zonier. Encore une fois le modèle de gradient boosting capte bien mieux les effets des classes dans la majorité des cas que le GLM. Sur le graphique ci-dessous, on observe que la prime prédite par le modèle de gradient boosting est bien plus proche de la prime réelle que de celle prédite par le GLM sur la majorité des classes.

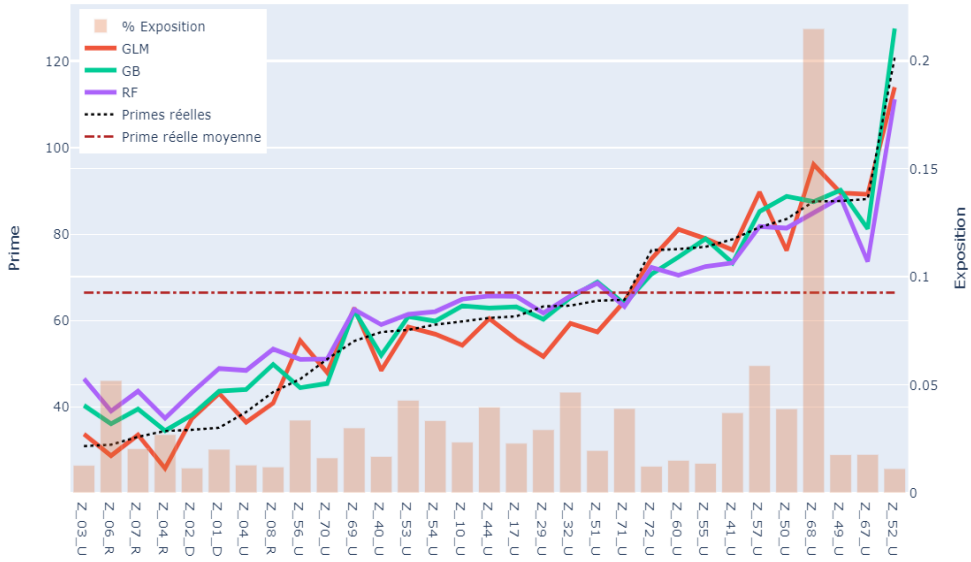
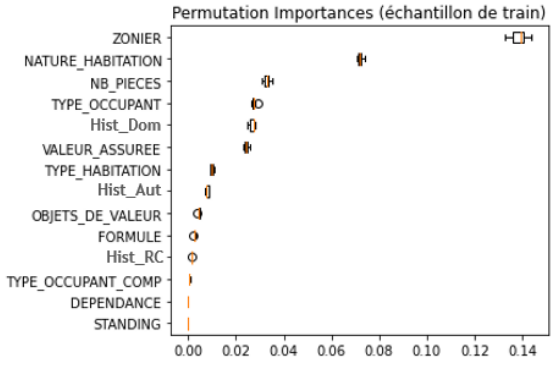


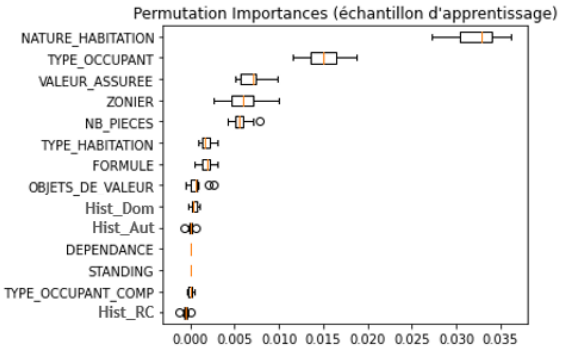
FIGURE 1 – Variation de la prime pure en fonction de la zone géographique

Défense du modèle choisi grâce à la transparence :
La deuxième partie de ce mémoire consistait à défendre le choix d'un modèle plus performant bien que moins transparent sur la façon dont les fréquences et les coûts sont prédits. Avec le modèle de gradient boosting, il est impossible dans un premier temps de connaître l'impact de chaque variable dans la prédiction. Grâce aux importances de variables (PFI), aux accumulated local effect (ALE) et aux valeurs shapley, l'impact des variables va désormais être éclairé et

permettre au modèle d'être explicable, intelligible et transparent. Dans cette synthèse, seule la transparence pour le modèle coûts sera mise en avant à l'exception de l'importance des variables.

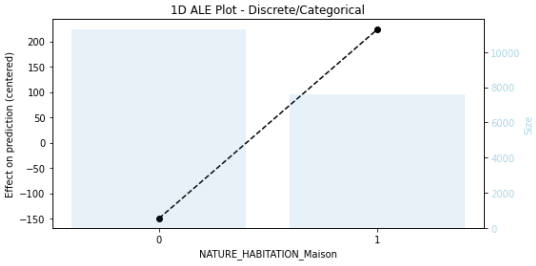


(a) Modèle de fréquence : Importance des variables sur les données d'apprentissage

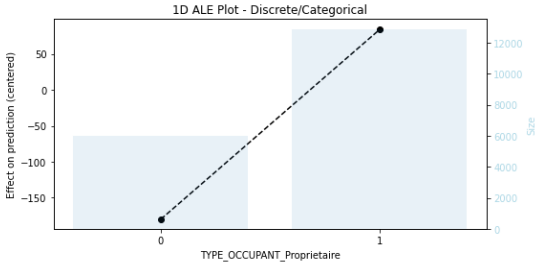


(b) Modèle de coûts : Importance des variables sur les données d'apprentissage

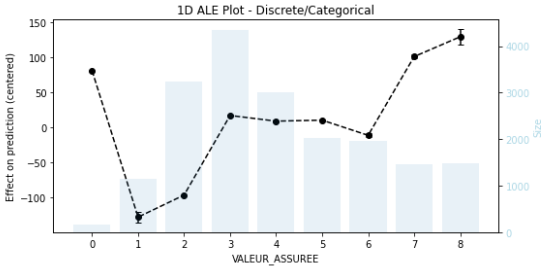
L'idée derrière la méthode d'importance des variables et de considérer qu'une variable est importante si toute chose égale par ailleurs, lorsque sa valeur est modifiée alors l'erreur de prédiction au globale sur toute la base en apprentissage augmente de façon importante.
Ensuite, pour savoir l'effet moyen d'une variable sur la prédiction, les ALE sont le meilleur choix, car c'est la méthode qui gère le mieux les variables corrélées par rapport aux PDP et ICE. Par exemple, grâce aux graphiques ALE, il est possible de savoir comment une modalité impactera la moyenne des prédictions des coûts par exemple ; concernant la variable "nature d'habitation", être en maison va avoir un impact de +200 sur la prédiction des coûts alors qu'être en appartement aura un impact de -150.



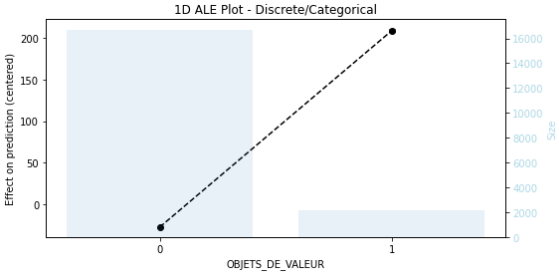
(a) Nature d'habitation



(b) Type d'occupant



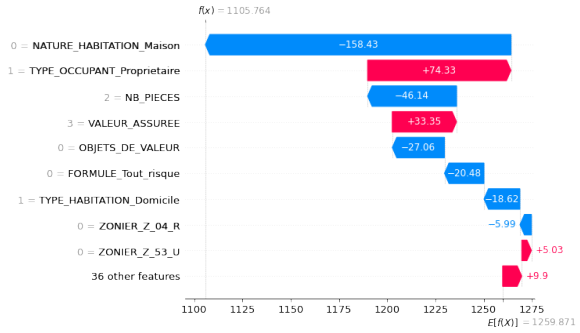
(c) Valeur assurée



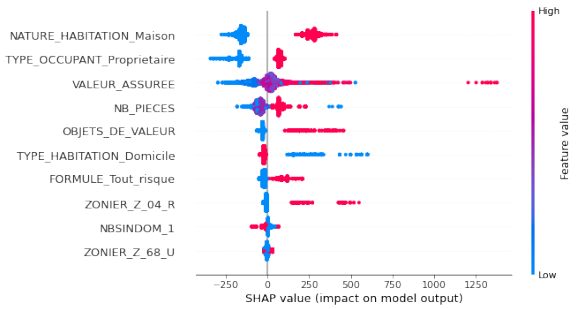
(d) Montant en objets de valeur

FIGURE 3 – ALE plot : modèle de coûts

Enfin les valeurs shapley permettent de donner vision à la fois globale et locale. La vision locale est donnée par l'impact de chaque variable dans la prédiction pour chaque individu (grâce à la décomposition additive des valeurs shapley). La vision globale est observée en faisant la moyenne des valeurs shapley pour chaque variable et ainsi, de la même manière que les PFI, elles permettent d'indiquer l'importance des variables. Cette méthode a été préférée à LIME, car elle est fondée sur des bases mathématiques solides issues de la théorie des jeux, ainsi du point de vue du décideur ou du régulateur cette méthode peut être utilisée.

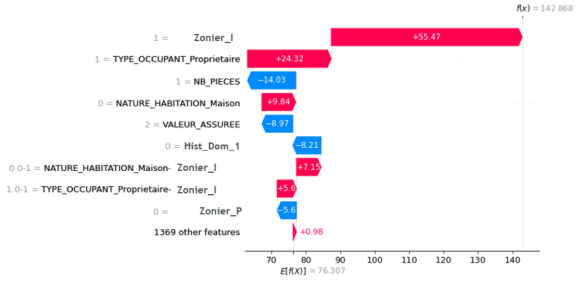


(a) Méthode locale : décomposition additive de la prédiction des coûts pour une exposition au risque

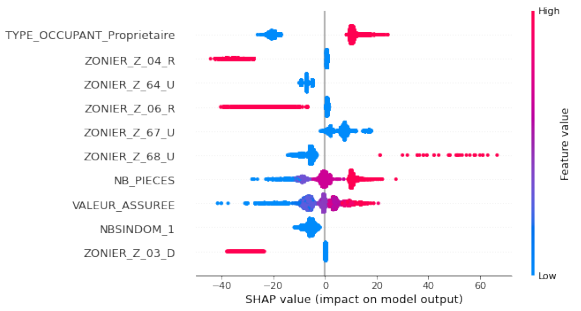


(b) Méthode globale : effet moyen des variables

Enfin étant donné que les valeurs shapley étaient à disposition pour le modèle de fréquence et de coûts, celles-ci ont été couplées pour obtenir des valeurs shapley pour la prime pure, ceci est rendu possible de par la structure additive de la méthode. Cette méthode est détaillée dans le mémoire en paragraphe § (4.5.5.1)



(a) Méthode locale : décomposition additive de la prédiction de la prime pure pour une exposition au risque



(b) Méthode globale : effet moyen des variables

Grâce aux valeurs shapley obtenues pour la prime pure, celle-ci devient transparente de la même manière qu'avec les modèles de coûts et de fréquence. On peut alors observer l'effet des variables les plus importantes au niveau global, et pour chaque individu la décomposition additive de la prime pure. Par exemple, on remarque ainsi que la nature d'habitation n'est pas importante pour la prime pure, ce qui s'explique par le fait que les effets de cette variable sont opposés pour le modèle de fréquence et de coûts.

L'ensemble des études réalisées a permis de trouver un meilleur modèle en termes de prédiction que le GLM classique, notamment celui-ci a montré qu'il arrivait à capter des effets à une maille beaucoup plus fine que le GLM. Grâce aux algorithmes de transparences mises en œuvre, les prédictions du modèle de gradient boosting sont explicables. Cet algorithme vérifie donc le côté transparent, explicable et intelligible demandé par le décideur, le régulateur et l'assuré.

Comme étude complémentaire à ce mémoire, traiter plus précisément la gestion des risques est primordiale. Pour cela, une check-list peut être créée, celle-ci consistant à stresser les données pour permettre la construction d'un intervalle de confiance à la prime potentielle future et au ratio S/P. Une autre étude potentielle consiste à inclure le traitement des sinistres forfaitaires et graves dans les algorithmes de machine learning.

Executive summary

This paper focuses on machine learning methods for pricing water damage coverage in home insurance. The objective is that, based on explanatory variables such as the type of dwelling, the number of rooms, the type of occupant and the amount of insured value, a pure premium is set more efficiently than GLM, while ensuring the transparency of the pricing model. To achieve this, a portfolio of just over one million insureds over eight years was used.

Implementation of an efficient pricing model :
A classic approach was implemented, with pure premium modeling using an independently calibrated frequency (number of claims per year) and cost (average cost of a claim) model. A data processing was implemented before performing the modeling, which consisted of removing lump-sum claims and capping the claims using extreme value theory methods. Finally, costs were adjusted to compensate for the decrease in costs over time and an oversampling method was applied to high frequencies to better capture them. The hyperparameters of the machine learning models were then optimized by a time block validation process. The scores obtained on the outputs from the block validation process are highlighted below. In particular, it can be observed that for the frequency models, random forests and gradient boosting show a gain in terms of gini of about 1 point. And that for the cost models, the contribution of machine learning methods is more remarkable, with a benefit of 6 points (i.e. a score twice as high) for machine learning methods compared to conventional GLM.

	Frequency models			Costs models		
Métrique	RF	GB	GLM	RF	GB	GLM
Gini	0.3545	0.3572	0.3435	0.1297	0.1267	0.0633
True/Pred	1.0013	1.0003	0.9978	0.9850	0.9862	1.0083
RMSE	0.1205	0.1203	0.1210	1540	1544	1549
MAE	0.0556	0.0550	0.0562	826	829	819

TABLE 6 – Frequency and cost model : comparison of models on block validation outputs

The study of model comparison went further, focusing on the study of scores on more specific segments. This involves setting the terms of the three most important variables for the cost and frequency models and observing the metrics on these segments. Through this process, it is possible to see whether a model predicts a segment really well or whether it does not capture the effects of a segment at all. For the frequency model will be displayed below the output table for the gini metric and for the cost model for the True/Pred metric.

NATURE HABITATION	Hist_Dom	TYPE OCCUPANT	% expo	% sinistre	Gini RF	Gini GB	Gini GLM
Appartement	0	Locataire	0.282	0.103	0.289	0.292	0.286
Appartement	0	Proprietaire	0.161	0.160	0.268	0.270	0.259
Appartement	1	Locataire	0.032	0.082	0.237	0.231	0.228
Appartement	1	Proprietaire	0.043	0.181	0.218	0.224	0.212
Maison	0	Locataire	0.080	0.055	0.335	0.332	0.311
Maison	0	Proprietaire	0.309	0.201	0.257	0.262	0.231
Maison	1	Locataire	0.007	0.023	0.239	0.240	0.236
Maison	1	Proprietaire	0.087	0.195	0.181	0.187	0.175

TABLE 7 – Frequency model : comparison by pair of modalities

For frequency, the gradient boosting model captures better than the GLM model on all segments and is slightly better than the random forest model.

Nature habitation	Type d'occupant	Valeur assurée	% nb	% charge	T/P RF	T/P GB	T/P GLM
Maison	Proprietaire	]13700 ;20500]	0.061	0.079	0.949	0.967	1.169
Maison	Proprietaire	]27300 ;41000]	0.058	0.076	0.920	0.935	1.048
Maison	Proprietaire	>54600	0.049	0.074	0.977	0.971	1.066
Appartement	Proprietaire	]6800 ;13700]	0.084	0.074	1.046	1.037	1.001
Maison	Proprietaire	]6800 ;13700]	0.053	0.072	1.018	1.019	1.303
Maison	Proprietaire	]20500 ;27300]	0.052	0.071	0.977	0.976	1.152
Maison	Proprietaire	]41000 ;54600]	0.047	0.068	0.986	0.989	1.088
Appartement	Locataire	]3500 ;6800]	0.090	0.056	1.000	1.021	0.843
Appartement	Locataire	]6800 ;13700]	0.080	0.056	1.029	1.035	0.877
Appartement	Proprietaire	]13700 ;20500]	0.060	0.052	0.987	0.954	0.919

TABLE 8 – Cost model : comparison by pair of modalities

For the cost model, the two machine learning models performed significantly better than the GLM model across all segments.

Finally, to finish the analysis of the models, the scores on pure premiums were studied. For this, several combinations of the product of a frequency model and a cost model were analysed . The ratio that was of particular interest to us was the S/P (Total claims burden on the sum of predicted premiums). In the same way as for the frequency and cost models, a segment analysis was carried out. One for the most important features of the frequency model and one for the most important features of the cost model.

Nature habitation	Hist_Dom	Type d'occupant	% exposition	S/P RF	S/P GB	S/P GLM
Appartement	0	Locataire	0.282	0.979	1.044	0.832
Appartement	0	Proprietaire	0.161	1.044	0.976	0.952
Appartement	1	Locataire	0.032	1.003	1.004	0.954
Appartement	1	Proprietaire	0.043	1.073	0.988	0.838
Maison	0	Locataire	0.080	0.869	0.966	0.969
Maison	0	Proprietaire	0.309	0.931	0.962	1.143
Maison	1	Locataire	0.007	0.905	0.964	1.095
Maison	1	Proprietaire	0.087	0.979	0.957	1.029

TABLE 9 – Cost model × frequency : comparison of S/P by pair of modalities

Nature habitation	Type d'occupant	Valeur assurée	% exposition	S/P RF	S/P GB	S/P GLM
Appartement	Locataire	]3500 ;6800]	0.114	0.971	1.043	0.803
Appartement	Locataire	]6800 ;13700]	0.076	1.018	1.056	0.932
Maison	Proprietaire	]6800 ;13700]	0.073	0.961	1.008	1.276
Maison	Proprietaire	]13700 ;20500]	0.068	0.952	0.966	1.208
Appartement	Locataire	]0 ;3500]	0.062	0.925	1.064	0.715
Maison	Proprietaire	]20500 ;27300]	0.058	0.937	0.940	1.095
Maison	Proprietaire	]27300 ;41000]	0.056	0.920	0.915	1.008
Appartement	Proprietaire	]6800 ;13700]	0.053	1.084	1.031	1.026
Appartement	Proprietaire	]3500 ;6800]	0.041	1.026	1.003	1.019
Maison	Proprietaire	>54600	0.041	0.949	0.920	0.953

TABLE 10 – Cost model × frequency : comparison of S/P by pair of modalities

Machine learning models give better results than GLM on frequency and cost-specific segments. In addition, for the final model, the choice was made to combine the gradient boosting (frequency x cost) models, since although giving similar results on the cost-specific segments, gradient boosting gives better results on the frequency-specific segments.
We ended this model study with a pure premium analysis by zoner, where we can observe once again that the gradient boosting model captures class effects much better in the majority of cases than GLM. In the graph below, we can see that the predicted premium of the gradient boosting model is much closer to the actual premium than to the predicted premium of the GLM.

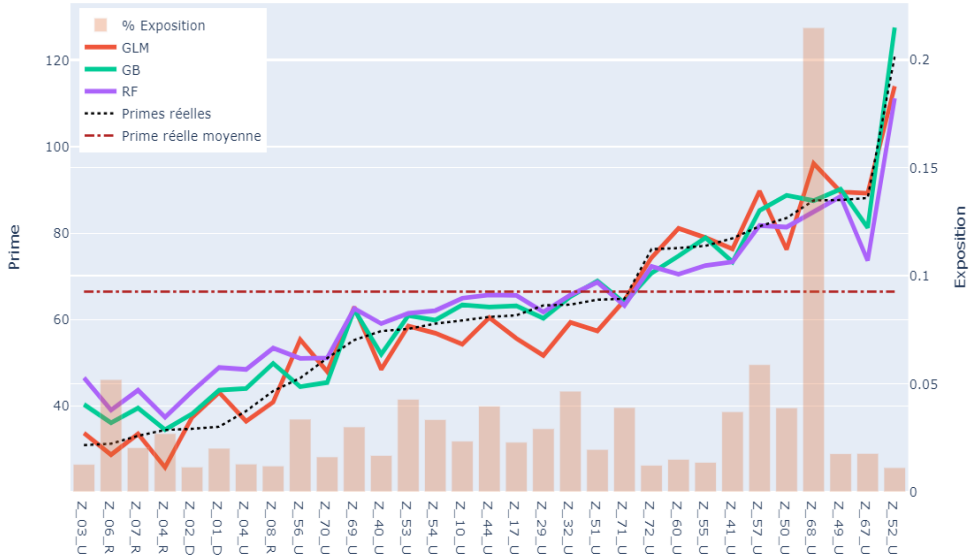
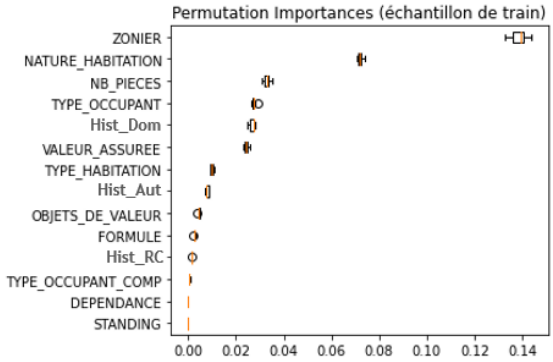
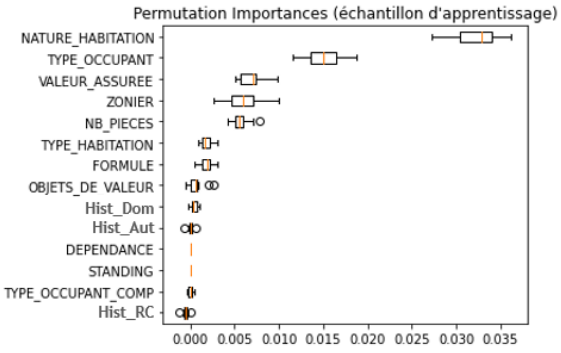


FIGURE 6 – Variation of the pure premium according to the geographical area

Defending the chosen model through transparency :
The second part of this brief was to defend the choice of a more efficient model although less transparent on how frequency and costs are predicted. With the gradient boosting model, it is initially impossible to know the impact of each variable in the prediction. Thanks to features importance (PFI), accumulated local effect (ALE) and shapley values, the impact of the variables will now be illuminated and allow the model to be explainable, intelligible and transparent. In this synthesis, only the transparency of the cost model will be highlighted, with the exception of the importance of the variables.

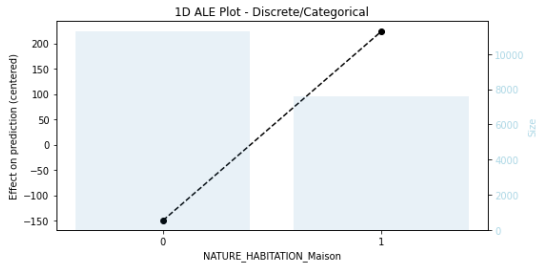


(a) Frequency model : Importance of features on learning data

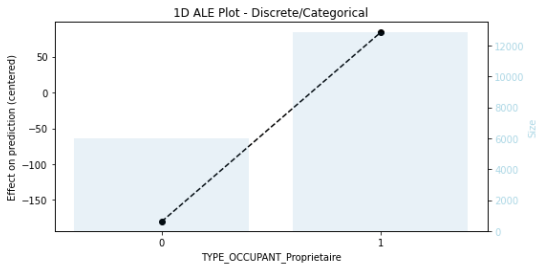


(b) Cost Model : Importance of features on learning data

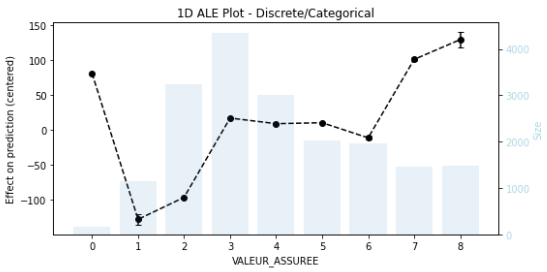
The idea behind the method of features importance and to consider that a variable is important if everything else is equal, when its value is changed then the overall prediction error over the entire learning base increases significantly.
Then to find out the average effect of a feature on the prediction, it turned out that ALE were the best choice. This is the method that best manages correlated variables with respect to PDP and ICE. Using ALE plots, it is possible to know how a modality will impact the average cost prediction for example ; being in a house will have an impact of +200 on the cost prediction while being in a flat will have an impact of -150.



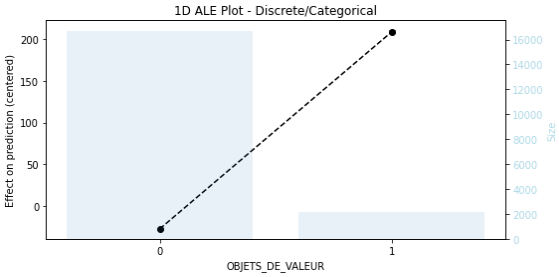
(a) Type of residence



(b) Type of occupant



(c) Insured value

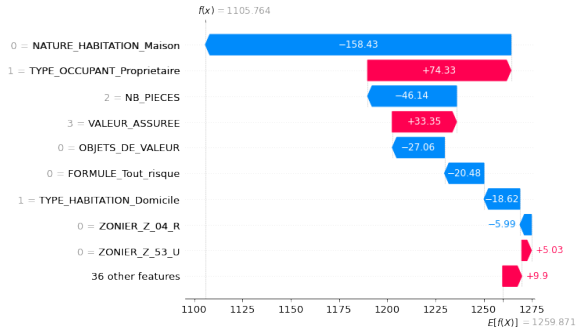


(d) Amount in valuable items

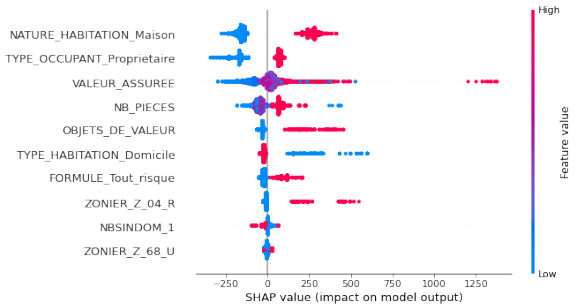
FIGURE 8 – ALE plot : costs model

Finally, the last transparency method made it possible to give both a global and a local vision. The shapley values give an additive decomposition according to each feature for the prediction of an individual. The local vision is given by the impact of each feature in the prediction for each individual. The global view is observed by averaging the shapley values for each feature

and thus in the same way that PFI indicates the importance of the features. This method has been preferred to LIME because it is based on solid mathematical foundations from game theory, so from the point of view of the decision-maker or regulator this method can be used.

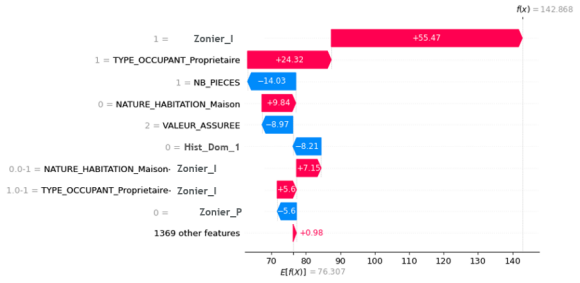


(a) Local method : additive decomposition of individual cost prediction

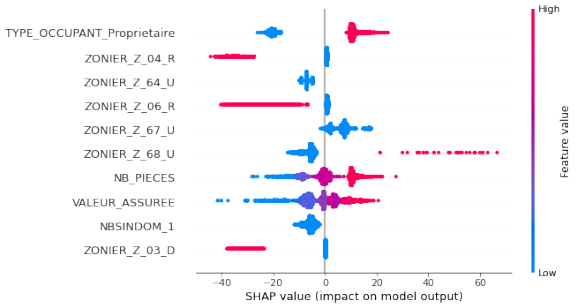


(b) Global method : features mean effect

Finally, since shapley values were available for the frequency and cost model and by the additive structure of the method, these were coupled to obtain shapley values for the pure premium. This method is described in detail in the § (4.5.5.1)



(a) Local method : additive decomposition of the pure premium prediction for an individual



(b) Global method : average effect of variables

Thanks to this work on the shapley values for the pure premium, the premium becomes transparent in the same way as the cost and frequency models. We can then observe the effect of the most important variables at the global level, and for each individual the additive decomposition of the pure premium can be observed. It can thus be seen that the type of residence is not important for the pure premium, which explains why the effects of this feature are opposite for the frequency and cost model.

All the studies carried out have led to a better model in terms of prediction than the classic GLM, in particular this one has shown that it was able to capture effects at a much finer mesh than the GLM. Thanks to the transparency algorithms implemented, the predictions of the gradient boosting model can be explained. This algorithm therefore verifies the transparent, explainable and intelligible side requested by the decision-maker, the regulator and the insured.

As a complementary study to this paper, addressing risk management more precisely is of paramount importance. For this, a checklist can be created, which consists in stressing the data

to allow the construction of a confidence interval for the future potential premium and the S/P ratio. Another potential study consists of including the processing of lump-sum and serious claims in machine learning algorithms.

TABLE DES MATIÈRES

1 Généralité sur l'assurance multirisques habitation (MRH) et étude du portefeuille **21**
1.1 L'assurance MRH 22
1.1.1 Les principales garanties 22
1.1.2 Critères tarifaires utilisées 23
1.1.3 Quelques chiffres sur l'assurance MRH 24
1.2 Données MRH 24
1.3 Statistiques descriptives 25
1.3.1 Corrélation entre les variables 27
1.3.2 Statistiques sur les montants et le nombre de sinistres 29
1.3.3 Observation de la fréquence en fonction des modalités des variables . . . 31
1.3.4 Observation de la sévérité en fonction des modalités des variables 35

2 Traitement des données et modélisation classique des sinistres **40**
2.1 Traitement des données 40
2.1.1 Traitement des sinistres graves 40
2.1.2 Traitement des sinistres forfaitaires 44
2.1.3 Visualisation des données finales 46
2.2 Modélisation classique des sinistres 47
2.2.1 Modèle Coûts x Fréquence 47
2.2.2 Modèle linéaire généralisé 49

3 Apport du machine learning dans la modélisation des sinistres **52**
3.1 Méthodes de machine learning : Théorie 52
3.1.1 Généralité sur l'apprentissage supervisé 52
3.1.2 Notion de sur-apprentissage et de sous apprentissage 55

3.1.3 Notion de validation par bloc (k-folds) . . . 56
3.1.4 Random forest . . . 57
3.1.5 Gradient boosting . . . 60
3.1.6 Méthode d'oversampling (sur-échantillonnage) . . . 62
3.1.7 Principe de sélection des hyperparamètres des modèles . . . 63
3.2 Pré-traitement des données avant la modélisation . . . 64
3.2.1 Statistiques sur les données de test et d'apprentissage . . . 64
3.2.2 Préprocessing mis en place . . . 65
3.3 Définition des métriques utilisées . . . 67
3.4 Calibration des modèles . . . 69
3.4.1 Paramètres du gradient boosting . . . 69
3.4.2 Paramètres des forêts aléatoires . . . 71
3.4.3 Hyperparamètres sélectionnés . . . 71
3.4.4 Analyses des résultats des différents modèles . . . 76
3.5 Étude de la performance des modèles . . . 79
3.5.1 Étude des erreurs par classe d'individus . . . 79
3.5.2 Travail sur le couplage des modèles de fréquence et de coûts . . . 85
3.5.3 Choix du modèle final . . . 90

4 Défense des nouveaux modèles grâce à la transparence 91
4.1 Intérêt réglementaire de la transparence . . . 91
4.2 Aspects littéraires de la transparence . . . 92
4.2.1 La loyauté des algorithmes . . . 93
4.2.2 L'équité des algorithmes . . . 93
4.2.3 L'explicabilité des algorithmes . . . 93
4.2.4 L'intelligibilité des algorithmes . . . 94
4.3 Compromis entre interprétabilité et efficacité . . . 94
4.4 Définition mathématiques des méthodes transparences . . . 96
4.4.1 Permutation Feature Importance (PFI) . . . 96
4.4.2 Accumulated Local Effects (ALE) . . . 98
4.4.3 Valeur Shapley . . . 100
4.4.4 SHAP . . . 104
4.5 Vérification de la transparence des modèles calibrés . . . 107
4.5.1 PFI . . . 107
4.5.2 Tracé des prédictions . . . 109
4.5.3 ALE plot . . . 111
4.5.4 SHAP . . . 112
4.5.5 SHAP pour la prime pure . . . 117
4.5.6 Construction d'un tableau de prime . . . 121

5 Proposition de deux ouvertures sur la gestion des risques 123
5.1 Construction d'une check-list . . . 123
5.2 Proposition de modélisation des sinistres graves et forfaitaires via le machine learning . . . 124

A Lexique des abréviations 131

B Annexes descriptives 132
B.1 Analyses descriptives : garantie incendie . . . 132
B.1.1 V Cramer . . . 135

C GLM : résultats 136
C.1 Modélisation de la fréquence . . . 136
C.2 Modélisation des coûts . . . 138
C.3 Prime pure pour GLM . . . 139

D XGBoost 140

E Graphiques de primes pures par variable 144

F Autres méthodes de transparence 146
F.1 Partial Dependant Plot (PDP) . . . 146
F.2 Individual Conditional Expectation (ICE) . . . 148

G Vérification des méthodes de transparence pour le GLM 149
G.1 ALE plot . . . 150
G.2 Valeurs Shapley . . . 150

H Application des méthodes de transparence aux modèles de forêts aléatoires 151
H.1 PFI . . . 151
H.2 Tracé des prédictions . . . 152
H.3 ALE plot . . . 154
H.4 Valeurs Shapley . . . 156

INTRODUCTION

De nos jours, proposer un contrat d'assurance à un tarif concurrentiel est de plus en plus compliqué. C'est pourquoi avoir une compréhension des risques sous-jacents à son portefeuille et une bonne capacité de prédiction est primordiale. Les assureurs ont la chance d'avoir des données en quantités suffisantes pour créer des modèles suffisamment robustes pour permettre une prédiction adéquate du risque.

Par ailleurs, les méthodes classiques telles que les modèles linéaires généralisés ne semblent plus suffisantes pour tarifer un contrat de la meilleure des façons. Grâce à de nouvelles méthodes de prédictions, les risques sous-jacents aux contrats d'assurances peuvent être mieux compris et donc les tarifs proposés peuvent être plus individualisés. En outre ces nouvelles méthodes de prédictions telles que les méthodes de gradient boosting ou de forêts aléatoires sont dites boite noire. Bien qu'elles améliorent la prédiction, il est difficile d'interpréter ce qui se passe à l'intérieur de ces modèles. Il s'agit de faire un compromis entre interprétabilité et performance. C'est pourquoi d'un point de vue de transparence ces méthodes ne sont que peu utilisées pour le moment.

Dans un premier temps, il conviendra donc de modéliser la fréquence et les coûts moyens grâce aux forêts aléatoires et aux gradient boosting ainsi qu'avec des GLM pour pouvoir comparer par la suite ces modèles. Au-préalable, le portefeuille aura été analysé et les coûts auront été écrêtés et les forfaits retirés pour permettre une bonne modélisation de ceux-ci. La comparaison de ces modèles s'effectuera en observant les métriques choisies sur le portefeuille en général, mais aussi en fonction de certaines variables tarifaires fixées. Ceci pour pouvoir observer si les nouveaux modèles captent mieux certains effets que les GLM au niveau global, mais aussi à une maille plus fine. Le choix du modèle final se fera en observant l'efficacité des modèles sur la prédiction de la prime pure.

Une fois les modèles obtenus, il s'agira de défendre les modèles dits *boite noire* grâce à des

méthodes de transparence. Pour cela des algorithmes aussi bien au niveau local que global seront utilisés pour comprendre comment les modèles sont arrivés aux valeurs prédites, mais aussi à détecter de potentiels biais dans les modèles crées. Pour ces nouveaux modèles, il faudra vérifier l'adéquation aux quatre mots d'ordre suivant : loyauté, équité, explicabilité et intelligibilité.

Enfin, un travail sera effectué au niveau de la prime pure. Dans un premier temps, une méthode de transparence sera appliquée à celle-ci pour déterminer l'influence des variables dans la tarification. Dans un second temps, les données du portefeuille mis à disposition seront stressées pour vérifier la stabilité du ratio sinistre sur prime.

CHAPITRE

1

GÉNÉRALITÉ SUR L'ASSURANCE MULTIRISQUES HABITATION (MRH) ET ÉTUDE DU PORTEFEUILLE

L'assurance est une opération par laquelle une personne (l'assureur) s'engage à réaliser une prestation, dans le cadre d'un contrat d'assurance, au profit d'un individu (l'assuré) lors de la survenance d'un risque et moyennant le paiement d'une cotisation ou d'une prime. Le risque constitue l'objet de l'assurance. Un assuré contracte une assurance pour se prémunir contre des événements dommageables (maladie, incendie, vol, décès...) ou contre les risques liés à certains objets qu'il possède (automobile, habitation...).
En cas de survenance d'un risque, l'assureur s'oblige à verser une prestation sous forme d'argent soit à l'assuré, soit à un tiers (celui qui a subi le dommage dont l'assuré est responsable), soit au bénéficiaire dans le cadre d'une assurance-vie. En contrepartie de cette prestation, l'assuré verse une prime ou une cotisation à l'assureur. Cette somme correspond au coût du risque et aux frais de fonctionnement de l'assureur.
L'assurance IARD (incendies, accidents, risques divers) a pour caractéristique de fonctionner en gestion des primes par répartition. Cela signifie que les primes de l'ensemble des assurés servent à payer les sinistres de la communauté des assurés au titre du même exercice. De nombreux produits sont rattachés au secteur de l'assurance non-vie (IARD) mais nous traiterons seulement l'assurance multirisque habitation.

1.1 L'assurance MRH

Lors d'un sinistre dans un logement, l'assurance habitation permet de couvrir les frais liés aux réparations immobilières, mais elle couvre aussi les biens mobiliers endommagés ou perdus. En cas d'événement affectant un logement voisin comme un dégât des eaux, ce contrat couvre également votre responsabilité d'occupant et prend en charge les dommages accidentels corporels et matériels causés aux tiers.
Fortement recommandée pour le propriétaire, l'assurance habitation est obligatoire pour le locataire. L'article 7 de la loi n°89-462 du 6 juillet 1989 lui impose en effet la souscription à une assurance habitation. Il a l'obligation de s'assurer pour les risques locatifs afin de couvrir les dommages causés au logement du propriétaire par :

- un incendie ;
- une explosion ;
- un dégât des eaux.

En revanche, l'assurance des risques locatifs ne couvre pas les dommages causés à des voisins et aux biens du locataire lui-même.
Pour un propriétaire, assurer son bien immobilier et ses biens mobiliers contre les risques accidentels qui peuvent survenir est le moyen de sauvegarder son patrimoine. Les conséquences d'un incendie ou d'une fuite d'eau peuvent être financièrement très lourdes.

1.1.1 Les principales garanties

Afin de couvrir l'ensemble des situations pouvant conduire à la survenue d'un sinistre dans un logement, qu'elles soient du fait du propriétaire, du locataire ou d'un tiers visiteur, les assureurs incluent dans leur contrat des garanties spécifiques prenant en charge des risques divers et variés. ci-dessous sont détaillées les garanties comprises de bases dans un contrat d'assurance MRH :

- **Responsabilité civile** : C'est la garantie qui intervient en cas de dommages causés à des tiers à la fois matériels et corporels, et ce, sans plafond d'indemnisation. À la fois contractuelle et délictuelle, elle couvre l'assuré et ses ayants droit vivant sous le même toit.
- **Incendie** : La garantie incendie est comprise dans tous les contrats d'assurance multirisques habitation. Elle couvre non seulement les dommages liés à un incendie, mais également ceux causés par une explosion. Les dommages causés par la foudre, la fumée, les secours et les premiers sauvetages sont eux aussi couverts au titre de cette garantie.
- **Dégât des eaux** : couverture des dégâts causés par l'eau à vos biens immobiliers et mobiliers suite à des infiltrations, des fuites ou des débordements d'appareils (machine à laver...).
- **Vol** : couverture des effractions et des vols dans votre logement. Les sociétés d'assurances peuvent exiger la mise en place de certains moyens de protection pour accorder la garantie vol. Un renforcement des moyens de protection de l'habitation peut être demandé, en particulier en présence de biens de valeur ou en cas de vols à répétition. En général seuls les vols par effraction, avec menaces ou violence ou bien par le personnel de maison sont considérés par la garantie. L'assurance porte sur les biens mobiliers qui sont dans

le logement et détenu par l'assuré, mais aussi en fonction de la formule les objets de valeurs et bijoux.

- **Évènements climatiques et catastrophes naturelles** : prise en charge des dommages occasionnés par les tempêtes, le poids de la neige, la grêle ou les inondations, généralement suite à la parution d'un arrêté de catastrophes naturelles.
- **Bris de vitres** : prise en charge des dommages sur les parties vitrées (portes et fenêtres) ou le mobilier (four, écran TV, etc.).
- **Attentats et actes de terrorisme :** obligatoire depuis septembre 1986, elle couvre les biens face aux dommages dus à un attentat, un sabotage, une émeute, un acte de terrorisme ou encore un mouvement populaire.

Les garanties complémentaires proposées par l'ensemble des organismes d'assurance sont sensiblement les mêmes. Voici une liste non exhaustive des options les plus fréquemment proposées.

- **Assistance** : suite à un accident au domicile de l'assuré, elle couvre les transports à l'hôpital, la garde des enfants et des animaux, ... suite à un sinistre touchant le domicile, elle peut couvrir le rapatriement d'urgence au domicile si l'assuré est absent au moment du sinistre et que sa présence est nécessaire pour effectuer des démarches administratives et les frais d'hébergement provisoire si le logement de l'assuré est inhabitable...
- **Vol d'objets de valeur/Bijoux :** Indemnisation des objets de valeur suite à un vol par effraction, avec menaces ou agression, une intrusion clandestine ou l'usage de fausses clés. Cette garantie est parfois comprise dans le contrat de base.
- **Défense pénale et recours :** Prise en charge des remboursements de frais juridiques engagés par l'assuré dans le cadre d'une poursuite pénale ou d'un litige lié à l'application d'une des garanties du contrat d'assurance.
- **Protection juridique :** La garantie protection juridique permet que l'assuré soit représenté et défendu par son assureur dans le cadre d'une procédure de justice opposant l'assuré à un tiers.
- **Piscine :** Cette garantie protège les structures et revêtements, la motorisation, les panneaux solaires destinés à chauffer l'eau, le matériel de sécurité, pompage, chauffage et épuration de l'eau et tous les équipements complémentaires. Cette garantie s'étend aux jacuzzis, spas et saunas situés hors du logement de l'assuré.

1.1.2 Critères tarifaires utilisées

A partir des données transmises par l'assuré à l'assureur et des critères qu'il a défini, l'assureur fixe une prime cohérente en fonction du bien assuré et de l'assuré. Les critères faisant varier la prime d'assurance sont les suivants :

- **le bien à assurer** : Type d'habitation, surface du logement, nombre de pièces, dépendances, résidence principale/secondaire, zone géographique
- **le contenu à assurer** : Valeur des biens mobiliers, objet de valeurs
- **le souscripteur** : Locataire/Propriétaire, présence d'un colocataire, historique de sinistralité.
- **le type de produit d'assurance** : Formule, c'est à dire le type de couverture du

contrat (Tous risques/Essentiel/Confort).

La tarification dans ce domaine de l'assurance fonctionne en général garantie par garantie, certaines garanties étant de type faible fréquence/forte sévérité et d'autre étant forte fréquence/faible sévérité. La prime finale est une agrégation des primes des différentes garanties couvertes.

Certaines variables de tarifications vont avoir un rôle plus ou moins importants suivant la garantie étudiée. Par exemple pour la garantie "vol", les variables sur le contenu à assurer vont être primordiales.

Enfin, certaines variables peuvent être utilisées différemment suivant les garanties étudiées. Par exemple, la variable de zone géographique aura un rôle central pour les garanties catastrophes naturelles et catastrophes technologiques, mais ne pourra servir que d'indicateur ou de variable de contrôle pour d'autres garanties.

1.1.3 Quelques chiffres sur l'assurance MRH

En 2018, le nombre de contrats MRH s'élevait à 41.9 millions et connaissait une progression de 2.0 % (+1.6 % pour les contrats occupants et +4.2 % pour les contrats non-occupants). Dans le même temps, le nombre de logements en 2018 est estimé à 36.8 millions, et a connu une hausse de 1.1 %. L'ensemble des cotisations des contrats MRH toutes garanties confondues (y compris Catastrophes naturelles) est estimé à 10,5 milliards d'euros. Le marché des contrats MRH enregistre une hausse des cotisations de 3.5 % et la prime moyenne a augmenté de 1.6 %. En 2018, la fréquence globale des contrats MRH (y compris les catastrophes naturelles) est en hausse de 7.9% par rapport à 2017. Parallèlement, le coût moyen des sinistres est orienté à la hausse (+ 9.2 %) après une baisse sensible en 2017 (- 7.2 %). (Source : FFA - Fédération Française des Assurances)

1.2 Données MRH

Le portefeuille mise à disposition comporte 1 065 000 d'expositions au risque sur les années 2011 à 2018 soit 156 865 assurées en moyenne par an. Aucune valeur manquante a été constatée. L'exposition moyenne est de 0.85 années (soit environ 10 mois), sachant que l'exposition maximale est d'un an. Par ailleurs, certains individus présents dans le portefeuille présentent une exposition anormalement faible (égale à 1 jour), ils seront écartés du portefeuille pour la modélisation de la fréquence de sinistre.

Les critères tarifaires présents dans la base ne sont que des données catégorielles. Une segmentation ayant été faite en amont sur les données numériques.

Un tableau récapitulatif des variables de tarification présentes dans la base se trouvent ci-dessous :

Libellé de la variable	Type de variable	Renseignement	Commentaires
FORMULE	Qualitative		Niveau de garantie du contrat : Tous risque/Essentiel/Confort
TYPE HABITATION	Qualitative	Bien assuré	Domicile principal ou non
NATURE HABITATION	Qualitative	Bien assuré	Appartement ou maison
NOMBRE DE PIECES	Qualitative	Bien assuré	
TYPE OCCUPANT	Qualitative	Assuré	Propriétaire ou locataire
COMPLÉMENTAIRE TYPE OCCUPANT	Qualitative	Assuré	Présence de colocataire
VALEUR ASSURÉE	Qualitative	Contenu à assurer	
OBJET DE VALEUR	Qualitative	Contenu à assurer	
ZONIER Niveau 1	Qualitative	Bien assuré	
ZONIER Niveau 2	Qualitative	Bien assuré	Zonier Simplifié
STANDING	Qualitative	Bien assuré	Présence de piscine
DEPENDANCE	Qualitative	Bien assuré	Présence de dépendance
Hist_Dom	Quantitative	Assuré	Présence ou non d'un sinistre dans un historique de 3 ans
Hist_RC	Quantitative	Assuré	Présence ou non d'un sinistre dans un historique de 3 ans
Hist_Aut	Quantitative	Assuré	Présence ou non d'un sinistre dans un historique de 3 ans

TABLE 1.1 – Variables de tarification - MRH

1.3 Statistiques descriptives

En préalable à la modélisation, il est primordial de faire une étude sur le portefeuille aussi bien graphique que statistique.
Pour l'étude graphique, il s'agit de représenter la fréquence de sinistre pour chaque modalité des variables explicatives catégorielles, ainsi que le pourcentage d'expositions de chaque modalité. Cette étude a pour but de voir l'impact potentiel de certaines variables de tarification sur les variables cibles. Ainsi certaines variables pourront d'ores et déjà être écartées des modèles du fait de leur intérêt limité. Dans un second temps, il s'agira de regarder le coût des sinistres en fonction des modalités des variables catégorielles.

Le tableau ci-dessous présente les variables explicatives ainsi que l'exposition des modalités de ces variables explicatives. Pour plus de lisibilité, la variable ZONIER contenant un trop grand nombre de modalités a été écartée. Pour des besoins d'anonymisation, les sommes d'expositions ont été arrondies.

Variable	Modalité	Exposition	% Exposition
FORMULE	Confort	8.65×10^5	0.80
FORMULE	Essentiel	3.96×10^4	0.04
FORMULE	Tout_risque	1.59×10^5	0.16
TYPE_HABITATION	Autres_que_domicile	1.47×10^5	0.13
TYPE_HABITATION	Domicile	9.17×10^5	0.87
NATURE_HABITATION	Appartement	5.55×10^5	0.53
NATURE_HABITATION	Maison	5.10×10^5	0.47
NB_PIECES	inf 20m2	8.20×10^4	0.07
NB_PIECES	inf 40m2	2.12×10^5	0.20
NB_PIECES	inf 80m2	4.51×10^5	0.43
NB_PIECES	sup 80m2	3.18×10^5	0.30
NATURE_HBT_NB_PIECES	Appartement - inf 20m2	6.97×10^4	0.06
NATURE_HBT_NB_PIECES	Appartement - inf 40m2	1.90×10^5	0.19
NATURE_HBT_NB_PIECES	Appartement - inf 80m2	2.62×10^5	0.25
NATURE_HBT_NB_PIECES	Appartement - sup 80m2	3.4×10^4	0.03
NATURE_HBT_NB_PIECES	Maison - inf 20m2	12.1×10^4	0.01
NATURE_HBT_NB_PIECES	Maison - inf 40m2	2.29×10^4	0.02
NATURE_HBT_NB_PIECES	Maison - inf 80m2	1.90×10^5	0.19
NATURE_HBT_NB_PIECES	Maison - sup 80m2	2.85×10^5	0.25
TYPE_OCCUPANT	Locataire	4.34×10^5	0.42
TYPE_OCCUPANT	Proprietaire	6.31×10^5	0.58
TYPE_OCCUPANT_COMP	Colocataire	1.12×10^4	0.01
TYPE_OCCUPANT_COMP	Sans_infos	1.05×10^5	0.99
VALEUR_ASSUREE	Sans_patmob	4.26×10^4	0.03
VALEUR_ASSUREE	]0 ;3500]	9.59×10^4	0.09
VALEUR_ASSUREE	]3500 ;6800]	2.34×10^5	0.22
VALEUR_ASSUREE	]6800 ;13700]	2.43×10^5	0.24
VALEUR_ASSUREE	]13700 ;20500]	1.46×10^5	0.14
VALEUR_ASSUREE	]20500 ;27300]	1.01×10^5	0.09
VALEUR_ASSUREE	]27300 ;41000]	8.59×10^4	0.08
VALEUR_ASSUREE	]41000 ;54600]	5.92×10^4	0.06
VALEUR_ASSUREE	>54600	5.68×10^4	0.05
OBJETS_DE_VALEUR	Faible_valeur	9.95×10^5	0.94
OBJETS_DE_VALEUR	Forte_valeur	7.03×10^4	0.06
ZONIER Niveau 2	Non_urbain	1.52×10^5	0.14
ZONIER Niveau 2	Urbain	9.13×10^5	0.86
STANDING	Piscine	4.58×10^4	0.04
STANDING	Sans_infos	1.02×10^6	0.96
DEPENDANCE	Absence	1.05×10^6	0.99
DEPENDANCE	Présence	1.01×10^4	0.01
Hist_Dom	0	8.85×10^5	0.83
Hist_Dom	1	1.18×10^5	0.17
Hist_RC	0	1.04×10^6	0.99
Hist_RC	1	2.42×10^4	0.01
Hist_Aut	0	1.01×10^6	0.96
Hist_Aut	1	5.69×10^4	0.04

TABLE 1.2 – Exposition des modalités des variables de tarification

- Les modalités de la variable FORMULE correspondent à des types de couverture souscrits par les assurés. La formule "Essentiel" correspondant à la couverture la plus faible et la formule "Tout risque" à la couverture la plus élevée.
- Pour des besoins d'anonymisation de données, les modalités de la variable nombre de pièces ont été modifiées en faisant une correspondance entre le nombre de pièces d'un logement et sa superficie.

1.3.1 Corrélation entre les variables

Il est utile de regarder la corrélation entre les différentes variables de la base. Étant donné que la base utilisée ne contient que des variables qualitatives, l'observation de la corrélation se fera par le biais du V de Cramer.
L'objectif de cette étape est de veiller à ce que les variables qui seront utilisées dans les modèles par la suite ne sont pas trop corrélées entre elles, car c'est une des hypothèses importantes des GLM (Generalized linear model). Sinon cela pourrait avoir un impact négatif sur la validité du modèle.

Analyse des sorties de V de Cramer :

Valeur du V de Cramer	Intensité de la relation
<0.10	Relation nulle ou très faible
≥ 0.10 *et* < 0.40	Relation moyenne
≥ 0.40 *et* < 0.70	Relation forte
>0.70	Relation robuste

TABLE 1.3 – Interprétation des statistiques de V Cramer

Nous représentons ci-dessous la matrice de V de Cramer obtenue pour les variables qualitatives :

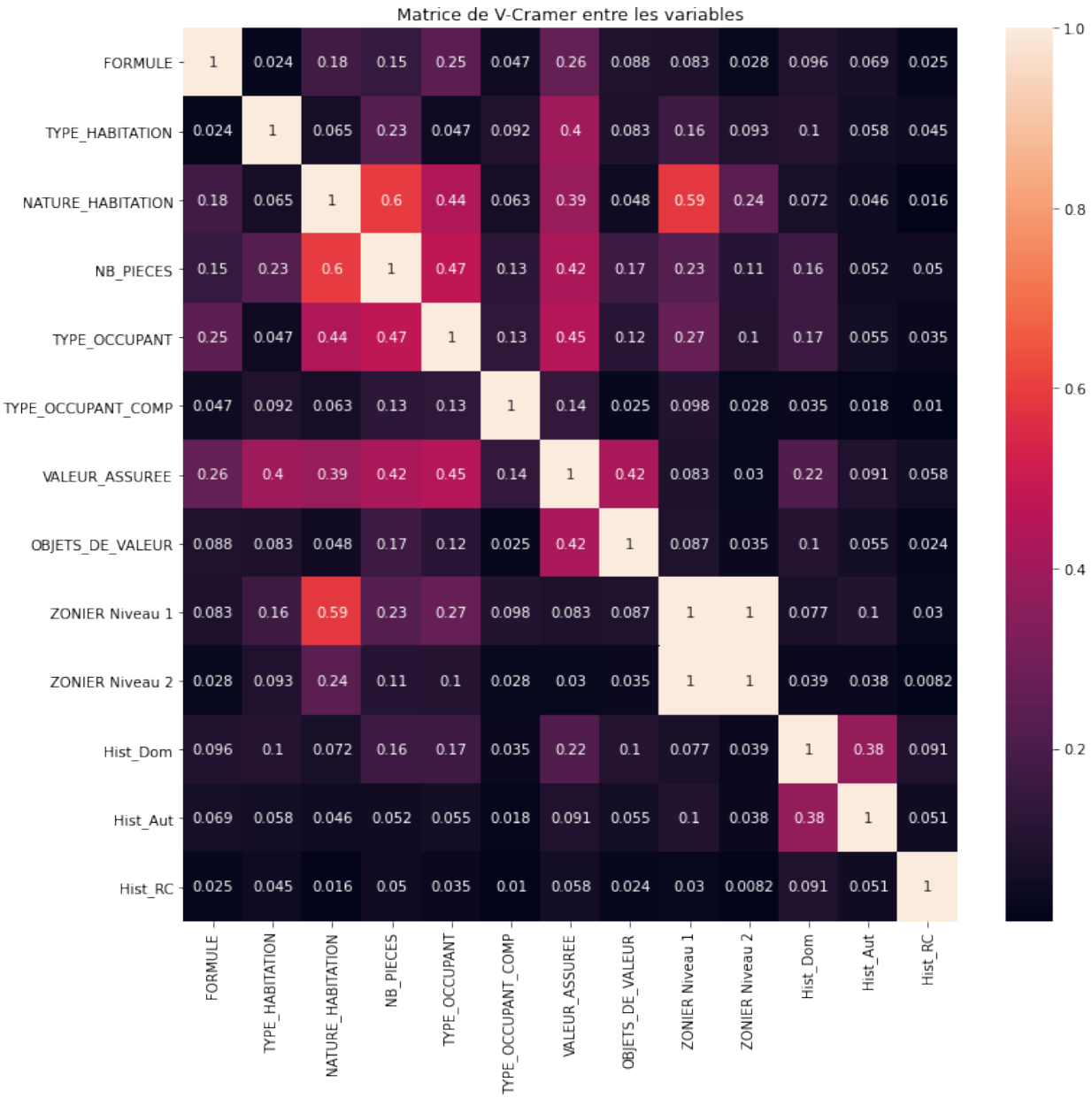


FIGURE 1.1 – Matrice de V Cramer

Les variables *Nature habitation* et *Nombre de pièces* ont un coefficient de Cramer de 0.62, ce qui correspond à une relation forte. Ces deux variables ont donc été couplées en une nouvelle variable *NatureHbt_NbPieces*. Cette nouvelle variable sera utilisée pour les GLM étant donné que ceux-ci sont très sensibles aux variables corrélées. Par ailleurs, certaines variables semblent avoir une relation moyenne-forte comme par exemple la valeur assurée avec le montant en objet de valeur, la nature d'habitation, le nombre de pièces ainsi que la type d'occupant. Ces autres variables ne seront pas couplées, la correspondance étant moins forte. Enfin, la variable de zone géographique de niveau 1 montre une relation robuste (V-Cramer = 1) à la variable de zone géographique de niveau 2, celle-ci ayant été créé à partir de la variable de zone géographique, c'est donc normal.

La matrice de V Cramer avec la nouvelle variable crée se trouve en annexe B.1.1. Après avoir réalisé le couplage de la variable *Nature habitation* et *Nombre de pièces*, la variable *Type occupant*

conserve une correspondance forte (V-Cramer = 0.52, voir B.1.1) avec la nouvelle variable *NatureHbt_NbPieces*. Ces deux variables ne sont pour autant pas jumelées pour des questions de lisibilité.

1.3.2 Statistiques sur les montants et le nombre de sinistres

Une vue d'ensemble des fréquences et des montants de sinistres par garantie (voir lexique des garanties en annexe A) est présentée dans les tableaux ci-dessous :

	Quantile 99	Somme	Moyenne	Écart type
ASS	21 000	8.5×10^6	2300	6068
BRIS	2 900	4.8×10^6	400	664
CNA N	32 000	6.4×10^6	5000	7886
CNA O	242 000	33.6×10^6	17000	49033
DDE	10 000	72.7×10^6	1300	2504
DIV	8 000	10.7×10^6	980	2814
IDC	4 800	2.2×10^6	370	1226
INC	342 800	26.3×10^6	17500	59045
PJ	23 200	6.6×10^6	2600	4194
RCC	104 600	9.5×10^6	8000	73692
RCM	15 500	16.7×10^6	1600	15072
REC	3 300	9.3×10^6	360	1197
TNG	12 100	20.2×10^6	1500	3165
VOL	11 600	32.0×10^6	1400	2670

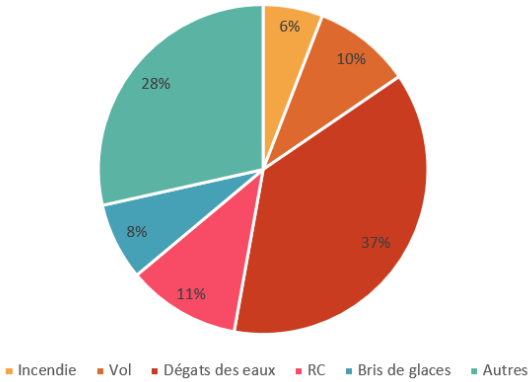
TABLE 1.4 – Statistiques sur les montants des sinistres par garantie

Garanties	Moyenne	Écart type	% individus avec sinistre
ASS	3.5×10^{-3}	0.068506	3.1×10^{-3}
BRIS	1.2×10^{-2}	0.124868	9.6×10^{-3}
CNA N	1.2×10^{-3}	0.041132	1.1×10^{-3}
CNA O	1.8×10^{-3}	0.049090	1.7×10^{-3}
DDE	5.4×10^{-2}	0.278588	4.6×10^{-2}
DIV	1.1×10^{-2}	0.121439	9.3×10^{-3}
IDC	5.5×10^{-3}	0.082901	4.9×10^{-3}
INC	1.5×10^{-3}	0.048995	1.3×10^{-3}
PJ	2.5×10^{-3}	0.059612	2.2×10^{-3}
RCC	1.1×10^{-3}	0.037744	1.0×10^{-3}
RCM	1.0×10^{-2}	0.116984	8.8×10^{-3}
REC	2.5×10^{-2}	0.188990	2.2×10^{-2}
TNG	1.2×10^{-2}	0.129135	1.1×10^{-2}
VOL	2.2×10^{-2}	0.173987	1.9×10^{-2}

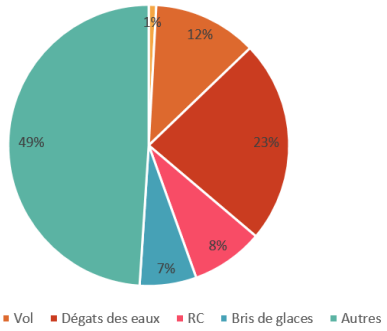
TABLE 1.5 – Statistiques sur les fréquences des sinistres par garantie

Comparaison des données du portefeuille aux données de la fédération française de l'assurance (FFA) :

Pour l'année 2018, il est possible de comparer les données du portefeuille à disposition avec les données à disposition sur le site de la FFA. Notamment, il est possible de comparer le poids de la fréquence par garantie dans la fréquence totale de sinistre du portefeuille par rapport



(a) Répartition des fréquences par garantie sur l'année 2018 en France



(b) Répartition des fréquences par garantie sur l'année 2018 dans le portefeuille

aux données globales. Le coût moyen des sinistres (MRH sur le graphique) pour l'ensemble des garanties est quant à lui légèrement inférieur pour les données FFA. On remarque que certaines garanties (RC et incendie) présentent des fréquences de sinistres bien inférieurs dans le portefeuille mis à disposition par rapport aux fréquences observées en France sur la même période. Le nombre de sinistres de la garantie dégâts des eaux est moins important en proportion dans le portefeuille mis à disposition que pour l'ensemble des assurés français.
Il est intéressant en même temps de comparer les fréquences pour certaines garanties entre les données mise à disposition et les données de la FFA.

Garanties	Données FFA	Données du portefeuille
Toutes garanties	93.7 ‰	183 ‰
BRIS	7.1 ‰	12.1 ‰
DDE	35.6 ‰	42.7 ‰
INC	5.5 ‰	1.58 ‰
RC	10.4 ‰	1.32 ‰
VOL	9.0 ‰	22.2 ‰

TABLE 1.6 – Comparaison des fréquences entre les données FFA et du portefeuille étudié pour certaines garanties (année 2018)

Pour l'année 2018, le portefeuille mis à disposition a une fréquence de sinistre significativement plus élevée que la fréquence française de sinistre pour la même année. Cela se remarque aussi au niveau local pour les garanties dégâts des eaux et vol qui ont des fréquences plus élevées, par ailleurs la garantie RC présente une fréquence plus faible.

Faisons le même type de comparaison pour le montant des sinistres.

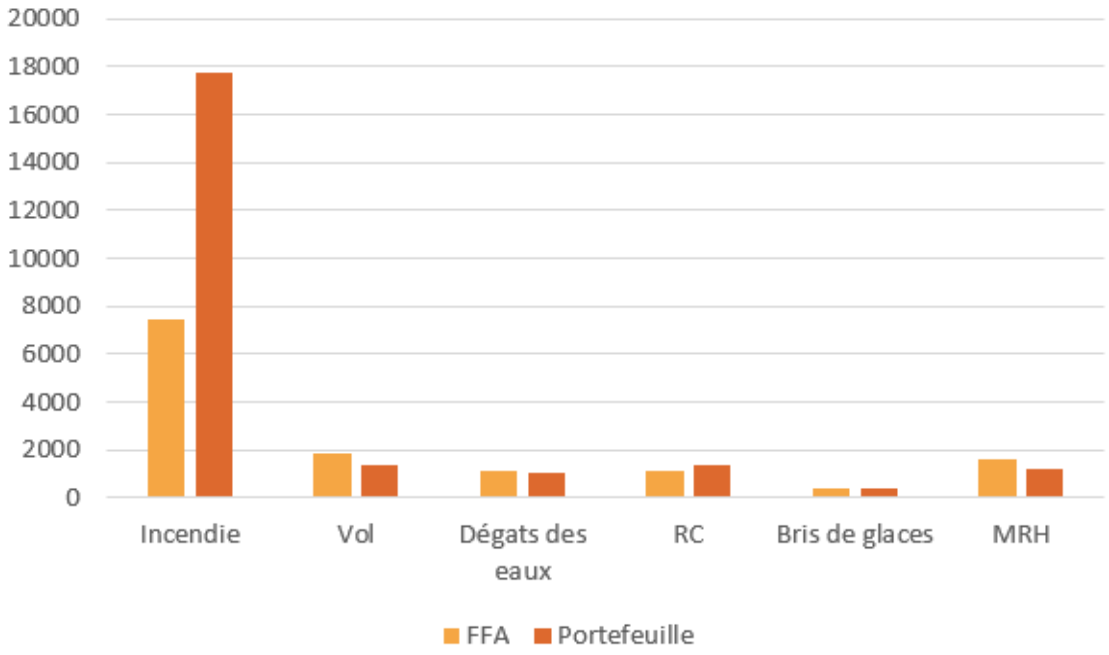


FIGURE 1.3 – Comparaison des coûts moyens des sinistres par garantie entre les données FFA et du portefeuille (année 2018)

La principale différence s'observe sur les coûts moyens pour les incendies qui sont nettement supérieurs dans le portefeuille mis à disposition par rapport aux données de la FFA. Pour les autres garanties, les montants de sinistres moyens sont similaires.
Après avoir comparé la fréquence et les coûts du portefeuille mis à disposition par rapport aux données de la FFA sur l'année 2018, on peut conclure qu'il y a certaines différences notables comme la fréquence moyenne bien plus élevée en générale. Ce portefeuille comporte peut-être des individus d'une certaine zone géographique ou d'un certain type qui sont en général plus sinistrés.

1.3.3 Observation de la fréquence en fonction des modalités des variables

Les graphiques ci-dessous sont ceux correspondants à la garantie dégâts des eaux. Les mêmes graphiques pour la garantie incendie sont présentés en annexe B.1.

Sensibilité à la variable "formule"

Une linéarité croissante de la fréquence de sinistre s'observe en fonction du niveau de garantie du contrat.

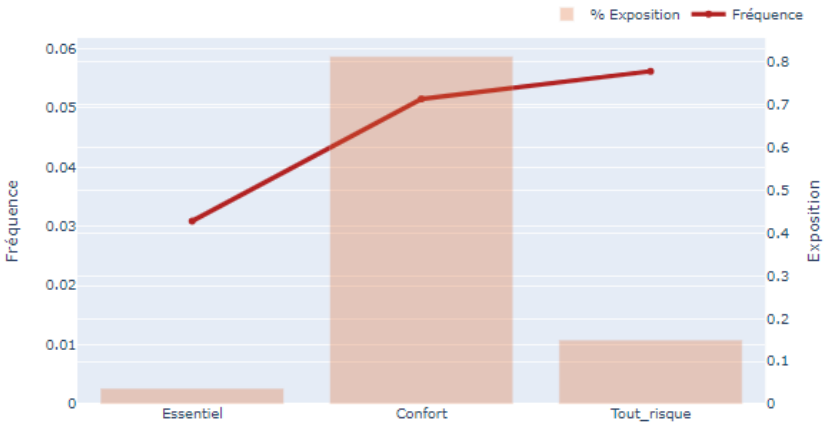


FIGURE 1.4 – Fréquence de sinistre par type de formule

Sensibilité à la variable "type d'habitation"

Pour la variable "Type d'habitation", la fréquence de sinistre est deux fois plus élevée pour la modalité domicile que pour la modalité *autres que domicile*.

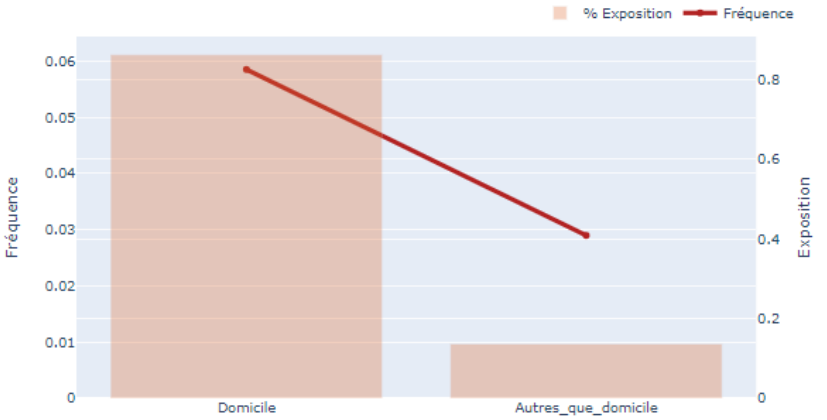


FIGURE 1.5 – Fréquence de sinistre par type d'habitation

Sensibilité à la variable "nature d'habitation"

Les modalités de la variable nature d'habitation sont représentées de la même manière en termes d'exposition dans le portefeuille, par ailleurs la fréquence de sinistre est plus faible pour la modalité maison que pour la modalité appartement. Dans les appartements, les logements peuvent subir des dégâts des eaux dues à des fuites survenues dans des appartements adjacents, ce qui peut expliquer cette fréquence supérieure.

FIGURE 1.6 – Fréquence de sinistre par nature d'habitation

Sensibilité à la variable "nombre de pièces"

La fréquence de sinistre est croissante avec le nombre de pièces. Plus un logement est grand, plus il y a de salle d'eau et donc la possibilité de dégâts des eaux y est importante.

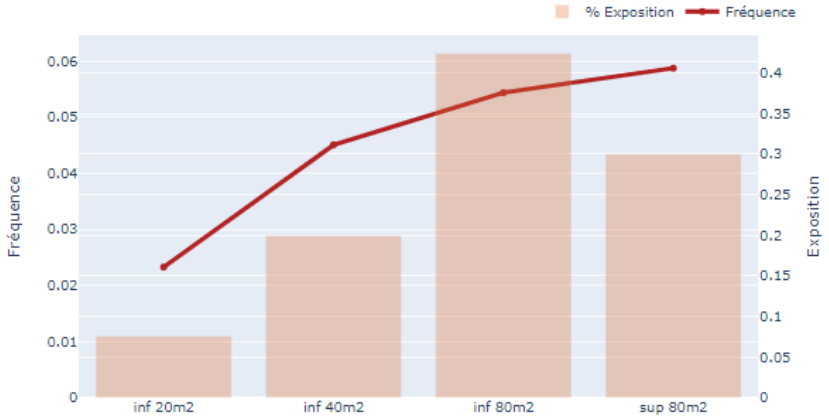


FIGURE 1.7 – Fréquence de sinistre par nombre de pièces

Sensibilité à la variable "type d'occupant"

Pour la variable type d'occupant, il est intéressant d'observer que la fréquence de sinistre est plus faible chez les locataires que chez les propriétaires. Le type d'occupant et le nombre de pièces sont corrélés d'après le graphique 1.1, et on observe une fréquence de sinistre plus élevé pour les logements ayant un nombre de pièces élevé. Or les propriétaires ont plus de pièces que les locataires, car on achète en général des biens de plus grande surface que ceux qu'on loue, ce qui peut expliquer cette différence de fréquence observée.

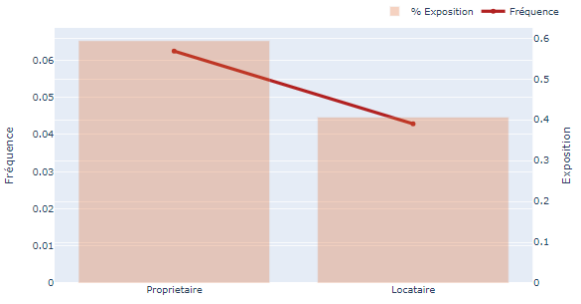


FIGURE 1.8 – Fréquence de sinistre par type d'occupant

Sensibilité à la variable "type d'occupant complémentaire"

La présence d'un colocataire semble avoir un effet à la baisse sur la fréquence de sinistre. Dans les collocations, il y a potentiellement plus de d'aller et venue, donc en permanence quelqu'un dans le logement. De ce fait, certains sinistres peuvent être évités par cette présence physique permanente.

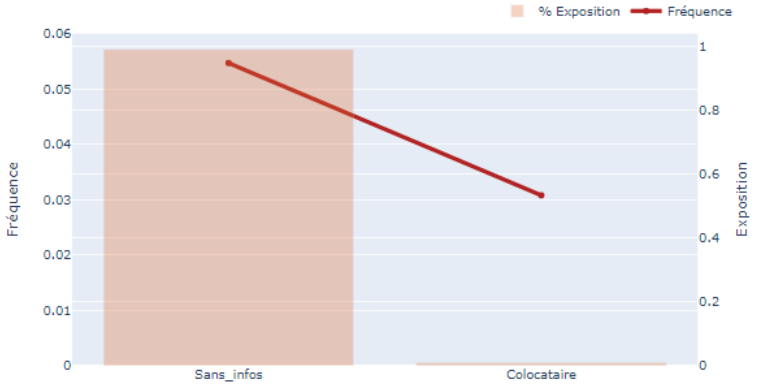


FIGURE 1.9 – Fréquence de sinistre par présence de colocataire

Sensibilité à la variable "valeur assurée"

Plus le montant de valeur assurée est élevé, plus la fréquence de sinistre est élevée.

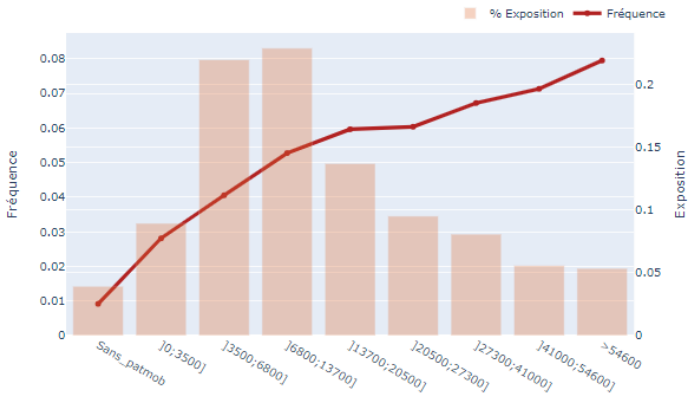


FIGURE 1.10 – Fréquence de sinistre par montant de valeur assurée

Sensibilité à la variable "zone géographique simplifiée"

La fréquence de sinistre est clairement plus élevée dans les milieux urbains. Dans les zones urbaines, les logements sont principalement des appartements, et nous avons observé une fréquence plus élevée pour ceux-ci, et cette différence de fréquence est cohérente avec celle observée sur la nature de l'habitation.

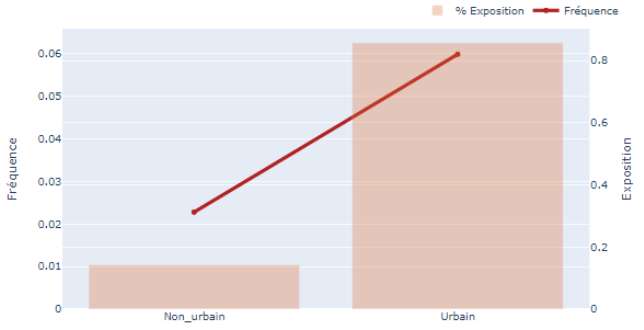


FIGURE 1.11 – Fréquence de sinistre par zone géographique simplifiée

Sensibilité à la variable "standing"

La fréquence de sinistre est légèrement plus forte pour les biens avec piscine que pour les biens sans piscine. Les piscines sont généralement signe de richesse, cette différence de fréquence peut donc être expliquée par le fait que les logements avec piscine ont en général une forte valeur assurée.

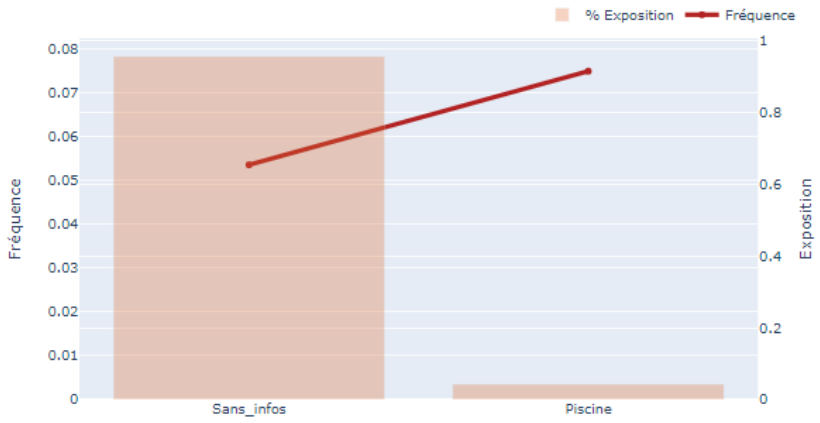


FIGURE 1.12 – Fréquence de sinistre par standing

Sensibilité à la variable "dépendance"

La modalité avec présence de grande dépendance semble avoir une fréquence de sinistre plus faible.

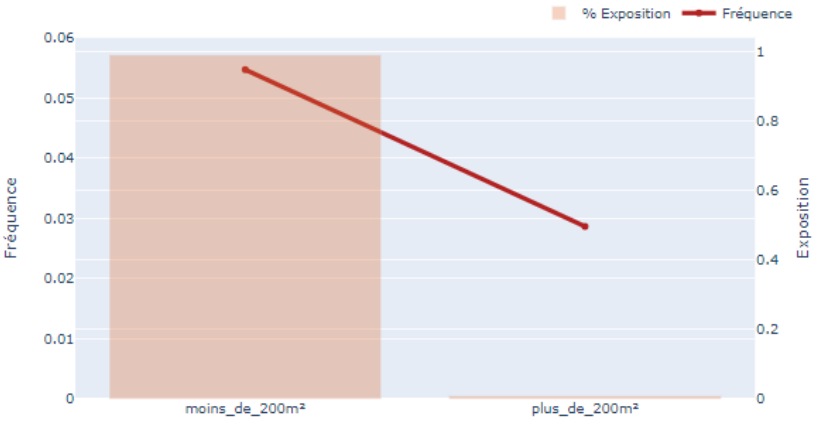


FIGURE 1.13 – Fréquence de sinistre par dépendance

1.3.4 Observation de la sévérité en fonction des modalités des variables

Après avoir observé la fréquence en fonction des modalités des différentes variables, il convient de faire de même pour la sévérité. Cette première approche graphique permettra d'avoir une première idée de l'impact de certaines modalités dans la prédiction des coûts dans les futurs modèles crées. Mais aussi pour permettre une certaine transparence des modèles "boites noires" en justifiant les résultats produits.
Les graphiques ci-dessous ont été tracés en prenant le log du montant des sinistres pour avoir plus de lisibilité du fait la présence de certains montants extrêmes dans la base.

Sensibilité à la variable "formule"

Les modalités Confort et Tout risque semblent avoir des montants de sinistre moyen plus élevés que pour la modalité Essentiel. Par ailleurs, la modalité Confort présente aussi une quantité élevée de valeurs significativement basses.

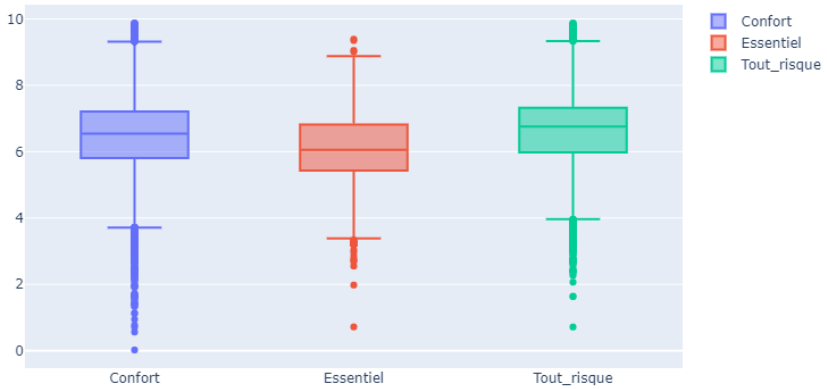


FIGURE 1.14 – Coût des sinistres par type de formule

Sensibilité à la variable "type d'habitation"

Le type d'habitation ne semble pas avoir d'effet significatif sur le coût des sinistres.

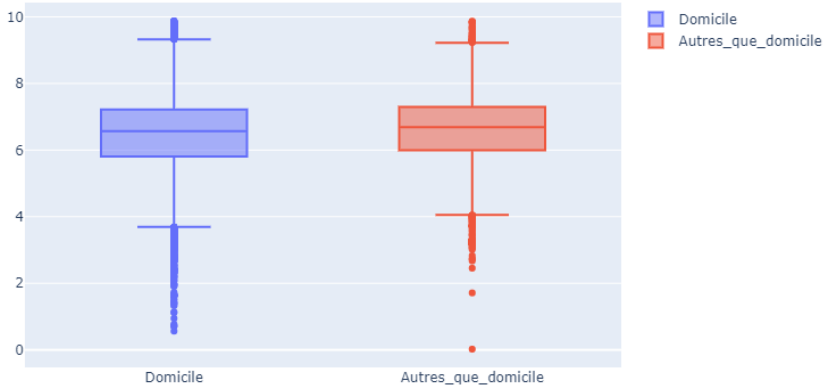


FIGURE 1.15 – Coût des sinistres par type d'habitation

Sensibilité à la variable "nature d'habitation"

La modalité maison a des montants de sinistre plus élevés, mais aussi plus dispersé que la modalité appartement. Ceci peut s'expliquer par des surfaces de biens plus hétérogènes chez les maisons et donc des coûts de sinistres plus dispersés.

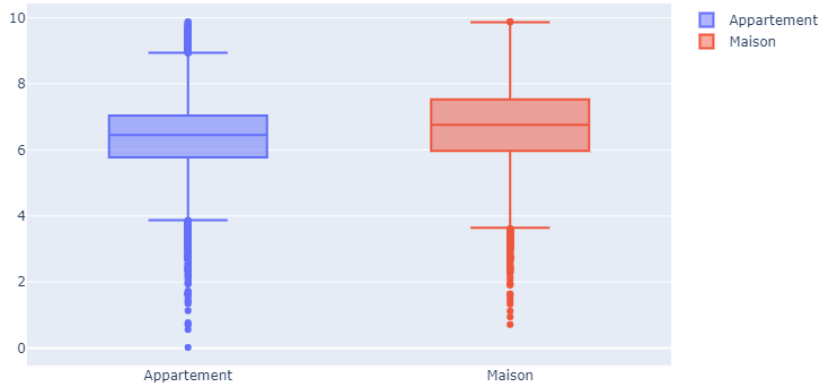


FIGURE 1.16 – Coût des sinistres par nature d'habitation

Sensibilité à la variable "nombre de pièces"

Une tendance croissante se dégage de ce graphique. Les logements ayant un nombre élevé de pièces ont des montants de sinistre plus élevés, mais aussi plus dispersés. Ceci peut s'expliquer par le fait que le nombre de pièces ait une faible correspondance avec la valeur assurée, et pour les dégâts des eaux une forte valeur assurée peut impliquer des forts montants de sinistre.

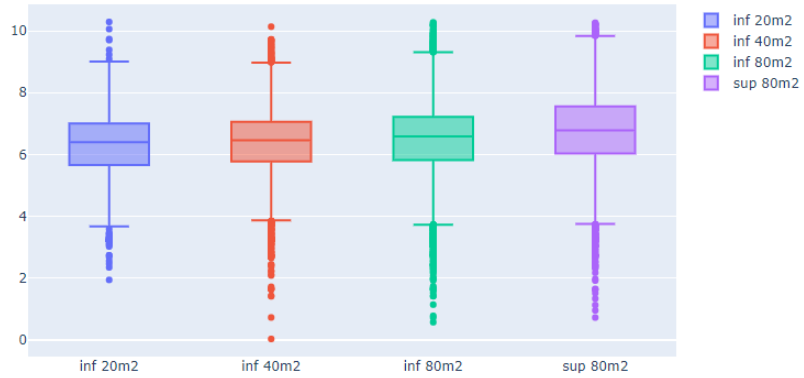


FIGURE 1.17 – Coût des sinistres par nombre de pièces

Sensibilité à la variable "type occupant"

Les propriétaires semblent avoir des montants de sinistres plus élevés que les locataires. De la même manière que précédemment, les propriétaires sont généralement dans des logements plus grands et donc ayant une plus forte valeur assurée.

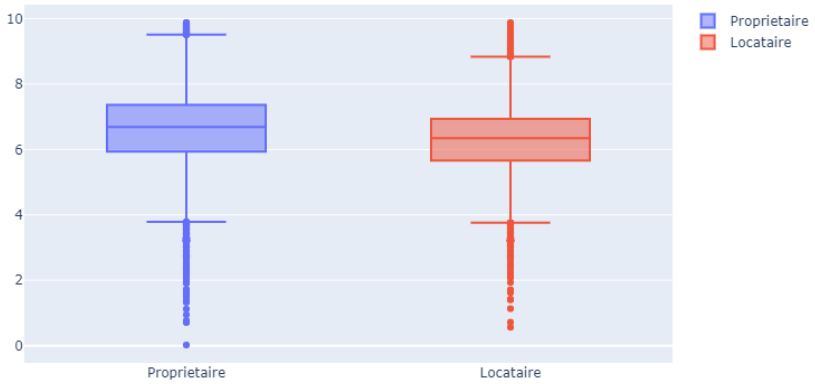


FIGURE 1.18 – Coût des sinistres par situation juridique

Sensibilité à la variable "type occupant complément"

Pour la variable complément de situation juridique, la modalité colocataire présente des montants de sinistres très peu dispersé et légèrement moins élevés que pour la modalité Sans infos. Ceci est cohérent avec le graphique précédent sur la variable type d'occupant.

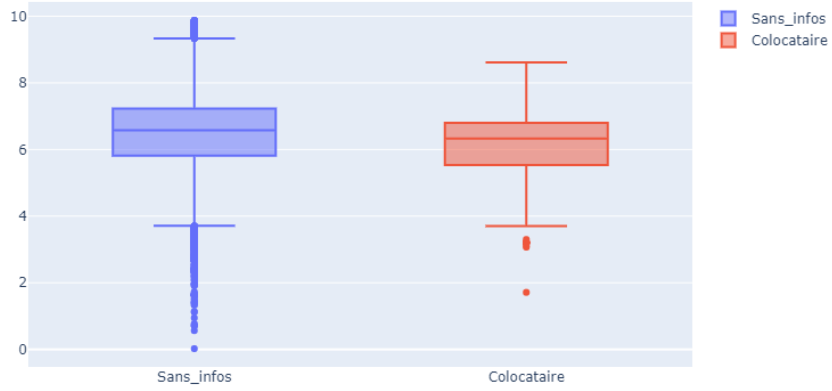


FIGURE 1.19 – Coût des sinistres par complément de situation juridique

Sensibilité à la variable "valeur assurée"

Une légère tendance croissante se détache de ce graphique en ce qui concerne la médiane du montant de sinistre et le montant de valeur assurée. C'est cohérent, plus la valeur assurée est élevée, plus les sinistres peuvent être élevés. Seule la modalité "Sans patrimoine" ne vérifie pas la tendance croissante, ce qui peut être dû au fait que ce sont des types de logements spéciaux.

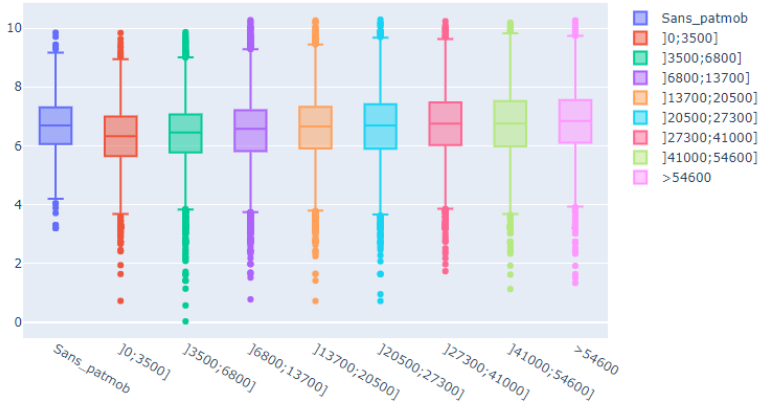


FIGURE 1.20 – Coût des sinistres par patrimoine mobilier

Sensibilité à la variable "objet de valeur"

La modalité représentant un fort montant d'objets de valeur présente des montants de sinistre plus élevés que pour un faible montant d'objets de valeur. Ceci s'explique de la même manière que précédemment pour la variable valeur assurée.

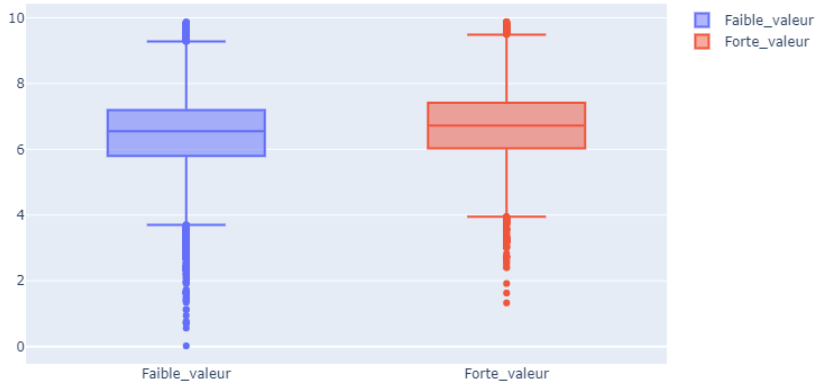


FIGURE 1.21 – Coût des sinistres par montant d'objet de valeur

Sensibilité à la variable "zone géographique simplifiée"

La variable zone géographique simplifiée ne semble pas avoir d'impact sur le montant des sinistres.

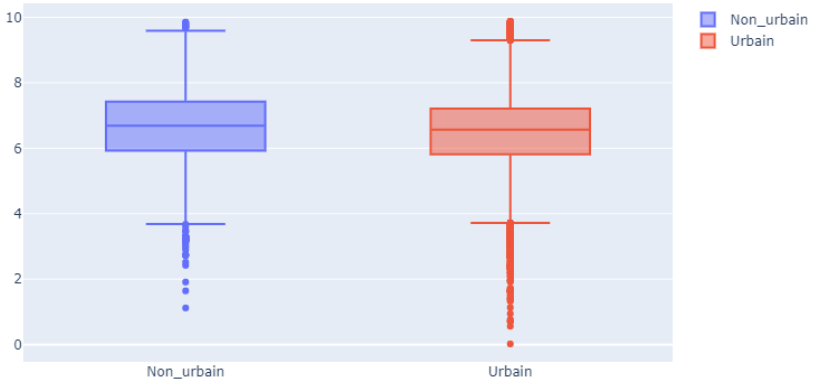


FIGURE 1.22 – Coût des sinistres par zone géographique simplifiée

Sensibilité à la variable "standing"

La présence d'une piscine dans le logement implique des montants de sinistres plus élevés en général. Ceci peut s'expliquer par le fait que la présence de piscine peut être un signe de richesse et donc les montants des sinistres pourront être plus élevés.

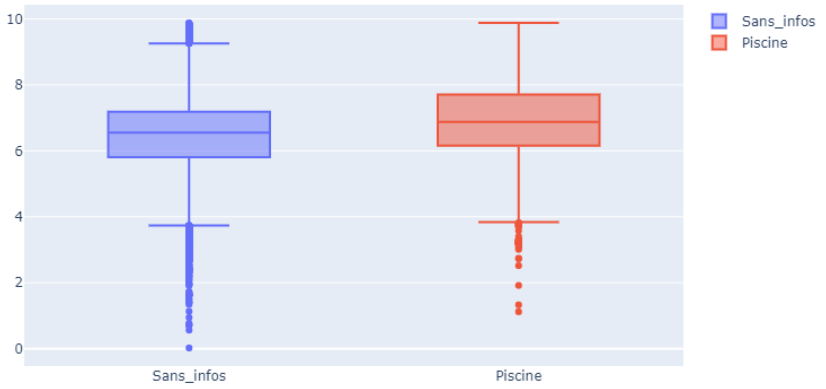


FIGURE 1.23 – Coût des sinistres par standing

Sensibilité à la variable "dépendance"

La présence d'une dépendance semble être vectrice de sinistres plus élevés en général.

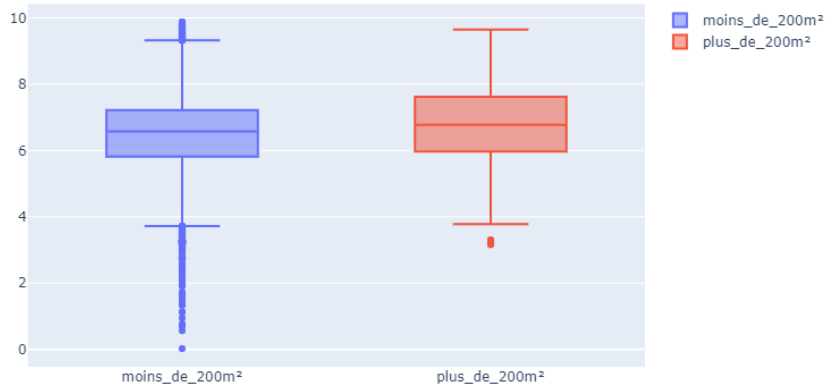


FIGURE 1.24 – Coût des sinistres par dépendance

CHAPITRE

2

TRAITEMENT DES DONNÉES ET MODÉLISATION CLASSIQUE DES SINISTRES

2.1 Traitement des données

L'objectif étant de ne conserver dans la base que les sinistres dit attritionnels, les sinistres vont être séparés en trois catégories : les graves, les forfaits et les attritionnels. Lors de la modélisation du coût moyen par des GLM ou d'apprentissage statistique les sinistres graves et forfaitaires pourront avoir un impact fort sur le modèle et le chambouler totalement. Le choix est donc fait de les traiter séparément.

2.1.1 Traitement des sinistres graves

Il existe de nombreux moyens de trouver le seuil à partir duquel on peut considérer qu'une variable a un comportement de loi Pareto Généralisée. L'approche retenue se déroule en trois phases :

- Observer le graphique de la fonction moyenne des excès pour détecter un seuil approximatif à partir duquel les points ont un comportement linéaire.
- Observer un QQ-plot pour déterminer le domaine d'attraction à une loi extrême.
- Calculer le seuil de manière statistique grâce à l'estimateur de Hill ou de Pickands suivant la valeur de l'indicateur de forme (ξ).

La loi de Pareto généralisée $GPD(\beta, \xi)$ est défini par la fonction de répartition :

$$G_{\beta,\xi}(x) \begin{cases} 1 - (1 + \xi \frac{x}{\beta})^{-1/\xi} & \text{si } \xi \neq 0 \\ 1 - e^{-\frac{x}{\beta}} & \text{si } \xi = 0 \end{cases}$$

où

$$\begin{aligned} x \geq 0 & \qquad \text{si } \xi \geq 0 \\ 0 \leq x \leq -\beta/\xi & \qquad \text{si } \xi < 0 \end{aligned}$$

Pour chacune des garanties présentes dans le portefeuille, le calcul du seuil a été effectué. Seul le processus pour la garantie dégâts des eaux sera détaillé, celui-ci étant identique pour chacune des garanties.

2.1.1.1 Fonction moyenne des excès

La fonction des excès moyen se définit de la manière suivante :

$$e(u) = E(X - u | X > u) \tag{2.1.1}$$

ou bien son approximation empirique s'écrit :

$$e_n(u) = \frac{\sum_{i=1}^{n} (x_i - u) \mathbb{1}_{\{x_i > u\}}}{\sum_{i=1}^{n} \mathbb{1}_{\{x_i > u\}}} \tag{2.1.2}$$

De plus, si les variables sont iid, la convergence du maximum des variables aléatoires correctement normalisées est équivalente à la convergence de la distribution des excès de seuil vers une loi Pareto Généralisée.
Pour la garantie dégâts des eaux, le graphique des excès moyens est présenté ci-dessous :

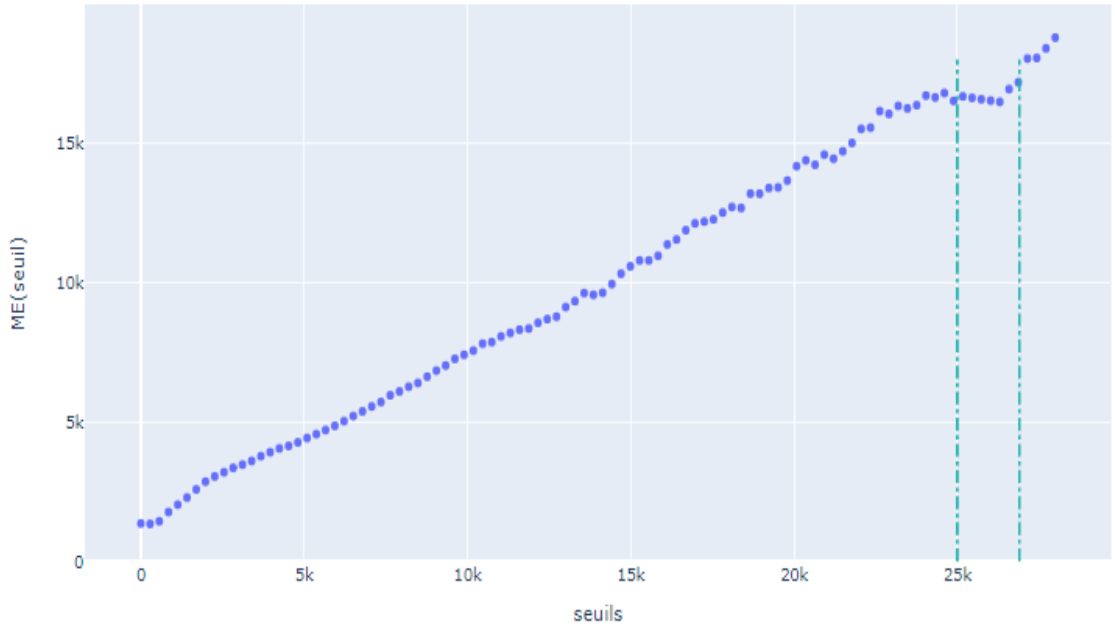


FIGURE 2.1 – Fonction moyenne des excès

Nous recherchons la valeur u à partir de laquelle la fonction $e_n(u)$ semble linéaire. La fonction moyenne des excès semble linéaire à partir d'un seuil aux alentours de 27K euro (entre les traits en pointillés).

2.1.1.2 Domaine d'attraction de la loi

Il préférable de donner une deuxième estimation du seuil, pour cela nous allons estimer le paramètre de queue afin de vérifier que nous avons bien $\xi < 1$ sans quoi la fonction moyenne des excès n'est pas cohérente et déterminer si $\xi > 0$, ce qui permettrait d'utiliser l'estimateur de Hill.
Tout d'abord il est nécessaire de déterminer le signe de ξ et ainsi le domaine d'appartenance de la variable.

Propriété : si une variable aléatoire X appartient au domaine d'attraction de la loi de Gumbel, alors la loi des excès de cette variable au-delà d'un certain seuil u converge vers une loi exponentielle lorsque $u \to \infty$.

Un graphique quantile-quantile (QQ plot) est une méthode pour vérifier l'adéquation d'une variable observée à une loi théorique donnée de fonction de répartition F continue. Si X a pour fonction de répartition F continue, alors $F(X) \sim U[0,1]$. En notant $X_{(1)} > ... > X_{(n)}$ la statistique d'ordre associé à une variable aléatoire, alors on a :

$$(F(X_{(i)}))_{i=1,...,n} = (U_{(i)})_{i=1,...,n}$$

qui est équivalent à :

$$(X_{(i)})_{i=1,...,n} = (F^{-1}(U_{(i)}))_{i=1,...,n}$$

Le QQ plot est donné par :

$$\{(X_{(i)})_{i=1,...,n} = (F^{-1}(1 - i/n)) : i = 1, ..., n\}$$

L'adéquation des observations à la loi de fonction de répartition F se traduit par la linéarité du nuage de points obtenu. Soit dans notre cas, avec la loi exponentielle :

$$\{(X_{(i)})_{i=1,...,n} = (-ln(i/n)) : i = 1, ..., n\}$$

Interprétation du QQ plot :

- Si la loi dont sont issues les observations appartient au domaine d'attraction maximum de la loi de **Gumbel**, alors les points du QQ plot sont approximativement alignés.
- Si la loi dont sont issues les observations appartient au domaine d'attraction maximum de la loi de **Weibull**, alors une forme convexe est observée.
- Si la loi dont sont issues les observations appartient au domaine d'attraction maximum de la loi de **Fréchet**, alors une forme concave est observée.

Le QQ plot obtenu pour la garantie dégâts des eaux présenté ci-dessous indique une légère concavité, ce qui traduit l'appartenance au domaine d'attraction de la loi de Fréchet et que donc $\xi > 0$.

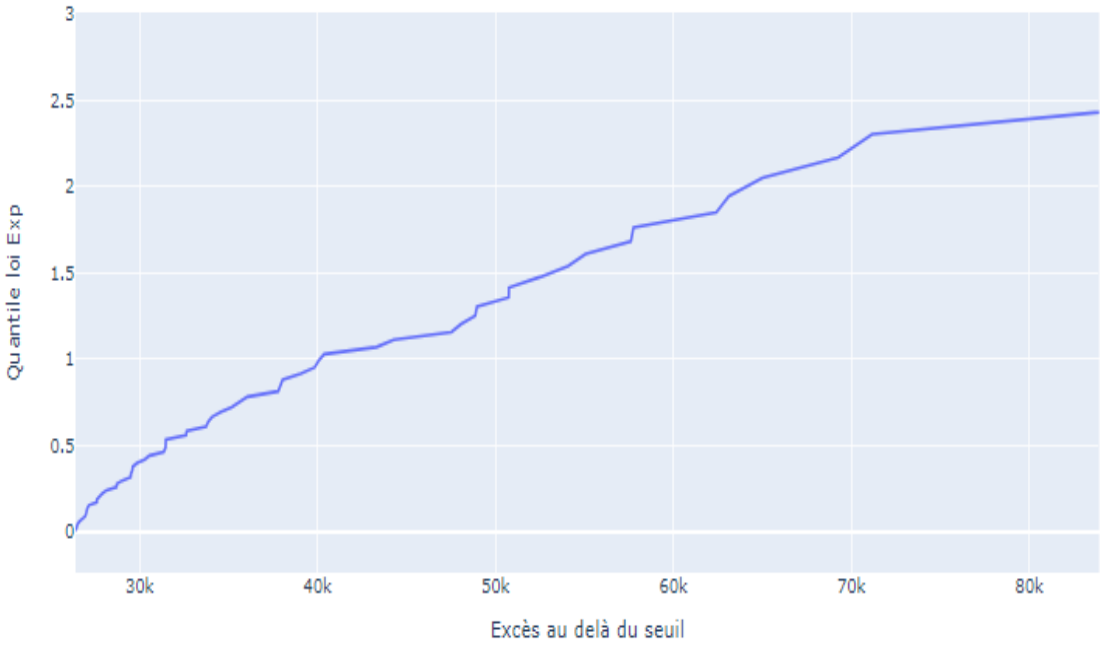


FIGURE 2.2 – Graphique quantile-quantile : excès de sinistre au-delà du seuil vs loi exponentielle

2.1.1.3 Deuxième estimation du seuil

L'estimateur de Hill qui est le plus utilisé en théorie des valeurs extrêmes est valide que lorsque $\xi > 0$. Il se définit de la manière suivante :

$$\xi_{k,n}^{Hill} = \frac{1}{k} \sum_{i=1}^{k} ln(X_{(i)}) - ln(X_{(k+1)})$$

Si $k \to \infty, k/n \to 0$ comme $n \to \infty$, les estimateurs sont faiblement convergents.
Si $\xi \in \mathbb{R}$, l'estimateur retenu parmi les différents estimateurs possible est celui de Pickands qui est défini de la manière suivante :

$$\xi_{k,n}^{Pickands} = \left[ln \frac{X_{(k)} - X_{(2k)}}{X_{(2k)} - X_{(4k)}} \right] / ln(2)$$

Ces estimateurs sont volatiles puis se stabilisent lorsque k est grand. Le seuil des graves peut être déterminé comme l'indice à partir duquel l'estimateur se stabilise. Sur le graphe obtenu, l'estimateur de Hill se stabilise aux alentours de l'indice 44, ce qui signifie que le seuil peut être considéré comme la 44ème valeur la plus grande de l'ensemble des données. Le seuil retenu est donc 29K euro.

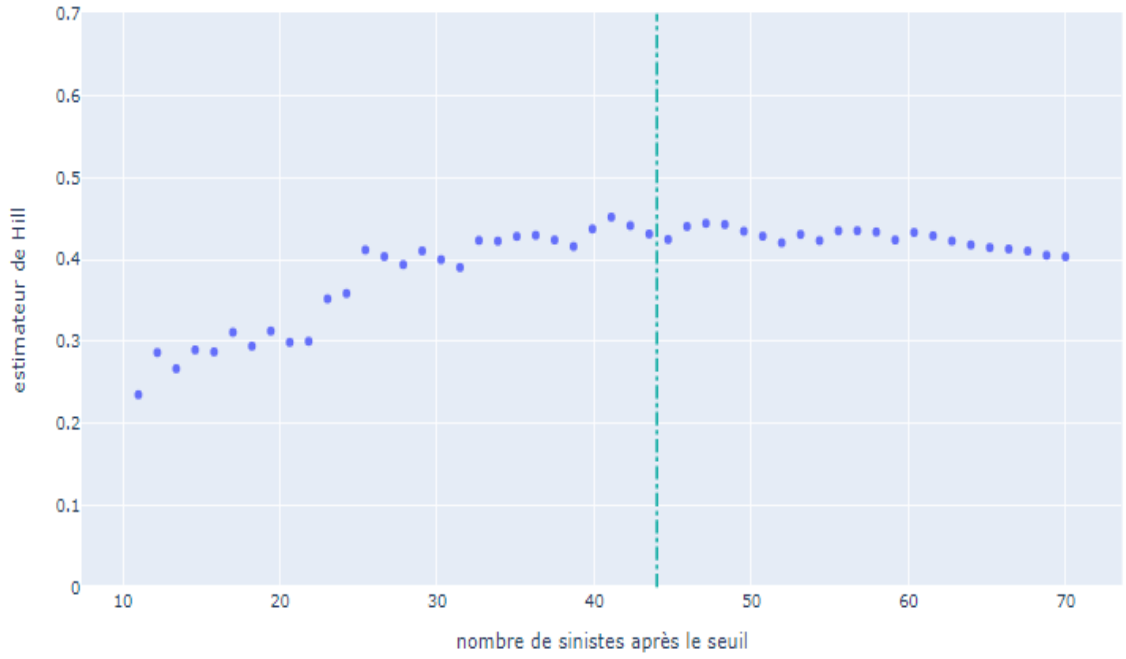


FIGURE 2.3 – Détermination du seuil : Estimateur de Hill

2.1.1.4 Récapitulatif de la détermination des seuils

Garantie	Seuil	% de graves présents
ASS	9771.23	2.78
BRIS	3932.90	0.54
CNA N	33595.75	1.10
CNA O	86815.86	4.27
DIV	7029.95	1.35
IDC	5335.60	0.78
INC	95925.78	4.83
PJ	10706.45	5.00
RCC	87406.33	1.43
RCM	12966.99	1.36
REC	3093.90	1.17
TNG	11598.14	1.07
VOL	16364.52	0.43
DDE	29514.22	0.09

TABLE 2.1 – Seuils de graves des différentes garanties

Les éléments de cette section ont été rédigés en se basant sur le cours de Christian Yann Robert concernant la théorie des valeurs extrêmes [17].

2.1.2 Traitement des sinistres forfaitaires

Pour les traitements sinistres forfaitaires, le partie a été pris de considérer qu'un sinistre est forfaitaire lorsqu'il apparaît plus de x % du temps dans le portefeuille pour une garantie donnée. La séparation des sinistres forfaitaires et des sinistres attritionnels a été faite après avoir écrêté les sinistres. Le seuil retenu arbitrairement est de 3 %. Pour certaines garanties présentes dans

le portefeuille, le taux de sinistres forfaitaires est très important et donc la modélisation du coût moyen de sinistre pour cette garantie serait mauvaise. Par exemple pour la garantie PJ qui comporte 44 % de sinistres forfaitaires, les graphiques ci-dessous représentent la distribution et la fonction de répartition empirique avant et après avoir enlevé les forfaits.

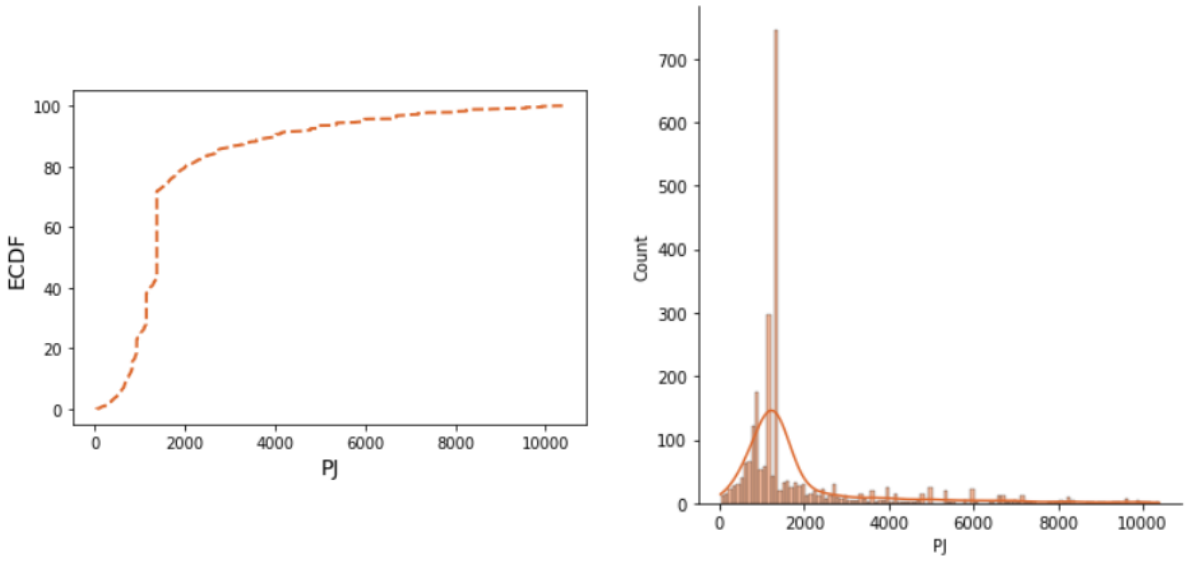


FIGURE 2.4 – Garantie PJ : Fonction de répartition empirique et distribution des sinistres

On observe bien les sauts sur la fonction de répartition empirique, ce sont ces données qu'il est nécessaire de mettre de côté et de traiter séparément.

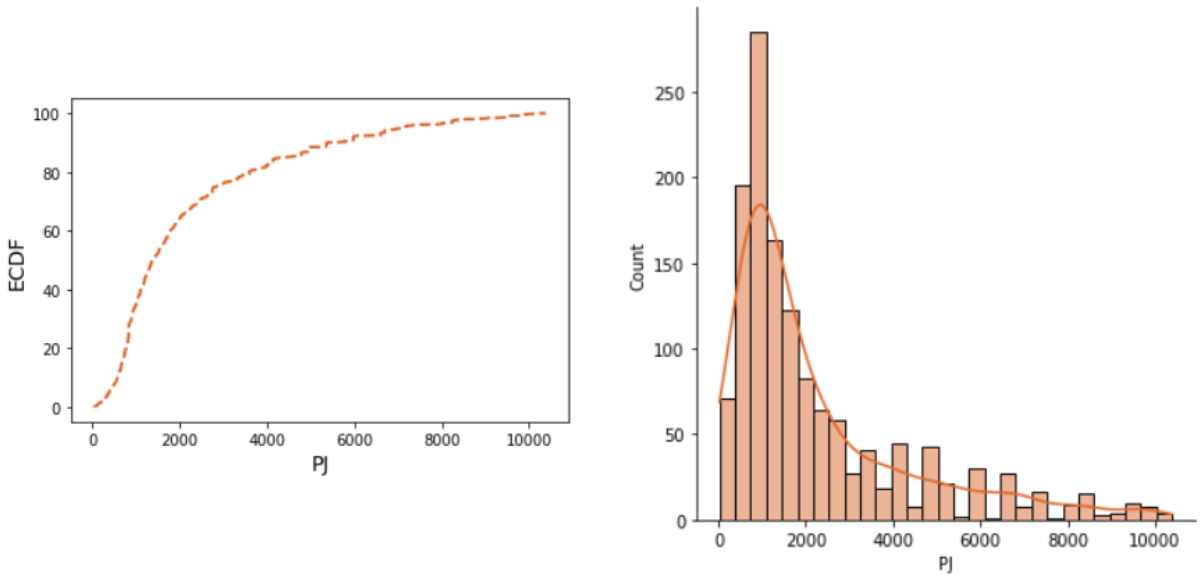


FIGURE 2.5 – Garantie PJ : Fonction de répartition empirique et distribution des sinistres sans les forfaits

Garantie	% de forfaits présents
ASS	20.71
BRIS	3.29
CNA N	37.80
CNA O	21.34
DIV	7.08
IDC	55.06
INC	7.04
PJ	44.54
RCC	30.36
RCM	22.38
REC	49.90
TNG	13.05
VOL	0
DDE	5.08

TABLE 2.2 – Séparation des forfaits et des attritionnels

2.1.3 Visualisation des données finales

Choix de la garantie étudiée :

Pour l'étude de ce portefeuille, le choix a été fait de ne traiter seulement qu'une garantie ; celle qui répond à trois critères : forte fréquence, faible sévérité et nombre important de données (>50 000 lignes de sinistre). La garantie dégâts des eaux est ainsi sélectionnée.

Pour cette garantie, les histogrammes ci-dessous représentent les coûts et la fréquence. Pour les coûts, l'abscisse maximale a été fixée à 4500 pour plus de clarté.

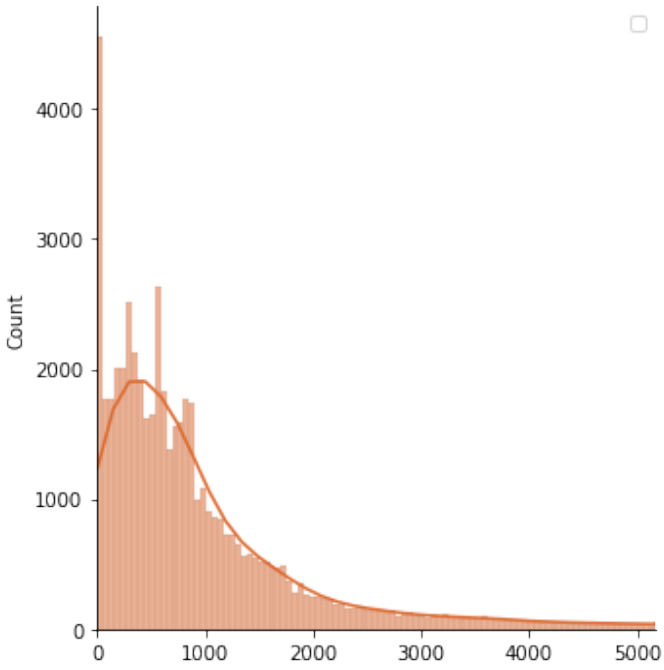


FIGURE 2.6 – Histogramme des coûts moyens attritionnels : DDE

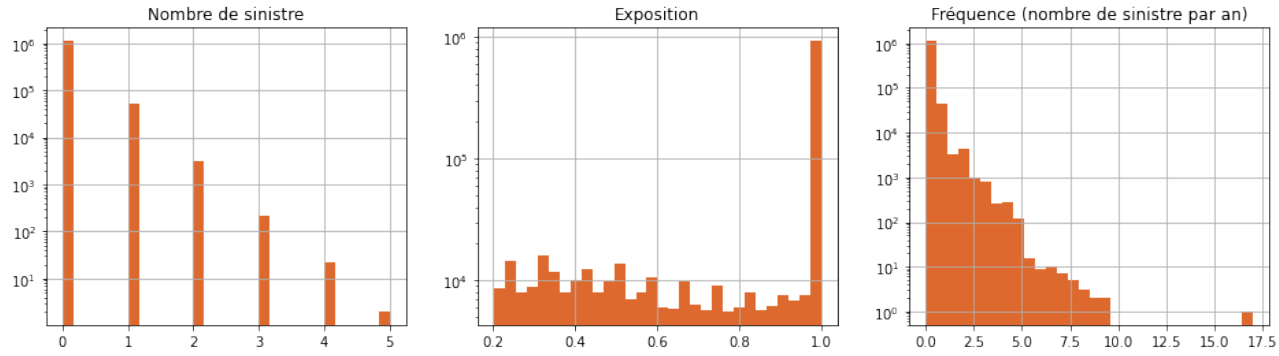


FIGURE 2.7 – Graphiques sur la fréquence : DDE

Notons que la fréquence moyenne de sinistre (par an) est de 0.0544, et que 95.2 % de l'exposition totale présent dans le portefeuille correspond à des individus sans sinistre.

2.2 Modélisation classique des sinistres

2.2.1 Modèle Coûts x Fréquence

2.2.1.1 Prime pure

L'assureur doit répondre à un problème d'optimisation simple :

$$\underset{P>0}{min} E[(X-P)^2] \tag{2.2.1}$$

où X est une variable aléatoire représentant le montant de sinistre probable sur une période donné pour un assuré donné, et où P est une constante correspondant au montant de prime demandé à un assuré donné.
Le résultat de ce problème d'optimisation est : $P = E(X)$. La prime pure correspond donc à l'espérance de la charge de sinistre pour un assuré donné.

2.2.1.2 Modélisation

Il nous faut donc à présent modéliser la charge de sinistre X. Selon le modèle général pour l'approche fréquence/sévérité, on définit la variable aléatoire X comme la somme :

$$X = \begin{cases} \sum_{i=1}^{N} B_i & \text{si } N > 0 \\ 0 & \text{si } N = 0 \end{cases} \tag{2.2.2}$$

où N est le nombre de sinistres pour une période donnée et les B_i représentent les montants de sinistres. On suppose que N est une variable discrète (dans $\mathbb{N}$) représentant la fréquence tandis que les B_i sont des variables continues positives représentant la sévérité (dans $\mathbb{R}_+$). Un couple de lois pour le nombre et le montant des sinistres définit un modèle fréquence/sévérité.
Nous faisons les hypothèses suivantes :

- H1 : Sévérité i.i.d : $(B_i)_i \overset{iid}{\sim} B$
- H2 : Fréquence indépendante : $\forall i, N \perp \mathrm{B}_i$

Ces deux hypothèses seront vérifiées avant la mise en application des GLM.
On peut alors, grâce au théorème de Wald, décomposer l'espérance des pertes de la façon suivante :

$$E(X) = E(N) \times E(B) \tag{2.2.3}$$

Cette formule met en avant l'idée que la prime pure est égale à la fréquence moyenne fois le coût moyen. Par la suite, il s'agira donc de faire un modèle de fréquence (variable cible : $\frac{Nombredesinistre}{Exposition}$) ainsi qu'un modèle de coût (variable cible : $\frac{Coûts}{Nombredesinistre}$) pour pouvoir obtenir cette prime pure. Plusieurs méthodes existent pour faire ces modèles, notamment les GLM, les Forets aléatoires et le Gradient Boosting. Ces trois méthodes seront comparées par la suite. Une autre méthode consiste à directement faire un modèle de prime pure, c'est à dire de modéliser comme variable cible $\frac{Coûts}{Exposition}$.

Zoom sur les variables cibles :
Habituellement lors de l'utilisation des GLM pour la tarification, par exemple pour la modélisation de la fréquence, la variable cible utilisée est le nombre de sinistre. L'exposition apparaît dans l'offset du GLM pour la modélisation de la fréquence. Alors pourquoi, dans ce mémoire pour la modélisation du nombre de sinistre, la variable cible prise est $\frac{Nombredesinistre}{Exposition}$? Pour rappel, sans offset, un GLM (avec fonction de lien logarithmique) s'écrit de la manière suivante :

$$log(Y_i) = \beta' X_i + \epsilon_i \tag{2.2.4}$$

Lorsque l'on souhaite que l'exposition ait un impact sur le modèle, il est nécessaire de la spécifier comme offset dans celui-ci. Cela donnera le modèle suivant :

$$log(Y_i) = \beta' X_i + \epsilon_i + log(\text{expo}_i) \tag{2.2.5}$$

$$\Leftrightarrow \qquad log(\frac{Y_i}{\text{expo}_i}) = \beta' X_i + \epsilon_i \tag{2.2.6}$$

Nous allons montrer que les deux approches suivantes sont équivalentes :

- variable cible = Y, offset = exposition
- variable cible = $\frac{Y}{exposition}$, poids = exposition

Considérons tout d'abord que la variable cible Y_i ait une distribution de Poisson de paramètre λ_i. La log-vraissemblance de ce modèle est la suivante (en ne conservant que les termes dépendants de β) :

$$l = \sum_{i=1}^{n} [y_i \beta' x_i - \text{expo}_i \; e^{\beta' x_i}] \tag{2.2.7}$$

Si l'on considère cette fois-ci un modèle de Poisson pondéré par les expositions et en prenant pour variable cible $\frac{Y_i}{\text{expo}_i}$, cela donnera la même log-vraissemblance :

$$l_{poids} = \sum_{i=1}^{n} \text{expo}_i [\frac{y_i}{\text{expo}_i} \beta' x_i - e^{\beta' x_i}] \tag{2.2.8}$$

On a donc démontré que l'approche (variable cible = $\frac{Y}{exposition}$, poids = exposition) est équivalente à l'approche classique. Ceci est nécessaire, du fait que dans la librairie scikit-learn qui

a été utilisée pour réaliser les modèles de machine learning, il n'est pas possible de définir un offset pour le modèle. Il est seulement possible de définir un poids.
De la même manière que pour la fréquence de sinistre, il est possible de montrer que la variable cible utilisée pour modéliser le montant des sinistres est équivalente à la méthode usuelle connue. La méthode utilisée correspond donc bien au modèle coûts/fréquence classique connu pour la tarification Non-Vie.

2.2.2 Modèle linéaire généralisé

Les modèles linéaires généralisés sont les modèles de base utilisés en tarification en assurance. Ils servent à faire le lien entre une variable à expliquer (fréquence ou coût) et des variables explicatives. Son avantage est qu'il est relativement simple à modéliser et qu'il est interprétable grâce à ses coefficients de sorties. Cette section est inspirée du cours d'Actuariat Non Vie [5]

2.2.2.1 Définition

Les modèles linéaires généralisés sont une généralisation du modèle linéaire Gaussien, obtenu en autorisant d'autres lois que la loi Gaussienne. Les lois possibles doivent appartenir à la famille exponentielle, i.e. dont la densité (ou mesure de probabilité dans le cas discret) s'écrit :

$$f(y|\theta,\phi) = \exp\left(\frac{y\theta - b(\theta)}{a(\phi)} + c(y,\phi)\right) \tag{2.2.9}$$

où a, b et c sont des fonctions et θ, ϕ les paramètres.
Cette famille de lois va s'avérer être particulièrement utile pour construire des modèles beaucoup plus généraux que le modèle Gaussien usuel. On suppose disposer d'un échantillon (Y_i, X_i), où les variables X_i sont des informations exogènes sur l'assuré ou sur le bien assuré, et où Y_i est la variable d'intérêt, qui sera par exemple :

- Une variable booléenne prenant les valeurs 0, 1, par exemple l'assuré i a-t-il été victime d'un accident l'an dernier, ou le sinistre i était-il très important,
- Une variable de comptage, à valeurs dans $\mathbb{N}$, par exemple le nombre d'accidents de l'assuré i l'an passé,
- Une variable positive, par exemple le coût du sinistre i

On peut montrer que :

— $E(Y_i) = \mu_i = b'(\theta_i)$
— $Var(Y_i) = \sigma^2 = b''(\theta_i)a(\phi)$

On supposera que, conditionnellement aux variables explicatives X, les variables Y sont iid. En particulier, on partira d'un modèle de la forme :

$$f(y_i|\theta_i,\phi) = \exp\left(\frac{y_i\theta_i - b(\theta_i)}{a(\phi)} + c(y_i,\phi)\right) \tag{2.2.10}$$

où l'on supposera que $g(\mu_i) = \eta_i = X_i'\beta$, pour une fonction de lien g$(\cdot)$ donnée, et où $\mu_i = E(Y_i)$. La fonction lien g est la fonction qui permet de lier les variables explicatives X à la prédiction μ, alors que la loi apparaît via la fonction variance, sur la forme de l'hétéroscédasticité et l'incertitude associée à la prédiction.

On peut noter que les paramètres θ_i sont liés au prédicateur linéaire par la relation $E(Yi) = b'(\theta_i) = g^{-1}(\eta_i)$.
Si $\theta_i = \eta_i$, ie $g(y) = (b')^{-1}(y)$, alors la fonction de lien est dite canonique.

2.2.2.2 Estimation

L'estimation du vecteur β et du paramètre ϕ se fait par la méthode du maximum de vraisemblance.
La log-vraissemblance s'écrit :

$$\log \mathcal{L}(\beta, \phi) = \sum_{i=1}^{n} \frac{y_i \theta_i(\beta) - b(\theta_i(\beta))}{a(\phi)} + \sum_{i=1}^{n} c(y_i, \phi) \tag{2.2.11}$$

avec $\theta_i = (b')^{-1}(g^{-1}(\eta_i))$.
Les équations du score dans le cas général sont donc

$$\forall j = 1, ..., d : \quad \sum_{j=1}^{d} \frac{y_i - \mu_i}{a(\phi_i) V(\mu_i)} \frac{x_{ij}}{g'(\mu_i)} = 0 \tag{2.2.12}$$

Dans le cas d'un lien canonique, les équations se simplifient

$$\forall j = 1, ..., d : \quad \sum_{j=1}^{d} \frac{y_i - \mu_i}{a(\phi_i)^2} x_{ij} = 0 \tag{2.2.13}$$

Le paramètre β est calculé par méthode numérique (Newton-Raphson). Pour le paramètre ϕ, il est usuel de faire certaines hypothèses, par exemple supposer que $a(x) = x$.

2.2.2.3 Procédure de sélection des variables

La sélection de variables est utile pour ne calibrer les GLM que sur les variables qui ont un effet bénéfique sur le modèle. Dans certains, cas ajouter une variable peut rendre le modèle moins bon. Trois méthodes classiques existent pour cette procédure de sélection de variables.

La Backward selection : Elle consiste à effectuer une régression avec toutes les variables explicatives possibles pour commencer. Ensuite à chaque étape, on enlève une variable et on regarde l'impact que cela a sur l'AIC du modèle. Le processus s'arrête lorsque le retrait de toute variable possible ne permet plus d'améliorer l'AIC.

La Forward selection : Il s'agit du même principe que la Backward selection, mais en sens inverse. Pour cette méthode, il s'agit de commencer avec un modèle trivial et de tester l'ajout de chaque variable explicative une à une. La variable ajoutée qui diminue le plus l'AIC est alors conservée. Ce processus est répété jusqu'à ce que l'ajout d'une variable n'entraîne plus d'amélioration de l'AIC.

La Stepwise selection : Il s'agit d'une méthode combinant le principe de la méthode Forward et la méthode Backward. À chaque étape, on ajoute la variable la plus significative et on

enlève celles qui ne sont plus significatives. Le processus s'arrête lorsque l'AIC est minimal.

Par ailleurs d'autres méthodes peuvent être aussi utilisées mais ne seront pas détaillées, comme le test de significativité de Wald, ou des méthodes de sélection de variable de type machine learning comme la méthode Lasso.

CHAPITRE

3

APPORT DU MACHINE LEARNING DANS LA MODÉLISATION DES SINISTRES

3.1 Méthodes de machine learning : Théorie

De nombreuses techniques d'apprentissage statistique se basent sur la combinaison de plusieurs classifieurs instables (par exemple, des réseaux de neurones, boosting, forêts aléatoires, ...). Des résultats ont montré que ces versions améliorées peuvent être plus performantes (i.e. un taux d'erreur plus faible).

3.1.1 Généralité sur l'apprentissage supervisé

3.1.1.1 Définition de base

Soit une base de données composée de n couples $Z_i = (X_i, Y_i)$ que l'on suppose être des réalisations indépendantes d'une même loi O inconnue. On écrira

$$Z_i = (X_i, Y_i) \overset{iid}{\sim} P$$

Les X_i appartiennent à un espace $\mathcal{X}$, et s'appellent les variables ou les features (en anglais). Typiquement $\mathcal{X} = \mathbb{R}^p$ pour un entier p. Les Y_i appartiennent à un espace $\mathcal{Y}$, et s'appellent les sorties ou target (en anglais). Typiquement $\mathcal{Y}$ est fini ou $\mathcal{Y}$ est un sous-ensemble de $\mathbb{R}$.

But de l'apprentissage supervisé : [2]
Prédire la sortie Y associée à toute nouvelle entrée X, où il est sous-entendu que la paire (X, Y) est une nouvelle réalisation de la loi P, cette réalisation étant indépendante des réalisations précédemment observées.

Une fonction de prédiction est une fonction mesurable de $\mathcal{X}$ dans $\mathcal{Y}$. Dans ce qui suit, on supposera que toutes les quantités que l'on manipulera sont mesurables. L'ensemble de toutes les fonctions de prédiction est noté $\mathcal{F}(\mathcal{X}, \mathcal{Y})$. La base de données $Z_1, ..., Z_n$ est appelée ensemble d'apprentissage.
Un algorithme d'apprentissage est une fonction qui à tout ensemble d'apprentissage renvoie une fonction de prédiction, c'est à dire une fonction de l'union

$$\bigcup_{n=1}^{\infty} \mathcal{Z}^n$$

dans l'ensemble $\mathcal{F}(\mathcal{X}, \mathcal{Y})$, où $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. C'est un estimateur de la "meilleure" fonction de prédiction, où le terme "meilleure" sera précisé ultérieurement.
Soit $l(y, y')$ la perte encourue lorsque la sortie réelle y et la sortie prédite est y'. La fonction $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ est appelée fonction de perte. Les deux exemples les plus fréquemment utilisés sont :

- **Exemple de la régression L_p :** $\mathcal{Y} = \mathbb{R}$ et $l(y, y') = |y - y'|^p$ où $p \geq 1$ est un réel fixé. Dans ce cas, on parle de régression L_p. La tâche d'apprentissage lorsque $p = 2$ est aussi appelé régression des moindres carrés.
- **Exemple de la classification :** $l(y, y') = \mathbb{1}_{(y \neq y')}$, c'est à dire $l(y, y') = 1$ si $y \neq y'$ et $l(y, y') = 0$ si $y = y'$. Un problème d'apprentissage, pour lequel cette fonction de perte est utilisée, est appelé un problème de classification. L'ensemble Y considéré en classification est le plus souvent fini, voir même de cardinal deux en classification binaire.

La qualité d'une fonction de prédiction $g : \mathcal{X} \to \mathcal{Y}$ est mesurée par son erreur de généralisation :

$$R_P(g) = \mathbb{E}_P[l(Y, g(X)] \tag{3.1.1}$$

Le risque est donc l'espérance par rapport à la loi P de la perte encourue sur la donnée (X, Y) par la fonction de prédiction g.
La meilleur fonction de prédiction est la fonction de $\mathcal{F}(\mathcal{X}, \mathcal{Y})$ minimisant le risque R_P :

$$g_P^* \in \underset{g \in \mathcal{F}(\mathcal{X}, \mathcal{Y})}{argmin} R_P(g) \tag{3.1.2}$$

Une telle fonction g_P^* n'existe pas nécessairement mais existe pour les fonctions de pertes usuelles, notamment celles que l'on considérera par la suite. Cette "meilleure" fonction sera appelée prédicteur de Bayes. Elle dépend de la probabilité inconnue P et par conséquent sera inconnue.

Consistance d'un algorithme :
On rappelle qu'un algorithme d'apprentissage g est une application

$$g : \bigcup_{n=1}^{\infty} \mathcal{Z}^n \to \mathcal{F}(\mathcal{X}, \mathcal{Y})$$

Par conséquent, $g = \{g_n : n \in \mathbb{N}_*\}$ où chaque $g_n : \mathcal{D}_n \mapsto \hat{g}_n \in \mathcal{F}(\mathcal{X}, \mathcal{Y})$ avec $\mathcal{D}_n = (Z_1, ..., Z_n)$. Comme $\mathcal{D}_n$ est aléatoire, il en est de même pour $\hat{g}_n(x)$ quelle que soit la valeur $x \in \mathcal{X}$. par conséquent, la quantité

$$R_P(\hat{g}_n) = \int_{\mathcal{X}\times\mathcal{Y}} l(y, \hat{g}_n(x)) dP(x, y)$$

est aléatoire. On peut donc considérer l'espérance de cette variable aléatoire par rapport à $\mathcal{D}_n \overset{iid}{\sim} P$, noté $\mathbb{E}_P[R_P(\hat{g}_n)]$

Un algorithme d'apprentissage est dit consistant par rapport à P si et seulement si

$$\mathbb{E}_P[R_P(\hat{g}_n)] \underset{n\to\infty}{\longrightarrow} R_P(g_P^*)$$

Un algorithme d'apprentissage est dit universellement consistant si et seulement s'il est consistant par rapport à toute probabilité P sur $\mathcal{Z}$.

3.1.1.2 Risque empirique et compromis biais-variance

On rappelle que le risque d'une fonction de prédiction $g : \mathcal{X} \to \mathcal{Y}$ est défini par :

$$R_P(g) = \mathbb{E}_P[l(Y, g(X)]$$

Le but d'un algorithme d'apprentissage est de trouver une fonction de prédiction dont le risque est aussi faible que possible (autrement dit aussi proche que possible du risque des prédicteurs de Bayes).

La distribution P générant les données étant inconnue, le risque R_P et les prédicteurs de Bayes sont inconnues. Néanmoins, le risque $R_P(g)$ peut être estimé par son équivalent empirique :

$$\hat{R}_n(g) = \frac{1}{n}\sum_{i=1}^{n} l(Y_i, g(X_i)) \tag{3.1.3}$$

Si l'on suppose $\mathbb{E}_P[l(Y, g(X))^2] < \infty$, alors la loi forte des grands nombres et le théorème centrale limite permettent d'affirmer que :

$$\hat{R}_n(g) \underset{n\to\infty}{\overset{p.s}{\longrightarrow}} R_P(g), \quad \sqrt{n}\{\hat{R}_n(g) - R_P(g)\} \underset{n\to\infty}{\overset{L}{\longrightarrow}} \mathcal{N}(0, Var[l(Y, g(X))])$$

Pour toute fonction de prédiction g, la variable aléatoire $\hat{R}_n(g)$ effectue donc des déviations en $O(1/\sqrt{n})$ autour de sa moyenne $R_p(g)$. Puisque nous cherchons une fonction qui minimise le risque $R_p(g)$ et puisque ce risque est approché par le risque empirique $\hat{R}_n$, il est naturel de considérer l'algorithme d'apprentissage, dit de minimisation du risque empirique, défini par :

$$\hat{g}_{n,\mathcal{G}} = \underset{g\in\mathcal{G}}{argmin}\hat{R}_n(g) \tag{3.1.4}$$

où $\mathcal{G}$ est un sous ensemble de $\mathcal{F}(\mathcal{X}, \mathcal{Y})$. Prendre $\mathcal{G} = \mathcal{F}(\mathcal{X}, \mathcal{Y})$ n'est pas une bonne idée. Tout d'abord, cela entraîne un problème de choix puisqu'en général, pour tout ensemble d'apprentissage, il existe une infinité de fonctions de prédiction minimisant le risque empirique. Par ailleurs et surtout, si on prend l'algorithme du plus proche voisin comme minimiseur du risque empirique (en régression des moindres carrés ou en classification), alors on peut montrer que

cet algorithme est "loin" d'être universellement consistant.
Prendre $\mathcal{G} = \mathcal{F}(\mathcal{X}, \mathcal{Y})$ mène en général à un sur-apprentissage dans la mesure où l'algorithme résultant a un risque empirique qui peut être très inférieur à son risque réel (même lorsque la taille de l'ensemble d'apprentissage tend vers l'infini).
En pratique, il faut prendre $\mathcal{G}$ suffisamment grand pour pouvoir raisonnablement approcher toute fonction tout en ne le prenant pas trop grand pour éviter que l'algorithme sur-apprenne. La grandeur de l'ensemble $\mathcal{G}$ est appelée capacité ou complexité. Un autre point de vue consiste à rajouter à $\hat{R}_n(g)$ une pénalisation, quand par exemple, la fonction g est trop irrégulière. Ces deux approches sont en fait proches l'une de l'autre.
Soit g une fonction minimisant le risque $\mathcal{G}$:

$$g^*_{P,\mathcal{G}} = \underset{g \in \mathcal{G}}{R_P(g)} \tag{3.1.5}$$

On suppose le minimum atteint pour simplifier. La fonction $g^*_{P,\mathcal{G}}$ est appelée prédicteur de Bayes sur $\mathcal{G}$. D'après l'inégalité

$$R_P(\hat{g}_{n,\mathcal{G}}) \geq R_P(g^*_{P,\mathcal{G}}) \geq R_P(g^*_P)$$

l'excès de risque de $\hat{g}_{n,\mathcal{G}}$ se décompose en deux termes positifs, appelés erreur stochastique (ou erreur d'estimation) et erreur systématique (ou biais) :

$$R_P(\hat{g}_{n,\mathcal{G}}) - R_P(g^*_P) = \underbrace{R_P(\hat{g}_{n,\mathcal{G}}) - R_P(g^*_{P,\mathcal{G}})}_{\text{erreur stochastique}} + \underbrace{R_P(g^*_{P,\mathcal{G}}) - R_P(g^*_P)}_{\text{erreur systématique}} \tag{3.1.6}$$

Plus $\mathcal{G}$ est grand, plus l'erreur d'approximation est faible mais plus l'erreur d'estimation est en général grande. Il y a donc un compromis à trouver dans le choix de $\mathcal{G}$. Ce compromis est souvent appelé compromis "biais-variance", où le terme de variance provient du lien entre l'erreur stochastique et la variabilité de l'ensemble d'apprentissage que l'on a supposé étant une réalisation de variables aléatoires iid.

3.1.2 Notion de sur-apprentissage et de sous apprentissage

Les modèles de machine learning sont souvent sujets à des problèmes de sur-apprentissage. Du fait que ces modèles arrivent à différencier à une maille extrêmement fine, le modèle peut apprendre par cœur sur les données mises à sa disposition [7]. Un modèle trop complexe va mal se généraliser et bien qu'il y ait une erreur très faible sur les données d'apprentissage, il y aura des erreurs très fortes sur les données de validation.
Au contraire, si la classe de fonction considérée par l'algorithme d'apprentissage n'est pas assez "riche" pour pouvoir décrire la diversité présente dans les données, on peut se retrouver en situation de sous-apprentissage.
La courbe d'erreur par rapport à l'échantillon d'apprentissage est décroissante, quant à celle de l'échantillon de validation, elle est décroissante puis croissante. Le point à partir duquel la courbe de l'échantillon de validation devient croissante est le point optimal qu'il faudra conserver. Le dessin ci-dessous illustre parfaitement la situation :

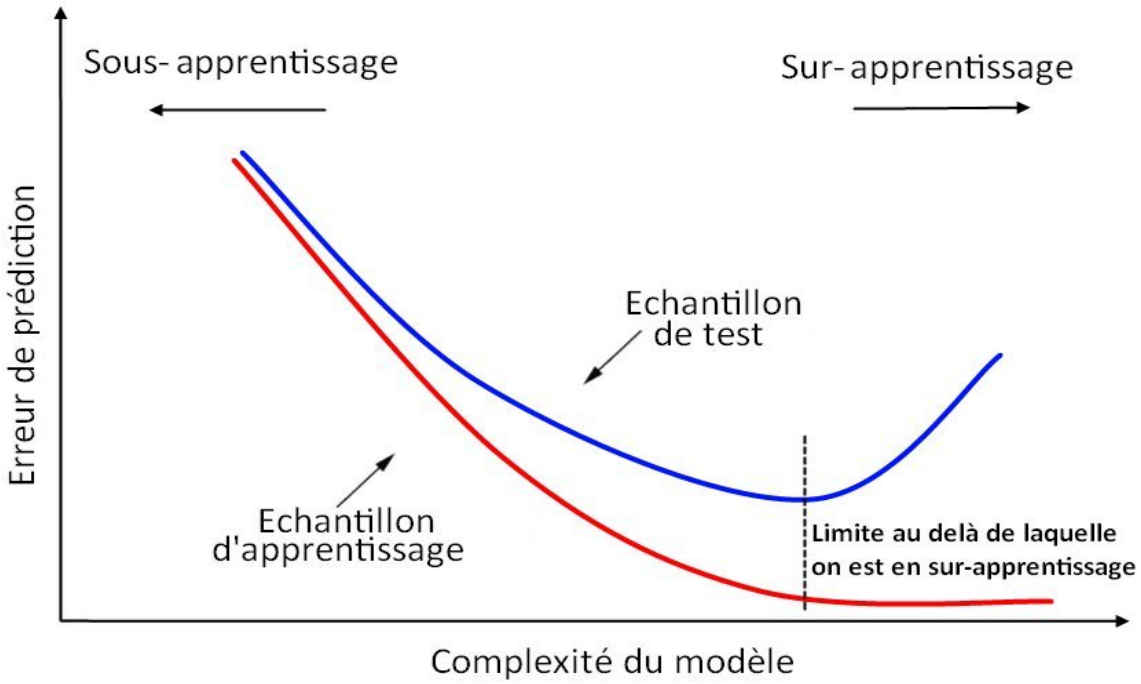


FIGURE 3.1 – Sur-apprentissage et sous-apprentissage (source : [7])

3.1.3 Notion de validation par bloc (k-folds)

Pour éviter ce phénomène de sur-apprentissage, une des méthodes possibles est la validation croisée. Cette méthode sera en particulier utilisée pour sélectionner les hyperparamètres optimaux des modèles de machine learning et donc le meilleur modèle.
Notre base de données ayant été au préalable découpée en base d'apprentissage et base de test, la validation par bloc s'effectuera sur la base d'apprentissage. Cette méthode consiste à découper la base d'apprentissage en k bloc, $k-1$ étant utilisés pour l'apprentissage à chaque étape et un étant utilisé pour la validation. Une fois l'apprentissage terminé, on effectue la prédiction grâce à l'échantillon de validation et l'erreur associé à cette prédiction est calculée (erreur calculée grâce aux métriques choisies). Nous itérons ceci k fois, et nous obtenons ainsi k erreurs. L'erreur finale est la moyenne des erreurs :

$$Erreur = \frac{1}{k}\sum_{i=1}^{k} Erreur_i$$

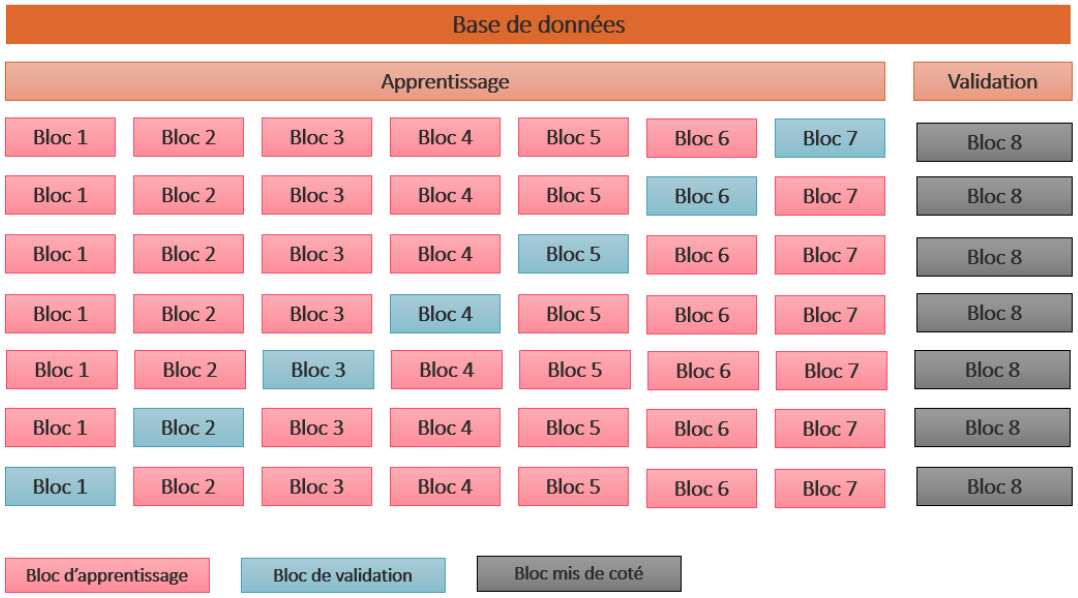


FIGURE 3.2 – Méthode de validation par bloc usuelle

Dans le cadre de ce mémoire, pour conserver l'aspect temporel propre aux données, les blocs ont été découpés en fonction des années présentes dans le portefeuille. Le processus peut donc se résumer par le schéma ci-dessous :

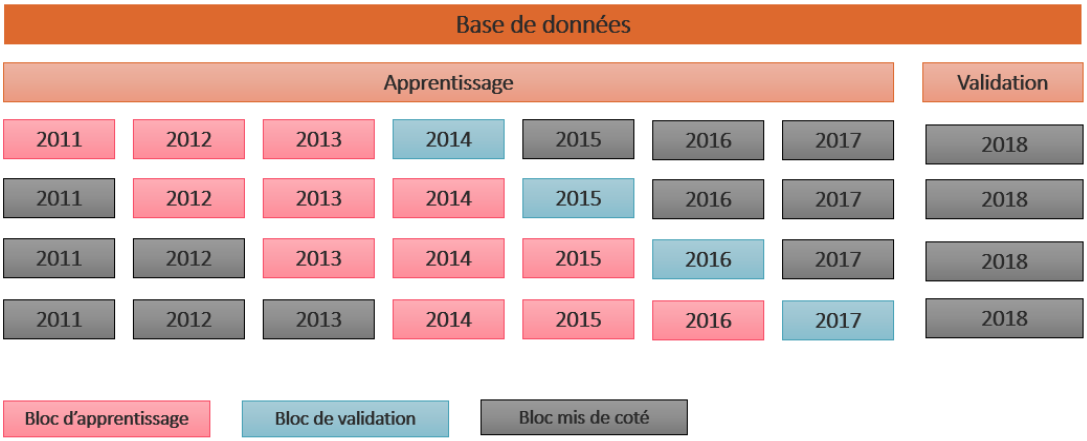


FIGURE 3.3 – Méthode de validation par bloc utilisée

3.1.4 Random forest

3.1.4.1 Définition générale des forêts aléatoires

Les forêts aléatoires, introduites par Breiman (2001), sont un algorithme d'apprentissage statistique très performant à la fois pour les problèmes de classification et de régression.

Définition : Forêts aléatoires (source [8])
Notations :
$L_n = \{(X_1, Y_1), ..., (X_n, Y_n)\}$ va iid de même loi que (X, Y).
$X \in \mathbb{R}^p (variables)$
$Y \in S$, $S = \mathbb{R}$ si régression et $S = \{1, ..., L\}$ si classification
Objectif : construire $\hat{h} : \mathbb{R}^p \to S$
$\{\hat{h}(., \Theta_l, 1 \leq l \leq q\}$ collection de prédicteur par arbre, $(\Theta_l)_{1 \leq l \leq q}$ va iid indépendantes de L_n.
Prédicteur de forêts aléatoires $\hat{h}$ obtenu en agrégeant la collection d'arbres.

$$\hat{h}(x) = \frac{1}{q} \sum_{l=1}^{q} \hat{h}(x, \Theta_l) \text{ en régression}$$

$$\hat{h}(x) = \underset{1 \leq c \leq L}{argmax} \sum_{l=1}^{q} \mathbb{1}_{\hat{h}(x, \Theta_l) = c} \text{ en classification}$$

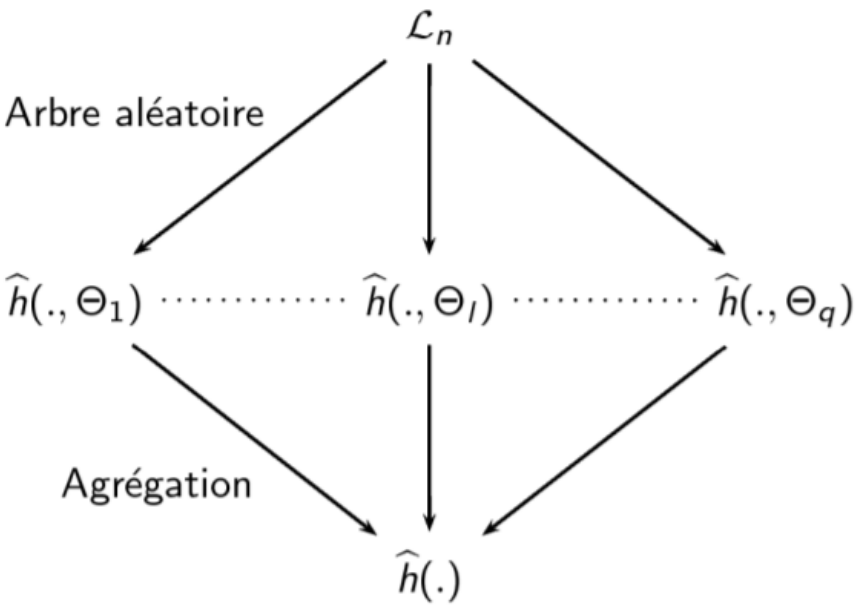


FIGURE 3.4 – Schéma général des forêts aléatoires (source [8])

3.1.4.2 Bagging

La méthode du Bagging a été introduite par Breiman en 1996. Le mot Bagging est la contraction des mots Boostrap et Aggregating. Étant donné un échantillon d'apprentissage L_n et une méthode de prédiction (appelé règle de base), qui construit sur L_n un prédicteur $\hat{h}(., L_n)$. Le principe du bagging est de tirer indépendamment plusieurs échantillon bootstrap $(L_n^{\Theta_1}, ..., L_n^{\Theta_q})$, d'appliquer la règle de base sur chacun d'eux pour obtenir une collection de prédicteur $(\hat{h}(., L_n^{\Theta_1}), ..., \hat{h}(., L_n^{\Theta_q}))$, et enfin d'agréger ces prédicteurs de base.
L'idée du Bagging, est qu'en appliquant la règle de base sur différents échantillons bootstrap, on en modifie les prédictions, et donc on construit ainsi une collection variée de prédicteurs. L'étape d'agrégation permet alors d'obtenir un prédicteur performant.

Un échantillon boostrap $L_n^{\Theta_l}$ est, par exemple, obtenu en tirant aléatoirement n observations avec remise dans l'échantillon d'apprentissage L_n, chaque observation ayant une probabilité $1/n$ d'être tirée. La variable Θ_l répresente alors ce tirage aléatoire. Une deuxième méthode classique d'obtenir un échantillon boostrap est de tirer aléatoirement k observations sans remise dans L_n, avec $k < n$. La figure 3.5 représente le principe de cette méthode.

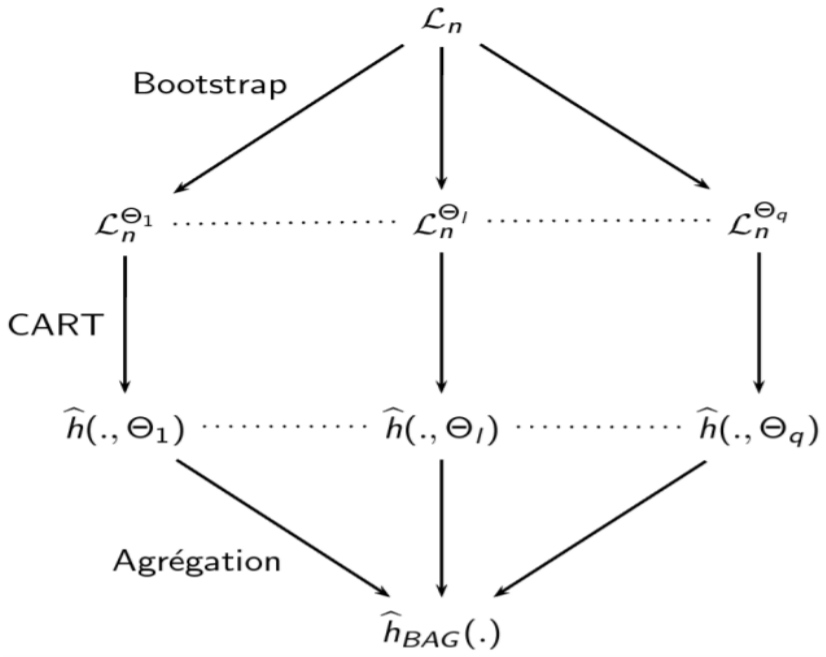


FIGURE 3.5 – Schéma du Bagging avec pour règle de base un arbre CART (source [8])

3.1.4.3 Erreur out of bag (OOB)

Définition : Forêts aléatoires (source [8])
Pour prédire X_i, on agrège uniquement les prédicteur $\hat{h}(.,\Theta_l)$ construits sur des échantillons boostrap ne contenant pas (X_i, Y_i).
$\Rightarrow$ Erreur OOB :

$$\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \text{ en régression}$$

$$\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}_{Y_i \neq \hat{Y}_i} \text{ en classification}$$

Cette estimation obéit aux mêmes contraintes que celles des estimateurs classiques de l'erreur de généralisation (par échantillon test ou par validation croisée), au sens où les données prédites sont les données qui n'ont pas été rencontrées au préalable par le prédicteur utilisé. Un avantage de l'erreur OOB par rapport aux estimateurs classique est qu'elle ne nécessite pas de découpage de l'échantillon d'apprentissage. Ce découpage est en quelque sorte inclus dans la génération des différents échantillons bootstrap. Cependant, il faut bien noter que pour chaque observation ce n'est pas le même ensemble d'arbres qui est agrégé. En conséquence, cette erreur estime l'erreur de généralisation d'une forêt, mais elle n'utilise jamais les prédictions de la forêt elle-même, mais

plutôt celles de prédicteurs qui sont des agrégations d'arbres de cette forêt. Par la suite, nous adopterons l'erreur OOB pour regarder l'importance des variables car ces deux notions sont très liées.

3.1.4.4 Importance des variables

Lorsqu'il ne s'agit pas seulement d'éliminer des variables du modèle ou de retenir seulement des variables explicatives suffisantes pour obtenir une bonne prédiction, on peut être intéressé par construire une hiérarchie de toutes les variables fondée sur l'importance vis-à-vis de la variable réponse. Un tel indice d'importance fournit donc un classement des variables, de la plus importante à la moins importante.
Dans ce cadre, une des mesures largement utilisée pour mesurer l'importance d'une variable donnée est l'accroissement moyen de l'erreur d'un arbre dans la forêt lorsque les valeurs observées de cette variable sont permutées au hasard dans les échantillons OOB.

Fixons $j \in \{1, ..., p\}$ et détaillons le calcul de l'indice d'importance de la variable X^j. Considérons un échantillon boostrap $L_n^{\Theta_l}$ et l'échantillon OOB_l associé, ie l'ensemble des observations qui n'apparaissent pas dans $L_n^{\Theta_l}$. On calcule $errOOB_l$, l'erreur commise sur OOB_l par l'arbre construit sur $L_n^{\Theta_l}$. On permute ensuite aléatoirement les valeurs de la j-ième variable dans l'échantillon OOB_l. Ceci donne un échantillon perturbé, noté $\widetilde{OOB_l}^j$. On calcule enfin $err\widetilde{OOB_l}^j$, l'erreur sur l'échantillon perturbé. On effectue ceci pour tous les échantillons bootstrap.
L'importance de la variable X^j, noté $VI(X^j)$, est définie par la différence entre l'erreur moyenne d'un arbre sur l'échantillon OOB perturbé et celle sur l'échantillon OOB :

$$VI(X^j) = \frac{1}{q}\sum_{l=1}^{q}(err\widetilde{OOB_l}^j - errOOB_l)$$

Ainsi, plus les permutations aléatoires de la j-ième variable engendrent une forte augmentation de l'erreur, plus la variable est importante. A l'inverse, si les permutations n'ont quasiment aucun effet sur l'erreur (voire même la diminuent ce qui fait que VI peut être légèrement négative), la variable est considérée comme très peu importante.

3.1.5 Gradient boosting

Cet algorithme se décompose en deux principes, le Boosting et le l'algorithme de descente de gradient. L'idée du Boosting est d'améliorer la performance d'un prédicteur dit faible, c'est à dire un modèle peu coûteux dont la prédiction est à peine supérieure à la moyenne. En itérant, l'information fournie par chaque prédicteur est agrégée au prédicteur final. Le principe est qu'en regroupant de nombreuses règles de décisions peu efficaces, on arrive à obtenir une règle finale très efficace.

3.1.5.1 Avantages et limites

Cette méthode de machine learning a un fondement mathématique poussé, c'est la raison pour laquelle nous allons l'étudier. Cet algorithme a une capacité à prendre en compte l'interaction entre les différentes variables explicatives mises à sa disposition. En effet, parmi certaines variables qui seront fournies en entrée de l'algorithme, il peut exister des liens plus ou moins importants. D'une manière plus générale, lorsque l'on traite une base de données avec un grand nombre de variables, il est intéressant de comprendre comment les variables interagissent entre elles. Ces correspondances ont un impact direct sur le choix du modèle et la qualité de la modélisation. Dans le cas des GLM, leur modélisation reste possible tant qu'un couple de variables n'est pas lié à 100 %. Par ailleurs, en général, pour avoir un modèle de qualité, ces liens sont traités en amont. En utilisant une approche de Gradient Boosting, il n'est plus nécessaire de traiter au-préalablement les variables potentiellement liées.
Toutefois, l'algorithme Gradient Boosting étant considéré comme un modèle boite noire, il est difficile d'interpréter le résultat obtenu par cet algorithme. Autrement dit, on ne sait pas ce qui a conduit à ce résultat plutôt qu'un autre résultat.

3.1.5.2 Algorithme de descente de gradient

Cet algorithme est une technique itérative qui permet d'approche la solution d'un problème d'optimisation convexe.

Soit f une fonction deux fois différentiable, d'espace de départ $\mathbb{R}^p$ (muni d'une norme associée $\|.\|$ et du produit scalaire $< . , . >$), et d'espace d'arrivée R qu'on souhaite minimiser. On note $f'(x)$ et $\nabla f(x)$ respectivement la dérivée et le gradient de f en x.
Algorithme :
Initialisation : Soit x_0 un point initial $\in \mathbb{R}$, et un seuil de tolérance $\epsilon \leq 0$. L'algorithme va alors calculer de manière itérative des valeurs x_1, x_2, ... jusqu'à la convergence vers l'optimum.

- Simulation : calcul de $\nabla f(x_k)$
- Test d'arrêt : $\|\nabla f(x_k)\| \leq \epsilon$
- Calcul du pas : $\alpha_k > 0$ par une règle de recherche sur f en x le long de la direction $-\nabla f(x_k)$
- Nouvel itéré : $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$

Appliquons à présent cet algorithme d'optimisation à la fonction perte du modèle considéré. Notons l notre fonction de perte utilisé pour comparer la valeur prédite $f(x_i) = \hat{y}_i$ avec la valeur réelle y_i. On peut par exemple prendre pour fonction de perte la norme 2 : $l(y_i, f(x_i) = (y_i - f(x_i))^2$. Soit L la fonction de perte globale, définie comme la somme des fonctions de coûts individuelles : $L(y, f) = \sum_{i=1}^{n} l(y_i, f(x_i)$.
L'objectif sera alors de minimiser cette fonction de perte globale par rapport aux paramètres de la fonction f associé au modèle à l'aide de la descente de gradient vu précédemment. Ainsi si l'on note f_m le classifieur à l'étape m, on obtient la formule suivante :

$$f_m(x) = f_{m-1}(x) - \alpha \nabla l(y, f(x))$$

3.1.5.3 Principe de l'algorithme de Gradient Boosting

L'algorithme de gradient boosting généralise la méthode précédente avec d'autre fonction de perte. Dans le cas de la régression, l'algorithme se présente de la manière suivante [6] :

- initialisation par un modèle de base : $f_0(x) = \underset{\theta}{argmin} \sum_{i=1}^{n} L(y_i, \theta)$ (correspond à la moyenne des observations de y pour une fonction de perte correspondant à la norme euclidienne.
- pour $m \in \{1, ..., M\}$:
  1. calcul des pseudo résidus : $r_{i,m} = -\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}|_{f(x)=f_{m-1}(x)}, \forall i \in \{1, ..., n\}$
  2. ajustement d'un classifieur $h_m(x)$ (faible, un arbre par exemple) entraîné sur les données $(x_i, r_{i,m})_{1 \leq i \leq n}$, où $(x_i)_{1 \leq i \leq n}$ et $(r_{i,m})_{1 \leq i \leq n}$ correspondent respectivement aux variables explicatives et aux sorties.
  3. calcul du prochain classifieur : $f_m(x) = f_{m-1}(x) + \theta_m h_m(x)$, avec θ_m résultat du problème d'optimisation suivant : $\theta_m = \underset{\theta}{argmin} \sum_{i=1}^{n} L(y_i, f_{m-1}(x_i) + \theta h_m(x_i))$

A la sortie de l'algorithme, on obtient : $f_M(x)$ qui est notre classifieur final.

Dans cet algorithme il s'agit plutôt que de chercher le meilleur classifieur dans d'autres méthodes, de chercher le meilleur pas de descente θ. Les fonctions de pertes généralement utilisées pour cet algorithme sont :

- erreur quadratique : $l(y_i, f(x_i)) = \frac{1}{2}(y_i - f(x_i))^2$. Elle présente l'avantage d'être sensible aux écarts, mais l'inconvénient d'être peu robuste aux valeurs aberrantes.
- erreur absolue : $l(y_i, f(x_i)) = |y_i - f(x_i)|$. Elle présente l'avantage d'être robuste aux valeurs aberrantes, mais moins sensible aux écarts.

On appelle Tree Gradient Boosting les cas particuliers où les modèles h_m sont des arbres de décision.

3.1.6 Méthode d'oversampling (sur-échantillonnage)

Un ensemble de données est considéré comme déséquilibré lorsque qu'il y a un biais sévère dans la distribution des classes, comme par exemple 1 pour 100 ou 1 pour 1000 de la classe minoritaire à la classe majoritaire. Dans notre cas, ce biais s'observera sur la base de fréquence, étant donné que la base est composée majoritairement d'individus sans sinistre. Ce biais dans l'ensemble de données d'apprentissage influence de nombreux algorithmes de machine learning, conduisant généralement à ignorer la classe minoritaire. Ceci est problématique car c'est généralement la classe minoritaire sur laquelle les prédictions sont les plus importantes.
Plusieurs méthodes et algorithmes existent, notamment, les algorithmes ADASYN et SMOTE, la règle du plus proche voisin, le sur-échantillonnage aléatoire ou la variation des poids.

- SMOTE : l'idée est de créer de nouveaux exemples de données minoritaires en interpolant les exemples déjà existants.
- ADASYN : s'appuie sur la méthodologie de SMOTE, mais se focalise sur les données de la classe minoritaire qui sont difficile à apprendre. Plus une donnée est difficile à

apprendre, plus des données synthétiques proches de celle-ci seront générées.

- Changement de poids : dans de nombreux algorithmes d'apprentissage de scikit learn, il est possible de mettre un poids sur les données. Cette méthode consistera donc à augmenter artificiellement le poids des données de la classe minoritaire.

3.1.7 Principe de sélection des hyperparamètres des modèles

Pour être efficace, un modèle de machine learning doit être calibré avec des paramètres adéquats. Pour notre étude, la base d'apprentissage a été découpée par année pour constituer les groupes de la validation croisée. Ceci ayant pour objectif de capter des effets propres aux années. Le choix du nombre d'années d'entraînement a été fixé à trois ou à quatre. Par la suite, la méthode donnant les meilleurs scores sera conservée.

Numéro de groupe	Base d'apprentissage	Base de validation
1	2011-2012-2013	2014
2	2012-2013-2014	2015
3	2013-2014-2015	2016
4	2014-2015-2016	2017

TABLE 3.1 – Groupes de validation croisée : 3 années d'historique

Numéro de groupe	Base d'apprentissage	Base de validation
1	2011-2012-2013-2014	2015
2	2012-2013-2014-2015	2016
3	2013-2014-2015-2016	2017

TABLE 3.2 – Groupes de validation croisée : 4 années d'historique

Pour sélectionner les hyperparamètres optimaux du modèle, il faut donc réaliser une validation par bloc pour chaque combinaison possible des paramètres. Ainsi, pour chaque combinaison de paramètres, une erreur sera obtenue, et la combinaison d'hyperparamètres optimaux correspondra à celle minimisant l'erreur. Dans le cas des forêts aléatoires et du gradient boosting, l'ensemble des combinaisons à tester est trop élevé en général si l'on souhaite optimiser plusieurs paramètres en même temps. Il s'agira donc de tester x combinaisons (x fixé à 1000) aléatoire parmi l'ensemble des combinaisons possibles.

3.2 Pré-traitement des données avant la modélisation

Pour réduire les temps de calculs, les bases de coûts et de nombre de sinistres ont été agrégées, c'est-à-dire que les individus ont été regroupés par classe. Chaque correspondant à un n-uplet unique de caractéristiques. Ceci est très facile dans le cas de notre portefeuille étant donné que les variables de celui-ci sont seulement composées de variables catégorielles. Pour la base des coûts lors de l'agrégation, le montant des sinistres a été sommé à l'intérieur de chaque classe et il a été retenu dans le même temps le nombre de sinistres pour chaque classe, dans le but de réaliser le modèle de coût sur la variable de coûts moyens. Pour la base du nombre de sinistres, lors de l'agrégation le nombre de sinistres ainsi que les expositions ont été sommés par classe, dans le but de réaliser le modèle de fréquence sur la variable de fréquence de sinistre (annuelle).

3.2.1 Statistiques sur les données de test et d'apprentissage

Il est utile d'observer l'allure des données présentes dans les échantillons d'apprentissages et de test. Étant donné qu'une validation croisée par bloc par année sera faite, il est important d'observer les statistiques sur les données par bloc de validation croisée. Rappelons que les données d'apprentissages sont les années de 2011 à 2017, et que les données de validation sont celle de l'année 2018.

Échantillon	Moyenne des coûts	Fréquence moyenne
Train	1277.58	0.0533
Test	1195.01	0.0649

TABLE 3.3 – Moyennes sur les données d'apprentissages et de validations

On observe que les coûts moyens des sinistres sont légèrement plus élevés sur la base d'apprentissage que sur la base de validation. Par ailleurs, la fréquence est en moyenne plus faible sur la base d'apprentissage que sur la base de test.
On peut de la même manière observer les moyennes sur chaque bloc de la validation croisée pour le scénario avec 3 années d'historique et 4 années d'historique.

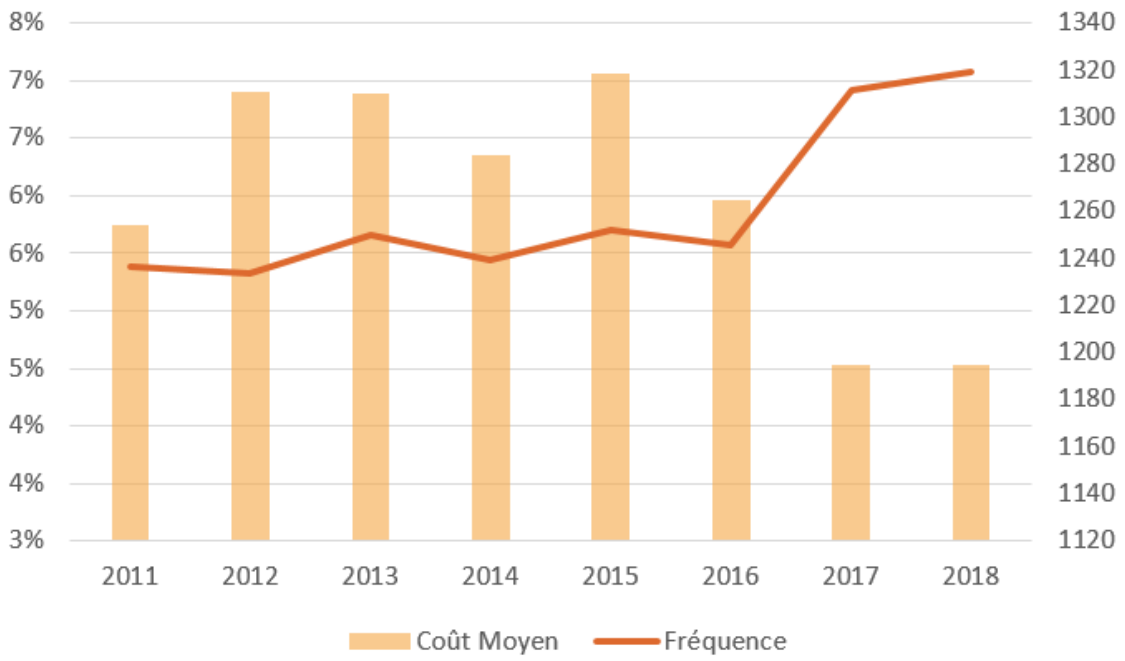


FIGURE 3.6 – Évolution des coûts moyens et des fréquences par année

Blocs train	Coûts	Fréquence	Blocs test	Coûts	Fréquence
2011, 2012, 2013	1291.56	0.0510	2014	1284.27	0.0516
2012, 2013, 2014	1301.88	0.0515	2015	1318.73	0.0524
2013, 2014, 2015	1304.30	0.0520	2016	1264.59	0.0531
2014, 2015, 2016	1288.97	0.0523	2017	1194.79	0.0643

TABLE 3.4 – Moyennes sur les blocs : 3 années d'historique

Blocs train	Coûts	Fréquence	Blocs test	Coûts	Fréquence
2011, 2012, 2013, 2014	1289.77	0.0512	2015	1318.73	0.0524
2012, 2013, 2014, 2015	1305.96	0.0517	2016	1264.59	0.0531
2013, 2014, 2015, 2016	1294.41	0.0523	2017	1194.79	0.0643

TABLE 3.5 – Moyennes sur les blocs : 4 années d'historique

Pour les deux scénarios possibles, les moyennes de coûts et de fréquences semblent stables par bloc, exceptés pour le dernier bloc de test (année 2017), où la fréquence semble plus forte et où les coûts semblent moins élevés en moyenne.

3.2.2 Préprocessing mis en place

Étant donné que le portefeuille mis à disposition n'est constitué que de données catégorielles, il est nécessaire de traiter les données. En effet les modèles utilisés sous scikit-learn ne peuvent pas

prendre en entrée des données comportant des *string*. Certaines données étaient initialement des données numériques comme la VALEUR_ASSUREE, mais celles-ci avaient déjà été segmentées en amont avant la prise en main du portefeuille. Concernant l'encoding deux méthodes différentes ont été utilisées, certaines variables catégorielles comportant plus d'informations que d'autres. Par exemple la variable VALEUR_ASSUREE conserve une notion d'ordre, il a donc été décidé de la convertir en nombres entiers ordinaux correspondant à un segment de valeur assurée croissant. D'autres variables comme la NATURE_HABITATION ne comporte pas d'ordre, ces variables ont donc été *One-hot encodé*, ie cela crée une colonne binaire pour chaque catégorie et renvoie une matrice creuse. Les variables de présence de dépendance et de piscine n'ont pas été retenues dans le modèle pour éviter un sur-apprentissage, celles-ci ayant une classe nettement surreprésentée (cela aurait empêché le modèle de "bien apprendre" des données).

Traitement de la variable ZONIER : Bien qu'une variable ZONIER simplifié (ZONIER_SIMP) soit présente dans la base initiale, celle-ci n'a finalement pas été retenue. Elle était trop simplifiée et faisait perdre trop d'informations. La variable ZONIER a donc été simplifiée d'une autre manière pour la prendre en compte dans le modèle. Initialement la variable ZONIER étant composée de 83 classes, celle-ci est très complexe et est composée de plein de modalités sous-représentées (<1 %). Pour la transformation de la variable ZONIER, il s'agit donc de considérer que lorsque qu'une modalité est présente dans moins de 1 % des cas alors on lui affectera la modalité majoritaire comme s'il s'agissait d'une modalité de référence. Grâce à ce processus, cette variable passe d'une variable catégorielle inexploitable composée de 83 modalités à une variable plus explicite composées de 30 modalités environ.

Ci-dessous un récapitulatif des encodages choisis pour chacune des variables :

- Ordinal encodée : VALEUR_ASSUREE, OBJETS_DE_VALEUR, NB_PIECES
- One-hot encodée : FORMULE, TYPE_HABITATION, NATURE_HABITATION, TYPE_OCCUPANT, TYPE_OCCUPANT_COMP, Hist_Dom, Hist_RC, Hist_Aut

Les variables indiquant la présence de piscine et de dépendance (respectivement STANDING et DEPENDANCE) n'ont pas été utilisées lors de la calibration des modèles. Cette décision a été prise car ces deux variables comportent seulement deux modalités et dont une majoritairement présente et pour éviter de perturber trop les modèles avec un potentiel sur-apprentissage.

Ajustement des coûts :
Pour les coûts, sur les données du graphique 3.6, on a observé une décroissance des coûts sur les derniers exercices à disposition dans le portefeuille. Pour pallier cette décroissance, lors de l'apprentissage des modèles, les données cibles à apprendre ont été ajustées d'un coefficient. Ce coefficient peut être calculé de deux manières différentes. Soit être calculé pour chaque bloc de validation, notamment lors de la validation par bloc, ou bien sur l'ensemble des données d'apprentissage donc de 2011 à 2017. Pour ces deux méthodes, ce coefficient est calculé en divisant la charge totale des années 1 à k par la charge totale des années 0 à $k-1$, où les années de l'échantillon d'apprentissage vont de 0 à k. En particulier, pour la deuxième méthode, sur les

années 2011 à 2017, ce coefficient est finalement de 0.976.
Exemple : ce coefficient est appliqué de manière différente suivant l'année de l'instance pris en compte. Pour l'année y de l'échantillon d'apprentissage, ses coûts moyens sont multipliés par 0.976^{x-y} *où x est l'année de l'échantillon de validation.*
Dans la suite de ce mémoire, la méthode retenue est celle où les coefficients sont ajustés pour chaque bloc, celle-ci ayant donné de meilleurs résultats.

3.3 Définition des métriques utilisées

Les métriques suivantes seront utilisées pour la sélection et la comparaison des modèles. Grâce à ces métriques, il est aussi possible de déterminer les hyperparamètres optimaux lors d'une validation croisée.
Nous définissons ici les métriques utilisées dans le rapport.
Les métriques sans notion de poids :

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}|y_i - \hat{y}_i|$$

$$\text{True/Pred} = \frac{\frac{1}{n}\sum_{t=1}^{n} y_i}{\frac{1}{n}\sum_{t=1}^{n} \hat{y}_i}$$

Les métriques avec notion de poids :

$$\text{PRMSE} = \sqrt{\frac{1}{\sum_{i=1}^{n} w_i}\sum_{i=1}^{n} w_i(y_i - \hat{y}_i)^2}$$

$$\text{PMAE} = \frac{1}{\sum_{i=1}^{n} w_i}\sum_{t=1}^{n} w_i|y_i - \hat{y}_i|$$

$$\text{True/Pred} = \frac{\frac{\sum_{t=1}^{n} w_i y_i}{\sum_{i=1}^{n} w_i}}{\frac{\sum_{t=1}^{n} w_i \hat{y}_i}{\sum_{i=1}^{n} w_i}}$$

Métrique Gini pour la prédiction de la sévérité :
Une autre métrique a été utilisée, il s'agit de l'indice de Gini. En général connu pour la classification, il peut servir aussi pour les variables continues lorsque l'on ne souhaite pas prédire parfaitement, mais seulement prédire dans un bon intervalle. Pour cela, on peut vérifier l'adéquation de la prédiction à une classe, cette classe correspondant à un quantile de l'échantillon de validation. Le nombre de découpes de la variable réelle sera fixé comme paramètre de cette métrique et ajusté arbitrairement.

Algorithme :

- On commence par diviser notre variable cible de validation (y_{val}) et notre variable de prédiction (y_{pred} en k segments croissant $([q_0, q_{1/k}], ..., [q_{(k-1)/k}, q_1])$, où $q_{1/k}$ correspond au quantile $1/k$ de la variable cible de l'échantillon de validation.
- variables : y_{val}, y_{pred}, $poids$, k
- $y_{val,cut}(i) = j$ si $y_{val}(i) \in [q_j, q_{j+1/k}], \forall i \in [1, n]$
- $y_{pred,cut}(i) = j$ si $y_{pred}(i) \in [q_j, q_{j+1/k}], \forall i \in [1, n]$
- sortie = 0
- Pour dec de 1 à k :
  1. $y_{val,dec}(i) = 1$ si $y_{val,cut}(i) = dec$, 0 sinon, $\forall i \in [1, n]$
  2. $y_{pred,dec}(i) = 1$ si $y_{pred,cut}(i) = dec$, 0 sinon, $\forall i \in [1, n]$
  3. $poids_{dec} = \sum_{i=1}^{n} w_i \mathbb{1}_{\{y_{val,dec}=dec\}}$
  4. $sortie = sortie + poids \times Gini(y_{val,dec}, y_{pred,dec}, poids)$
- $sortie = \frac{sortie}{\sum_{i=1}^{n} poids_i}$

sortie correspond à notre indice de Gini pour des données continues.

Métrique Gini pour la prédiction de la fréquence :

Dans le cadre de la prédiction de la fréquence, il s'agit de mesurer si les individus ont bien été classés du plus risqué au moins risqué. Il s'agit donc avec cette métrique de refléter la capacité d'un modèle à classer les prédictions indépendamment de leurs valeurs absolues et donc d'évaluer uniquement leur pouvoir de classement. Pour comparer les modèles de ce point de vue, il s'agira de calculer l'indice de Gini de la proportion cumulée de sinistres vs la proportion cumulée d'exposition pour les échantillons classés selon les prédictions du modèle, du plus sûr au plus risqué selon chaque modèle.

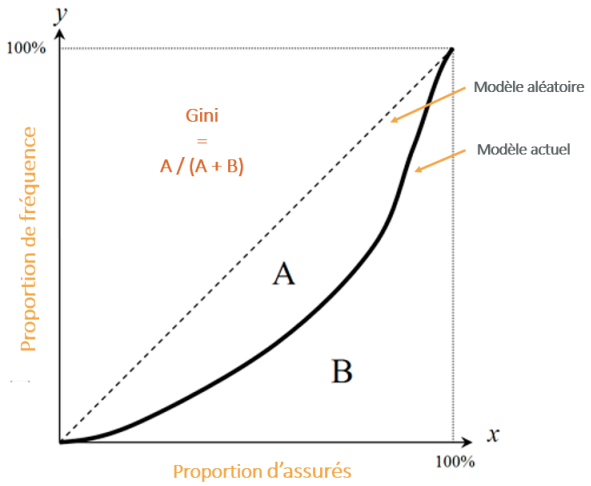


FIGURE 3.7 – Illustration du coefficient de gini pour la fréquence

Description étape par étape de la méthode :

- Trier les fréquences prédites dans l'ordre croissant.
- Ordonner les fréquences réelles et les expositions suivant l'ordre trouvé précédemment.
- Calculer les vecteurs cumulés des fréquences réelles et des expositions ordonnés à l'étape précédente.
- Calculer le gini grâce à l'area under curve (AUC) des deux vecteurs créés à l'étape précédente.

3.4 Calibration des modèles

Rappelons que les données du portefeuille sont issues de huit années, 2011 à 2018. L'objectif est de prédire la sinistralité de l'année 2018 grâce aux années 2011 à 2017. Les années 2011 à 2017 correspondent à l'échantillon dit de développement sur lequel seront calibrés par le biais d'une validation croisée les hyperparamètres des modèles souhaités. L'objectif de cette validation croisée sera d'éviter le sur-apprentissage tout en optimisant le modèle. En s'appuyant sur la méthode ci-dessous (3.8), pour la validation croisée, il s'agit de répéter les étapes 1 et 2 respectivement d'apprentissage et de test sur les blocs qui nous intéressent et qui ont été définis précédemment (3.1 et 3.2).

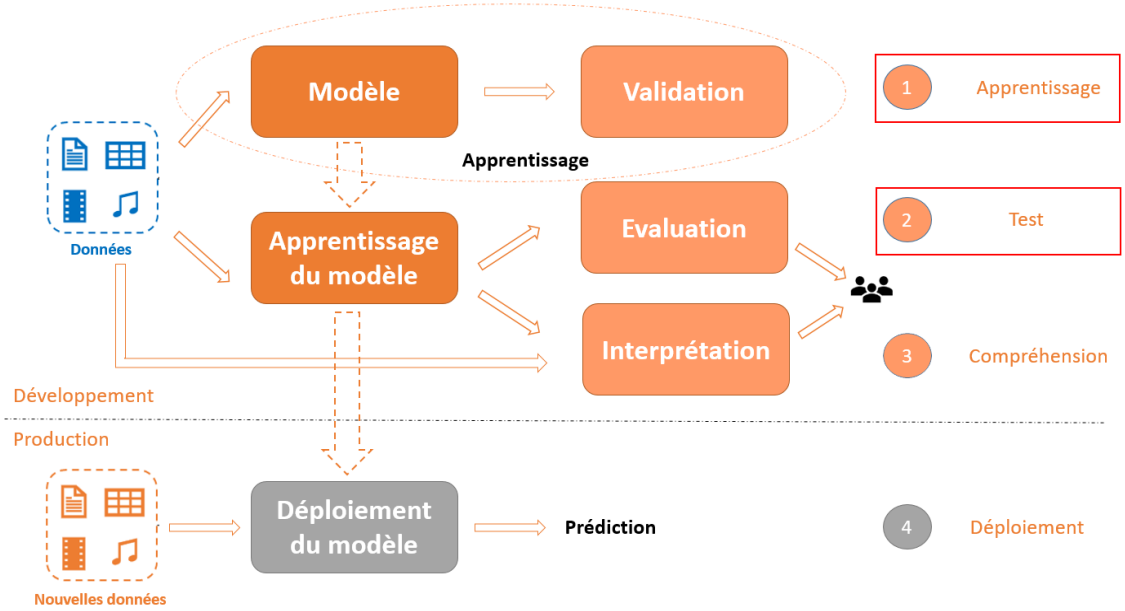


FIGURE 3.8 – Méthodologie machine learning

3.4.1 Paramètres du gradient boosting

Avant d'optimiser les paramètres du Gradient Boosting de scikit-learn, il convient de comprendre l'utilité de chacun de ces paramètres. Commençons par expliciter les paramètres spécifiques à l'arbre, ils affecteront chaque arbre individuel du modèle.

1. min_samples_split
   - Définit le nombre minimum d'échantillons (ou d'observations) qui sont requis dans un nœud pour être pris en compte pour le fractionnement.
   - Utile pour contrôler le sous-ajustement et le sur-ajustement. Des valeurs trop élevées empêchent le modèle d'apprendre des relations qui pourraient être très spécifiques à l'échantillon particulier sélectionné pour un arbre. Il faut donc l'ajuster à l'aide de validation croisée.
2. min_samples_leaf
   - Définit les échantillons (ou observations) minimum requis dans un nœud terminal ou une feuille.

- Utilisé pour contrôler le sur-ajustement similaire à min_samples_split.

3. max_depth
   - La profondeur maximale d'un arbre.
   - Utilisé pour contrôler le sur-ajustement car une profondeur trop élevée permettra au modèle d'apprendre des relations très spécifiques à un échantillon particulier.
   - Doit être réglé en utilisant une validation croisée.
4. max_features
   - Le nombre de variables à prendre en compte lors de la recherche de la meilleure répartition. Ceux-ci seront sélectionnés au hasard.
   - Des valeurs trop élevées peuvent entraîner un sur-ajustement, mais cela dépend du cas par cas.
   - Dans notre cas, des valeurs jusqu'à 80 % du nombre total de variables seront testées.

Ensuite viennent les paramètres de boosting qui affectent l'opération de boost du modèle.

1. learning_rate
   - Cela détermine l'impact de chaque arbre sur le résultat final. Le Gradient Boosting fonctionne en commençant par une estimation initiale qui est mise à jour en utilisant la sortie de chaque arbre. Le paramètre d'apprentissage contrôle l'ampleur de ce changement dans les estimations.
   - Des valeurs inférieures sont généralement préférées car elles rendent le modèle robuste aux caractéristiques spécifiques de l'arbre et lui permettent ainsi de bien se généraliser. Par ailleurs, elles exigeraient un plus grand nombre d'arbres pour modéliser toutes les relations et coûteraient cher en calcul.
2. n_estimators
   - Le nombre d'arbres séquentiels à modéliser.
   - Bien que le Gradient Boosting soit assez robuste à un grand nombre d'arbres, il peut toujours surajuster à un moment donné. Par conséquent le nombre d'arbres devra être réglé en utilisant une validation croisée.
3. subsample
   - La fraction des observations à sélectionner pour chaque arbre. La sélection se fait par échantillonnage aléatoire.
   - Des valeurs légèrement inférieures à 1 rendent le modèle robuste en réduisant la variance.
   - Les valeurs typiques d'environ 0.8 fonctionnent généralement bien mais peuvent être affinées par validation croisée.

En dehors des paramètres mentionnés ci-dessus, il en existe d'autres qui affectent la fonctionnalité globale :

1. loss : fait référence à la fonction de perte à minimiser dans chaque fractionnement, la rmse et mae seront testées.
2. warm_start : permet d'ajuster des arbres supplémentaires sur les ajustements précédents d'un modèle.

3.4.2 Paramètres des forêts aléatoires

De la même manière que pour le gradient boosting, l'optimisation des hyperparamètres des forêts aléatoires est primordiale pour avoir un bon modèle. Comme pour le gradient boosting, certains paramètres sont propres aux arbres et affectent chaque arbre individuel du modèle. Beaucoup de paramètres sont en commun avec le gradient boosting, notamment min_samples_split, min_samples_leaf, max_depth, max_features, n_estimators et warm_start.
Le paramètre bootstrap auquel il faut s'intéresser d'un peu plus près indique si des échantillons bootstrap sont utilisés lors de la création d'arbres. Si ce paramètre est indiqué à False, l'ensemble des données est utilisé pour construire chaque arbre. 3.1.4.2

3.4.3 Hyperparamètres sélectionnés

Aussi bien pour les modèles de forêts aléatoires que de Gradient Boosting, les scénarios avec 3 années d'historique et avec 4 années d'historique ont été comparés. Pour ces deux méthodes, le scénario donnant les meilleurs résultats a été retenu.
La sélection des meilleurs hyperparamètres se fait en minimisant la métrique de Gini tout en contrôlant la variance de la RMSE par bloc de validation croisée ainsi que le ratio $True/Pred$. Il ne s'agit donc pas simplement de regarder quelle combinaison d'hyperparamètres donne le meilleur Gini, mais bien de regarder d'autres métriques dans le même temps. Il s'agit en quelque sorte d'une optimisation du Gini sous contrainte d'un résultat convenable en termes de $True/Pred$ et de variance de RMSE. Il est utile de minimiser la variance de la RMSE lors de la validation croisée pour prendre des hyperparamètres qui ne vont pas sur-performer sur certains blocs et sous-performer sur d'autres. Grâce à cette méthode, on va avoir tendance à limiter le sur-apprentissage.

Paramètres d'oversampling pour la fréquence :
Après plusieurs tests sur le modèle de fréquence, il s'est avéré que ceux-ci captaient mal les fortes fréquences, il a donc été décidé d'augmenter le poids (en multipliant l'exposition par un coefficient supérieur à 1) des instances où la fréquence était supérieure à la médiane des fréquences sur l'ensemble de la base d'apprentissage. Pour cette augmentation de poids, en parallèle de la validation par bloc, pour chaque hyperparamètrage testé, il s'agissait de tester l'ensemble des coefficients d'oversampling parmi la liste suivante : [1.075, 1.1, 1.15, 1.175].

3.4.3.1 Sélection des hyperparamètres et du nombre d'années en historique pour les modèles de forêt aléatoire :

La sélection d'hyperparamètres optimaux pour le modèle de fréquence et pour les deux scénarios est résumée dans le tableau suivant :

Hyperparamètres	Scénario 3 ans	Scénario 4 ans
max_depth	36	40
max_features	4	2
min_samples_split	444	344
min_samples_leaf	2	2
n_estimators	90	63
coeff_oversampling	1.15	1.175

TABLE 3.6 – Modèle de fréquence/forêts aléatoires : hyperparamètres optimaux

Notons bien tout d'abord, que pour le scénario avec trois années d'historique, une fois la meilleure combinaison d'hyperparamètres sélectionnée, le modèle a été appris sur les trois dernières années de l'échantillon d'apprentissage, ie 2015, 2016 et 2017. De la même manière, pour le scénario avec quatre années d'historique, le modèle a été appris sur les années 2014, 2015, 2016 et 2107. Observons donc désormais les scores pour ces deux scénarios possibles dans le tableau ci-dessous :

	Train		Test	
Métrique	Scénario 3 ans	Scénario 4 ans	Scénario 3 ans	Scénario 4 ans
Gini	0.393	0.381	0.360	0.3599
True/Pred	0.941	0.933	1.068	1.084
RMSE	0.1216	0.1196	0.1341	0.1343
MAE	0.0567	0.0562	0.0628	0.0630

TABLE 3.7 – Modèle de fréquence/forêts aléatoires : score des différents scénarios

Bien que le scénario avec 4 années d'historique semble donner des meilleurs scores sur les données d'apprentissage pour la RMSE et la MAE, il est moins bon sur les scores Gini et True/Pred. Par ailleurs sur les données de test, les scores MAE et RMSE sont meilleurs (plus faibles) pour le scénario avec 3 ans, ce qui peut mettre en avant un léger sur-apprentissage du modèle causé par le nombre d'années en apprentissage plus élevé. Nous ferons donc le choix de conserver le scénario avec 3 années d'historique.

Pour les deux scénarios, la sélection d'hyperparamètres optimaux pour le modèle de coûts est résumée dans le tableau suivant :

Hyperparamètres	Scénario 3 ans	Scénario 4 ans
max_depth	8	16
max_features	41	39
min_samples_split	104	4
min_samples_leaf	12	32
n_estimators	277	123

TABLE 3.8 – Modèle de coûts/forêts aléatoires : hyperparamètres optimaux

De la même manière que pour le modèle de fréquence, les modèles de coûts sont appris sur trois ou quatre années d'historique en fonction du scénario choisi. Observons donc désormais les scores pour ces deux scénarios possibles dans le tableau ci-dessous :

	Train		Test	
Métrique	Scénario 3 ans	Scénario 4 ans	Scénario 3 ans	Scénario 4 ans
Gini	0.1407	0.1065	0.0967	0.0903
True/Pred	1.0499	1.0624	0.9767	0.9866
RMSE	1523	1537	1372	1364
MAE	779	787	740	727

TABLE 3.9 – Modèle de coûts/forêts aléatoires : score des différents scénarios

Pour le modèle de coûts, les hyperparamètres retenus sont donc ceux du scénario avec trois ans d'historiques.

Pour les forêts aléatoires, aussi bien pour les coûts que pour les fréquences, nous testons d'ajouter ou non le paramètre *bootstrap* à cette sélection d'hyperparamètres.
Nous présentons dans le tableau ci-dessous les scores obtenus avec l'utilisation ou non du paramètre *bootstrap* pour le modèle de fréquence :

	Train		Test	
Métrique	Avec Bootstrap	Sans Bootstrap	Avec Bootstrap	Sans Bootstrap
Gini	0.393	0.383	0.360	0.358
True/Pred	0.941	0.939	1.068	1.066
RMSE	0.1216	0.1221	0.1341	0.1343
MAE	0.0567	0.0573	0.0628	0.0631

TABLE 3.10 – Modèle de fréquence/forêts aléatoires : score suivant bootstrap ou non

Pour le modèle de fréquence, le choix final a donc été d'utiliser la méthode Bootstrap. Bien que les True/Pred sur les données de validation soit légèrement plus proche de 1 sur les données de validation sans bootstrap, les Gini sont plus faibles sur les données d'apprentissage et de

validation
Il convient de faire la même chose pour le modèle de coûts, nous présentons donc dans le tableau ci-dessous les scores obtenus avec l'utilisation ou non du paramètre *bootstrap* :

	Train		Test	
Métrique	Avec Bootstrap	Sans Bootstrap	Avec Bootstrap	Sans Bootstrap
Gini	0.1407	0.1281	0.0967	0.1041
True/Pred	1.0499	1.0499	0.9767	0.9769
RMSE	1523	1523	1372	1384
MAE	779	780	740	746

TABLE 3.11 – Modèle de coûts/forêts aléatoires : score suivant bootstrap ou non

Pour le modèle de coûts, le choix final a donc été de ne pas utiliser de méthode Bootstrap. Bien que le Gini sur les données d'apprentissage s'améliore, celui-ci diminue sur les données de test. Il y a donc un sur-apprentissage en utilisant cette fonctionnalité.

3.4.3.2 Sélection des hyperparamètres et du nombre d'années en historique pour les modèles de gradient boosting :

Nous procéderons de la même manière que pour la sélection des hyperparamètres des modèles de forêts aléatoires.
Pour les deux scénarios, la sélection d'hyperparamètres optimaux pour le modèle de fréquence est résumée dans le tableau suivant :

Hyperparamètres	Scénario 3 ans	Scénario 4 ans
n_estimators	213	285
min_samples_split	464	404
min_samples_leaf	22	32
max_depth	10	4
max_features	4	10
subsample	0.8	0.80
learning_rate	0.04	0.08
loss	ls	ls
coeff_oversampling	0.15	0.175

TABLE 3.12 – Modèle de fréquence/Gradient Boosting : hyperparamètres optimaux

Il est intéressant ici de souligner le fait que pour les deux scénarios, l nombre maximal de variable des arbres du modèle de Gradient Boosting optimale est de 4 ou de 10 suivant le scénario choisi. Cela met donc bien en avant le fait que le Gradient Boosting se sert d'arbre peu complexe pour construire un modèle complexe. Les modèles de fréquence sont appris sur

trois ou quatre années d'historique en fonction du scénario choisi. Observons donc désormais les scores pour ces deux scénarios possibles dans le tableau ci-dessous :

	Train		Test	
Métrique	Scénario 3 ans	Scénario 4 ans	Scénario 3 ans	Scénario 4 ans
Gini	0.377	0.433	0.363	0.350
True/Pred	0.942	0.935	1.067	1.085
RMSE	0.1220	0.116	0.1337	0.1343
MAE	0.0568	0.0531	0.0624	0.0620

TABLE 3.13 – Modèle de fréquence/Gradient Boosting : score des différents scénarios

Bien que le scénario avec 4 années d'historique semble donner des meilleurs scores sur les données d'apprentissage notamment en termes de Gini, il est moins performant sur les données de test. Pour le modèle de fréquence, les hyperparamètres retenus sont donc ceux du scénario avec 3 années d'historique.

Pour les deux scénarios, la sélection d'hyperparamètres optimaux pour le modèle de coûts est résumée dans le tableau suivant :

Hyperparamètres	Scénario 3 ans	Scénario 4 ans
n_estimators	146	93
min_samples_split	404	444
min_samples_leaf	2	22
max_depth	2	10
max_features	40	2
subsample	0.7	0.8
learning_rate	0.31	0.33
loss	ls	ls

TABLE 3.14 – Modèle de coûts/Gradient boosting : hyperparamètres optimaux

Observons donc désormais les scores pour ces deux scénarios possibles dans le tableau ci-dessous :

	Train		Test	
Métrique	Scénario 3 ans	Scénario 4 ans	Scénario 3 ans	Scénario 4 ans
Gini	0.1269	0.1387	0.0854	0.0802
True/Pred	1.0506	1.0603	0.9756	0.9805
RMSE	1528	1511	1376	1384
MAE	782	780	747	750

TABLE 3.15 – Modèle de coûts/Gradient Boosting : score des différents scénarios

Bien que le scénario avec 4 années d'historique semble donner des meilleurs scores sur les données d'apprentissage, par ailleurs, il est moins performant sur les données de test. Il semble donc qu'il y ait un net sur-apprentissage. Le scénario retenu sera donc le scénario avec trois années d'historique.

Synthèse :
Une synthèse des hyperparamètres obtenus pour chacun des différents modèles est réalisée dans le tableau ci-dessous.

	Forêts aléatoires		Gradient Boosting	
Hyperparamètres	Fréquence	Coûts	Fréquence	Coûts
n_estimators	90	277	213	146
max_depth	36	8	10	2
max_features	4	41	4	40
min_samples_split	444	104	464	404
min_samples_leaf	2	12	22	2
subsample	-	-	0.8	0.7
learning_rate	-	-	0.04	0.31
loss	-	-	ls	ls
Années d'historique	3 ans	3 ans	3 ans	3 ans
coeff_oversampling	0.15	-	0.15	-

TABLE 3.16 – Synthèse des hyperparamètres des modèles de machine learning

3.4.4 Analyses des résultats des différents modèles

Maintenant que les modèles ont été calibrés et que les scénarios ont été choisis pour les modèles de coûts et de fréquence, nous pouvons comparer dans un tableau récapitulatif les forêts aléatoires et les gradients boosting aux GLM. La modélisation des GLM est placée en annexe (C) celle-ci n'étant pas l'objectif principal du mémoire.

	Train			Test		
Métrique	RF	GB	GLM	RF	GB	GLM
Gini	0.3932	0.3765	0.3451	0.3597	0.3630	0.3485
True/Pred	0.9417	0.9419	0.9391	1.0682	1.0669	1.0643
RMSE	0.1217	0.1221	0.1237	0.1341	0.1337	0.1343
MAE	0.0567	0.0568	0.0588	0.0628	0.0624	0.0636

TABLE 3.17 – Modèle de fréquence : comparaison des modèles

	Train				Test			
	Réelle	RF	GB	GLM	Réelle	RF	GB	GLM
Moyenne	0.05656	0.06006	0.060015	0.06023	0.06466	0.06053	0.06056	0.06075

TABLE 3.18 – Modèle de fréquence : comparaison des moyennes prédites vs Vraies valeurs

Pour appuyer ces résultats, nous pouvons aussi analyser ci-dessous les scores issus des prédictions des blocs de validation croisée. Il s'agira donc de score de validation, et c'est un des meilleurs moyens de comparer les modèles.

Métrique	RF	GB	GLM
Gini	0.3545	0.3572	0.3435
True/Pred	1.0013	1.0003	0.9978
RMSE	0.1205	0.1203	0.1210
MAE	0.0556	0.0550	0.0562

TABLE 3.19 – Modèle de fréquence : comparaison des modèles sur les sorties de la validation par bloc

Pour avoir une autre vision des différents modèles, nous observons ci-dessous un graphique des prédictions des blocs de la validation par bloc pour les trois modèles :

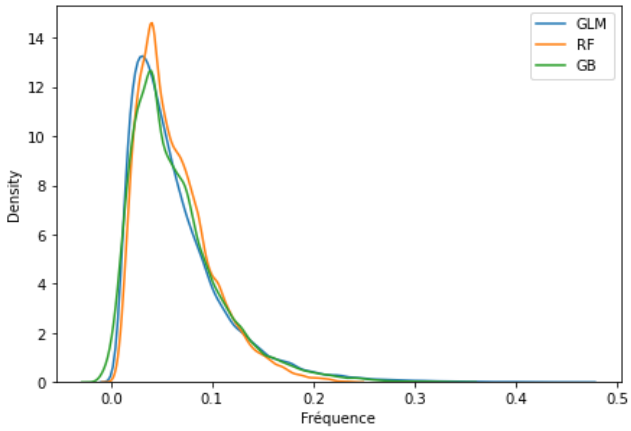


FIGURE 3.9 – Tracé des prédictions issues de la validation par bloc (ie 2014-2015-2015-2017)

Les trois modèles ont une distribution de prédiction très similaire, ce qui explique des scores obtenus similaire.
Le modèle qui est le plus performant sur les données d'apprentissage est le modèle de forêts aléatoires. Sur les données de validation, il s'agit du modèle de Gradient Boosting. On observe que les deux modèles de machine learning sont nettement meilleurs sur les données d'apprentissages que le GLM. Mais bien que restant plus performant que le GLM sur les données de validation, la différence de score est moindre. Le modèle retenu pour la fréquence sera le modèle de **gradient boosting**.

La même comparaison est effectuée pour les modèles de coûts, dont les résultats se trouvent dans le tableau ci-dessous :

	Train			Test		
Métrique	RF	GB	GLM	RF	GB	GLM
Gini	0.1281	0.1269	0.0440	0.1041	0.0854	0.0493
True/Pred	1.0499	1.0506	1.0652	0.9769	0.9756	0.9989
RMSE	1523	1528	1559	1384	1376	1370
MAE	780	782	791	746	747	728

TABLE 3.20 – Modèle de coûts : comparaison des modèles

	Train				Test			
	Réelle	RF	GB	GLM	Réelle	RF	GB	GLM
Moyenne	1260.42	1200.55	1199.73	1183.24	1195.01	1223.29	1224.87	1196.27

TABLE 3.21 – Modèle de coûts : comparaison des moyennes prédites vs Vraies valeurs

Comme pour la fréquence, nous observons ci-dessous les scores obtenus sur les sorties issues de la validation par bloc.

Métrique	RF	GB	GLM
Gini	0.1297	0.1267	0.0633
True/Pred	0.9850	0.9862	1.0083
RMSE	1540	1544	1549
MAE	826	829	819

TABLE 3.22 – Modèle de coûts : comparaison des modèles sur les sorties de la validation par bloc

Pour avoir une autre vision des différents modèles de coûts, nous observons ci-dessous un graphique des prédictions des blocs de la validation par bloc pour les trois modèles :

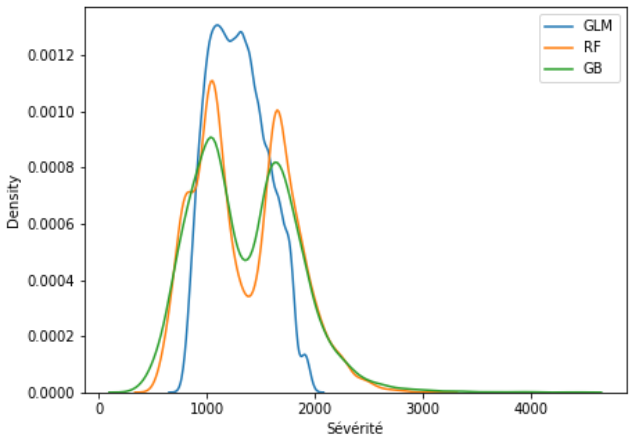


FIGURE 3.10 – Tracé des prédictions issues de la validation par bloc (ie 2014-2015-2015-2017)

Il est clair ici que le GLM fait des prédictions dans un intervalle beaucoup plus restreint que les modèles de forêts aléatoires et de gradient boosting. On peut donc imaginer que lorsque l'on regardera les résultats des modèles modalités par modalité, celui-ci sera moins bon sur certains segments où il est nécessaire de prédire des coûts faibles ou élevés.
Le modèle le plus performant sur les données d'apprentissage et de validation est les forêts aléatoires. Le modèle gradient boosting semble légèrement sur-apprendre, étant donné que son efficacité est moins bonne sur les données de validation par rapport aux forêts aléatoires, alors que les deux modèles avaient des Gini semblables sur les données d'apprentissage. Toutefois, sur les données de sorties de validation par bloc, le modèle de gradient boosting est tout aussi bon que le modèle de forêts aléatoires. Aussi bien le Gradient Boosting que les forêts aléatoires semblent surpasser le GLM pour la modélisation des coûts sur les données d'apprentissage que sur les données de validation. Pour le modèle de coûts, nous retiendrons donc le modèle de **forêts aléatoire** ou **gradient boosting** en second plan.

Enfin, pour les trois modèles étudiés, le constat est le même sur la fréquence et les coûts, étant donné que les fréquences et les coûts sur l'année 2018 (données de test) sont en moyenne respectivement plus faible et plus élevée, les modèles ont tous tendances à prédire sur les données de test une fréquence trop faible et un coût trop élevé. On peut penser qu'il s'agit ici d'un sur-apprentissage, mais il s'agit surtout d'une année différente des autres années en termes de fréquence et de coûts, d'où cette erreur de prédiction.

3.5 Étude de la performance des modèles

3.5.1 Étude des erreurs par classe d'individus

Outre les erreurs sur l'ensemble des données d'apprentissage et de validation, pour pouvoir observer où les modèles créés sont performants, il est nécessaire d'observer les erreurs pour certaines classes. C'est-à-dire prendre une certaine partie du portefeuille, par exemple les individus habitant dans une maison, et regarder les métriques associées aux prédictions sur cette partie. Grâce à cela, on peut observer si les modèles de Gradient Boosting et de forêts aléatoires performent plus que les GLM sur certaines classes.
Dans cette sous-partie, les erreurs ont été calculées sur les prédictions issues des blocs de validation croisée, donc c'est comme s'il s'agissait d'erreur de validation.

3.5.1.1 Erreurs par classe unique

Étude de la fréquence :

Features	Suffixe	% expo	% sinistre	Gini RF	Gini GB	Gini GLM
FORMULE	Confort	0.812	0.643	0.355	0.357	0.344
FORMULE	Essentiel	0.040	0.035	0.336	0.344	0.342
FORMULE	Tout_risque	0.148	0.322	0.334	0.337	0.318
TYPE_HABITATION	Autres_que_domicile	0.142	0.121	0.310	0.316	0.305
TYPE_HABITATION	Domicile	0.858	0.879	0.338	0.341	0.325
NATURE_HABITATION	Appartement	0.518	0.526	0.364	0.366	0.361
NATURE_HABITATION	Maison	0.482	0.474	0.309	0.313	0.291
NB_PIECES	inf 20m2	0.078	0.043	0.456	0.470	0.398
NB_PIECES	inf 40m2	0.195	0.132	0.354	0.356	0.353
NB_PIECES	inf 80m2	0.423	0.384	0.350	0.352	0.343
NB_PIECES	sup 80m2	0.304	0.441	0.302	0.304	0.284
TYPE_OCCUPANT	Locataire	0.401	0.264	0.351	0.353	0.343
TYPE_OCCUPANT	Proprietaire	0.599	0.736	0.337	0.340	0.323
TYPE_OCCUPANT_COMP	Colocataire	0.010	0.015	0.275	0.269	0.290
TYPE_OCCUPANT_COMP	Sans_infos	0.990	0.985	0.354	0.356	0.343
VALEUR_ASSUREE	Sans_patmob	0.040	0.013	0.589	0.620	0.500
VALEUR_ASSUREE	]0 ;3500]	0.095	0.067	0.322	0.326	0.317
VALEUR_ASSUREE	]13700 ;20500]	0.138	0.154	0.319	0.318	0.302
VALEUR_ASSUREE	]3500 ;6800]	0.214	0.104	0.308	0.310	0.306
VALEUR_ASSUREE	]20500 ;27300]	0.095	0.132	0.310	0.314	0.293
VALEUR_ASSUREE	]27300 ;41000]	0.082	0.137	0.309	0.313	0.289
VALEUR_ASSUREE	]6800 ;13700]	0.227	0.153	0.321	0.323	0.316
VALEUR_ASSUREE	]41000 ;54600]	0.056	0.121	0.282	0.289	0.270
VALEUR_ASSUREE	>54600	0.054	0.119	0.304	0.307	0.288
OBJETS_DE_VALEUR	Faible_valeur	0.933	0.772	0.345	0.348	0.333
OBJETS_DE_VALEUR	Forte_valeur	0.067	0.228	0.289	0.294	0.280
Hist_Dom	0	0.832	0.519	0.324	0.326	0.309
Hist_Dom	1	0.168	0.481	0.273	0.276	0.263
Hist_Aut	0	0.944	0.748	0.335	0.338	0.323
Hist_Aut	1	0.056	0.252	0.280	0.284	0.269
Hist_RC	1	0.022	0.102	0.343	0.344	0.333
Hist_RC	0	0.978	0.898	0.353	0.356	0.342

TABLE 3.23 – Modèle de fréquence : comparaison par modalité (1/2)

Features	Suffixe	% expo	% sinistre	T/P RF	T/P B	T/P GLM
FORMULE	Confort	0.812	0.643	1.006	1.006	1.004
FORMULE	Essentiel	0.040	0.035	0.929	1.000	1.004
FORMULE	Tout_risque	0.148	0.322	0.990	0.972	0.967
TYPE_HABITATION	Autres_que_domicile	0.142	0.121	0.894	0.979	0.972
TYPE_HABITATION	Domicile	0.858	0.879	1.011	1.002	1.000
NATURE_HABITATION	Appartement	0.518	0.526	1.028	1.008	1.005
NATURE_HABITATION	Maison	0.482	0.474	0.962	0.989	0.987
NB_PIECES	inf 20m2	0.078	0.043	0.801	0.958	0.911
NB_PIECES	inf 40m2	0.195	0.132	0.925	0.987	1.017
NB_PIECES	inf 80m2	0.423	0.384	1.022	1.019	1.013
NB_PIECES	sup 80m2	0.304	0.441	1.044	0.988	0.979
TYPE_OCCUPANT	Locataire	0.401	0.264	0.962	1.017	1.021
TYPE_OCCUPANT	Proprietaire	0.599	0.736	1.021	0.993	0.987
TYPE_OCCUPANT_COMP	Colocataire	0.010	0.015	0.793	0.898	0.747
TYPE_OCCUPANT_COMP	Sans_infos	0.990	0.985	1.003	1.001	1.000
VALEUR_ASSUREE	Sans_patmob	0.040	0.013	0.531	0.838	0.453
VALEUR_ASSUREE	]0 ;3500]	0.095	0.067	0.945	1.010	1.003
VALEUR_ASSUREE	]13700 ;20500]	0.138	0.154	1.049	1.026	1.087
VALEUR_ASSUREE	]3500 ;6800]	0.214	0.104	0.985	1.023	0.997
VALEUR_ASSUREE	]20500 ;27300]	0.095	0.132	0.991	0.968	0.979
VALEUR_ASSUREE	]27300 ;41000]	0.082	0.137	1.035	0.993	0.991
VALEUR_ASSUREE	]6800 ;13700]	0.227	0.153	0.999	1.005	1.056
VALEUR_ASSUREE	]41000 ;54600]	0.056	0.121	1.057	1.004	0.968
VALEUR_ASSUREE	>54600	0.054	0.119	1.004	0.953	0.878
OBJETS_DE_VALEUR	Faible_valeur	0.933	0.772	0.999	1.007	1.004
OBJETS_DE_VALEUR	Forte_valeur	0.067	0.228	1.018	0.953	0.955
Hist_Dom	0	0.832	0.519	0.982	1.008	1.005
Hist_Dom	1	0.168	0.481	1.049	0.983	0.980
Hist_Aut	0	0.944	0.748	0.992	1.000	0.997
Hist_Aut	1	0.056	0.252	1.073	1.003	1.005
Hist_RC	1	0.022	0.102	1.033	0.960	0.962
Hist_RC	0	0.978	0.898	1.000	1.002	0.999

TABLE 3.24 – Modèle de fréquence : comparaison par modalité (2/2)

Les gini sont dans l'ensemble meilleurs pour les deux modèles de machine learning sur la majorité des modalités. On peut remarquer par contre en particulier que les modèles semblent mieux capter les effets sur les appartements que sur les maisons, il y a une différence de gini de 6 points environs entre ces deux modalités pour l'ensemble des modèles.

Étude des coûts :

Features	Suffixe	% nb	% charge	Gini RF	Gini GB	Gini GLM
FORMULE	Confort	0.803	0.782	0.127	0.129	0.065
FORMULE	Essentiel	0.026	0.016	0.068	0.005	0.006
FORMULE	Tout_risque	0.171	0.202	0.080	0.076	0.058
TYPE_HABITATION	Autres_que_domicile	0.075	0.082	0.113	0.111	0.012
TYPE_HABITATION	Domicile	0.925	0.918	0.130	0.129	0.069
NATURE_HABITATION	Appartement	0.607	0.485	0.072	0.076	0.061
NATURE_HABITATION	Maison	0.393	0.515	0.019	0.017	0.004
NB_PIECES	inf 20m2	0.037	0.023	0.007	0.037	0.004
NB_PIECES	inf 40m2	0.169	0.126	0.069	0.095	0.055
NB_PIECES	inf 80m2	0.450	0.404	0.099	0.114	0.041
NB_PIECES	sup 80m2	0.344	0.446	0.036	0.034	0.003
TYPE_OCCUPANT	Locataire	0.315	0.223	0.065	0.090	0.071
TYPE_OCCUPANT	Proprietaire	0.685	0.777	0.067	0.068	0.025
TYPE_OCCUPANT_COMP	Colocataire	0.005	0.003	0.003	0.000	0.000
TYPE_OCCUPANT_COMP	Sans_infos	0.995	0.997	0.129	0.126	0.062
VALEUR_ASSUREE	]0 ;3500]	0.057	0.038	0.056	0.075	0.056
VALEUR_ASSUREE	]3500 ;6800]	0.171	0.130	0.088	0.118	0.061
VALEUR_ASSUREE	Sans_patmob	0.008	0.010	0.079	0.057	0.000
VALEUR_ASSUREE	]13700 ;20500]	0.162	0.163	0.080	0.087	0.018
VALEUR_ASSUREE	]20500 ;27300]	0.109	0.121	0.048	0.060	0.008
VALEUR_ASSUREE	]6800 ;13700]	0.230	0.214	0.123	0.152	0.042
VALEUR_ASSUREE	]27300 ;41000]	0.106	0.119	0.040	0.033	0.026
VALEUR_ASSUREE	]41000 ;54600]	0.078	0.099	0.043	0.060	0.000
VALEUR_ASSUREE	>54600	0.079	0.107	0.033	0.042	0.023
OBJETS_DE_VALEUR	Faible_valeur	0.880	0.844	0.138	0.139	0.067
OBJETS_DE_VALEUR	Forte_valeur	0.120	0.156	0.049	0.061	0.016
Hist_Dom	0	0.698	0.686	0.151	0.153	0.066
Hist_Dom	1	0.302	0.314	0.080	0.073	0.051
Hist_Aut	0	0.967	0.966	0.133	0.131	0.064
Hist_Aut	1	0.033	0.034	0.038	0.014	0.015
Hist_RC	0	0.877	0.880	0.141	0.139	0.064
Hist_RC	1	0.123	0.120	0.064	0.060	0.041

TABLE 3.25 – Modèle de coûts : comparaison par modalité (1/2)

Features	Suffixe	% nb	% charge	Gini RF	Gini GB	Gini GLM
FORMULE	Confort	0.803	0.782	0.983	0.985	0.999
FORMULE	Essentiel	0.026	0.016	0.909	0.965	0.803
FORMULE	Tout_risque	0.171	0.202	1.001	0.992	1.070
TYPE_HABITATION	Autres_que_domicile	0.075	0.082	1.014	0.993	1.220
TYPE_HABITATION	Domicile	0.925	0.918	0.982	0.986	0.993
NATURE_HABITATION	Appartement	0.607	0.485	1.002	1.002	0.909
NATURE_HABITATION	Maison	0.393	0.515	0.969	0.972	1.124
NB_PIECES	inf 20m2	0.037	0.023	0.935	0.930	0.883
NB_PIECES	inf 40m2	0.169	0.126	0.985	1.002	0.935
NB_PIECES	inf 80m2	0.450	0.404	0.981	0.984	0.958
NB_PIECES	sup 80m2	0.344	0.446	0.991	0.987	1.092
TYPE_OCCUPANT	Locataire	0.315	0.223	1.001	1.001	0.865
TYPE_OCCUPANT	Proprietaire	0.685	0.777	0.980	0.982	1.059
TYPE_OCCUPANT_COMP	Colocataire	0.005	0.003	0.775	0.843	0.677
TYPE_OCCUPANT_COMP	Sans_infos	0.995	0.997	0.986	0.987	1.010
VALEUR_ASSUREE	]0 ;3500]	0.057	0.038	0.947	1.014	0.909
VALEUR_ASSUREE	]3500 ;6800]	0.171	0.130	0.978	1.000	0.950
VALEUR_ASSUREE	Sans_patmob	0.008	0.010	1.034	1.021	1.524
VALEUR_ASSUREE	]13700 ;20500]	0.162	0.163	0.973	0.967	1.017
VALEUR_ASSUREE	]20500 ;27300]	0.109	0.121	0.975	0.965	1.027
VALEUR_ASSUREE	]6800 ;13700]	0.230	0.214	1.033	1.029	1.043
VALEUR_ASSUREE	]27300 ;41000]	0.106	0.119	0.933	0.942	0.971
VALEUR_ASSUREE	]41000 ;54600]	0.078	0.099	0.998	0.989	1.020
VALEUR_ASSUREE	>54600	0.079	0.107	0.991	0.977	1.023
OBJETS_DE_VALEUR	Faible_valeur	0.880	0.844	0.985	0.989	0.993
OBJETS_DE_VALEUR	Forte_valeur	0.120	0.156	0.987	0.969	1.098
Hist_Dom	0	0.698	0.686	0.992	0.985	1.013
Hist_Dom	1	0.302	0.314	0.971	0.989	0.997
Hist_Aut	0	0.967	0.966	0.987	0.987	1.009
Hist_Aut	1	0.033	0.034	0.931	0.960	0.983
Hist_RC	0	0.877	0.880	0.984	0.985	1.011
Hist_RC	1	0.123	0.120	0.992	0.998	0.991

TABLE 3.26 – Modèle de coûts : comparaison par modalité (2/2)

Comme pour les modèles de fréquences, les gini sont dans l'ensemble meilleurs pour les deux modèles de machine learning sur la majorité des modalités. On peut remarquer par contre en particulier que les modèles semblent mieux capter les effets sur les appartements que sur les maisons, il y a une différence de gini de 6 points environs entre ces deux modalités pour l'ensemble des modèles. Le modèle de Gradient boosting semble bien capter le segment]6800 ;13700] pour le montant de valeur assuré, avec un gini de 0.152, supérieur aux autres modèles.
Sur certains segments où le GLM ne semble pas bien prédire (où l'on observe un ratio True/Pred éloigné de 1), comme la Formule Essentiel, les types d'occupant colocataire, les forts montants en objets de valeur, les modèles de machine learning ont des ratios nettement plus proches de 1. Ces segments sont en général peu représentés dans la base. Ces observations appuient donc le fait que les modèles de machine learning arrive à capter des effets à une maille plus fine et appuient les observations faites suite au graphique 3.10.

3.5.1.2 Erreurs par classes multiples

Pour les coûts et les fréquences, grâce à PFI (voir § 4.5.1), les trois variables les plus importantes de chaque modèle ont été sélectionnées. Dans cette section, les métriques associées aux prédictions ont été calculées sur des sous-ensembles où trois variables ont été fixées à une modalité

particulière. Ainsi, si les trois variables étaient composées de 2, 3 et 2 modalités, un tableau de 12 lignes a été obtenu. Une ligne représentant une combinaison possible des modalités des trois variables les plus influentes du modèle observé.
Les données cibles et de prédictions pour calculer ces erreurs par classe sont issues de la validation par bloc. Ainsi, c'est comme s'il s'agissait d'erreurs de validation, il n'est donc pas anormal d'observer des erreurs bien plus élevées que celles obtenues précédemment sur la base d'apprentissage.

Étude de la fréquence :

Variables retenues : Nature d'habitation, Type d'occupant et Hist_Dom

NATURE HABITATION	Hist_Dom	TYPE OCCUPANT	% expo	% sinistre	Gini RF	Gini GB	Gini GLM
Appartement	0	Locataire	0.282	0.103	0.289	0.292	0.286
Appartement	0	Proprietaire	0.161	0.160	0.268	0.270	0.259
Appartement	1	Locataire	0.032	0.082	0.237	0.231	0.228
Appartement	1	Proprietaire	0.043	0.181	0.218	0.224	0.212
Maison	0	Locataire	0.080	0.055	0.335	0.332	0.311
Maison	0	Proprietaire	0.309	0.201	0.257	0.262	0.231
Maison	1	Locataire	0.007	0.023	0.239	0.240	0.236
Maison	1	Proprietaire	0.087	0.195	0.181	0.187	0.175

TABLE 3.27 – Modèle de fréquence : comparaison par couple de modalités (2/2)

NATURE HABITATION	Hist_Dom	TYPE OCCUPANT	% expo	% sinistre	T/P RF	T/P GB	T/P GLM
Appartement	0	Locataire	0.282	0.103	0.971	1.032	0.988
Appartement	0	Proprietaire	0.161	0.160	1.043	0.998	1.029
Appartement	1	Locataire	0.032	0.082	1.002	0.961	1.098
Appartement	1	Proprietaire	0.043	0.181	1.133	1.008	0.953
Maison	0	Locataire	0.080	0.055	0.864	1.037	1.072
Maison	0	Proprietaire	0.309	0.201	0.960	0.992	0.990
Maison	1	Locataire	0.007	0.023	0.940	0.981	1.251
Maison	1	Proprietaire	0.087	0.195	1.004	0.970	0.945

TABLE 3.28 – Modèle de fréquence : comparaison par couple de modalités (2/2)

Étude des coûts :

Variables retenues : Nature d'habitation, Type d'occupant et valeur assurée
Pour le modèle de coûts, le tableau sorti est composé de 36 lignes, donc pour plus de clarté nous afficherons seulement les dix segments comportant le plus de charge de sinistre.

Nature habitation	Type d'occupant	Valeur assurée	% nb	% charge	Gini RF	Gini GB	Gini GLM
Maison	Proprietaire	]13700 ;20500]	0.061	0.079	0.000	0.000	0.000
Maison	Proprietaire	]27300 ;41000]	0.058	0.076	0.011	0.012	0.000
Maison	Proprietaire	>54600	0.049	0.074	0.000	0.000	0.000
Appartement	Proprietaire	]6800 ;13700]	0.084	0.074	0.007	0.026	0.001
Maison	Proprietaire	]6800 ;13700]	0.053	0.072	0.031	0.021	0.000
Maison	Proprietaire	]20500 ;27300]	0.052	0.071	0.007	0.009	0.000
Maison	Proprietaire	]41000 ;54600]	0.047	0.068	0.000	0.000	0.000
Appartement	Locataire	]3500 ;6800]	0.090	0.056	0.013	0.040	0.084
Appartement	Locataire	]6800 ;13700]	0.080	0.056	0.051	0.027	0.000
Appartement	Proprietaire	]13700 ;20500]	0.060	0.052	0.034	0.000	0.066

TABLE 3.29 – Modèle de coûts : comparaison par couple de modalités (1/2)

Nature habitation	Type d'occupant	Valeur assurée	% nb	% charge	T/P RF	T/P GB	T/P GLM
Maison	Proprietaire	]13700 ;20500]	0.061	0.079	0.949	0.967	1.169
Maison	Proprietaire	]27300 ;41000]	0.058	0.076	0.920	0.935	1.048
Maison	Proprietaire	>54600	0.049	0.074	0.977	0.971	1.066
Appartement	Proprietaire	]6800 ;13700]	0.084	0.074	1.046	1.037	1.001
Maison	Proprietaire	]6800 ;13700]	0.053	0.072	1.018	1.019	1.303
Maison	Proprietaire	]20500 ;27300]	0.052	0.071	0.977	0.976	1.152
Maison	Proprietaire	]41000 ;54600]	0.047	0.068	0.986	0.989	1.088
Appartement	Locataire	]3500 ;6800]	0.090	0.056	1.000	1.021	0.843
Appartement	Locataire	]6800 ;13700]	0.080	0.056	1.029	1.035	0.877
Appartement	Proprietaire	]13700 ;20500]	0.060	0.052	0.987	0.954	0.919

TABLE 3.30 – Modèle de coûts : comparaison par couple de modalités (2/2)

L'indice de Gini crée n'est pas consistant pour les données comportant trop peu d'individus. Donc dans notre cas en divisant notre base en 36 segments, et ces segments comportant au maximum 8 % de la charge de sinistre dans la base étudiée, bien qu'affiché, il est inutile de s'attarder sur le tableau (1/2) ci-dessus indiquant les Gini par segment.
Mis à part sur le segment Appartement, Propriétaire et]6800 ;13700] où le GLM performe mieux que les modèles de machine learning, sur les autres segments les modèles de machine learning performent nettement plus que le GLM. Ceci se remarque particulièrement sur les segments avec une charge de sinistre plus faible, comme le segment Appartement, Locataire et]3500 ;6800]. Sur les autres segments qui ne sont pas dans le top 10 qui est affiché, le constat est le même.

3.5.2 Travail sur le couplage des modèles de fréquence et de coûts

3.5.2.1 Étude sur l'ensemble des sorties

En plus des métriques habituelles comme le Gini, la MAE et la RMSE, une métrique intéressante à observer est le ratio du montant des sinistres sur la somme des primes pures (S/P). Ces ratios seront observés pour les différents modèles étudiés sur les prédictions obtenues lors de la validation par bloc.

Métrique	RF	GB	GLM
S/P	0.981	0.980	0.983
Gini	0.3384	0.3412	0.3261
RMSE	3230	3292	3308
MAE	1625	1630	1658

TABLE 3.31 – Comparaison des modèles coûts × fréquence

Pour obtenir ces scores, les RMSE, les MAE et les Gini ont été calculés sur les primes prédites. On observe que les modèles de machine learning surpassent le GLM sur l'ensemble des métriques sauf sur les S/P où le GLM donne un score légèrement supérieur. Nous allons étudier

ceci plus en profondeur dans la sous-partie suivante en étudiant les scores par segment.

3.5.2.2 Étude sur des segments

S/P par classe unique :

Features	Suffixe	% exposition	S/P RF	S/P GB	S/P GLM
FORMULE	Confort	0.812	0.971	0.973	0.969
FORMULE	Essentiel	0.040	0.891	1.010	0.840
FORMULE	Tout_risque	0.148	1.030	1.002	1.058
TYPE_HABITATION	Autres_que_domicile	0.142	0.890	0.950	1.141
TYPE_HABITATION	Domicile	0.858	0.989	0.982	0.971
NATURE_HABITATION	Appartement	0.518	1.026	1.000	0.888
NATURE_HABITATION	Maison	0.482	0.941	0.961	1.093
NB_PIECES	inf 20m2	0.078	0.748	0.886	0.809
NB_PIECES	inf 40m2	0.195	0.901	0.982	0.936
NB_PIECES	inf 80m2	0.423	0.978	0.995	0.946
NB_PIECES	sup 80m2	0.304	1.025	0.971	1.047
TYPE_OCCUPANT	Locataire	0.401	0.964	1.022	0.877
TYPE_OCCUPANT	Proprietaire	0.599	0.986	0.968	1.018
TYPE_OCCUPANT_COMP	Colocataire	0.010	0.614	0.720	0.500
TYPE_OCCUPANT_COMP	Sans_infos	0.990	0.982	0.981	0.986
VALEUR_ASSUREE	Sans_patmob	0.040	0.635	0.876	0.729
VALEUR_ASSUREE	]0 ;3500]	0.095	0.929	1.060	0.905
VALEUR_ASSUREE	]3500 ;6800]	0.214	0.951	1.006	0.928
VALEUR_ASSUREE	]6800 ;13700]	0.227	1.016	1.031	1.074
VALEUR_ASSUREE	]13700 ;20500]	0.138	1.004	0.986	1.080
VALEUR_ASSUREE	]20500 ;27300]	0.095	0.955	0.926	0.982
VALEUR_ASSUREE	]27300 ;41000]	0.082	0.955	0.930	0.941
VALEUR_ASSUREE	]41000 ;54600]	0.056	1.052	1.014	0.972
VALEUR_ASSUREE	>54600	0.054	0.981	0.920	0.886
OBJETS_DE_VALEUR	Faible_valeur	0.933	0.979	0.990	0.977
OBJETS_DE_VALEUR	Forte_valeur	0.067	0.989	0.926	1.020
Hist_Dom	0	0.832	0.968	0.982	1.000
Hist_Dom	1	0.168	1.010	0.974	0.948
Hist_Aut	0	0.944	0.976	0.978	0.989
Hist_Aut	1	0.056	1.019	0.990	0.942
Hist_RC	1	0.022	0.971	0.917	0.933
Hist_RC	0	0.978	0.981	0.982	0.985

TABLE 3.32 – Modèle de coûts × fréquence : comparaison par modalité des S/P

On remarque que pour la plupart des segments, les modèles de machine learning donne des ratios S/P plus proches de 1, et notamment le modèle de gradient boosting qui est dans la plupart des cas meilleur que le modèle de forêts aléatoires. Sur certains segments représentant une faible exposition comme la formule *Essentiel*, on observe que le gradient boosting performe nettement plus que le GLM (1.010 contre 0.84).

S/P par classe multiples :

Nous allons commencer par fixer les trois variables les plus importantes du modèle de fréquence, soit la nature d'habitation, le type d'occupant et Hist_Dom. Deux de ces variables sont en commun avec les plus importantes pour les modèles de coûts.

Nature habitation	Hist_Dom	Type d'occupant	% exposition	S/P RF	S/P GB	S/P GLM
Appartement	0	Locataire	0.282	0.979	1.044	0.832
Appartement	0	Proprietaire	0.161	1.044	0.976	0.952
Appartement	1	Locataire	0.032	1.003	1.004	0.954
Appartement	1	Proprietaire	0.043	1.073	0.988	0.838
Maison	0	Locataire	0.080	0.869	0.966	0.969
Maison	0	Proprietaire	0.309	0.931	0.962	1.143
Maison	1	Locataire	0.007	0.905	0.964	1.095
Maison	1	Proprietaire	0.087	0.979	0.957	1.029

TABLE 3.33 – Modèle de coûts × fréquence : comparaison des S/P par couple de modalités

Le même constat que précédemment peut être effectué ; le modèle de gradient boosting performe plus en termes de S/P en comparaison des S/P. Aussi bien sur le segment le plus important (Maison, 0, Propriétaire) où le S/P est de 0.96 pour le gradient boosting contre 1.14 et 0.93 respectivement pour le GLM et les forêts aléatoires, que sur un segment plus faiblement représenté (Appartement, 1, Propriétaire, 4.3 % d'exposition) où le S/P est de 0.98 pour le gradient boosting contre 0.83 et 1.07 respectivement pour le GLM et les forêts aléatoires.
Dans un second temps, réalisons de même pour les trois variables les plus importantes pour le modèle de coûts, soit la nature d'habitation, le type d'occupant et le montant en valeur assurée. On obtient le tableau de résultats suivant pour les 10 segments les plus exposés :

Nature habitation	Type d'occupant	Valeur assurée	% exposition	S/P RF	S/P GB	S/P GLM
Appartement	Locataire	]3500 ;6800]	0.114	0.971	1.043	0.803
Appartement	Locataire	]6800 ;13700]	0.076	1.018	1.056	0.932
Maison	Proprietaire	]6800 ;13700]	0.073	0.961	1.008	1.276
Maison	Proprietaire	]13700 ;20500]	0.068	0.952	0.966	1.208
Appartement	Locataire	]0 ;3500]	0.062	0.925	1.064	0.715
Maison	Proprietaire	]20500 ;27300]	0.058	0.937	0.940	1.095
Maison	Proprietaire	]27300 ;41000]	0.056	0.920	0.915	1.008
Appartement	Proprietaire	]6800 ;13700]	0.053	1.084	1.031	1.026
Appartement	Proprietaire	]3500 ;6800]	0.041	1.026	1.003	1.019
Maison	Proprietaire	>54600	0.041	0.949	0.920	0.953

TABLE 3.34 – Modèle de coûts × fréquence : comparaison des S/P par couple de modalités

Le modèle de gradient boosting est dans la majorité des segments meilleurs ou aussi bon que le GLM. Sur certains segments, cette différence est frappante, par exemple sur le segment Maison, propriétaire et $]6800, 13700]$, le modèle de gradient boosting a un S/P de 1.008 contre un S/P de 1.276 pour le GLM.

3.5.2.3 Étude des primes pures

Jetons un œil aux primes pures prédites pour les différents modèles calibrés et si celles-ci sont plus ou moins proches de la prime réelle qui aurait dû être donnée.

	Réelle	RF	GB	GLM
Ensemble des données	66.46	67.78	67.84	67.60
Données avec coût réel >0	130.35	68.53	69.52	69.22

TABLE 3.35 – Primes pures des différents modèles

On observe que bien que les GLM prédisent des primes pures plus proche de la prime réelle sur l'ensemble des données sans sinistre en réalité, ceux-ci sont moins bons en termes de prédiction sur les données où il y a en réalité un montant de sinistre positif. Le GLM capte donc moins bien la présence de sinistre dans les données, et les potentiels sinistres.

Primes pures par classe unique :

Features	Suffixe	% exposition	PP RF	PP GB	PP GLM
FORMULE	Confort	0.812	65.9	65.7	66.1
FORMULE	Essentiel	0.040	30.2	26.7	32.0
FORMULE	Tout_risque	0.148	88.0	90.4	85.6
TYPE_HABITATION	Autres_que_domicile	0.142	42.6	39.9	33.2
TYPE_HABITATION	Domicile	0.858	72.0	72.5	73.3
NATURE_HABITATION	Maison	0.482	75.3	73.8	64.9
NB_PIECES	inf 20m2	0.078	26.3	22.2	24.4
NB_PIECES	inf 40m2	0.195	47.6	43.7	45.9
NB_PIECES	sup 80m2	0.304	95.2	100.6	93.2
TYPE_OCCUPANT	Locataire	0.401	38.3	36.1	42.1
TYPE_OCCUPANT	Proprietaire	0.599	87.5	89.1	84.7
TYPE_OCCUPANT_COMP	Colocataire	0.010	28.8	24.6	35.4
TYPE_OCCUPANT_COMP	Sans_infos	0.990	68.2	68.3	68.0
VALEUR_ASSUREE	Sans_patmob	0.040	26.8	19.4	23.3
VALEUR_ASSUREE	]0 ;3500]	0.095	28.6	25.0	29.3
VALEUR_ASSUREE	]3500 ;6800]	0.214	42.3	40.0	43.4
VALEUR_ASSUREE	]13700 ;20500]	0.138	78.3	79.7	72.8
VALEUR_ASSUREE	]27300 ;41000]	0.082	100.8	103.5	102.2
VALEUR_ASSUREE	>54600	0.054	135.3	144.3	149.8
OBJETS_DE_VALEUR	Faible_valeur	0.933	61.4	60.7	61.5
OBJETS_DE_VALEUR	Forte_valeur	0.067	157.0	167.7	152.2
Hist_Dom	0	0.832	56.7	55.8	54.8
Hist_Dom	1	0.168	122.8	127.4	130.8

TABLE 3.36 – Modèle de coûts × fréquence : comparaison des primes pures par modalité

Pour des soucis de confidentialités, les primes de certaines modalités ne sont pas affichées.

Primes pures par classe multiples :
Tout d'abord pour les trois variables les plus importantes du modèle de fréquence :

Nature habitation	Hist_Dom	Type d'occupant	% exposition	PP réelles	PP RF	PP GB	PP GLM
Appartement	Locataire	0	0.282	33.487	34.215	32.071	40.267
Appartement	Locataire	1	0.032	88.711	88.471	88.327	92.996
Appartement	Proprietaire	1	0.043	162.821	151.803	164.737	194.288
Maison	Proprietaire	0	0.309	68.172	73.192	70.850	59.651
Maison	Proprietaire	1	0.087	122.615	125.295	128.173	119.102

TABLE 3.37 – Modèle de coûts × fréquence : comparaison des primes pures par couple de modalités

Pour des soucis de confidentialités, les primes pour certains couples ne sont pas affichées.
Les primes pures prédites sont en moyenne plus faibles pour les maisons que pour les appartements. Sur les graphiques de statistiques descriptives (1.3.4 et 1.3.3), des fréquences plus faibles mais des coûts plus forts étaient observés pour les maisons par rapport aux appartements. Les

primes pures plus faibles doivent donc être dues principalement à la fréquence plus faible ici.
Enfin pour les trois variables les plus importantes pour le modèle de coût :

Nature habitation	Type d'occupant	Valeur assurée	% exposition	PP réelles	PP RF	PP GB	PP GLM
Appartement	Locataire	]3500 ;6800]	0.114	32.705	33.693	31.342	40.717
Appartement	Locataire	]6800 ;13700]	0.076	48.668	47.791	46.089	52.226
Maison	Proprietaire	]6800 ;13700]	0.073	65.556	68.200	65.026	51.391
Maison	Proprietaire	]13700 ;20500]	0.068	76.716	80.562	79.454	63.511
Appartement	Locataire	]0 ;3500]	0.062	19.153	20.706	17.997	26.777
Maison	Proprietaire	]20500 ;27300]	0.058	81.630	87.130	86.803	74.549
Maison	Proprietaire	]27300 ;41000]	0.056	89.962	97.796	98.325	89.226
Appartement	Proprietaire	]6800 ;13700]	0.053	93.331	86.087	90.525	90.933
Appartement	Proprietaire	]3500 ;6800]	0.041	67.914	66.172	67.695	66.679
Maison	Proprietaire	>54600	0.041	120.947	127.424	131.525	126.932

TABLE 3.38 – Modèle de coûts × fréquence : comparaison des primes pures par couple de modalités

En fixant la nature d'habitation et le type d'occupant, on observe que la prime est croissante en fonction de la valeur assurée. Or sur les graphiques des statistiques descriptives (1.3.4 et 1.3.3), les fréquences et les coûts étaient croissant en fonction de la valeur assurée. Donc les modèles ont bien capté ce phénomène.
Enfin aussi bien en fixant les variables les plus importantes pour le modèle de fréquence que pour le modèle de coûts, on a pu observer que le modèle qui prédisait le mieux sur ces segments est le modèle de gradient boosting. Bien que sur certains segments il ne soit pas le meilleur, sur la majorité des segments il s'en sort mieux que le GLM et le modèle de forêts aléatoires.

Focus sur les primes pures par zonier :
Étant donné que les différents indices pour les zones géographiques ont été écartés des tableaux précédents pour des raisons de lisibilité, nous allons quand même faire un focus sur l'influence de la zone géographique dans la tarification. Ci-dessous est donc présenté un graphique comparant la prime pure en fonction de la zone géographique pour les différents modèles testés.

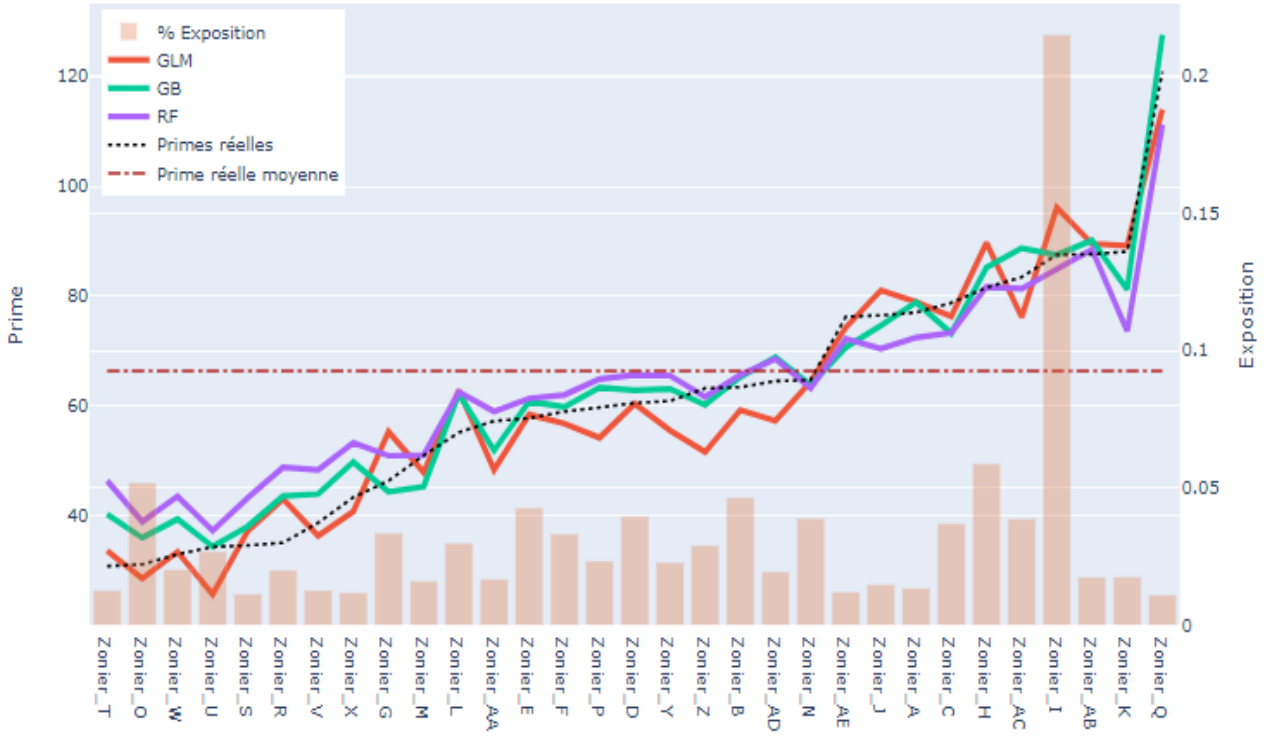


FIGURE 3.11 – Variation de la prime pure en fonction de la zone géographique

On peut donc aisément conclure que les modèles de machine learning sont plus proche de la prime réelle que le GLM. Bien que pour certaines zones géographiques, le GLM semble mieux prédire comme sur le zonier *Zonier_D*, ce n'est pas le cas sur la majorité des zones géographiques.

3.5.3 Choix du modèle final

Après toutes ces analyses de résultats, le modèle qui s'avère donner les meilleurs résultats est le modèle de **gradient boosting**. Ce choix est appuyé notamment en regardant les scores obtenus au niveau local où l'on observe que ce modèle arrive à capter les effets à une maille beaucoup plus fine. Le prochain enjeu consistera à défendre le côté transparent de ce modèle. Ce modèle étant un modèle de machine learning dit *boite noire*, il est nécessaire d'appuyer ce choix par des algorithmes de transparences que nous verrons dans le prochain chapitre. Par ailleurs, en étudiant les ratios True/Pred et S/P par classe multiples et uniques, nous avons d'ores et déjà commencé à répondre au problème d'explicabilité.

CHAPITRE

4

DÉFENSE DES NOUVEAUX MODÈLES GRÂCE À LA TRANSPARENCE

4.1 Intérêt réglementaire de la transparence

Outre les aspects pratiques personnels de la compréhension de son risque et des résultats des modèles boite noire, une certaine transparence est demandée aussi bien par le régulateur, l'assuré et le décideur. Pour valider un modèle de tarification, il est nécessaire d'avoir le contrôle sur ce modèle, et de comprendre pourquoi certaines catégories vont être amenées à avoir une prime plus élevée que d'autres. Les GLM par le biais de leurs coefficients (voir § 4.2), permettent cette explicabilité et cette intelligibilité, contrairement aux modèles boites noires. L'enjeu est donc, de la même manière qu'un GLM avec ses coefficients de sortie, d'être transparent sur l'impact qu'auront les variables dans la prédiction finale.
Le règlement général sur la protection des données (RGPD) prévoit des règles restrictives dans le cas des décisions automatisées pour éviter que l'homme ne subisse ces décisions émanant uniquement de machines. L'article 22 ([16]) du RGPD encadre ainsi les processus de prise de décision entièrement automatisés, lorsqu'elles produisent des effets juridiques ou des effets significatifs. Les décisions automatisées peuvent intervenir dans de nombreux domaines comme la finance et l'assurance, par exemple, une décision de refus de crédit peut avoir pour seul fondement l'utilisation d'un algorithme qui applique automatiquement certains critères à la situation financière du demandeur, sans aucune intervention humaine.
L'Autorité de contrôle prudentiel et de résolution (ACPR) quant à elle préconise quatre principes d'évaluations des algorithmes et outils d'intelligence artificielle (IA) :

- Traitement adéquat des données

- Performance
- Stabilité
- Explicabilité

Ces principes représentent des objectifs liés entre eux par des relations nécessitant un arbitrage : il est en général impossible de maximiser les quatre objectifs simultanément. Ces quatre principes constituent donc en quelque sorte des points cardinaux permettant d'orienter la conception et le développement d'un algorithme d'IA :

FIGURE 4.1 – Points cardinaux de la conception d'algorithme d'IA (source [4])

Les enjeux de traitements des données et de performance sont traités dans les sections 2.1 et 3.2 en partie. Le principe de stabilité consiste à garantir la persistance de la qualité et des caractéristiques d'un algorithme au cours du temps. Les attentes en termes de stabilité sont d'autant plus importantes dans le cas du ML que le nombre de dimension capturée afin de constituer l'ensemble des variables prédictives est en général bien plus important que dans des modèles prédictifs classiques. La stabilité d'un algorithme de ML s'entend avant tout comme stabilité temporelle. En effet la distribution des données peut changer au fil du temps et dégrader la performance de l'algorithme si celui n'est pas ré-entraîné. Cette stabilité ne sera pas étudiée en profondeur dans ce mémoire, mais le fait de faire une validation croisée par année permet d'assurer un minimum de stabilité temporelle. L'enjeu d'explicabilité est important avec d'une part les contraintes réglementaires et d'autre part les enjeux sociaux tels que la non-discrimination qui seront traités dans cette section.

4.2 Aspects littéraires de la transparence

Pour mieux cadrer le sujet de la transparence il est nécessaire de distinguer sous ce terme quatre notions indépendantes : la loyauté, l'équité, l'intelligibilité et l'explicabilité.

4.2.1 La loyauté des algorithmes

Un algorithme est dit loyal si la fonctionnalité affichée par son créateur auprès des utilisateurs correspond à la fonctionnalité connue du créateur. Si le créateur dissimule une fonctionnalité dont il a une claire conscience qu'elle est remplie par l'algorithme, alors celui-ci sera déloyal [15]. De nombreux usages actuels des algorithmes font malheureusement preuve d'une telle déloyauté. Par exemple : tarification volatile, personnalisation de l'offre commerciale réalisée selon des critères différents des critères affichés, recommandation de trajets basés sur la présence de points d'intérêts commerciaux, réponse à des requêtes sur un moteur de recherche donnant la priorité à des liens sponsorisés plutôt qu'à la pertinence... On peut d'une autre manière parler de déloyauté lorsque l'algorithme, bien qu'il remplisse les fonctionnalités affichées, remplit une autre fonctionnalité dont l'utilisateur n'a pas été clairement informé. On peut citer par exemple la collecte de données de position de l'utilisateur à des fins commerciales.

4.2.2 L'équité des algorithmes

Un algorithme est dit équitable si les résultats qu'il produit n'induisent pas un effet discriminant ou biais à l'égard d'une catégorie particulière de la population. La discrimination peut être introduite de manière consciente ou inconsciente, prévisible ou imprévisible au niveau de la conception de l'algorithme. Elle peut aussi être introduite pour un algorithme de machine learning au niveau de l'apprentissage sur un jeu de données et donc peut donc dépendre des discriminations intrinsèques aux données elles-mêmes. [15]

4.2.3 L'explicabilité des algorithmes

Un algorithme est dit explicable s'il est possible de donner à l'ensemble des utilisateurs, quel que soit leur bagage éducatif, une vision claire des procédures employées et des fonctionnalités remplies par l'algorithme. L'explicabilité est un enjeu non seulement pour s'adresser à un public sans aucun bagage technique en informatique, mais aussi à des personnes n'ayant pas les capacités psychologiques, les compétences cognitives ni même l'expérience de la vie d'un adulte. L'explicabilité est un enjeu majeur particulièrement pour les modèles de machine learning. Pour ces modèles, le créateur de l'algorithme peut être incapable d'expliquer comment son algorithme a effectué son apprentissage et comment il prend les décisions. Par exemple si ces algorithmes étaient utilisés dans des situations où ils devraient prendre des décisions importantes telles que conduire une voiture, accorder un prêt ou un entretien d'embauche, ou même refuser une mise en liberté, il serait parfois impossible de fournir la justification de la décision prise aux parties concernées. Dans ces cas-là, la prise de décision peut être vue comme une boite noire impossible à contester en dehors du rejet de cette méthode de machine learning. Ceci va à contre-courant des droits du citoyen de demande d'informations et de formulation de recours. Une telle situation n'est pas concevable d'un point de vue RGPD ; l'explicabilité doit donc être au centre des attentions lors de la création d'un algorithme de machine learning. [15]

4.2.4 L'intelligibilité des algorithmes

Un algorithme est dit intelligible s'il est possible au concepteur de comprendre son fonctionnement et de vérifier s'il satisfait bien les propriétés désirées. L'intelligibilité est une forme d'explicabilité fondamentale, qui porte sur la capacité des concepteurs à s'expliquer l'algorithme qu'ils conçoivent. L'intelligibilité constitue un enjeu majeur de la discipline informatique en général. Il n'est pas exagéré de dire que l'une des questions les plus fondamentales de l'informatique est "comment savoir si le programme fait bien ce qu'il est censé faire ?" Il se pose de manière particulièrement aiguë pour les IA, et en particulier les IA basées sur des mécanismes de machine learning. [15]
Ces concepts peuvent être regroupés en deux familles différentes. Les concepts d'explicabilité et d'intelligibilité renvoient à des propriétés épistémiques des algorithmes. Les concepts de loyauté et d'équité renvoient à des propriétés normatives ou prescriptives, de la relation entre créateur et utilisateur, et des effets sociaux des usages. On a donc affaire à deux familles de significations bien distinctes du concept de transparence : une famille descriptive renvoyant à des propriétés des algorithmes, une famille prescriptive renvoyant à des propriétés de leurs usages.

4.3 Compromis entre interprétabilité et efficacité

Les modèles de machine learning sont connus pour être plus efficace que les GLM en général pour la prédiction. Par ailleurs, derrière ce gain en efficacité, une perte importante en interprétabilité est observée. Ils vont beaucoup plus loin que les GLM dans la prédiction en arrivant à capter à une maille beaucoup plus fine certaines caractéristiques influentes. C'est pourquoi à l'échelle humaine, on considère ces modèles de machine learning comme boite noire. Par exemple les méthodes de forêts aléatoires, en faisant une agrégation d'arbre, perdent le côté interprétable de l'arbre. Il est quasiment impossible de retracer le chemin qui a mené à une prédiction en particulier de la même manière qu'avec un arbre. Lors du calibrage des paramètres de ces modèles boite noire, un jeu peut être effectué pour ne pas sélectionner les paramètres permettant la meilleure efficacité, mais plutôt ceux conservant un côté interprétable.

Le principe d'un modèle de machine learning est de laisser le modèle générer des règles lui-mêmes à partir de données d'apprentissages qui lui montrent les prises de décisions passées dans un certain contexte. Le modèle va chercher à apprendre de la meilleure des manières ces données.
Le principal problème vient des données qui peuvent contenir des biais qui vont être appris et propagés dans le modèle d'une façon nettement plus conséquente qu'avec les GLM. Ces biais sont difficiles à détecter car la plupart du temps ils sont inconscients.
Quelques exemples célèbres mettent en avant les problèmes de propagations de biais des modèles de machine learning :

- Le moteur de prédiction de récidive COMPAS (2016) utilisait indirectement l'origine ethnique des condamnés (notamment leur couleur de peau) comme expliquant leur propension à la récidive

- Le moteur de recrutement d'Amazon (2017) écartait d'office les femmes en reproduisant le recrutement passé
- L'usage de Watson (IA d'IBM) pour la recommandation de traitement du cancer (2018) a été jugé peu efficace, voire négatif car il ne prenait pas en compte les autres pathologies
- Le modèle de conduite autonome d'Uber a conduit à la mort d'un piéton (2018) à la suite d'une mauvaise reconnaissance de la part de l'IA qui avait jugé que ce piéton ne ressemblait pas un humain.
- Le chatbot évolutif censé apprendre des subtilités de langage en interagissant avec des humains (Microsoft 2016) : moins de 24h après sa mise en service, il faisait l'apologie du régime nazi après avoir été " entraîné " par des internautes.

Un autre exemple (source [3]) connu est celui du biais dans les images conduisant à de très bonnes performances en termes de score mais ayant produit en réalité un modèle totalement inutilisable. Comme illustration, un modèle de réseau de neurones convolutionnels (CNN) a été entraîné sur une base de 2000 photos composées d'avions et de motos extraites de la base de données Caltech 101. Ces données ont été biaisées volontairement en surpondérant les photos de motos présentant un encadrement blanc.

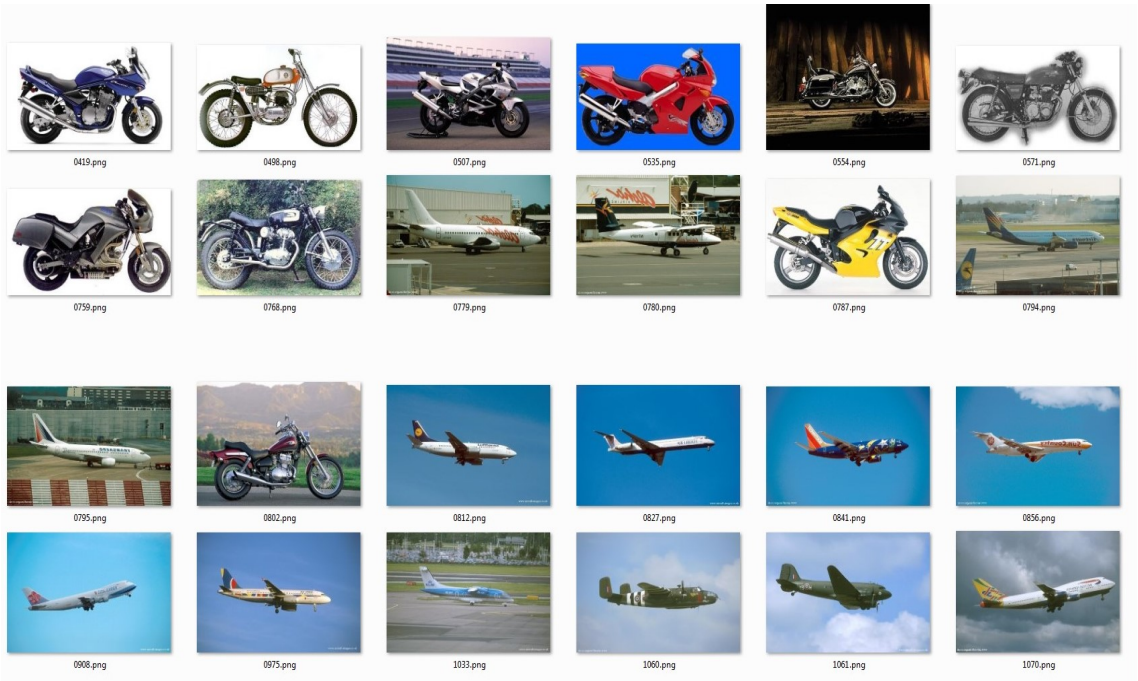


FIGURE 4.2 – Exemple de jeu de données biaisé

Le modèle affiche une précision de 94 % sur la base de test, ce qui semble indiquer qu'il distingue correctement les motos des avions même sur des photos qu'il n'a jamais observées auparavant. Pour s'assurer de sa stabilité, un algorithme d'interprétabilité (DeepLift) est appliqué pour détecter les pixels d'importance pour chaque photo. L'application sur une photo de moto est présentée ci-après :

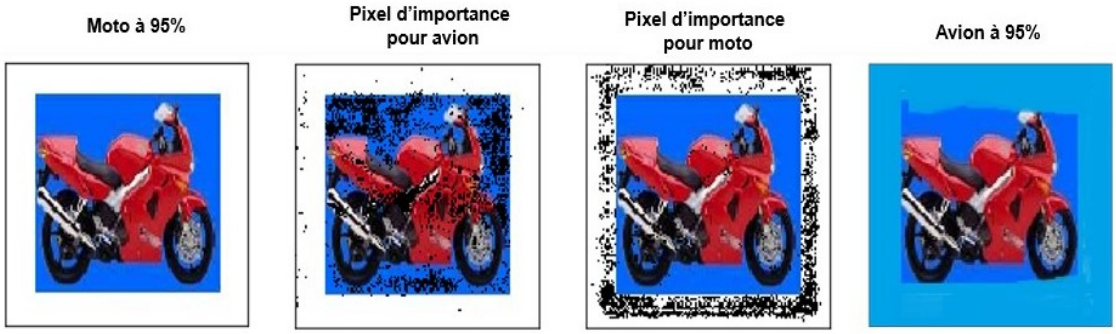


FIGURE 4.3 – Interprétabilité du modèle : DeepLift

Le modèle entraîné va considérer que cette image est bien une moto avec une probabilité de 95 %, donc ne pas se tromper. Par contre, l'application de l'algorithme de transparence a montré que pour cette photo les pixels ayant le plus d'impact dans la prédiction ne sont pas les roues ni d'autres parties de la moto, mais le cadre blanc présent autour de la moto. En perturbant ces pixels, le modèle devient "aveugle" et considère à 95 % que cette moto est en fait un avion. Le modèle a donc détecté le biais présent dans les données et s'en sert pour distinguer les avions des motos : il est donc inutilisable en pratique. Sans l'application du modèle d'interprétabilité et en se fiant uniquement à la performance de classification du modèle, ce phénomène n'aurait pas pu être détecté. Les algorithmes d'interprétabilité permettent de comprendre comment les décisions ont été prises et permettent donc de détecter la présence ou non de biais indésirables.

4.4 Définition mathématiques des méthodes transparences

Le tableau ci-dessous récapitule l'ensemble des méthodes de transparence d'algorithme qui seront étudiés dans cette section 4.4 :

Nom de la méthode	Article de recherche d'origine	Local/Global
Permutation feature importance (PFI) 4.4.1	L. Breiman (2001) [1]	Global
Partial dependant plot (PDP) F.1	Jerome H.Friedman (2001) [10]	Global
Individual conditional expectation (ICE) F.2	Alex Goldstein et al.(2015) [9]	Global
Accumulated local effect (ALE) 4.4.2	Daniel W.Apley (2016) [18]	Global
Valeur Shapley 4.4.3	Scott M.Lundberg (2017) [13]	Local et global

TABLE 4.1 – Revue des méthodes de transparence des modèles de machine learning

4.4.1 Permutation Feature Importance (PFI)

La PFI mesure l'augmentation de l'erreur de prédiction du modèle après avoir permuté les valeurs de la variable, ce qui brise la relation entre la variable et le résultat réel.

Théorie :

L'importance d'une variable est mesurée en calculant l'augmentation de l'erreur de prédiction

du modèle après avoir permuté la variable. Une variable est dite "importante" si le mélange de ses valeurs augmente l'erreur du modèle, car si tel est le cas, alors cela signifie que le modèle s'est bien servi de cette variable dans la prédiction. Une variable est dite "sans importance", si le mélange de ses valeurs laisse l'erreur du modèle inchangée, car si tel est le cas, alors le modèle a ignoré cette variable pour la prédiction. Cette notion a été introduite par Breiman (2001) [1] pour les forêts aléatoires. Sur la base de cette idée, Fisher, Rudin et Dominici (2018) ont proposé une version agnostique de l'importance des variables et l'ont appelée *model reliance*.

Algorithme de PFI basé sur Fisher, Rudin et Dominici :

Entrées : Modèle entraîné f, matrice de données X, vecteur cible y, fonction d'erreur $L(y, f)$

1. Estimation de l'erreur d'origine : $err^{origine} = L(y, f(X))$
2. pour chaque variable, $j = 1, ..., p$:
   - générer une matricer permutée : X^{perm} en permutant la variable j dans la matrice X. Cela rompt l'association entre la variable j et la vraie sortie y
   - estimer l'erreur : $err^{perm} = L(y, f(X^{perm}))$ basée sur les prédictions de la matrice permutée
   - calculer la PFI : $FI^j = \frac{err^{perm}}{err^{origine}}$. (Une autre alternative peut être aussi utilisée : $FI^j = err^{perm} - err^{origine}$)
3. trier les variables par ordre décroissant de FI

Avantages :

- La PFI fournit un aperçu global et compressé du comportement du modèle.
- Un aspect positif de l'utilisation du taux d'erreur au lieu de la différence d'erreur c'est que les mesures de PFI sont comparables entre les différents problèmes.
- La mesure de l'importance prend automatiquement en compte toutes les interactions avec d'autres variables. En permutant la fonction, les effets d'interaction avec d'autres variables sont également détruits. Cela signifie que l'importance de la variable de permutation prend en compte à la fois l'effet de la variable principale et les effets de l'interaction sur les performances du modèle. C'est également un inconvénient, car l'importance de l'interaction entre deux variables est incluse dans les mesures de l'importance des deux variables. Cela signifie que l'importance des variables ne correspond pas à la baisse totale des performances, mais que la somme est plus importante. Ce n'est que s'il n'y a pas d'interaction entre les variables, comme dans un modèle linéaire, que les importances s'additionnent approximativement.
- Les PFI ne nécessitent pas de ré-entraînement du modèle, seule la permutation des variables peut donc prendre du temps avec cette méthode.

Inconvénients :

- Il est très difficile de savoir si des données d'entraînement ou de test doivent être utilisées pour calculer l'importance de la variable.
- L'importance de la variable de permutation est liée à l'erreur du modèle. Ce n'est pas mauvais en soi, mais dans certains cas, ce n'est pas ce qui est nécessaire. Dans certains

cas, il peut être préférable de savoir dans quelle mesure la sortie du modèle varie pour une variable sans considérer ce que cela signifie pour les performances. Par exemple, si l'on souhaite savoir si la sortie du modèle est robuste lorsque qu'on manipule les variables. Dans ce cas, la diminution des performances du modèle lorsqu'une variable est permutée n'est pas intéressante, mais la part de la variance de sortie du modèle qui est expliquée par chaque variable l'est plus. La variance du modèle (expliquée par les variables) et l'importance des variables sont fortement corrélées lorsque le modèle se généralise bien (c'est-à-dire qu'il ne se sur-ajuste pas).

- La PFI dépend du mélange de la variable, ce qui ajoute un caractère aléatoire à la mesure. Lorsque la permutation est répétée, les résultats peuvent varier considérablement. Répéter la permutation et faire la moyenne des mesures d'importance sur les répétitions stabilise la mesure, mais augmente le temps de calcul.
- Si les variables sont corrélées, la PFI peut être biaisée par des instances de données irréalistes. L'ajout d'une variable corrélée peut diminuer l'importance de la variable associée en répartissant l'importance entre les deux variables.

4.4.2 Accumulated Local Effects (ALE)

Si les variables d'un modèle d'apprentissage automatique sont corrélées, le tracé de dépendance partielle ne peut pas être approuvé. Le calcul d'un graphique de dépendance partielle pour une variable qui est fortement corrélée avec d'autres variables implique de faire la moyenne des prédictions d'instances de données artificielles qui sont peu probables en réalité. Cela peut considérablement biaiser l'effet estimé de la variable.
Une idée pour corriger ce biais pourrait consister à calculer l'impact moyen au travers de la distribution conditionnelle, on appelle cela les M-plots :

$$\hat{f}_{x_S,M}(x_S) = \mathbb{E}_{X_C|X_S}\left[\hat{f}(X_S, X_C)|X_S = x_S\right] \tag{4.4.1}$$

$$= \int_{x_C} \hat{f}(x_S, x_C)\mathbb{P}(x_C|x_S)dx_C \tag{4.4.2}$$

Ainsi, les M-plots empêcheront de calculer les prédictions moyennes sur des couples invraisemblables. Par ailleurs, les M-plots ne résolvent que partiellement le problème des PDP car ils conduisent à mélanger le pouvoir prédictif de toutes les variables corrélées. Les ALE plot résolvent ceci en calculant sur la distribution conditionnelle des différences de prédictions et non plus des prédictions moyennes.

$$\hat{f}_{x_S,ALE}(x_S) = \int_{z_{0,1}}^{x_S} \mathbb{E}_{X_C|X_S}\left[\hat{f}^S(X_S, X_C)|X_S = z_S\right] dz_S - \text{constant} \tag{4.4.3}$$

$$= \int_{z_{0,1}}^{x_S} \int_{x_C} \hat{f}^S(z_S, x_C)\mathbb{P}(x_C|z_S)dx_C dz_S - \text{constant} \tag{4.4.4}$$

La formule révèle trois différences par rapport aux M-Plots. Premièrement, on fait la moyenne des changements des prévisions, et non des prévisions elles-mêmes. Le changement est défini comme le gradient (mais plus tard, pour le calcul réel il sera remplacé par les différences des

prédictions sur un intervalle).

$$\hat{f}^S(x_S, x_C) = \frac{\delta \hat{f}(x_S, x_C)}{\delta x_S}$$

La deuxième différence est l'intégrale supplémentaire sur z. Nous accumulons les gradients locaux sur la plage des variables de l'ensemble S, ce qui nous donne l'effet de la variable sur la prédiction. Pour le calcul proprement dit, les z sont remplacés par une grille d'intervalles sur laquelle on calcule les changements de prédiction. Au lieu de faire directement la moyenne des prédictions, la méthode ALE calcule les différences de prédiction conditionnelles aux variables de S et intègre la dérivée sur les variables de S pour estimer l'effet. La dérivée (ou différence d'intervalle) isole l'effet de la variable d'intérêt et bloque l'effet des variables corrélées. La troisième différence entre les graphiques ALE et les M-plots est que l'on soustrait une constante des résultats. Cette étape centre le tracé ALE afin que l'effet moyen sur les données soit nul. Toutefois, un problème demeure : tous les modèles de machine learning n'ont pas forcément un gradient comme par exemple les forêts aléatoires. Mais comme on le verra, le calcul proprement dit fonctionne sans gradients et utilise des intervalles.

Estimation :

La description de l'estimation sera faite seulement pour une seule variable numérique dans un premier temps. La variable est divisée en plusieurs intervalles et on calcule les différences entre les prédictions. Cette procédure approxime les gradients et fonctionne aussi pour les modèles sans gradient.

On estime d'abord l'effet non centré :

$$\hat{\tilde{f}}_{j,ALE}(x) = \sum_{k=1}^{k_j(x)} \frac{1}{n_j(k)} \sum_{i:x_j^{(i)} \in N_j(k)} \left[f(z_{k,j}, x_{-j}^{(i)}) - f(z_{k-1,j}, x_{-j}^{(i)}) \right] \tag{4.4.5}$$

Décomposition de la formule de droite à gauche : Le nom **accumulated local effect** reflète bien tous les composants individuels de cette formule. A la base, la méthode ALE calcule les différences de prédictions, grâce à quoi on remplace la variable d'intérêt par des valeurs z de la grille. La différence de prédiction est l'effet que la variable a pour une instance sur un certain intervalle. La somme de droite additionne les effets de toutes les instances dans un intervalle qui apparaît dans la formule comme voisinage $N_j(k)$. On divise cette somme par le nombre d'instances dans cet intervalle pour obtenir la différence moyenne des prédictions pour cet intervalle. Cette moyenne dans l'intervalle est couverte par le terme **Local** dans le nom ALE. Le symbole de somme de gauche signifie que l'on accumule les effets moyens sur tous les intervalles. L'ALE (non centré) d'une valeur de variable qui se trouve par exemple dans le troisième intervalle, est la somme des effets des premiers, deuxième et troisième intervalles. Le mot **accumulé** dans l'ALE en est le reflet.

Cet effet est centré de sorte que l'effet moyen soit nul :

$$\hat{f}_{j,ALE}(x) = \hat{\tilde{f}}_{j,ALE}(x) - \frac{1}{n} \sum_{i=1}^{n} \hat{\tilde{f}}_{j,ALE}(x_j^{(i)}) \tag{4.4.6}$$

La valeur de l'ALE peut être interprétée comme l'effet principal de la variable à une certaine valeur comparé à la moyenne de prédiction des données. Par exemple, une estimation d'ALE

de -2 pour $x_j = 3$ signifie que lorsque la j-ième variable a la valeur 3, alors la prédiction est inférieure de 2 par rapport à la prédiction moyenne.
Les quantiles de la distribution de la variable sont utilisés comme une grille qui définit les intervalles. L'utilisation des quantiles garantit qu'il y a le même nombre d'instances de données dans chacun des intervalles. Les quantiles ont l'inconvénient que les intervalles peuvent avoir des longueurs très différentes. Cela peut conduire à des tracés ALE étranges si la variable d'intérêt est très biaisée, par exemple de nombreuses valeurs faibles et seulement quelques valeurs très élevées.

ALE pour les variables catégorielles :

La méthode des ALE a besoin par définition que les valeurs des variables aient un ordre, car la méthode accumule les effets dans une certaine direction. Les variables catégorielles n'ont aucun ordre naturel. Pour calculer un tracé ALE pour une variable catégorielle, nous devons en quelque sorte créer ou trouver un ordre.
L'ordre des catégories influence le calcul et l'interprétation des ALE. Lorsqu'un ordre est clair pour les variables comme par exemple pour la variable "VALEUR_ASSUREE", alors le calcul de l'ALE est simple. Quand aucun ordre clair ne se dégage alors il faut réussir à classer les modalités en fonction de leur similitude avec les autres variables. La distance entre deux modalités est la somme des distances de chaque variable. La distance entre les éléments compare soit la distribution cumulative dans les deux modalités, également appelée distance de Kolmogorov-Smirnov (pour les éléments numériques), soit les tables de fréquences relatives (pour les éléments catégoriels). Une fois que les distances entre toutes les modalités sont toutes calculées, une mise à l'échelle multidimensionnelle est réalisée pour réduire la matrice de distance à une mesure de distance unidimensionnelle. Cela donne un ordre des catégories basé sur la similarité. Pour rendre cela un peu plus clair, voici un exemple : Supposons que l'on a les deux variables catégorielles "saison" et "météo" et une caractéristique numérique "température". Pour la première variable catégorielle (saison), on souhaite calculer les ALE. La variable a les modalités "printemps", "été", "automne", "hiver". On commence à calculer la distance entre les modalités "printemps" et "été". La distance est la somme des distances sur les variables température et météo. Pour la température, nous prenons toutes les instances avec la saison "printemps", nous calculons la fonction de distribution cumulative empirique puis nous faisons de même pour les instances avec la saison "été" et enfin nous mesurons leur distance avec la statistique Kolmogorov-Smirnov. Pour la fonction météo, nous calculons pour toutes les instances "printemps" les probabilités pour chaque type de temps puis nous faisons de même pour les instances "été" et ensuite nous additionnons les distances absolues dans la distribution de probabilité. Si "printemps" et "été" ont des températures et des conditions météorologiques très différentes, la distance totale par catégorie est grande. [14]

4.4.3 Valeur Shapley

Une prédiction peut être expliquée en supposant que chaque valeur de variables d'une instance est un "joueur" dans un jeu où la prédiction est le gain. Les valeurs Shapley, issues de la théorie des jeux, indiquent comment répartir équitablement le gain entre les variables. Pour comprendre

le principe des valeurs Shapley, on peut commencer par les étudier dans un cas maîtrisé tel que les modèles linéaires.

Intuition de la valeur Shapley avec les modèles linéaires :

Dans un modèle linéaire, il est facile de calculer les effets individuels. La prédiction d'un modèle linéaire pour une instance est la suivante :

$$\hat{f}(x) = \beta_0 + \beta_1 x_1 + ... + \beta_p x_p$$

où x est l'instance pour laquelle nous souhaitons calculer les contributions. Chaque x_j est une valeur de variables, avec $j \in \{1, ..., p\}$. β_j est le poids correspondant à la variable j. La contribution ϕ_j de la j-ième variable sur la prédiction $\hat{f}(x)$ est :

$$\phi_j(\hat{f}) = \beta_j x_j - \mathbb{E}(\beta_j X_j)$$

où $\mathbb{E}(\beta_j X_j)$ est l'estimation de l'effet moyen pour la variable j. La contribution est la différence entre l'effet de la variable moins l'effet moyen. Si l'on additionne toutes les contributions de variable pour une instance, le résultat est le suivant :

$$\sum_{j=1}^{p} \phi_j(\hat{f}) = \sum_{j=1}^{p} (\beta_j x_j - \mathbb{E}(\beta_j X_j)) \tag{4.4.7}$$

$$= (\beta_0 + \sum_{j=1}^{p} \beta_j x_j) - (\beta_0 + \sum_{j=1}^{p} \mathbb{E}(\beta_j X_j)) \tag{4.4.8}$$

$$= \hat{f}(x) - \mathbb{E}(\hat{f}(X)) \tag{4.4.9}$$

C'est la valeur prédite pour la donnée x moins la prédiction moyenne.

Est-il possible de faire de même pour n'importe quel type de modèle ? La valeur Shapley venant de la théorie des jeux coopératif est une solution pour calculer les contributions des variables pour des prédictions individuelles pour chaque modèle de machine learning.

Théorie sur la valeur Shapley :

La valeur Shapley d'une valeur de variable est sa contribution au gain, pondérée et additionnée sur toutes les combinaisons de valeurs de variables possibles :

$$\phi_j(val) = \sum_{S \subseteq \{x_1,...,x_p\} \setminus \{x_j\}} \frac{|S|!(p - |S| - 1)!}{p!} (val(S \cup \{x_j\}) - val(S)) \tag{4.4.10}$$

où S est un sous-ensemble des variables utilisées dans le modèle, x est le vecteur des valeurs de variables de l'instance à expliquer et p est le nombre de variables. $val_x(S)$ est la prédiction des valeurs de variables de l'ensemble S qui sont marginalisées par rapport aux variables non incluses dans l'ensemble S :

$$val_x(S) = \int \hat{f}(x_1, ..., x_p) d\mathbb{P}_{x \notin S} - \mathbb{E}_X(\hat{f}(X)) \tag{4.4.11}$$

La valeur Shapley est la seule méthode d'attribution qui satisfait les propriétés Efficacité, Symétrie, Dummy (Facticité) et Additivité, qui ensemble peuvent être considérées comme une définition d'un paiement équitable.

Efficacité : Les contributions des variables doivent correspondre à la différence de la prédiction pour x et la moyenne des prédictions.

$$\sum_{j=1}^{p} \phi_j(val) = \hat{f}(x) - \mathbb{E}_X(\hat{f}(X))$$

Symétrie : Les contributions de deux valeurs de variables j et k doivent être les mêmes si elles contribuent de manière égale à toutes les coalitions possibles. Si

$$val(S \cup \{x_j\}) = val(S \cup \{x_k\})$$

pour tous $S \subseteq \{x_1, ..., x_p\} \setminus \{x_j, x_k\}$, alors $\phi_j = \phi_k$

Dummy : Une variable j qui ne change pas la valeur prédite - quelle que soit la coalition de valeurs d'entité à laquelle elle est ajoutée - doit avoir une valeur Shapley de 0. Si

$$val(S \cup \{x_j\}) = val(S)$$

pour tous $S \subseteq \{x_1, ..., x_p\}$, alors $\phi_j = 0$

Additivité : Pour un jeu avec des gains combinés $val + val^+$, les valeurs Shapley respectives sont les suivantes : $\phi_j + \phi_j^+$

Supposons qu'une forêt aléatoire a été entraînée, ce qui signifie que la prédiction est une moyenne de plusieurs arbres de décision. La propriété d'Additivité garantit que pour une valeur de variable, on peut calculer la valeur Shapley pour chaque arbre individuellement, en faire la moyenne et obtenir la valeur Shapley pour la valeur de variable pour la forêt aléatoire.

Estimation de la valeur Shapley :

Toutes les coalitions possibles de valeurs de variables doivent être évaluées avec et sans j-ième entité pour calculer la valeur Shapley exacte. Pour plus qu'un petit nombre de variables, la solution exacte à ce problème devient problématique car le nombre de coalitions possibles augmente de façon exponentielle à mesure que de nouvelles variables sont ajoutées. Strumbelj et coll. (2014) [19] proposent une approximation grâce à un échantillonnage de Monte Carlo :

$$\hat{\phi}_j = \frac{1}{M} \sum_{m=1}^{M} (\hat{f}(x_{+j}^m) - \hat{f}(x_{-j}^m)) \tag{4.4.12}$$

où j est l'indice de la variable que l'on souhaite expliquer, $M \in \mathbb{N}$ est le nombre d'itérations choisi et $\hat{f}(x_{+j}^m)$ est la prédiction pour x, mais avec un nombre aléatoire de variables remplacées par un point z aléatoire, à l'exception de la valeur de la variable j. Le vecteur x_{-j}^m est presque identique à x_{+j}^m, mais la valeur x_j^m est aussi pris dans un échantillon z.

Estimation approximative de la valeur Shapley pour une seule variable :

- Variables requises : nombres d'itérations M, instance d'intérêt x, indice de la variable j, matrice de données X et le modèle de machine learning f
- Pour tout $m = 1, ..., M$:
  1. Tirer une instance aléatoire z de la matrice de données X
  2. Choisir une permutation aléatoire O dans l'ensemble des permutations possible
  3. Ordonner l'instance x : $x_o = (x_{(1)}, ..., x_{(p)})$
  4. Ordonner l'instance z : $z_o = (z_{(1)}, ..., z_{(p)})$
  5. construire deux nouvelles instances :
     — avec la variable j : $x_{+j} = (x_{(1)}, ..., x_{(j-1)}, x_{(j)}, z_{(j+1)}, ..., z_{(p)})$
     — sans la variable j : $x_{-j} = (x_{(1)}, ..., x_{(j-1)}, z_{(j)}, z_{(j+1)}, ..., z_{(p)})$
  6. calcul de la contribution marginale : $\phi_j^m = \hat{f}(x_{+j}) - \hat{f}(x_{-j})$
- Sortie : calcul de la valeur Shapley comme la moyenne : $\phi_j(x) = \frac{1}{M} \sum_{m=1}^{M} \phi_j^m$

Tout d'abord, on sélectionne une instance d'intérêt x, une variable j et le nombre d'itérations M. Pour chaque itération, une instance aléatoire z est sélectionnée parmi les données et un ordre aléatoire des variables est généré. Deux nouvelles instances sont créées en combinant les valeurs de l'instance d'intérêt x et de l'échantillon z. L'instance x_{+j} est l'instance qui nous intéresse, mais toutes les valeurs dans l'ordre avant la variable j sont remplacés par les valeurs de variables de l'échantillon z. L'instance x_{-j} est la même que l'instance x_{+j} mais en plus, la variable j est remplacée par la valeur de la variable j de l'échantillon z. La différence entre la prédiction et la boite noire est calculée, et enfin toutes ces différences sont moyennées pour donner la valeur Shapley estimée de la variable j pour l'instance x.

Avantages :

La différence entre la prédiction et la prédiction moyenne est équitablement répartie entre les valeurs des variables de l'instance grâce à la propriété d'Efficacité des valeurs Shapley. Cette propriété distingue la valeur Shapley des autres méthodes telles que LIME. LIME ne garantit pas que la prédiction est équitablement répartie entre les variables. La valeur Shapley est peut-être la seule méthode qui fournit une explication complète. Dans les situations où la loi exige l'explicabilité, la valeur de Shapley peut être la seule méthode juridiquement conforme, car elle est basée sur une théorie solide et distribue équitablement les effets.

La valeur Shapley permet des explications contrastées. Au lieu de comparer une prédiction à la prédiction moyenne de l'ensemble des données, on peut la comparer à un sous-ensemble ou même à un seul point des données. Cette contrastivité est également quelque chose que les modèles locaux comme LIME n'ont pas.

La valeur de Shapley est la seule méthode d'explication avec une théorie solide. Les axiomes efficacité, symétrie, facticité et additivité donnent à l'explication une base raisonnable. Des méthodes telles que LIME supposent un comportement linéaire du modèle d'apprentissage automatique localement, mais il n'y a pas de théorie expliquant pourquoi cela devrait fonctionner.

Inconvénients :
La valeur Shapley nécessite beaucoup de temps de calcul. Dans la plupart des cas seule la solution approximative est réalisable. Un calcul exact de la valeur de Shapley est coûteux en calcul car il y a 2^k coalitions possibles des valeurs des variables et l'absence d'une variable doit être simulée en tirant des instances aléatoires, ce qui augmente la variance pour l'estimation des valeurs de Shapley. Le nombre exponentiel des coalitions est traité en échantillonnant les coalitions et en limitant le nombre d'itérations M. La diminution de M réduit le temps de calcul, mais augmente la variance de la valeur Shapley. Il n'y a pas de bonne règle empirique pour le nombre d'itérations M. M doit être suffisamment grand pour estimer avec précision les valeurs de Shapley, mais suffisamment petit pour terminer le calcul dans un temps raisonnable. La valeur Shapley renvoie une valeur simple par variable, mais pas de modèle de prédiction comme LIME. Cela signifie qu'il ne peut pas être utilisé pour faire des déclarations sur les changements de prédiction pour les changements dans l'entrée, tels que : "Si je devais gagner 300 € de plus par an, ma cote de crédit augmenterait de 5 points."
Enfin, comme beaucoup d'autres méthodes d'interprétation basées sur la permutation, la valeur Shapley souffre de l'inclusion d'instances irréalistes lorsque les variables sont corrélées.

4.4.4 SHAP

SHAP (SHapley Additive exPlanations) de Lundberg et Lee ([13]) est une méthode pour expliquer les prédictions individuelles. SHAP est basée sur la théorie des jeux optimaux des valeurs Shapley.
Définition de SHAP Le but de SHAP est d'expliquer la prédiction d'une instance x en calculant la contribution de chaque variable à la prédiction. La méthode SHAP calcule les valeurs Shapley à partir de la théorie des jeux à coalition. Les valeurs de variable d'une instance agissent comme des joueurs dans une coalition. Les valeurs Shapley indiquent comment répartir équitablement le paiement (équivalent à la prédiction) entre les variables. Un joueur peut être une valeur de variable individuelle par exemple, ou également un groupe de valeurs de variables. SHAP apporte comme innovation le fait que la valeur Shapley est représentée comme une méthode additive d'attribution de variable, un modèle linéaire. SHAP spécifie l'explication de la manière suivante :

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j$$

où g est le modèle d'explication, $z' \in \{0,1\}^M$ est le vecteur de coalition, M est la taille maximale de la coalition et $\phi_j \in \mathbb{R}$ est l'attribution de variable pour une variable j. En ce qui concerne la coalition, une entrée 1 signifiera que la variable est "présente" et 0 que la variable est "absente". Rappelons que les valeurs Shapley sont la seule solution satisfaisant les propriétés d'efficacité, de symétrie, de facticité et d'additivité. SHAP les satisfait également car il calcule les valeurs Shapley. De plus SHAP décrit les trois propriétés suivantes :

Précision locale :

$$f(x) = g(x') = \phi_0 + \sum_{j=1}^{M} \phi_j x'_j$$

si l'on définit $\phi_0 = \mathbb{E}_X(\hat{f}(X))$ et qu'on fixe x_j à 1, c'est la propriété d'efficacité des valeurs Shapley. Seulement avec un nom différent et en utilisant le vecteur de coalition.

$$f(x) = \phi_0 + \sum_{j=1}^{M} \phi_j x'_j = \mathbb{E}_X(\hat{f}(X)) + \sum_{j=1}^{M} \phi_j$$

Termes manquants :

Les termes manquants indiquent que lorsqu'une variable est manquante, alors celle-ci obtient une attribution de 0. Notons bien que x'_j fait référence à la coalition où la valeur 0 représente l'absence d'une valeur de variable. Cette propriété ne fait pas partie de celles de base des valeurs Shapley classiques, pourquoi est-elle donc présente dans SHAP. Lundberg [12] appelle ceci une "petite propriété comptable". Une variable manquante pourrait en théorie avoir une valeur Shapley arbitraire sans nuire à la propriété de précision locale car elle est multipliée par $x'_j = 0$. La propriété de termes manquants impose aux variables manquantes d'obtenir une valeur Spaley égale à 0 ; en pratique cela n'est pertinent que si une variable est constante.

Cohérence :

Laisser $f_x(z') = f(h_x(z'))$ et $z'_{\setminus j}$ indique que $z'_j = 0$. Pour deux modèles f et f' qui satisfont :

$$f'_x(z') - f'_x(z'_{\setminus j}) \geq f_x(z') - f_x(z'_{\setminus j})$$

pour toutes les entrées $z' \in \{0,1\}^M$, alors :

$$\phi_j(f', x) \geq \phi_j(f, x)$$

La propriété de cohérence indique que si un modèle change de sorte que la contribution marginale d'une valeur de variable augmente ou reste la même (indépendamment des autres varaibles), la valeur Shapley augmente également ou reste la même.

TreeSHAP : (source [14])

Lundberg et al ([12]) ont proposé TreeSHAP une variante de SHAP pour les modèles d'apprentissage automatiques basés sur les arbres tels que les arbres de décisions, les forêts aléatoires et les arbres à gradient boosté (gradient boosting). TreeSHAP a été présenté comme une alternative rapide et spécifique au modèle KernelSHAP (modèle non étudié dans ce mémoire), mais il s'est avéré qu'il pouvait produire des attributions de variables non intuitives.

TreeSHAP définit la valeur de la fonction en utilisant l'espérance conditionnelle $\mathbb{E}_{X_S|X_C}(f(x)|x_S)$ au lieu de l'espérance marginale. Le problème avec l'espérance conditionnelle est que les variables qui n'ont aucune influence sur la fonction de prédiction f peuvent avoir une TreeSHAP estimée différente de 0. L'estimation différente de 0 peut apparaître quand la variable est corrélée avec une autre variable qui a de l'influence sur les prédictions.

Intuition du fonctionnement TreeSHAP :
Une intuition du mode de fonctionnement de TreeSHAP et du mode de calcul de la prédiction attendue pour un seul arbre, une instance x et un sous-ensemble de variables S sera donnée dans ce paragraphe. Si toutes les variables sont conditionnées, si S est l'ensemble de toutes les variables, alors la prédiction du nœud dans lequel tombe l'instance x serait la prédiction attendue. Si aucune condition n'est mise sur aucune variable, si S est vide, la moyenne pondérée des prédictions de l'ensemble des nœuds terminaux sera utilisée. Si S contient seulement certaines variables, mais pas toutes, les prédictions des nœuds inaccessibles sont ignorées. Inaccessible signifie ici que le chemin de décision qui mène à ce nœud contredit les valeurs dans x_S. A partir des nœuds terminaux restant, la moyenne des prédictions pondérée par la taille des nœuds (ie le nombre d'échantillons d'apprentissage dans ce nœud) est calculée. La moyenne des nœuds terminaux restants, pondérée par le nombre d'instances par nœud, est la prédiction attendue pour x sachant S. Le problème est que cette procédure doit être appliquée pour chaque sous-ensemble possible S des valeurs de variable. TreeSHAP calcule en temps polynomial contre un temps exponentiel pour KernelSHAP. L'idée est de pousser tous les sous-ensembles possibles S vers le bas de l'arbre en même temps. Pour chaque nœud de décision, une trace du nombre de sous-ensembles doit être conservée. Par exemple, lorsque la première division dans une arborescence est sur la variable x_3, tous les sous-ensembles qui contiennent la variable x_3 iront au nœud (celui où x va). Les sous-ensembles qui ne contiennent pas la variables x_3 vont au deux nœuds avec un poids réduit. Malheureusement, les sous-ensembles de tailles différentes ont des poids différents. L'algorithme doit garder une trace du poids global des sous-ensembles dans chaque nœud et cela complique clairement l'algorithme.

TreeSHAP possède les mêmes avantages que les valeurs Shapley de par sa construction. SHAP a de plus une implémentation rapide pour les modèles basés sur des arbres ce qui est un bon point étant donné que dans ce mémoire seront étudiés les forêts aléatoires et les gradient boostings basé sur les arbres.
Par ailleurs, TreeSHAP peut produire des attributions de variables contre-intuitives. Alors que TreeSHAP résout le problème de l'extrapolation à des points de données improbables, il introduit un nouveau problème. TreeSHAP modifie la fonction valeur en s'appuyant sur l'espérance conditionnelle prédite. Avec cette modification de la fonction valeur, les variables qui n'ont aucune influence sur la prédiction peuvent obtenir une valeur de TreeSHAP différente de 0.
Côté global de SHAP :
SHAP permettent d'obtenir une décomposition additive pour chaque individu du jeu de données. Une méthode classique consiste à tracer l'ensemble des valeurs shapley pour tous les individus du jeu de données et pour une variable prédictive fixée. Ainsi, de la même manière qu'avec les courbes ICE, il est possible de faire ressortir les effets globaux d'une variable prédictive sur l'ensemble du jeu de données en fonction de la valeur de cette variable prédictive. Pour des mêmes valeurs de cette variable prédictive, on peut donc observer sur ce type de graphique des contributions marginales complètement différentes.

4.5 Vérification de la transparence des modèles calibrés

Avant de commencer à défendre les nouveaux modèles de machine learning créés, revenons sur le processus qui a été mis en œuvre.

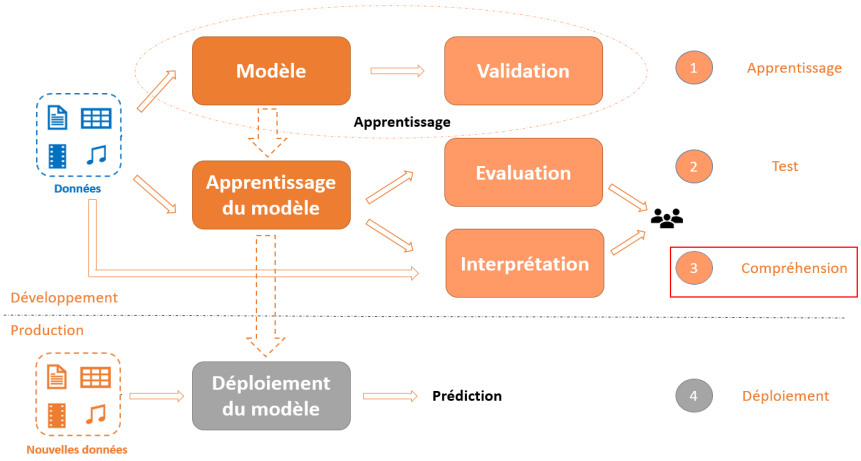
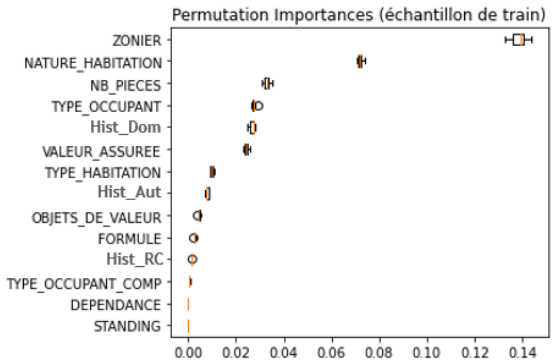


FIGURE 4.4 – Méthodologie machine learning

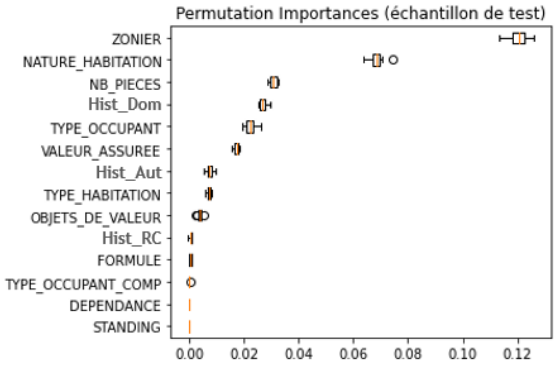
La partie d'apprentissage (1) et de test (2) ont déjà été effectuées, il ne reste donc désormais que la partie compréhension (3) en ce qui concerne le développement. Cette partie de compréhension est centrale, car elle permet de défendre les nouveaux modèles et de les analyser plus précisément. Il est d'ores et déjà possible de spécifier que les modèles créés pour la fréquence et les coûts vérifient le caractère *loyauté* des quatre principes énoncés au § 4.2. En effet, les algorithmes créent ne seront pas déloyaux dans le sens où ils prédiront des fréquences et des coûts seulement grâce aux variables tarifaires mises à disposition : il n'y a pas de méthode cachée en arrière-plan allant à l'encontre des futurs assurés.
Les graphiques mis en avant dans cette section seront ceux des modèles de gradient boosting aussi bien pour la fréquence que pour les coûts. Pour alléger le corps du mémoire, les graphiques correspondant aux modèles de forêts aléatoires seront placés en annexe H. Ces méthodes ont toutes été réalisées sur les années 2015, 2016 et 2017, c'est-à-dire la base d'apprentissage retenue lors de l'apprentissage du modèle.
Les méthodes de transparence d'ALE plot et de valeur Shapley ont été testées au préalablement en annexe G sur le GLM de fréquence pour vérifier que ces méthodes donnent des résultats cohérents. Ceci car avec les GLM il est possible de décomposer de manière multiplicative une prédiction faite et il est donc facile de connaître l'impacte d'une modalité dans la prédiction.

4.5.1 PFI

Une première analyse consiste à savoir l'importance de chaque variable dans la prédiction des modèles créés. Grâce à l'importance des variables, il sera désormais possible de déterminer quelles variables impacte le plus la prédiction de la fréquence et la prédiction des coûts. Grâce à cela, on saura donc que la prime pure sera en partie prédite en fonction des modalités de ces variables importantes.
Importance des variables pour la fréquence :



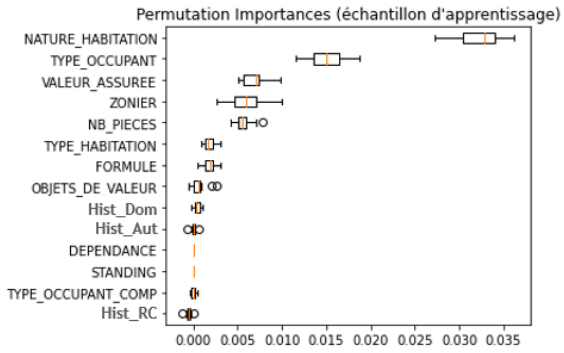
(a) Modèle de fréquence : Importance des variables sur les données d'apprentissage



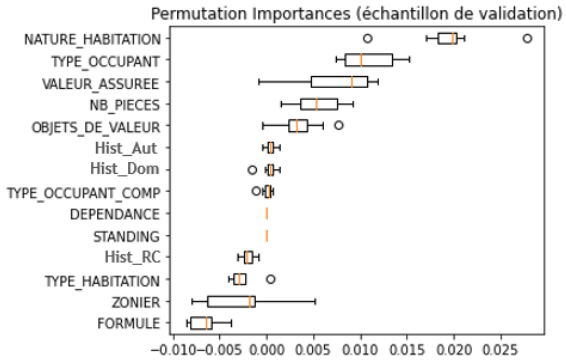
(b) Modèle de fréquence : Importance des variables sur les données de test

Les PFI pour le modèle de gradient boosting sur les fréquences indiquent que les trois variables les plus importantes sont la zone géographique, la nature d'habitation et le nombre de pièces. Ceci se vérifie aussi bien sur les données de test que sur les données d'apprentissage, ce qui est un bon point sachant que pour les PFI il est toujours délicat de savoir sur quel échantillon calculer les importances de variables. Les variables indiquant le montant de valeur assurée, le type d'occupant et Hist_Dom semblent aussi être importantes, mais restent un peu en retrait sur les échantillons d'apprentissage et de validation par rapport aux trois principales.
Le gradient boosting aussi bien que les forêts aléatoires désignent comme variables importantes la nature d'habitation, la zone géographique et Hist_Dom. Pour ces deux modèles, la valeur assurée et le nombre de pièces font aussi partie des variables importantes sur les données de test et d'apprentissage.

Importance des variables pour les coûts :



(a) Modèle de coûts : Importance des variables sur les données d'apprentissage



(b) Modèle de coûts : Importance des variables sur les données de test

Les PFI pour le modèle de gradient boosting sur les coûts indiquent que les trois variables les plus importantes sont la Nature d'habitation, le Type d'occupant et le montant en valeur assurée. Pour les données d'apprentissage, les PFI semblent moins dispersées que pour les données de test, ceci est dû au fait que le modèle ait été appris sur les données d'apprentissage. Le montant en objets de valeur semble aussi être une variable importante, mais reste en retrait des trois autres variables citées précédemment.

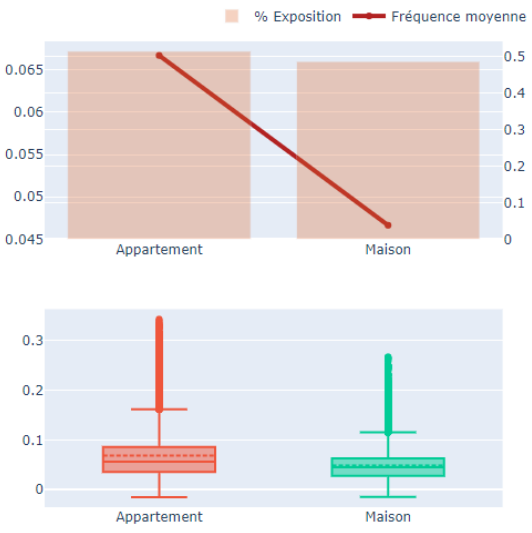
Les résultats obtenus pour les modèles de gradient boosting et de forêts aléatoires ont pour point commun de désigner tous les deux comme variable importante la nature d'habitation, le type d'occupant et le montant de valeur assurée. Le montant en objet de valeur semble aussi important pour les deux modèles en faisant à chaque fois partie du top cinq des variables importantes.

4.5.2 Tracé des prédictions

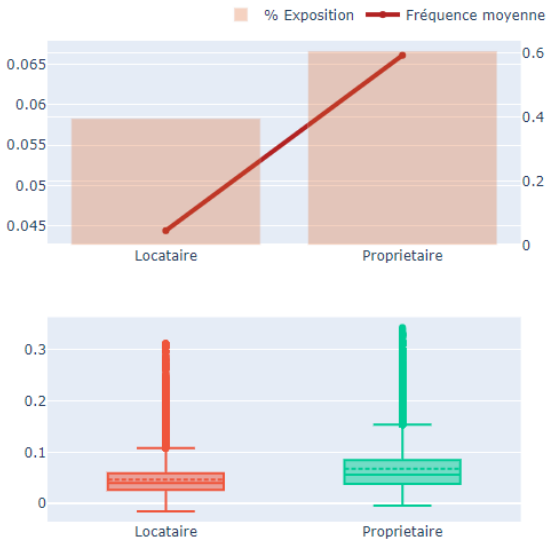
Dans un second temps, on va comparer les prédictions des modèles par modalité des variables tarifaires aux fréquences réelles et coûts réels par modalité. Pour cela nous nous attarderons seulement sur les variables les plus importantes définies précédemment, d'un côté pour plus de clarté et éviter d'encombrer le mémoire, et d'un autre coté car il est primordial que les modèles captent bien les effets des variables importantes. Si une même tendance s'observe en fonction des modalités, alors les modèles seront cohérents et auront dans l'ensemble capté le pouvoir prédictif de la variable tarifaire. Cette analyse des tracés des prédictions pourra notamment servir de réponse au caractère d'*équité* souhaité pour un algorithme. En effet si certaines modalités des variables tarifaires impliquent une prédiction des coûts moyens élevés en comparaison aux autres modalités, mais que ceci se vérifie dans le même temps sur les observations du portefeuille ; alors cet algorithme ne pourra pas être jugé inéquitable.

Les graphiques présentés ci-dessous sont divisés en deux parties, la première partie représente la distribution des modalités ainsi que la moyenne de la fréquence ou du coût pour une modalité fixée. La deuxième partie est un box-plot des prédictions des coûts ou des fréquences pour une modalité fixée.

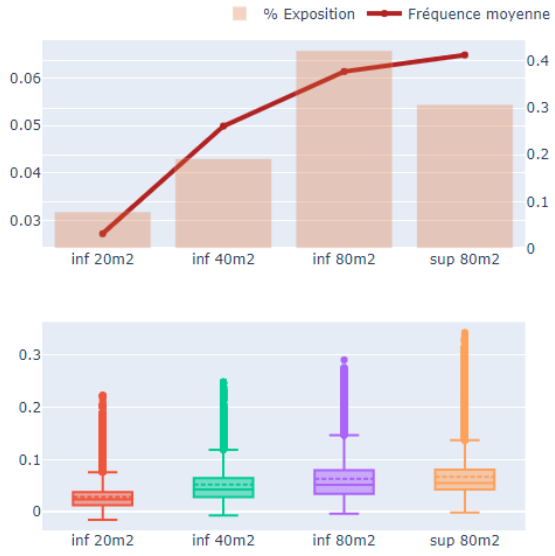
Tracé des prédictions pour le modèle de fréquence :



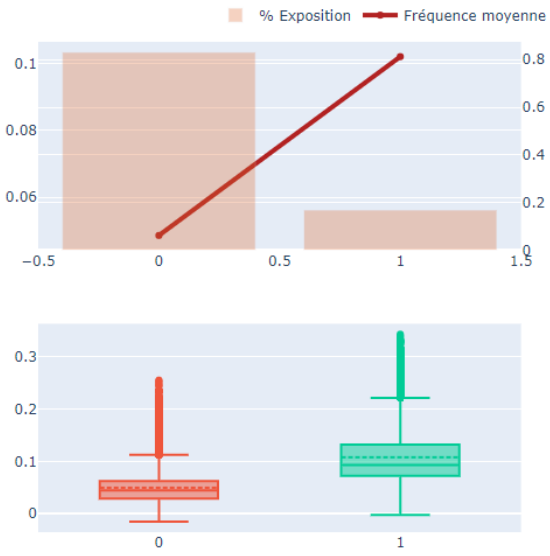
(a) Nature d'habitation : Prédictions vs valeurs réelles



(b) Type d'occupant : Prédictions vs valeurs réelles



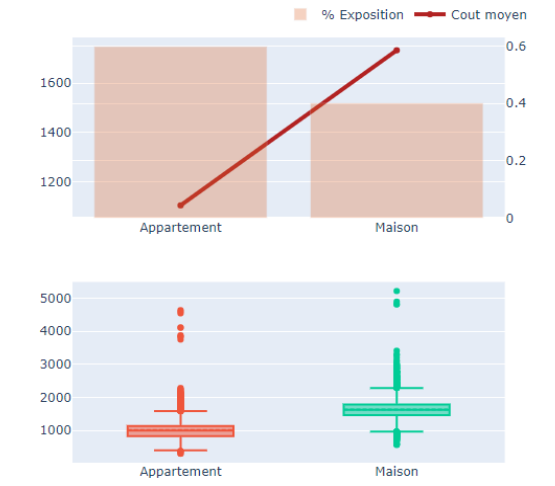
(a) Nombre de pièces : Prédictions vs valeurs réelles



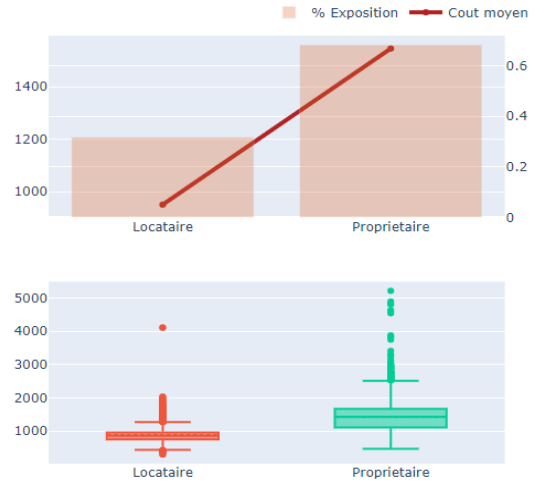
(b) Hist_Dom : Prédictions vs valeurs réelles

Pour les quatre variables les plus importantes du modèle de gradient boosting, les prédictions vont dans le même sens que les valeurs réelles. En particulier pour la valeur assurée, la même tendance linéaire croissante s'observe. Pour les forêts aléatoires (H.2), le même constat s'opère.

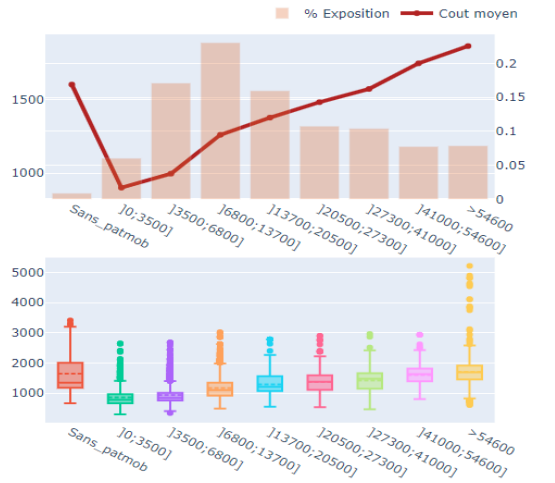
Tracé des prédictions pour le modèle de coûts :



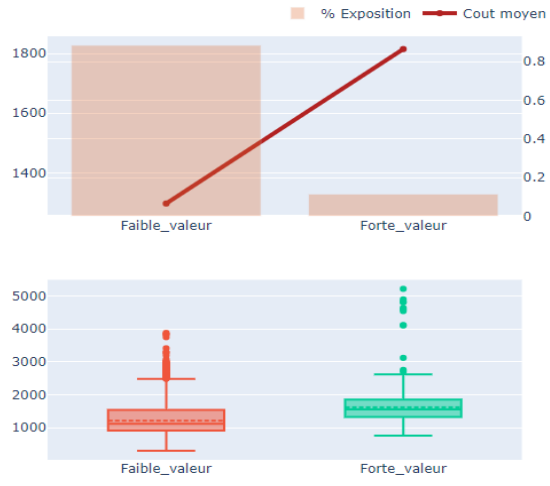
(a) Nature d'habitation : Prédictions vs valeurs réelles



(b) Type d'occupant : Prédictions vs valeurs réelles



(c) Valeurs assurée : Prédictions vs valeurs réelles



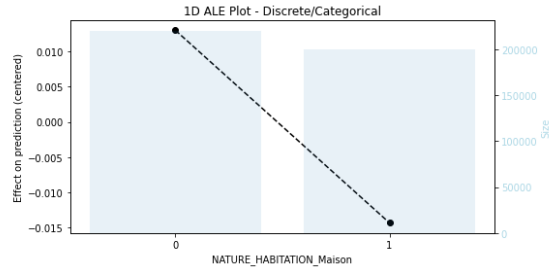
(d) Montant en objets de valeur : Prédictions vs valeurs réelles

Pour les quatre variables choisies, le modèle de gradient boosting semble faire les prédictions dans le même sens que les valeurs réelles. Pour le modèle de forêts aléatoires (H.2), le même constat s'opère.

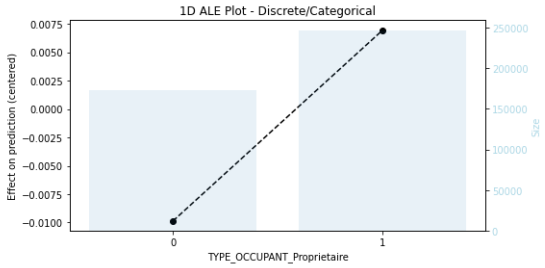
4.5.3 ALE plot

Les graphiques ci-dessous sont ceux qui vont le plus nous intéresser étant donné que comme énoncé en 4.4.2, c'est la seule méthode globale qui arrive à gérer d'une façon correcte les variables corrélées. Bien que les variables à disposition dans le portefeuille ne soient pas à 100 % corrélées ou en correspondance (V Cramer faible, voir 1.1), cette correspondance n'en reste pas moins différente de 0. Etant donné que des ALE plot ont été tracés, bien que réalisés, les PDP et les ICE plot ne seront pas affichés, car comme cela a été expliqué dans la partie théorique, les ALE plot gère mieux les corrélations.

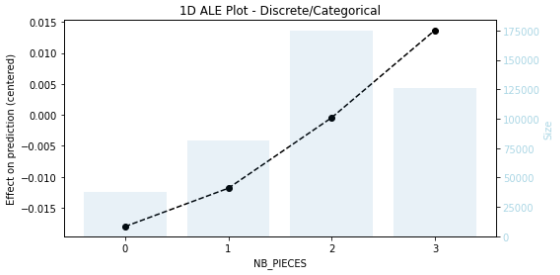
Tracé des prédictions pour le modèle de fréquence :



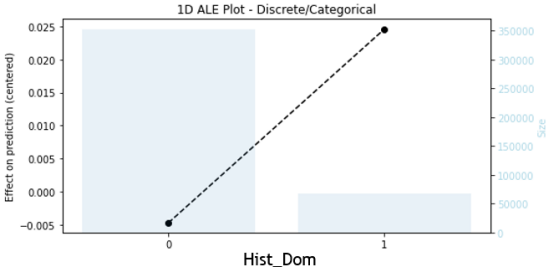
(a) Nature d'habitation



(b) Type d'occupant



(c) Nombre de pièces



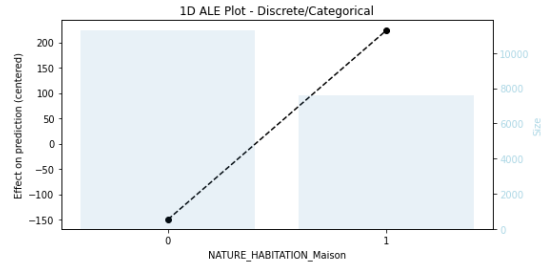
(d) Hist_Dom

FIGURE 4.10 – ALE plot : modèle de fréquence

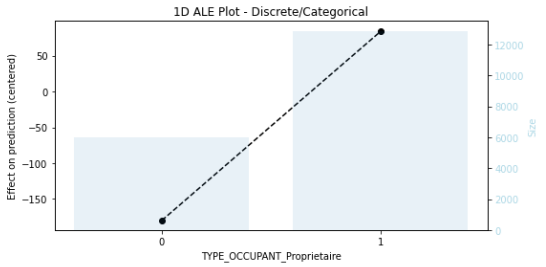
Les ALE plot pour le modèle de fréquence nous informent des effets suivant :

- Nature d'habitation : l'effet moyen d'être en Maison sur la prédiction des fréquences sera de -0.015, contre +0.010 pour un individu étant en appartement.
- Type occupant : l'effet moyen d'être propriétaire sur la prédiction de la fréquence sera de +0.0075, contre -0.010 pour un individu étant en location.
- Hist_Dom : l'effet moyen sur la prédiction d'avoir eu un sinistre les années passées est de +0.025, contre -0.005 pour un individu n'ayant pas eu de sinistre récemment.
- Nombres de pièces : l'effet moyen du nombre de pièces sur la prédiction de la fréquence varie de -0.015 pour une déclaration *inf 20 m2* à +0.015 pour la plus haute tranche *sup 80m2*.

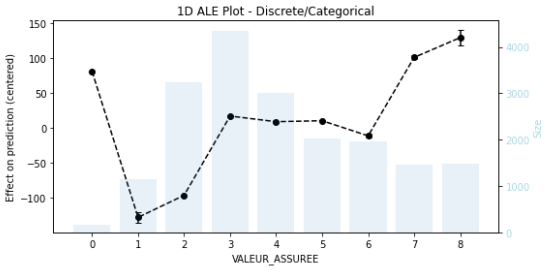
Tracé des prédictions pour le modèle de coûts :



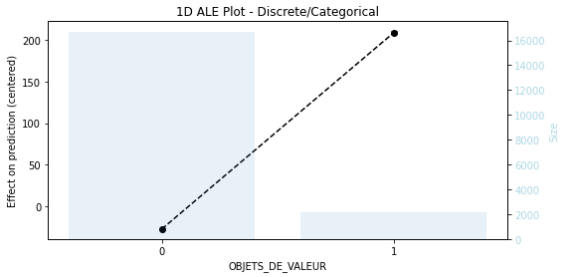
(a) Nature d'habitation



(b) Type d'occupant



(c) Valeur assurée



(d) Montant en objets de valeur

FIGURE 4.11 – ALE plot : modèle de coûts

Les ALE plot pour le modèle de coûts nous informent des effets suivants :

- Nature d'habitation : l'effet moyen d'être en Maison sur la prédiction des coûts sera de +200, contre -150 pour un individu étant en appartement.
- Type occupant : l'effet moyen d'être propriétaire sur la prédiction des coûts sera de +70, contre -150 pour un individu étant en location.
- Valeur assurée : l'effet de la valeur assurée sur la prédiction des coûts s'observe principalement pour les classes extrêmes. Appartenir à la classe *>54 600* de valeur assurée, va avoir un effet moyen de +150 sur la prédiction moyenne d'après le graphique. L'ALE plot nous montre par ailleurs la même tendance que sur les valeurs réelles.
- Objets de valeur : l'effet moyen d'avoir un fort montant en objet de valeur sur la prédiction des coûts sera de +200, contre un effet moyen presque nul pour un montant en objet de valeur faible déclaré.

Le modèle de gradient boosting semble plus pénaliser l'absence de déclaration de montant de valeur assurée (effet moyen : +100, voir H.3) que les forêts aléatoires (effet moyen -50). Par ailleurs, en comparant avec les données réelles, il semble que le gradient boosting ait mieux capté l'effet de cette modalité.

4.5.4 SHAP

Pour le modèle de fréquence et le modèle de coûts, les valeurs shapley ont été étudiées de façon locale et de façon global. Pour l'approche globale, nous nous sommes servis des variables les plus importantes trouvées en § 4.5.1, de la même manière que pour l'étude des scores par modèles

sur les couples de variables. Nous avons décidé de tracer l'impact de chaque variable pour un individu moyen dans le sous-ensemble où les modalités des variables les plus importantes ont été fixées. Une fois les modalités des trois variables les plus importantes fixées, nous avons pris l'individu le plus représentatif de ce sous-ensemble.

Étude du modèle de fréquence :

Pour l'étude de la fréquence, les variables les plus importantes sont les suivantes : Nature d'habitation, Hist_Dom, et type d'occupant.

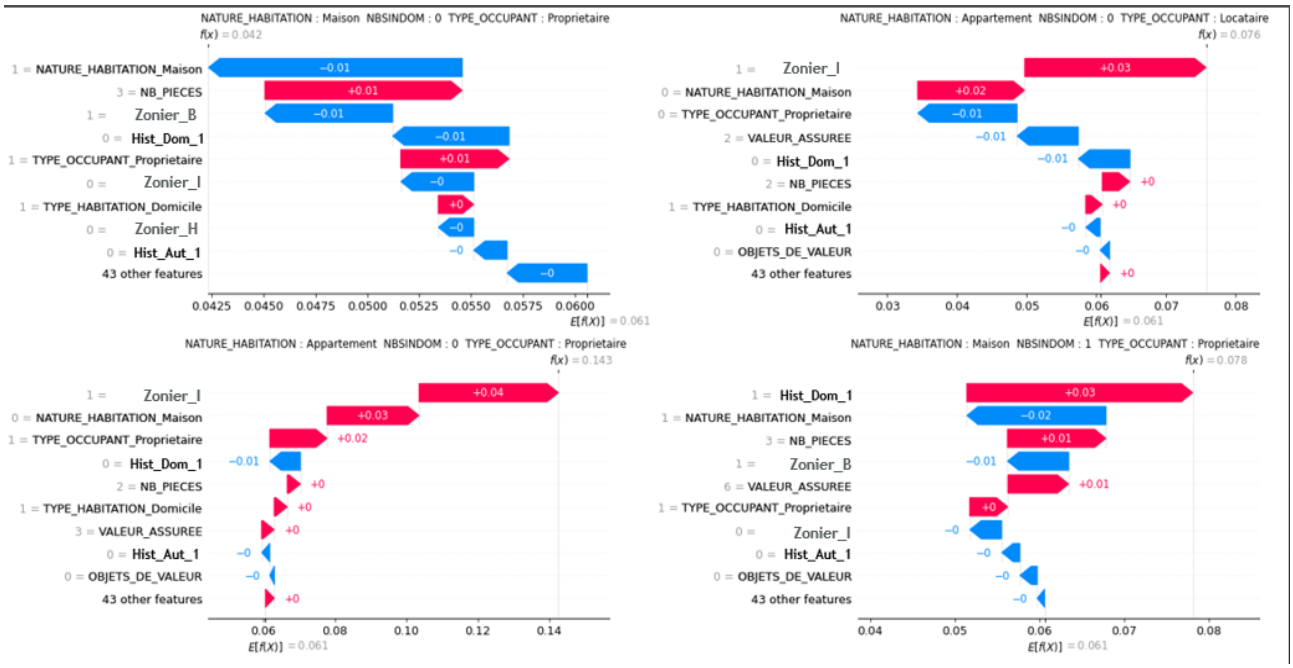


FIGURE 4.12 – Impact des différentes variables sur la prédiction de la fréquence pour 4 individus représentatifs

Ce graphique met bien en évidence l'effet des 10 variables les plus importantes sur la prédiction de la fréquence. On retrouve comme sur les PFI des modèles de fréquences (4.5a), que la nature d'habitation, le nombre de pièces, le type d'occupant et Hist_Dom ont un poids important dans la prédiction de la fréquence. De plus, on observe que les maisons impactent à la baisse la fréquence ($\approx$ -0.01) alors que les appartements impactent à la hausse la fréquence ($\approx$ +0.02, dans des ordres de grandeurs similaires à ceux observés (rappel : -0.015 et +0.01) sur les ALE plot (4.10). De la même manière, un Hist_Dom de 1 va augmenter la prédiction de la fréquence ($\approx$ 0.03), ce qui s'observe encore sur l'ALE plot de la fréquence (rappel : + 0.025). Pour le type d'occupant et le nombre de pièces, ces remarques se vérifient aussi. Ceci prouve donc bien la consistance des shap values. Grâce à ce graphique, on arrive à comprendre comment le modèle arrive à de telles prédictions, et aussi pourquoi certaines prédictions ont été faites, ie quelles variables ont influencé la prédiction faite.

Un autre graphique intéressant à observer et crée grâce au package SHAP est le résumé de l'impact de chaque variable sur la prédiction en fonction de sa valeur.

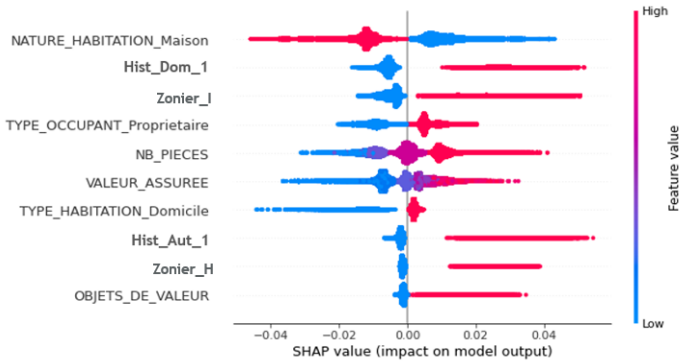


FIGURE 4.13 – Summary plot : modèle de fréquence

Analyse variable par variable :

- Nature d'habitation : une maison aura un impact à la baisse sur la prédiction de la fréquence (valeur shapley <0)
- Hist_Dom : la présence de sinistre dans un historique de trois ans aura un impact à la hausse sur la prédiction de la fréquence (valeur shapley >0). Pour l'absence de sinistre dans un historique de trois ans, impact à la baisse, mais avec une forte concentration de valeur shapley proche de 0.
- Type d'occupant : être propriétaire semble impacter la prédiction de la fréquence à la hausse.
- Nombre de pièces : avoir un nombre de pièces faible ou élevé a un impact respectivement à la baisse et à la hausse sur la prédiction de la fréquence. Avoir un nombre de pièces "moyen" n'a pas d'impact marqué (valeurs shapley centrées en 0).

Ces impacts à la hausse ou à la baisse observés vont tous dans le même sens que les valeurs réelles observées en section 1.3.3. Grâce à cela, on peut donc juger que le modèle calibré est transparent et réalise bien les prédictions attendues, ie il est dit intelligible.

Pour appuyer le choix pris de ne tracer l'impact des shap values seulement sur un individu après avoir fixé les trois variables les plus importantes, nous avons observé le poids des variables peu importantes sur le sous-ensemble de données des individus ayant les trois variables les plus importantes avec une modalité fixée en particulier. Pour cela, le poids de l'ensemble des variables moins celui des 9 les plus importantes (celles affichées sur le graphique 4.12) a été tracé ci-dessous :

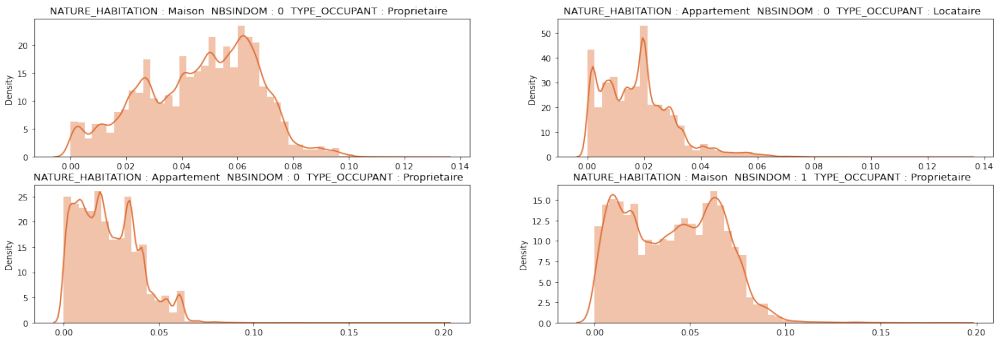


FIGURE 4.14 – Résidus shap values : modèle de fréquence

On observe donc que le poids des variables les moins influentes est quasiment négligeable dans la prédiction, il représente en moyenne 3 %. Si on enlève seulement les trois variables les plus importantes, ce montant monte à 10 %. Pour donner une prédiction approximative de la fréquence, il suffit donc de se focaliser sur les variables les plus importantes.

Étude du modèle de coûts :

Pour l'étude de la sévérité, les variables les plus importantes sont les suivantes : Nature d'habitation, le type d'occupant le montant de valeur assurée. Les graphiques analysés seront les mêmes que pour la fréquence.

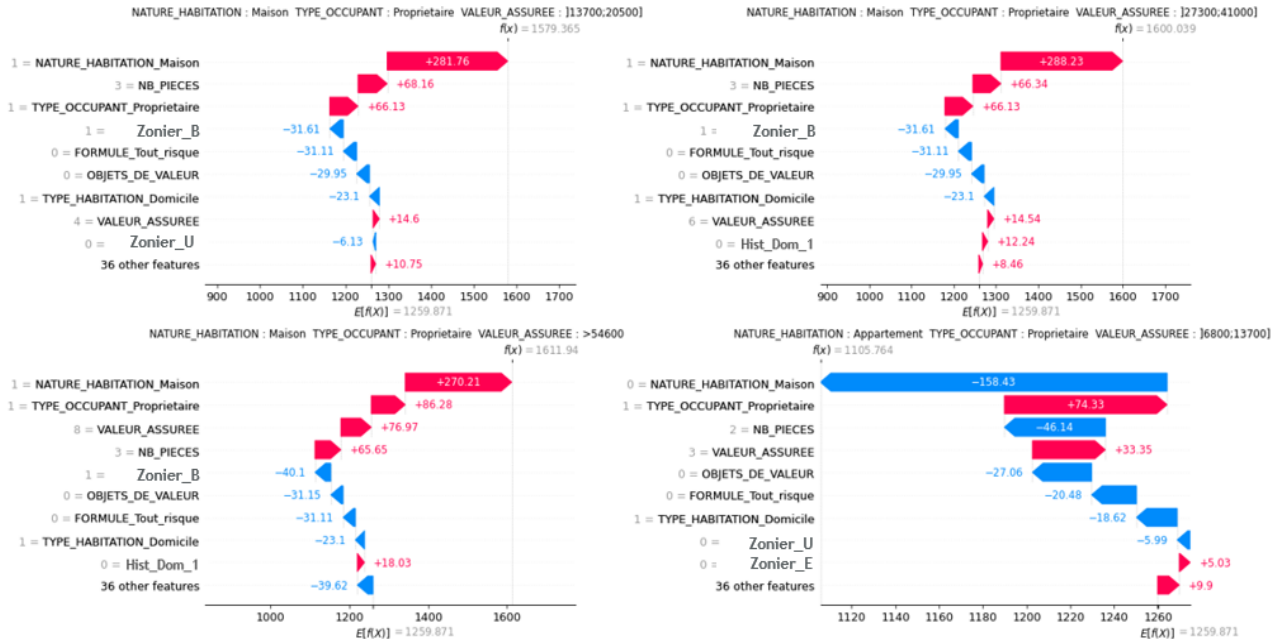


FIGURE 4.15 – Impact des différentes variables sur la prédiction des coûts pour 4 individus représentatifs

De manière encore plus marquée que pour le modèle de fréquence, l'impact des variables les plus influentes est prépondérant par rapport aux variables mineures. La nature d'habitation contribue dans la prédiction des coûts de manière nettement plus importante que les autres variables. De plus, on observe que les maisons impactent à la hausse (≈ 270) alors que les appartements impactent à la baisse (≈ 150) la fréquence dans des ordres de grandeurs équivalents (rappel : +250 et -150) à ceux observés sur les ALE plot (4.11). De la même manière, être propriétaire va augmenter la prédiction du coût (≈ 70), ce qui s'observe encore sur l'ALE plot des coûts (rappel : + 100). Pour le montant de valeur assurée ces remarques se vérifient aussi. Ceci prouve donc bien la consistance des shap values.

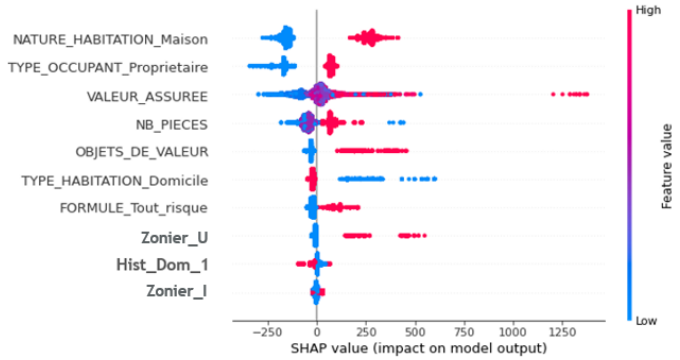


FIGURE 4.16 – Summary plot : modèle de coûts

Analyse variable par variable :

- Nature d'habitation : une maison aura un impact à la hausse sur la prédiction des coûts (valeur shapley >0)
- Type d'occupant : être propriétaire semble impacter la prédiction des coûts à la hausse. Par ailleurs, l'impact à la hausse est plus concentré contre un impact négatif très dispersé pour les locataires.
- Objets de valeur : avoir un fort montant en objet de valeur augmente la prédiction des coûts.
- Valeur assurée : pour les classes extrêmes, l'effet est marqué et est positif ou négatif pour respectivement les fortes valeurs assurées et les faibles valeurs assurées. Pour les valeurs assurées intermédiaires, aucun effet clair ne se dégage.
- Nombre de pièces : Un nombre de pièces élevé aura un impact faible à la hausse.

De la même manière que pour le modèle de fréquence, on affiche ci-dessous le poids des variables les moins influentes dans la prédiction des coûts.

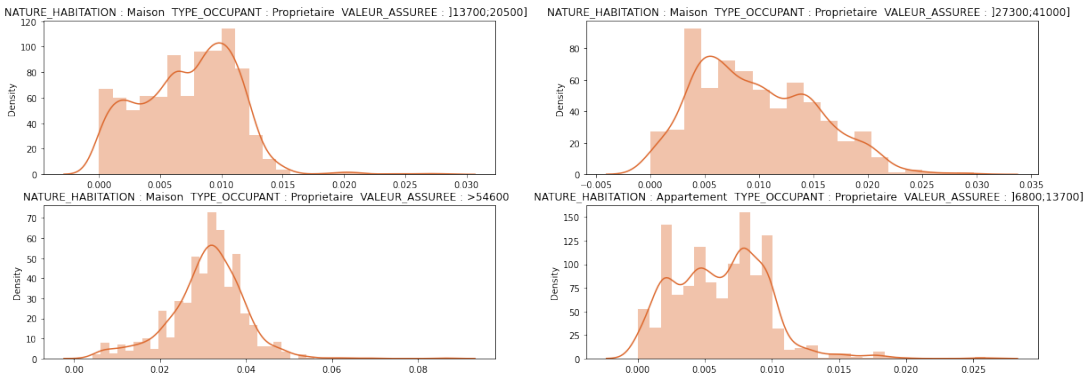


FIGURE 4.17 – Résidus shap values : modèle de coûts

On observe donc que le poids des variables les moins influentes est quasiment négligeable dans la prédiction, il représente en moyenne 1 % et cela pour les 4 classes d'individus observés. Si on enlève seulement les trois variables les plus importantes, ce montant monte à 5 %. Pour donner une prédiction approximative des coûts, on peut donc de se focaliser sur les variables les plus importantes. Par rapport à la fréquence, la prédiction des coûts dépend de beaucoup moins de variables.

Après avoir étudié la transparence des modèles de fréquence et de coûts, il s'avère que une partie des variables, les plus importantes sont communes aux deux modèles. Dans l'étude de la transparence de la prime pure (coûts x fréquence), on peut d'ores et déjà avoir une idée des variables qui seront les plus influentes.

4.5.5 SHAP pour la prime pure

4.5.5.1 Intérêt de SHAP pour la prime pure et théorie

Grâce à la section précédente (§ 4.5.4), il est possible d'observer pour quatre individus représentatifs la prédiction de la fréquence et des coûts de ceux-ci et comment les variables prédictives vont impacter cette prédiction. Mais une question se pose, peut-on obtenir une telle méthode de visualisation pour la prime pure ? Oui, il s'agit de coupler les valeurs shapley issues des modèles de fréquence et de coûts pour obtenir des valeurs shapley de prime pure.
Rappelons qu'avec les valeurs shapley, une prédiction se décompose de la manière suivante :

$$\hat{f}(x) = \mathbb{E}(\hat{f}(X)) + \sum_{j=1}^{p} \phi_j \tag{4.5.1}$$

où $\mathbb{E}(\hat{f}(X))$ est la prédiction moyenne du modèle f, et ϕ_j sont les valeurs shapley associées aux variables j, et p le nombre de variables prédictives. Ici, il s'agit des valeurs shapley associées aux variables *one-hot encodées*, donc une quarantaine de valeurs shapley sont données pour chaque instance.
Du fait que les prime pures se calculent comme le produit de la fréquence prédite fois le coût moyen prédit, on peut retrouver les valeurs shapley associées à chaque variable pour la prime pure. Décomposons ceci ci-dessous :

$$\hat{PP}(x) = (\mathbb{E}(\hat{f}_{freq}(X)) + \sum_{j=1}^{p} \phi_j) \times (\mathbb{E}(\hat{f}_{cots}(X)) + \sum_{j=1}^{p} \psi_j) \tag{4.5.2}$$

où ϕ_j et ψ_j correspondent respectivement aux valeurs shapley pour la fréquence et les coûts.
L'ensemble des composantes du produits sont présents dans la matrice ci-dessous (on note $\mathbb{E}(\hat{f}_{freq}(X))$ par ϕ et $\mathbb{E}(\hat{f}_{cots}(X))$ par ψ pour plus de lisibilité) :

$$\begin{bmatrix} \phi\psi & \phi_1\psi & \phi_2\psi & \dots & \phi_{p-1}\psi & \phi_p\psi \\ \phi\psi_1 & \phi_1\psi_1 & \phi_2\psi_1 & \dots & \phi_{p-1}\psi_1 & \phi_p\psi_1 \\ \phi\psi_2 & \phi_1\psi_2 & \phi_2\psi_2 & \dots & \phi_{p-1}\psi_2 & \phi_p\psi_2 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \phi\psi_{p-1} & \phi_1\psi_{p-1} & \phi_2\psi_{p-1} & \dots & \phi_{p-1}\psi_{p-1} & \phi_p\psi_{p-1} \\ \phi\psi_p & \phi_1\psi_p & \phi_2\psi_p & \dots & \phi_{p-1}\psi_p & \phi_p\psi_p \end{bmatrix}$$

Où les éléments sont groupés par couleur en fonctions de la variable avec laquelle ils interagissent :

- rouge : valeur de base, prime moyenne
- bleu : la somme des éléments bleus représentent l'effet de la première variable sur la prime moyenne.
- vert : la somme des éléments verts représentent l'effet de la deuxième variable sur la prime moyenne.
- violet : la somme des éléments violets représentent l'effet de la p-ième variable sur la prime moyenne.

Note : seules les interactions directes sont en couleur, les interactions croisées de deux variables sont sommées par couple de variables et ajoutées de la même manière.
Grâce à cette méthode, on dispose désormais des valeurs shapley pour chaque individu pour chaque variable présente dans la base, et pour chaque couple de variables présent dans la base ($\binom{p}{2}$ couples). Il n'y aura donc avec cette méthode plus p valeurs shapley, mais $p + \binom{p}{2}$ variables.

4.5.5.2 Application de la méthode au modèle de gradient boosting

Pour analyser les valeurs shapley pour la prime pure, nous allons tracer l'impact des variables sur la prime pure sur des individus représentatifs de la même manière que ce qui a été réalisé sur le graphique 4.12 par exemple. Pour cela, les individus représentatifs ont été sélectionnés dans le top 3 des variables les plus influentes pour le modèle de fréquence et pour le modèle de coûts. Les couples Nature d'habitation, Hist_Dom et type d'occupant (fréquence) et Nature d'habitation, type d'occupant et montant de valeur assurée (coûts) seront testés.

Impacte des variables importantes du modèle de fréquence sur le prime pure :

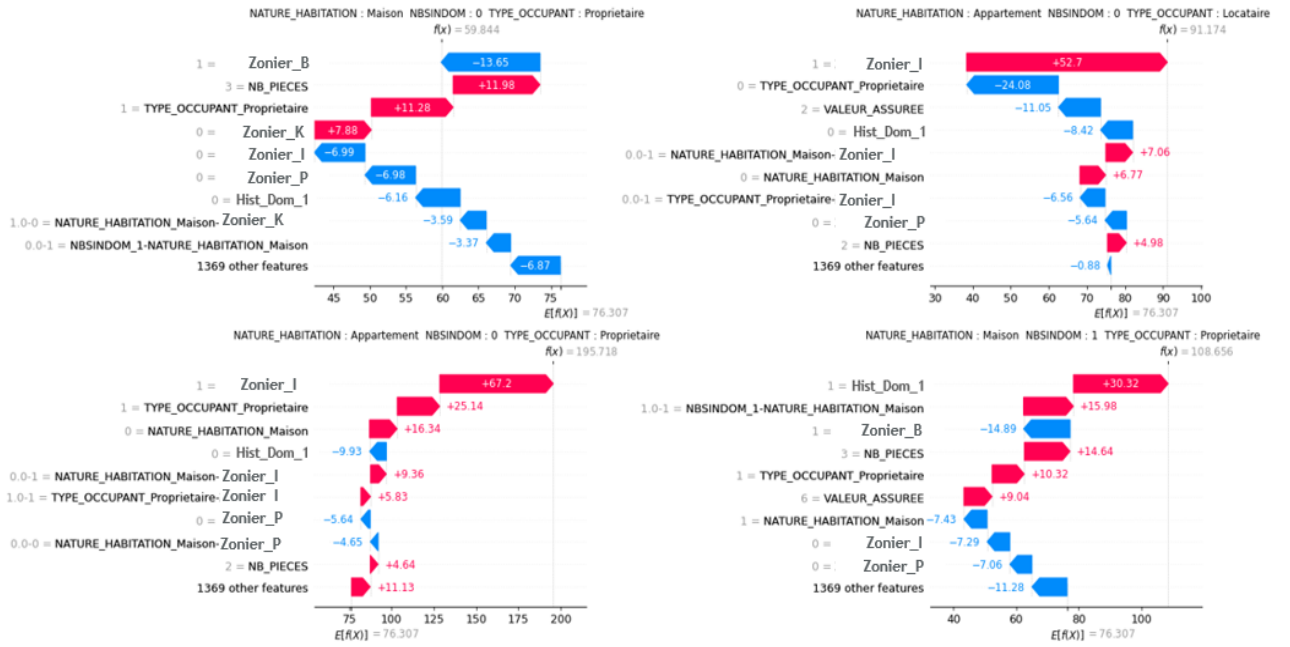


FIGURE 4.18 – Impact des différentes variables sur la prédiction de la prime pure pour 4 individus représentatifs

Ce graphique est particulièrement intéressant pour mettre en évidence comment la prime pure a été fixée pour les quatre individus étudiés. Le zonier semble particulièrement influent sur la prime finale avec des zones I et B qui impactent respectivement d'environ +55 et -14 la prime pure moyenne. Ces chiffres correspondent à l'effet direct et ne prennent pas en compte les effets croisés, ce qui explique que les valeurs ne sont pas totalement similaires à celle observées sur le graphique 3.11 de la prime pure par zonier.
De plus, pour les individus 1 et 4, il s'agit d'individus en maison et propriétaire, mais la zone géographique est aussi la même (Zone B), et ayant le même nombre de pièces ($> 80m2$). Le nombre de sinistres en historique (Hist_Dom) les différencie donc, respectivement 0 contre 1. On peut vraisemblablement penser que la prime sera plus élevée pour l'individu 4 que pour l'individu 1, ce qui se vérifie avec une prime pure de prédite de 59.8 € contre 108.6 €. La différence de prédiction s'explique en grande partie par l'impact de Hist_Dom dans le calcul de la prime pure (-6 contre +30). Sur les 48 € de différence de prime, 36€ sont bien expliqué par cette différence de Hist_Dom. Grâce à cette analyse, qui peut se généraliser à plein d'autres exemples, on peut dire que le modèle est intelligible et réalise les prédictions attendues.
Enfin, pour ces individus, les variables les plus influentes pour la prédiction de la prime pure sont le zonier, le nombre de pièces, le type d'occupant, la nature d'habitation et Hist_Dom.

Impacte des variables importantes du modèle de coûts sur le prime pure :

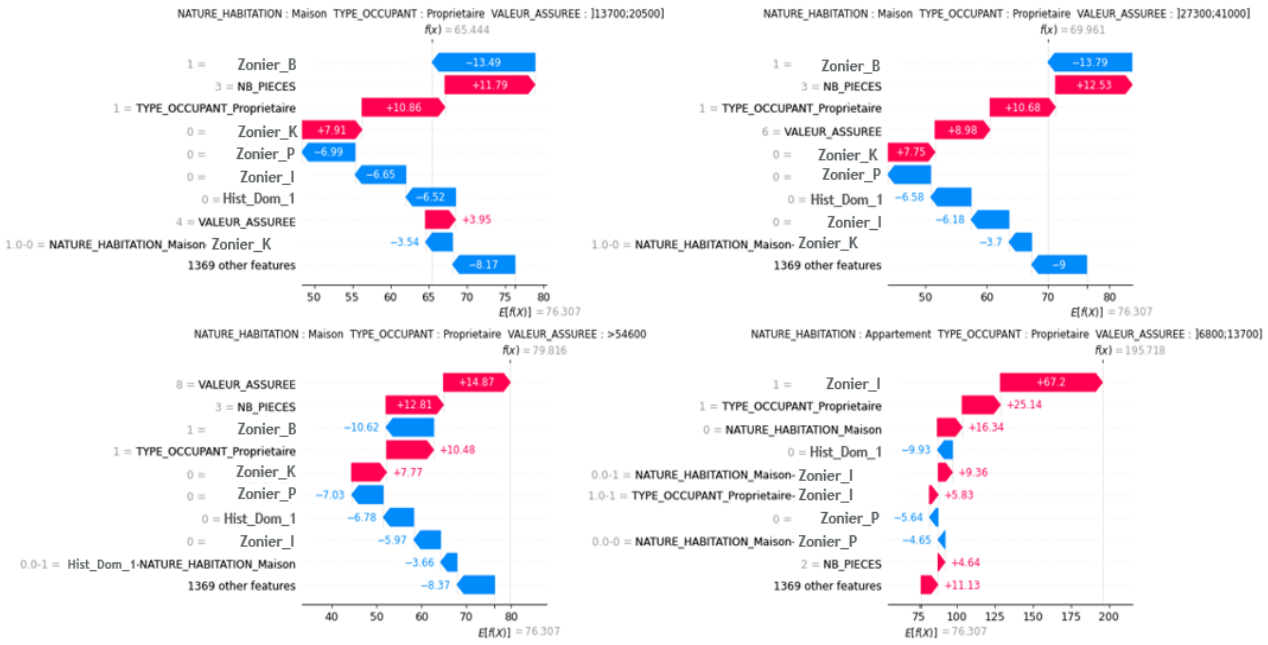


FIGURE 4.19 – Impact des différentes variables sur la prédiction de la prime pure pour 4 individus représentatifs

Une remarque intéressante pour les individus 1, 2 et 3, est qu'il s'agit d'individus en maison et propriétaire, mais aussi dans la même zone géographique (zone B), et ayant le même nombre de pièces ($> 80m2$) et aucun sinistre en historique (Hist_Dom $= 0$). Le montant de valeur assurée les différencie donc en partie. Les individus 1, 2 et 3 ont respectivement $]13700, 20500]$, $]27300, 41000]$ et > 54600 en montant de valeur assurée, on peut donc imaginer vraisembla-

blement que la prime sera croissante pour ces 3 individus en fonction du montant de valeur assurée. On observe bien ce phénomène, respectivement 65.4 €, 69.8 € et 79.8 € de prime pure, et ces différences s'expliquent bien grâce à l'impact de la variable correspondant au montant de valeur assurée en partie (respectivement +3.95, +8.98 et +14.87).
Pour l'individu 4, la prime pure élevée (195 €) est principalement dû à l'impact du zonier (+67.2€ pour la zone *I*). L'impact de cette zone est cohérent avec ce que l'on pouvait observer sur le graphique 3.11.
Enfin, pour ces individus, les variables les plus influentes pour la prédiction de la prime pure sont le zonier, le nombre de pièces, le type d'occupant, la valeur assurée et Hist_Dom.
Les primes pures sont donc calculées de manière cohérente pour l'ensemble des individus observés.

Pour faire une analyse au niveau global, une analyse du summary plot est nécessaire :

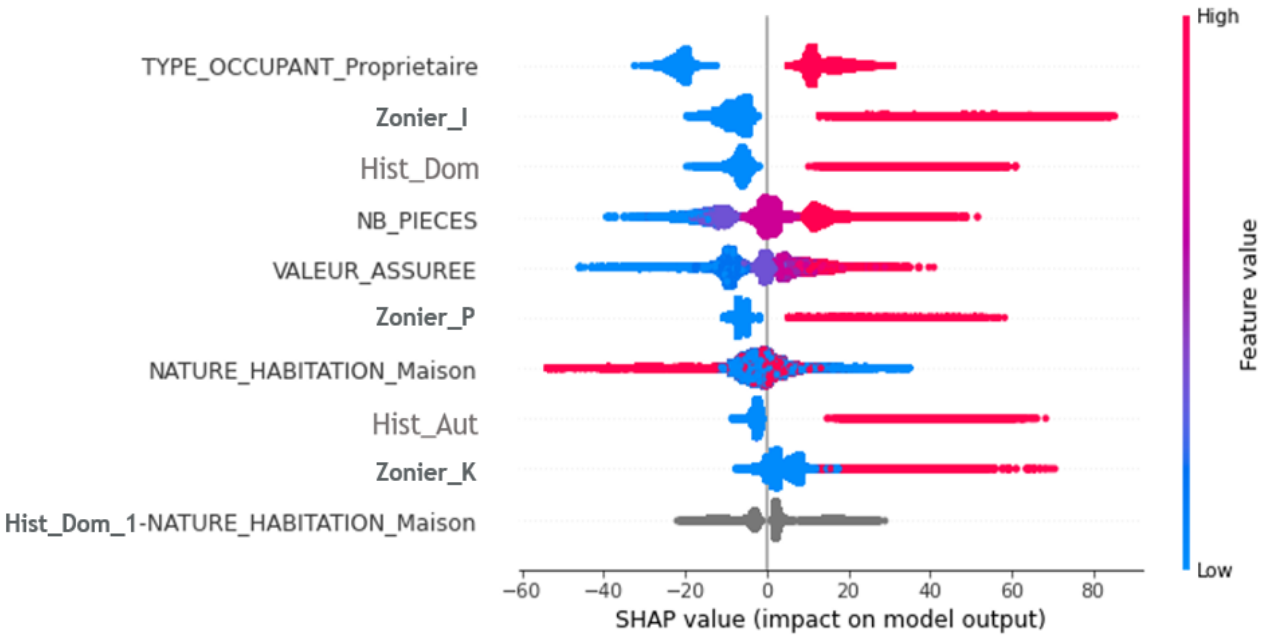


FIGURE 4.20 – Summary plot : Prime pure

Ce graphique met immédiatement en évidence le fait que le type d'occupant, le zonier, le nombre de pièces, le montant de valeur assurée et Hist_Dom sont les variables les plus importantes. On s'en doutait déjà en vue des graphiques 4.18 et 4.19 où la prime finalement s'écartait de la valeur de base en partie grâce à ces variables. Ce graphique est aussi intéressant dans le sens où on avait d'ores et déjà à disposition les summary plot pour les modèles de coûts et de fréquence, mais lorsque l'effet d'une variable n'était pas le même pour les coûts et pour la fréquence comme la nature d'habitation, alors rien ne pouvait être conclue quant à l'effet de cette variable sur la prime pure.
Analyse variable par variable :

- Type d'occupant : être propriétaire semble impacter la prédiction de la prime pure à la hausse (≈ +20 €). Par ailleurs, l'impact à la hausse pour les propriétaires est plus dispersé que l'impact à la baisse (≈ -20 €) pour les locataires. Enfin, cela correspond

à une différence de prime pure de 50 € environ en moyenne entre les modalités, ce qui correspond à l'écart observé sur le graphique E.2c.

- Zonier : chaque zonier semble impacter différemment la prime pure. Par exemple appartenir à la zone I impacte à la hausse la prime pure d'en moyenne 40 €. On observe sur le graphique 3.11 que les primes réelles pour ce zonier devraient être de ≈ 92€ contre une prime moyenne de ≈ 66 € sur les données de sortie de validation croisée. Le modèle de gradient boosting a donc bien capté la tendance de ce zonier, mais l'a surévalué.
- Nombre de pièces : impact à la baisse sur la prime pure par rapport à la moyenne pour les petites surfaces, et impact à la hausse pour les grands logements. Ceci était déjà observé sur les summary plot des modèles de fréquence et de coûts. Plus précisément, le summary plot indique que les logements $inf20m2$, $inf40m2$, $inf80m2$ et $sup80m2$ auront respectivement comme impact sur la prime pure ≈ -30 €, -10 €, 0 € et +20 €. Sur le graphique E.2d, on observait comme impact par rapport à la prime réelle -45 €, -20 €, 0 € et +30 €, donc bien que la tendance soit la même, l'effet n'est pas totalement directement expliqué par les valeurs shapley. Notons toutefois que le nombre de pièces peut encore impacter la prime pure de base en tant qu'interaction avec d'autres variables.
- Valeur assurée : la prime pure est impactée de façon linéaire croissante en fonction du montant de valeur assurée. La prime pure est plus impactée à la baisse pour les faibles montants en valeur assurée.
- Hist_Dom : l'effet de Hist_Dom=1 sur la prime pure est en moyenne de +50 € mais est très dispersé. L'effet de ne pas avoir de sinistre en historique est lui condensé autour de -10 €. Cette différence de prime pure en moyenne de de 60 € en les deux modalités coïncide bien avec le graphique E.2a où l'on observait une différence de 70 €.
- Nature d'habitation : pour cette variable l'effet n'est pas clair, bien que deux groupes se dégagent légèrement. Pour une partie des maisons, la prime pure prédite sera moins importante que pour une partie des appartements. Par ailleurs, pour une majorité des logements, la valeur shapley est proche de 0, ce qui peut s'expliquer par les effets opposés des modèles de coûts et de fréquence sur cette variable.

On peut conclure que le modèle de gradient boosting capte bien les effets souhaités et tarifie dans la bonne direction. Par ailleurs, l'ensemble des valeurs shapley directe ne peut pas expliquer toute la prime pure, dans certains cas certains couples de variable ont une valeur shapley importante. C'est ce qui peut expliquer les écarts observés ci-dessus entre les montants de valeurs shapley et les écarts réels observés.

4.5.6 Construction d'un tableau de prime

Pour les modèles linéaires grâce aux tableaux de coefficients, il est possible de retrouver aisément de manière multiplicative la fréquence et les coûts prédits, donc la prime pure pour un individu. Pour le modèle de machine learning retenu, ie gradient boosting pour la fréquence et le coût, nous pouvons réaliser un outil similaire. Grâce aux valeurs shapley obtenues précédemment sur la base d'apprentissage, nous avons désormais la décomposition additive de la prime pure pour chaque individu et donc pour chaque segment possible. Pour chaque segment, la moyenne des

valeurs shapley sera calculée, ce qui donnera l'impact moyen de chaque variable sur la prime pure et ceci sur chaque segment. Grâce à ce processus, pour chaque nouvel individu possible, on serait désormais capable d'indiquer une prime pure et comment la prime pure est impactée par les variables explicatives de cet individu. De ce fait, le modèle mis en place répondrait totalement aux attentes de transparence demandé par le décideur, le régulateur et l'assuré. Quel que soit le modèle de machine learning mis en œuvre, cette méthode restera la même.
Dans un second temps, on pourrait également comparer les sorties de manière multiplicative des GLM avec les valeurs shapley obtenues sur ce même modèle et ainsi observer si la méthode des valeurs shapley va bien dans le même sens que la méthode classique.

CHAPITRE

5

PROPOSITION DE DEUX OUVERTURES SUR LA GESTION DES RISQUES

5.1 Construction d'une check-list

Cette partie n'a pas été faite de manière appliquée, mais il en sera expliqué la démarche souhaitée à mettre en œuvre.
Grâce au tableau indiquant les primes par segment, nous pouvons d'ores et déjà calculer les primes pour les individus présents dans la base en 2018 et donc dans le même temps le ratio S/P.
La check-list consiste à stresser les données de l'année 2018 en changeant l'exposition dans la base de certains segments, par exemple augmenter le part des individus en maison par rapport à la part d'individus en appartement, et d'observer comment le S/P de cette année sera impacté. On pourra dans le même temps observer si les mouvements de S/P, à la baisse ou à la hausse, vont dans le même sens que les S/P trouvés pour ce segment dans le tableau 3.32. En partant d'un S/P pour l'année 2018 proche de 1, si notre modèle de tarification est consistant, en augmentant la part de maison dans la base, le S/P des données modifiées sera inférieur à 1 (le S/P trouvé pour les maisons dans le tableau 3.32 étant de 0.96, contre 1.00 pour les appartements).
Pour le stress des données, une méthode bootstrap pourra être utilisée pour piocher aléatoirement nos données avec pour contrainte la part d'exposition dans le portefeuille d'une certaine modalité à respecter. Grâce à ce tirage bootstrap, un intervalle de confiance des métriques utilisées lors de ce mémoire pourra être donné, en particulier du S/P qui nous intéresse pour cette étude. Pour chaque stress que l'on voudra effectuer sur le portefeuille, on pourra tirer B échantillons bootstrap et ainsi avec un intervalle de confiance de l'impact de chaque stress sur

le ratio S/P notamment.
Le but final est donc d'observer si des chocs à la hausse ou à la baisse sur la part dans le portefeuille de certains types de population va avoir un impact significatif sur le S/P de l'année à étudier. Grâce à cette étude, il sera désormais possible de déterminer quels individus sont préférables à avoir dans le portefeuille, et quels individus il est préférable d'éviter. Enfin, on pourra aussi déterminer un seuil critique à partir duquel, une trop grande part dans le portefeuille de certains individus causera un S/P trop supérieur à 1 et ce qui serait un scénario clairement défavorable pour l'assureur.
Un récapitulatif de la démarche est indiqué ci-dessous :

- Calcul du ratio S/P de base pour l'année 2018
- Définition des scénarios de stress à appliquer à la base
- Observation de la variation du S/P suite au stress
- Tirer des conclusions suite aux résultats obtenus, fixer des parts maximales d'exposition pour certains segments dans la base.

Il est de plus intéressant de pouvoir présenter une prime future dans un certain intervalle. Pour cela, de la même manière que précédemment, on peut tirer B échantillons bootstrap parmi les individus de l'année 2018 et noter la prime moyenne pour chaque échantillon. Grâce à cela, on peut obtenir un intervalle de confiance pour une prime potentielle future. On peut dans le même temps, pour ces échantillons bootstrap, fixer des contraintes comme la part d'individu en maison présents dans l'échantillon. On peut ainsi observer le spectre de la prime pure futur en fonction de la variation de la composition du portefeuille d'individu. Cet exercice est particulièrement intéressant pour réussir à cerner le risque intrinsèque à la volatilité potentielle de la composition du portefeuille.

5.2 Proposition de modélisation des sinistres graves et forfaitaires via le machine learning

La méthode utilisée pour gérer les sinistres graves et forfaitaires lors de ce mémoire consiste à les écarter du portefeuille pour les traiter séparément et ne modéliser seulement que la fréquence et les coûts des sinistres attritionnels. Dans cette ouverture, une autre méthode sera proposée de manière théorique qui consiste à les traiter au sein de la validation par bloc.
Dans cette méthode, les seuils de graves et de forfaits seront considérés comme des hyperparamètres et déterminés grâce à la méthode de validation par bloc en même temps que les hyperparamètres des modèles de gradient boosting et de forêts aléatoires. Un schéma permettant d'illustrer la méthode est affiché ci-dessous :

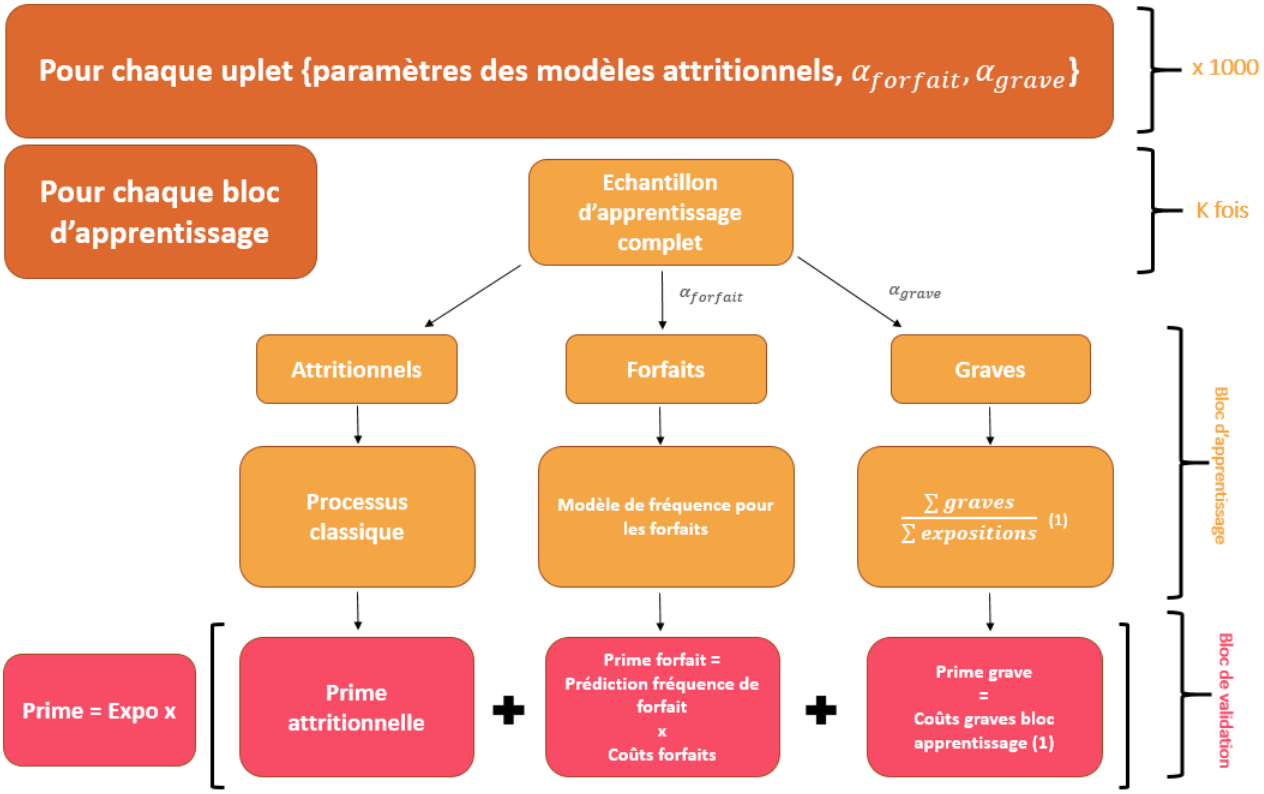


FIGURE 5.1 – Méthodologie validation par bloc incluant sinistres graves et forfaitaires

Focus sur le modèle de fréquence pour les forfaits : étant donné un seuil de forfait α, pour chaque bloc d'apprentissage, le sous-portefeuille composé des sinistres forfaitaires sera pris et une liste de coûts forfaitaires sera mise en évidence. Si on se place dans le bloc i et que la liste de forfaits est composées de m_i forfaits pour ce bloc, alors pour le modèle de fréquence, on réalisera un modèle de classification composé de $m_i + 1$ classe, une classe par forfait et une classe pour l'absence de sinistre forfaitaire. Les prédictions de ce modèle correspondront donc aux probabilités prédites d'appartenir aux classes possibles. La prime de forfait prédite pour l'individu j du bloc de validation i peut donc se décomposer de la manière suivante :

$$primeForfait_j = 0 \times predict_proba(X_j)_0 + \sum_{k=1}^{m_i} predict_proba(X_j)_k \times forfait_k \qquad (5.2.1)$$

où la classe 0 correspond à la classe "absence de sinistre forfaitaire" et X_j correspond aux variables prédictives de l'individu j.

Focus sur les graves : étant donné un seuil de grave α, pour chaque bloc d'apprentissage, le montant de grave en fonction de l'exposition est calculé. La prime des graves prédite pour le bloc de validation consiste donc à affecter à chaque individu au prorata de l'exposition le même montant qui a été appris sur le bloc d'apprentissage.

Grâce à cette méthodologie appliquée sur chaque bloc de validation, des primes pures incluant les graves et les forfaits sont prédites. De la même manière que précédemment, il est donc possible d'observer les métriques gini et S/P (cette fois ci incluant les forfaits et les graves) pour comparer les modèles. A la sortie de cette nouvelle validation par bloc on aura donc désormais

le meilleur modèle avec ses hyperparamètres optimaux, mais aussi les seuils de graves et de forfaits optimaux. De plus, au moment de l'analyse des métriques pour valider le modèle, on étudiera la volatilité des primes obtenues au titre des forfaits et des graves suivant les seuils retenus post validation par bloc. Mettre ces seuils en hyperparamétrage exigent tout de même une approche actuarielle traditionnelle en amont pour définir l'ensemble de définition de ces nouveaux hyperparamètres.

Un des avantages de cette méthode consiste à déterminer le seuil de forfait et de grave grâce à la validation par bloc et donc prendre en compte l'aspect temporel propre aux données. De plus, celle-ci permet d'inclure le traitement des forfaits et des graves au sein du processus et donc d'obtenir plus directement et intuitivement une prime pure finale avec ces types de sinistre.

CONCLUSION

Les méthodes de machine learning sont au cœur des sujets de recherches ces dernières années dans tous les domaines. Le secteur de l'assure n'y échappe pas, mais le manque d'interprétabilité de ces nouvelles méthodes n'est pas toujours compatible avec les normes mises en place par l'ACPR et le RGPD. Dans un secteur où il est nécessaire de comprendre et de maîtriser son risque, mais aussi d'avoir une certaine transparence vis-à-vis des régulateurs, des décideurs et des assurés sur la tarification pratiquée, l'usage et le bénéfice de telles méthodes n'est pas encore systématique. Bien qu'ayant en général un caractère prédictif plus efficace en captant des effets à une maille beaucoup plus fine, les modèles de machine learning perdent autant en interprétabilité voir plus qu'ils n'ont gagné en efficacité par rapport aux modèles linéaires généralisés classiques.

De nombreuses méthodes de machine learning existent, mais le choix s'est porté sur les forêts aléatoires et les gradient boostings dans ce mémoire. Il a été montré que ces méthodes apportaient un gain net en termes de gini sur la prédiction des coûts, et un gain plus minime sur la prédiction de la fréquence par rapport aux GLM. Cette métrique a été utilisée pour avoir une vision différente des métriques classiques utilisées habituellement (MAE et RMSE). Le modèle retenu, ie le gradient boosting, s'est avéré être plus performant sur la prédiction de la fréquence et des coûts aussi bien au niveau global, qu'à un niveau plus local c'est-à-dire par segment. Enfin, pour finir, les modèles ont été comparés en termes de précision sur la prime pure. Le modèle de gradient boosting s'est avéré être le plus performant avec un gain de 2 points par rapport au GLM en termes de gini. Par ailleurs, bien que le modèle de gradient boosting ait observé un S/P légèrement moins bon que le GLM, lorsqu'il s'agissait de regarder les S/P sur des segments précis par variable unique ou couplée, le modèle de gradient boosting a surpassé le GLM sur la majorité des segments. Ceci a permis d'appuyer le fait que les gradients boosting captent les effets à une maille plus fine que le GLM.

Pour défendre le modèle de gradient boosting qui a été jugé meilleur en termes de prédic-

tion, plusieurs méthodes de transparences ont été appliquées à cet algorithme pour rendre transparent celui-ci. Pour cela, deux méthodes globales, celles des PFI et des effets locaux accumulés (ALE), ont été utilisées. La première a permis de mettre en évidence les variables ayant un poids important dans la prédiction. La deuxième étant la méthode la plus adaptée du fait que celle-ci gère bien les corrélations entre les variables. Grâce aux ALE, l'impact moyen de chaque variable sur la prédiction de la fréquence et des coûts a pu être expliqué. De plus, une méthode à la fois locale et globale, celle des valeurs shapley, a été mise en œuvre. Cela a permis de décomposer de manière additive les prédictions et ainsi d'obtenir l'impact de chacune des variables dans la prédiction d'un individu en particulier. Son côté global a permis de mettre en avant de la même manière que les ALE l'effet moyen de chaque variable sur la prédiction et d'avoir un autre point de vue que avec les PFI sur l'importance des variables sur la prédiction.

Enfin les valeurs shapley ont été appliquées au croisement des modèles de fréquence et de coûts, ce qui a permis d'obtenir la décomposition additive de la prime pure et donc d'être transparent sur l'effet de chaque variable dans la tarification finale. Grâce à cette méthode, il est possible de vérifier aisément de la loyauté, de l'équité, de l'explicabilité et l'intelligibilité de l'algorithme. Un autre travail sur la prime pure mais qui n'a pas été réalisé en pratique consiste à maîtriser le risque sous-jacent au portefeuille mis à disposition. Pour cela, l'objectif serait de stresser les données suivant différents scénarios et d'observer la variation du ratio S/P. Grâce à des méthodes de tirages bootstrap, il serait même possible de donner un intervalle de confiance à ce ratio S/P.

Cette étude a été faite avec une garantie présentant une forte fréquence et une faible sévérité, mais il serait intéressant de déployer cette démarche à l'ensemble des garanties notamment avec une garantie présentant une faible fréquence et une forte sévérité comme la garantie incendie (statistiques descriptives en : B.1) par exemple. Ceci permettrait de voir si les modèles de machine learning s'adaptent aussi bien à ce type de garantie et si les méthodes de transparences s'appliquent encore.

BIBLIOGRAPHIE

[1] L. Breiman. "random forests", machine learning. 2001.

[2] Arnak Dalalyan. *Cours d'Apprentissage Statistique*. 2011.

[3] Institut des actuaires. 100%actuaires & 100% data-science, `https://www.institutdesactuaires.com/global/gene/link.php?doc_id=15779&fg=1`. 2019.

[4] Laurent Dupont and ACPR Pôle Fintech-Innovation. Gouvernance des algorithmes d'intelligence artificielle dans le secteur financier. 2020.

[5] Christophe Dutang. *Cours d'actuariat Non-Vie*. 2019.

[6] J. H. Friedman. Stochastic gradient boosting.

[7] Sébastien Gambs. *Survol de l'apprentissage machine*. 2011.

[8] Robin Genuer and Jean-Michel Poggi. Arbres cart et forêts aléatoires, importance et sélection de variables. 2017.

[9] Alex Goldstein and al. Peeking inside the black box : Visualizing statistical learning with plots of individual conditional expectation. 2015.

[10] Jerome H.Friedman.

[11] Scott M Lundberg. *SHAP packages*, 2018.

[12] Scott M Lundberg and al. Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv :1802.03888*, 2018.

[13] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. pages 4765–4774, 2017.

[14] Christoph Molnar. *Interpretable Machine Learning*. 2020.

[15] Maël Pégny and Issam Ibnouhsein. Quelle transparence pour les algorithmes d'apprentissage machine ? 2018.

[16] EU RGPD. Article 22, "décision individuelle automatisée, y compris le profilage".

[17] Christian Y. Robert. *Cours de théorie des valeurs extrêmes*. 2018.

[18] Daniel W.Apley and Jinhyu Zhu. Visualizing the effects of predictor variables in black box supervised learning models. 2016.

[19] Erik Štrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. 2014.

ANNEXE

A

LEXIQUE DES ABRÉVIATIONS

Abréviations	Garanties
ASS	Assistance
BRIS	Bris de glace
CNA N	Catastrophe naturelle standard
CNA O	Catastrophe naturelle Remarquable
DDE	Dégâts des eaux
DIV	Divers
IDC	Indemnisation dommages corporels
INC	Incendie
PJ	Protection juridique
RCC	Responsabilité civile corporelle
RCM	Responsabilité civile matérielle
REC	Recours
TNG	Tempête, neige, grêle
VOL	Vol

TABLE A.1 – Abréviations des garanties

Abréviations	Modèles
GLM	Modèle linéaire généralisé
GB	Gradient boosting
RF	Forêt aléatoire

TABLE A.2 – Abréviations des modèles

ANNEXE

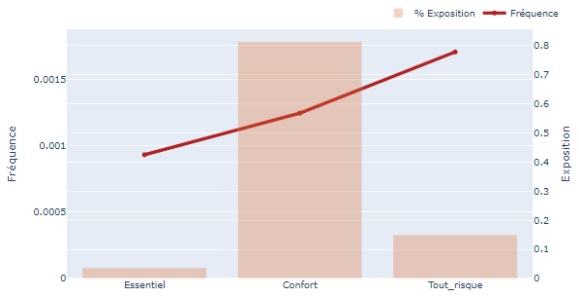
B

ANNEXES DESCRIPTIVES

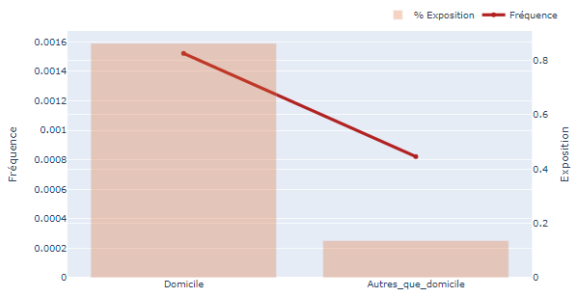
B.1 Analyses descriptives : garantie incendie

Les graphiques sur la fréquence et les coûts ont été réalisés de la même manière que pour les graphiques pour la garantie dégâts des eaux. Ces graphiques auraient pu être réalisés pour l'ensemble des autres garanties, mais il a été choisi de la présenter seulement pour la garantie incendie, celle-ci étant une garantie avec une faible fréquence et une forte sévérité et qui sera probablement étudiée de la même manière que la garantie dégâts des eaux par le futur.

Analyse de la fréquence des sinistres incendies en fonction des variables :

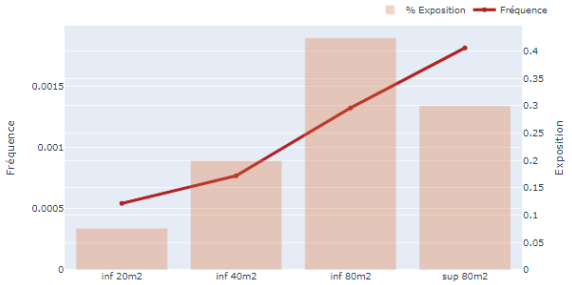


(a) Fréquence de sinistre par type de formule

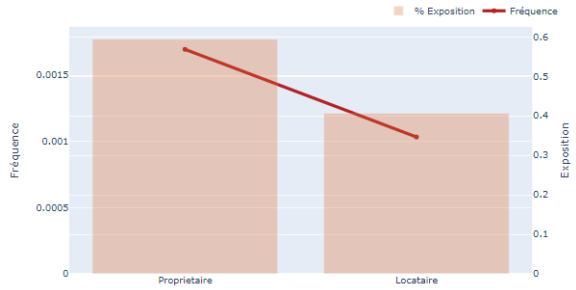


(b) Fréquence de sinistre par type d'habitation

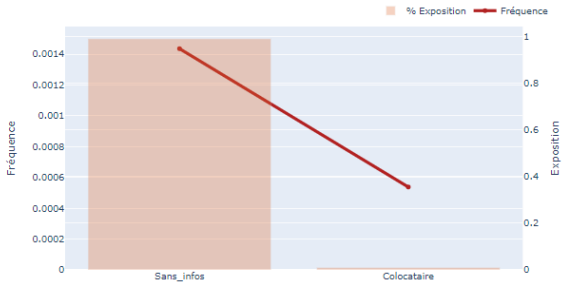
(a) Fréquence de sinistre par nature d'habitation



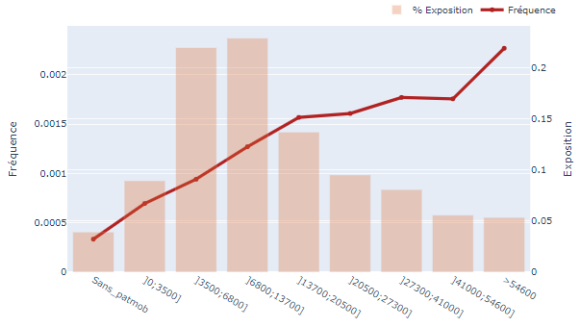
(b) Fréquence de sinistre par nombre de pièces



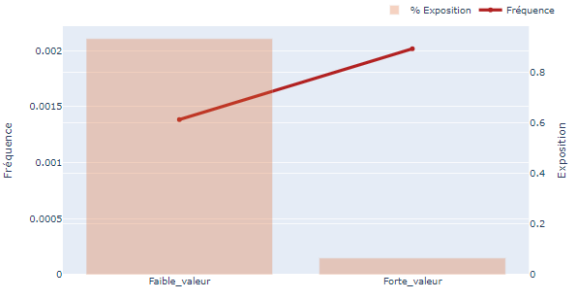
(a) Fréquence de sinistre par type d'occupant



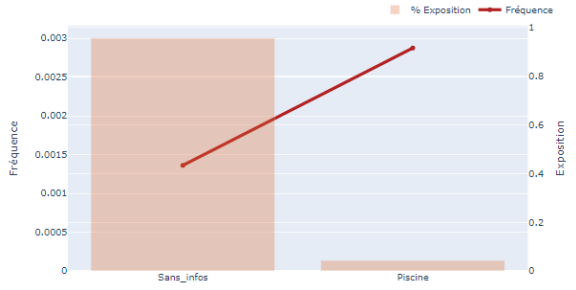
(b) Fréquence de sinistre par présence de colocataire



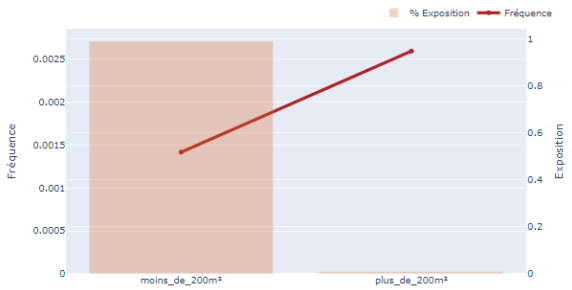
(a) Fréquence de sinistre par montant de valeur assurée



(b) Fréquence de sinistre par montant d'objet de valeur

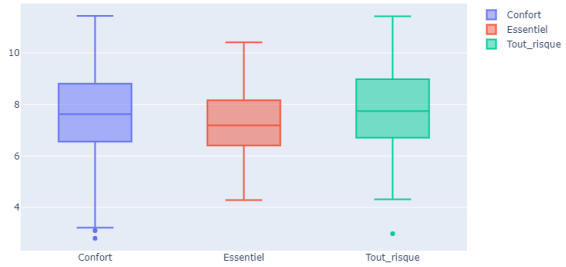


(a) Fréquence de sinistre par présence de piscine

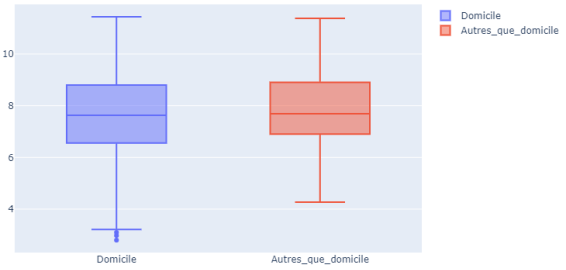


(b) Fréquence de sinistre par dépendance

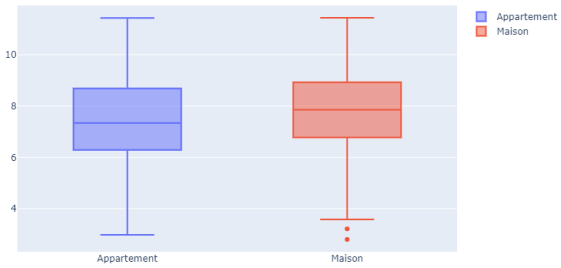
Analyse des coûts des sinistres incendies en fonction des variables :



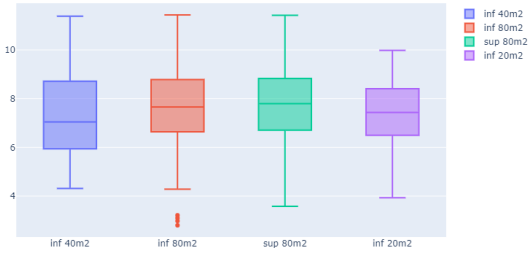
(a) Coûts des sinistres par type de formule



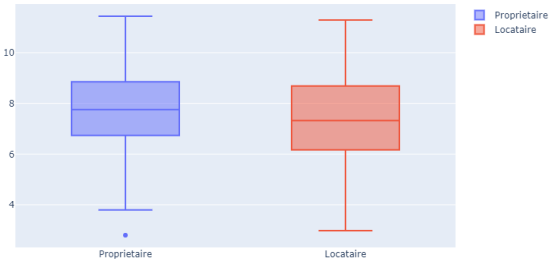
(b) Coûts des sinistres par type d'habitation



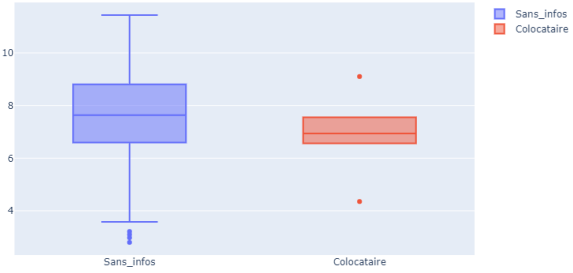
(a) Coûts des sinistres par nature d'habitation



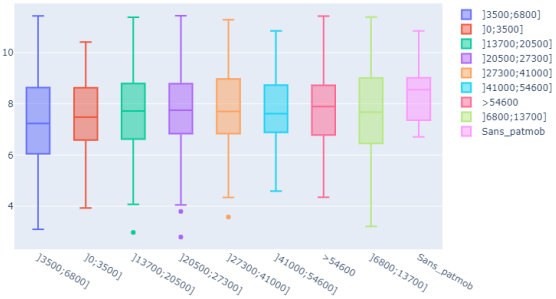
(b) Coûts des sinistres par nombre de pièces



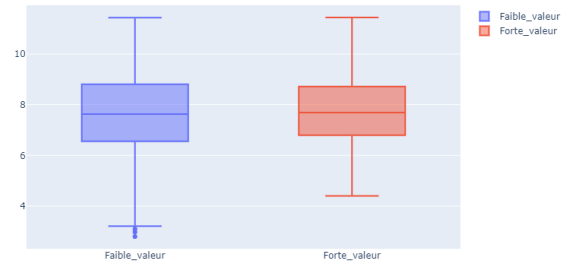
(a) Coûts des sinistres par type d'occupant



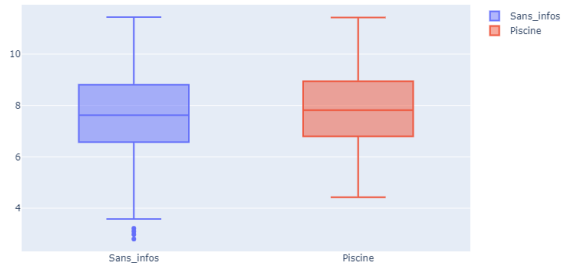
(b) Coûts des sinistres par présence de colocataire



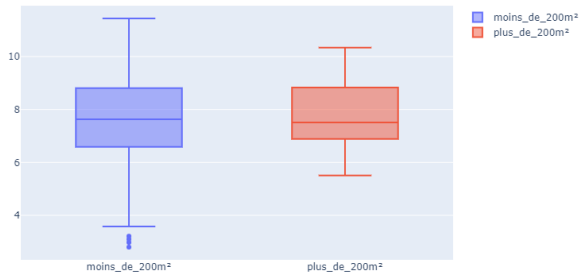
(a) Coûts des sinistres par montant de valeur assurée



(b) Coûts des sinistres par montant d'objet de valeur



(a) Coûts des sinistres par présence de piscine



(b) Coûts des sinistres par dépendance

B.1.1 V Cramer

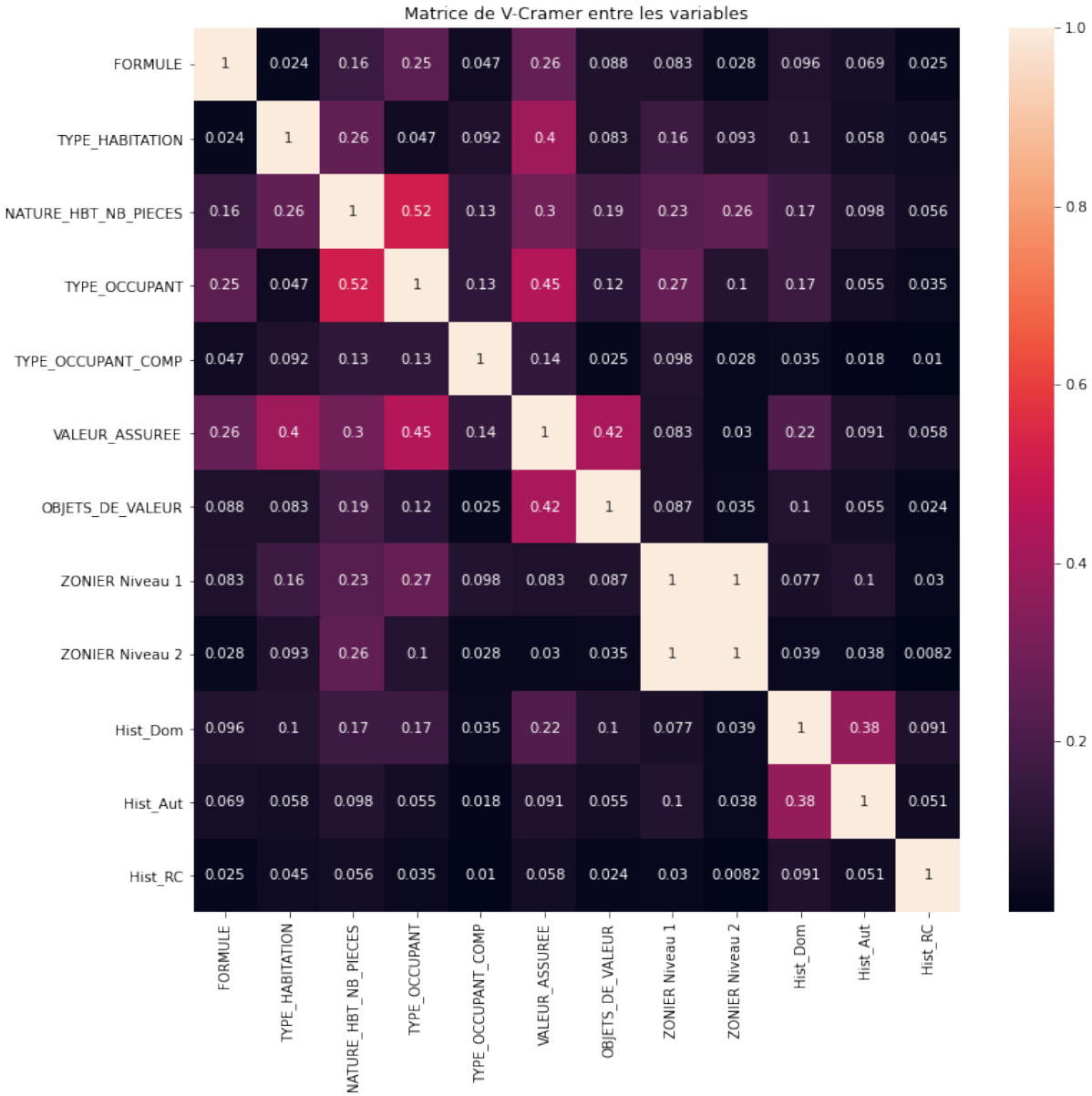


FIGURE B.11 – Matrice de V Cramer finale

ANNEXE

C

GLM : RÉSULTATS

Pour des questions d'anonymisation, l'ensemble des coefficients n'est pas affiché pour les grilles du modèle de fréquence et de coûts. Ces GLM ont été réalisés grâce à une démarche à mi-chemin entre la minimisation de l'AIC en approche actuarielle traditionnelle combinée à une démarche datascientist combinant cross-validation et tests des différentes combinaisons de variables en entrée et de famille de liens. Les GLM ont été réalisés grâce à un outils présent en interne qui est donc optimal. Cela permet donc de comparer les modèles de machine learning à des GLM optimaux et donc d'être sur le pied d'égalité.

C.1 Modélisation de la fréquence

Pour la modélisation de la fréquence, de la même manière que pour les modèles de machine learning, les données ont été agrégées. Par ailleurs, pour les GLM, étant donné que les variables correspondant aux nombres de pièces et à la nature de l'habitation correspondent fortement (V cramer élevé), celles-ci ont été jumelées. Le GLM réalisé a été fait avec une loi de Poisson et lien logarithme. La variable prédite correspond donc à l'exponentielle de l'intercept et de la somme des coefficients adéquates correspondant aux modalités de l'individu i. Une réduction de variable a été mise en œuvre grâce à la méthode dite Backward en se basant sur la métrique AIC. Cette sélection de variables a indiqué qu'aucune variable ne devait être retirée du modèle pour obtenir le modèle optimal. Pour réaliser ce modèle, les modalités de références choisies sont celles présentant la plus forte exposition, par exemple la modalité "Confort" pour la variable Formule.
Ci-dessous, est présenté un tableau récapitulatif des coefficients du GLM réalisé sur la fréquence pour la garantie dégâts des eaux.

feature	modalite	prc_expo	Pr	coef	Ajuster
(Intercept)			0	-3.010	
FORMULE	Confort	0.813	0	0	
FORMULE	Essentiel	0.037	0.00003	-0.132	
FORMULE	Tout_risque	0.150	0.447	-0.009	KO
NATURE_HABITATION	Appartement		0	0	
NATURE_HABITATION	Maison		0	0	
NATURE_HBT_NB_PIECES	Appartement - inf 20m2	0.065	0	-0.443	
NATURE_HBT_NB_PIECES	Appartement - sup 80m2	0.032	0	0.243	
NATURE_HBT_NB_PIECES	Maison - inf 80m2	0.178	0	-0.270	
NATURE_HBT_NB_PIECES	Maison - sup 80m2	0.266	0	0	
NB_PIECES	inf 20m2		0	0	
NB_PIECES	inf 40m2		0	0	
NB_PIECES	inf 80m2		0	0	
NB_PIECES	sup 80m2		0	0	
OBJETS_DE_VALEUR	Faible_valeur	0.934	0	0	
OBJETS_DE_VALEUR	Forte_valeur	0.066	0	0.127	
TYPE_HABITATION	Autres_que_domicile	0.136	0	-0.243	
TYPE_HABITATION	Domicile	0.864	0	0	
TYPE_OCCUPANT_COMP	Colocataire	0.011	0.008	-0.153	
TYPE_OCCUPANT_COMP	Sans_infos	0.989	0	0	
TYPE_OCCUPANT	Locataire	0.411	0	-0.329	
TYPE_OCCUPANT	Proprietaire	0.589	0	0	
VALEUR_ASSUREE	]0 ;3500]	0.090	0	-0.165	
VALEUR_ASSUREE	]27300 ;41000]	0.080	0.00001	0.078	
VALEUR_ASSUREE	]6800 ;13700]	0.230	0	0	
VALEUR_ASSUREE	>54600	0.053	0.0001	0.083	
ZONIER_SIMP	Non_urbain		0	0	
ZONIER_SIMP	Urbain		0	0	
ZONIER	Zonier_AD	0.020	0	-1.031	
ZONIER	Zonier_Q	0.011	0	-0.307	
ZONIER	Zonier_E	0.043	0	-0.912	
ZONIER	Zonier_F	0.033	0	-1.013	
ZONIER	Zonier_A	0.014	0	-0.526	
ZONIER	Zonier_G	0.034	0	-0.844	
ZONIER	Zonier_H	0.059	0	-0.408	
ZONIER	Zonier_K	0.018	0	-0.341	
ZONIER	Zonier_I	0.093	0	0	

TABLE C.1 – Fréquence : Coefficients du GLM

La colonne *Ajuster* indique si une variable n'est pas significative. Dans cette tableau toutes les variables ont été indiquées, même celles non prises en compte par le modèle comme le nombre de pièces et la nature de l'habitation.

Analyse des coefficients :

- Nombres de pièces/Nature habitation : en fixant le nature d'habitation à Appartement ou Maison, il se dégage une tendance linéaire croissante en fonction du nombre de pièces. La tendance linéaire était vérifiée sur les graphiques de fréquences pour les modalités de la variable correspondant au nombre de pièces (voir 1.6). De même en fixant le nombre de pièces, la fréquence prédite sera plus faible pour les maisons que pour les appartements, ce qui est concordance avec les vraies graphiques de fréquence (voir 1.5).
- Formule : le GLM prédit des fréquences plus faibles (coefficient négatif) pour la modalité

Essentiel que pour la modalité Tout-risque, ce qui va dans le même sens que le graphique 1.4.

- Valeur assurée : les coefficients du GLM pour la valeurs assurée indiquent que la fréquence prédite sera linéaire en fonction du montant de valeur assurée, ce qui coïncide bien avec le graphique des valeurs observées (voir 1.10). Le GLM capte donc bien les effets du montant de valeur assurée.
- pour les autres variables l'analyse se fait de la même manière...

C.2 Modélisation des coûts

Pour la modélisation des coûts, un modèle linéaire généralisé avec une loi Gamma et un lien logarithmique ont été utilisés. Pour la sélection de variables, une validation croisée a été réalisée sur l'ensemble des scénarios de choix de variable possibles. Le scénario donnant le meilleur AIC a été retenu. Celui a indiqué que les variables à retenir sont les suivantes : Formule, NATURE_HBT_NB_PIECES, Objets de valeur, Type d'occupant, Valeur assurée et Zonier. Pour ce modèle, les modalités de références choisies sont les modalités présentant le plus grand nombre de sinistres.

feature	modalite	nb_sin	prc_sin	Pr	Ajuster	coef
(Intercept)		0	0	0		6.963
FORMULE	Confort	37,212	0.810	0		0
FORMULE	Essentiel	1,108	0.024	0.001		-0.147
FORMULE	Tout_risque	7,609	0.166	0.006		0.051
NATURE_HBT_NB_PIECES	Appartement - inf 40m2	7,952	0.173	0.014		-0.051
NATURE_HBT_NB_PIECES	Appartement - sup 80m2	3,149	0.069	0.001		0.094
NATURE_HBT_NB_PIECES	Maison - inf 80m2	5,319	0.116	0		0.335
NATURE_HBT_NB_PIECES	Maison - sup 80m2	12,295	0.268	0		0.412
OBJETS_DE_VALEUR	Faible_valeur	40,600	0.884	0		0
OBJETS_DE_VALEUR	Forte_valeur	5,329	0.116	0		0.214
TYPE_HABITATION	Autres_que_domicile	3,420	0.074	0		0
TYPE_HABITATION	Domicile	42,509	0.926	0		0
TYPE_OCCUPANT_COMP	Colocataire	293	0.006	0		0
TYPE_OCCUPANT_COMP	Sans_infos	45,636	0.994	0		0
TYPE_OCCUPANT	Locataire	15,002	0.327	0.004		-0.091
TYPE_OCCUPANT	Proprietaire	30,927	0.673	0		0
VALEUR_ASSUREE	]0;3500]	2,539	0.055	0		-0.225
VALEUR_ASSUREE	]27300;41000]	4,623	0.101	0.009		0.069
VALEUR_ASSUREE	]6800;13700]	10,838	0.236	0		0
VALEUR_ASSUREE	>54600	3,599	0.078	0.0004		0.115
ZONIER	Zonier_R	258	0.006	0		0.081
ZONIER	Zonier_S	287	0.006	0		-0.250
ZONIER	Zonier_T	254	0.006	0		-0.045
ZONIER	Zonier_K	1,253	0.027	0		0.122
ZONIER	Zonier_I	8,652	0.188	0		0.101
ZONIER	Zonier_L	1,293	0.028	0		-0.027

TABLE C.2 – Coûts : Coefficients GLM

Analyse des coefficients :

- Formule : les coefficients du GLM pour les coûts indiquent des coûts prédits plus élevés pour les formules Tout risque et plus faible pour les formules Essentiel, ce qui correspond

bien à ce que l'on attendait en se basant sur le graphique 1.14.

- Nature d'habitation/Nombre de pièces : en fixant la nature d'habitation à appartement ou maison, on observer une tendance linéaire croissante des coefficients en fonction du nombre de pièces, ce qui est signe de coûts prédits croissants en fonction du nombre de pièces. Ceci est en accord avec le graphique 1.17. De la même manière, en fixant le nombre de pièces, les coefficients sont supérieurs pour la modalité maison que pour la modalité appartement, ce qui est à nouveau en accord avec le graphique 1.16.
- Valeur assurée : on observe des coefficients négatifs pour les modalités]0,3500] et]3500, 6400] et des coefficients positifs "élevés" pour les modalités Sans patrimoine,]41000,54000], >54000. Ceci correspond bien à ce que l'on peut observer sur le graphique 1.20 où les coûts moyens sont plus faibles pour les modalités énoncées avec coefficients négatifs et plus élevés pour les modalités avec coefficients positifs. Le modèle de GLM semble donc bien capter les effets de la valeur assurée.
- pour les autres variables l'analyse se fait de la même manière...

C.3 Prime pure pour GLM

Pour obtenir la prime pure d'un individu i grâce aux tables des GLM présentés ci-dessus, il faut réaliser le produit du modèle de fréquence et coûts.

$$Prime_Pure_i = exp(\text{intercept}_{freq} + \sum_{j=1}^{p} X_{i,j} \times \text{coeff}_{freq,j}) \times exp(\text{intercept}_{couts} + \sum_{j=1}^{p} X_{i,j} \times \text{coeff}_{couts,j}) \tag{C.3.1}$$

où $X_{i,j}$ est égal à 1 ou 0, si l'individu i présente la caractéristique associée aux coefficient j.

ANNEXE

D

XGBOOST

Pour un gain de temps par rapport à la méthode Gradient Boosting de scikit-learn, la méthode XGBoost aurait pu être utilisée. Celle-ci a été étudiée en théorie mais non mise en pratique. Une ouverture de ce mémoire pourrait être de traiter cette méthode et comparer les résultats ainsi que les temps de calcul pour des gains de productivités.

XGBoost est une manière optimisée et plus rapide de réaliser que l'algorithme de Tree Gradient Boosting. Ce dernier est un algorithme d'apprentissage supervisé qui tente de prédire le plus précisément possible une variable cible, en combinant de manière séquentielle les prédictions de plusieurs algorithmes dits faibles. Il se différencie des forêts aléatoires en travaillant de manière séquentielle, ce qui le rend plus lent, mais lui permet aussi de s'améliorer à chaque itération. Le Gradient Boosting tente d'optimiser à chaque itération une fonction objective constituée d'une fonction de perte et d'une fonction de régularisation. A chaque itération, le nouveau modèle prédit la partie résiduelle que l'on n'a pas su apprendre à l'itération précédente. Le résultat final réside en la somme des prédictions faites par chaque modèle.
L'intérêt du XGBoost réside dans la correction d'un problème majeur du Tree Gradient Boosting, à savoir le fait de considérer la perte potentielle pour chaque séparation possible (split) à un nœud donné pour créer une nouvelle branche. En effet, dans le cas où nous disposons de beaucoup de variables explicatives, il y a un grand nombre de splits possibles, ce qui engendre des temps de calcul élevés.
La fonction objective : fonction de perte et régularisation
Afin de former le modèle, nous devons définir la fonction objective pour mesurer l'adéquation du modèle aux données d'apprentissage.

$$obj(\theta) = L(\theta) + \Omega(\theta) \tag{D.0.1}$$

où L est la fonction de perte et Ω est le terme de régularisation. La perte mesure le degré de prédiction par rapport aux données d'entraînement. Un choix commun pour L est par exemple l'erreur quadratique moyenne. Tandis que le terme de régularisation contrôle la complexité du modèle, ce qui nous permet d'éviter le sur-ajustement. L'idée dernière ce problème est d'avoir à la fois un modèle performant et simple.

Formalisation mathématique :

Mathématiquement, on peut écrire le modèle sous la forme :

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{D.0.2}$$

où K est le nombre d'arbre, f est une fonction dans l'espace fonctionnel F, et F est l'ensemble de tous les panier possible. La fonction objective est donc donnée par :

$$obj(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{D.0.3}$$

Formation additive :

La première question que nous voulons poser : quels sont les paramètres des arbres ? On peut constater que ce qu'on doit apprendre, ce sont ces fonctions f_i, chacun contenant la structure de l'arbre et les partitions de feuille. L'apprentissage de la structure arborescente est beaucoup plus difficile que le problème d'optimisation traditionnel où on peut simplement prendre le gradient. Il est impossible d'apprendre tous les arbres à la fois. Au lieu de cela, nous utilisons une stratégie additive : corrigez ce que nous avons appris et ajoutez un nouvel arbre à la fois. Nous écrivons la valeur de prédiction à l'étape t comme $\hat{y}_i^{(t)}$. On obtient ensuite :

$$\begin{aligned}
\hat{y}_i^{(0)} &= 0 \\
\hat{y}_i^{(1)} &= f_1(x_i) = \hat{y}_i^{(0)} + f_1(x_i) \\
\hat{y}_i^{(2)} &= f_1(x_i) + f_2(x_i) = \hat{y}_i^{(1)} + f_2(x_i) \\
&\dots \\
\hat{y}_i^{(t)} &= \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i)
\end{aligned} \tag{D.0.4}$$

De cette dernière équation, on en déduit la fonction objective à optimiser à l'étape t :

$$\begin{aligned}
obj(\theta)^{(t)} &= \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t)}) + \sum_{i=1}^{t} \Omega(f_i) \\
&= \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) + constante
\end{aligned} \tag{D.0.5}$$

On cherche à chaque itération la fonction f_t qui permet d'optimiser la fonction objective définie ci-dessus (D.0.5). Si l'on écrit le développement de Taylor à l'ordre 2 de la fonction objective D.0.5, on obtient :

$$obj(\theta)^{(t)} = \sum_{i=1}^{n} [l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t) \tag{D.0.6}$$

où g_i et h_i sont définis de la manière suivante :

$$g_i = \partial_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$$
$$h_i = \partial^2_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$$

Ainsi en supprimant toutes les constantes, la fonction objective à l'étape t à optimiser devient :

$$\sum_{i=1}^{n} [g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t) \tag{D.0.7}$$

Finalement, à la fin de notre algorithme, notre estimation de y est donnée par $\hat{y}^{(K)}$.

Complexité du modèle :

Un terme important à ne pas oublier est le terme de régularisation. On doit définir la complexité de l'arbre $\Omega(f)$. Pour ce faire, il faut au préalable affiner la définition de l'arbre $f(x)$ comme :

$$f_t(x) = w_{q(x)}, w \in \mathbb{R}^T, q : \mathbb{R}^d \to \{1, ..., T\}$$

où w est le vecteur des scores sur les feuilles, q est une fonction affectant chaque point de données à la feuille correspondante, et T est le nombre de feuille. Dans XGBoost, la complexité est définie comme :

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2 \tag{D.0.8}$$

où γ et λ sont des paramètres de régularisation à choisir.

En reprenant la fonction objective à l'étape t de l'équation D.0.7 et en injectant la fonction de régularisation D.0.8, on obtient :

$$obj(\theta)^t \simeq \sum_{i=1}^{n} [g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2 \tag{D.0.9}$$
$$= \sum_{j=1}^{T} [(\sum_{i \in I_j} g_i) w_j + \frac{1}{2} (\sum_{i \in I_j} h_i + \lambda) w_j^2] + \gamma T \tag{D.0.10}$$

où $I_j = \{i | q(x_i) = j\}$ est l'ensemble des individus affectés à la même feuille j pour un arbre de structure q. En notant $G_j = \sum_{i \in I_j} g_i$ et $H_j = \sum_{i \in I_j} h_i$, on obtient :

$$obj(\theta)^t = \sum_{j=1}^{T} [G_j w_j + \frac{1}{2} (H_j + \lambda) w_j^2] + \gamma T \tag{D.0.11}$$

De là, pour une structure q donnée, et puisque les w_j sont indépendants entre eux. Pour une fonction q(x) donnée, on peut optimiser chaque terme de la somme, à savoir $G_j w_j + \frac{1}{2}(H_j + \lambda) w_j^2$. Finalement obtient :

$$\forall j \in \{1, ..., T\}, w_j^* = -\frac{G_j}{H_j + \lambda}$$
$$obj^*(q) = -\frac{1}{2} \sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T$$

La fonction objective obj^* mesure à quel point l'arbre associé à la fonction q est bon (en termes de prédiction et de complexité). Maintenant que l'on possède un moyen de mesurer la qualité d'un arbre, il reste à déterminer quels arbres choisir. Il n'est pas possible dans la pratique de tous les lister et de tous les tester. Une méthode utilisée en pratique est une optimisation étage par étage de l'arbre. L'idée est de regarder si l'ajout de nouvelles feuilles à un niveau donné fournit un gain comparé à l'arbre initial, tout en tenant compte de la régularisation. Dans le cas de l'ajout de deux feuilles à un niveau donné, il faut que le gain associé (régularisé) soit positif :

$$Gain = \frac{1}{2}[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda}] - \gamma \tag{D.0.12}$$

L'indice L réfère à la feuille gauche et l'indice R réfère à la feuille droite. Finalement, il faut que le gain d'ajout de ces deux feuilles, comparé à la situation précédente, soit supérieur au coefficient γ de régularisation, sinon il n'y a aucun intérêt à effectuer cette subdivision. Ainsi, pour construire un arbre, on cherche récursivement la meilleure division possible jusqu'à ce qu'on atteigne la profondeur maximale. Ensuite, on élague les nœuds avec un gain négatif avec une approche du bas vers le haut.

ANNEXE

E

GRAPHIQUES DE PRIMES PURES PAR VARIABLE

Dans cette annexe, seront présentés les graphiques des primes pures par variable. Les variables mises en avant sont les variables importantes. Ils serviront d'appui pour l'analyse des valeurs shapley pour la prime pure.

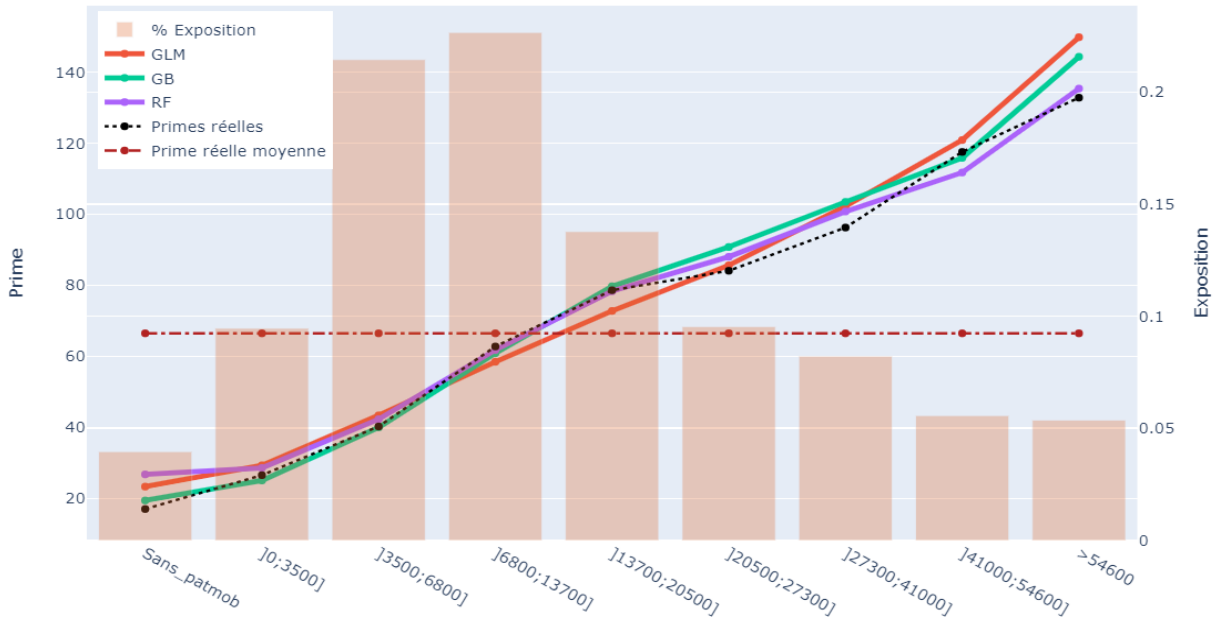
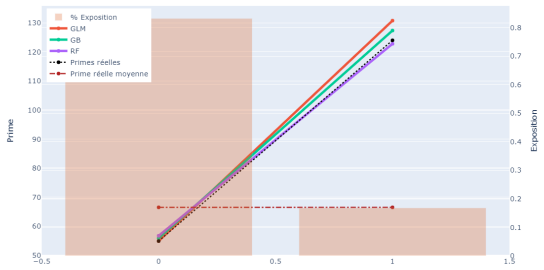
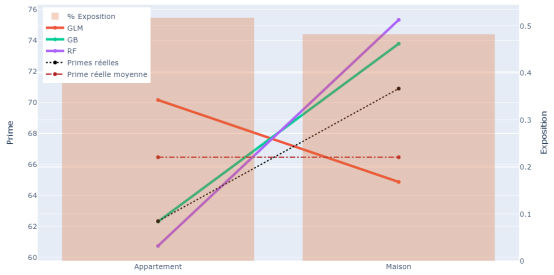


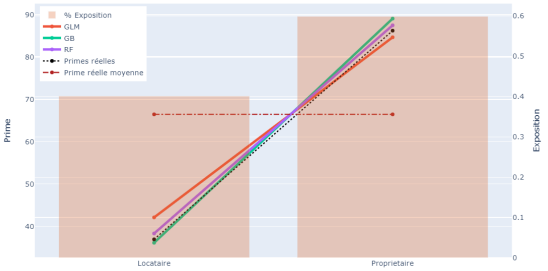
FIGURE E.1 – Prime pure en fonction du montant de valeur assurée



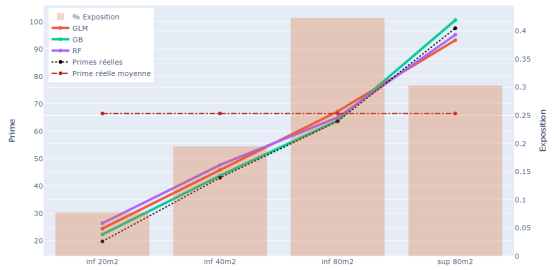
(a) Prime pure en fonction de Hist_Dom



(b) Prime pure en fonction de la nature d'habitation



(c) Prime pure en fonction du type d'occupant



(d) Prime pure en fonction du nombre de pièces

Le modèle de gradient boosting est le plus performant sur les variables indiquant le nombre de pièces, la nature d'habitation et le montant de valeur assurée. Sur les autres variables, il est prédit dans l'ensemble de manière correcte. On peut noter de plus que le GLM ne capte pas du tout l'effet de la nature d'habitation, comme l'on avait déjà pu l'observer dans le tableau récapitulant les S/P.

ANNEXE

F

AUTRES MÉTHODES DE TRANSPARENCE

Les méthodes de partial dependant plot et d'individual conditional expectation sont placées en annexe car les graphiques qu'elles permettent d'obtenir n'ont pas été affichés dans ce mémoire bien qu'ils aient été réalisés. Les accumulated local effect leur ont été préférés du fait qu'ils gèrent mieux les variables corrélées.

F.1 Partial Dependant Plot (PDP)

Le graphique de dépendance partielle (PDP) montre l'effet marginal d'une ou deux caractéristiques sur le résultat prédit d'un modèle d'apprentissage automatique. Un PDP peut montrer si la relation entre la variable cible à prédire et une variable est linéaire, monotone ou plus complexe. Par exemple, lorsqu'ils sont appliqués à un modèle de régression linéaire, les PDP montrent toujours une relation linéaire.
La fonction de dépendance partielle est définie comme suit [14] :

$$\hat{f}_{x_S}(x_S) = E_{x_C}[\hat{f}(x_S, X_C)] = \int \hat{f}(x_S, x_C) d\mathbb{P}(x_C) \tag{F.1.1}$$

Le x_S correspond aux variables pour lesquelles la fonction de dépendance partielle doit être tracée et x_C sont les autres variables utilisées dans le modèle d'apprentissage automatique $\hat{f}$. En général il n'y qu'une ou deux variables dans l'ensemble S. Les caractéristiques dans S sont celles pour lesquelles on veut connaître l'effet sur la prédiction. Les vecteurs de variables x_S et x_C combinés constituent l'espace total des variables prédictives. La dépendance partielle fonctionne en marginalisant la sortie du modèle de machine learning sur la distribution des variables de l'ensemble C, de sorte que la fonction montre une relation entre les variables de l'ensemble S qui nous intéressent et le résultat prédit. En marginalisant les autres variables,

nous obtenons une fonction qui ne dépend que des variables de S, interactions avec d'autres variables incluses.
La fonction PDP $\hat{f}_{x_S}$ est estimée en calculant la moyenne par une méthode de Monte Carlo :

$$\hat{f}_{x_S}(x_S) = \frac{1}{n}\sum_{i=1}^{n} \hat{f}(x_S, x_C^{(i)}) \tag{F.1.2}$$

Pour une ou plusieurs valeurs des variables S données, la fonction de dépendance partielle indique quel est l'effet marginal moyen sur la prédiction. Dans cette formule, $x_C^{(i)}$ correspond aux vraies valeurs du jeu de données pour les entités de l'ensemble C, et n correspond au nombre de lignes du jeu de données. Une des hypothèses importantes des PDP c'est que les variables de l'ensemble C ne doivent pas être corrélées avec celles de l'ensemble S. Si cette hypothèse est violée, la moyenne calculée pour la PDP va inclure des points de données qui sont invraisemblables (comme un studio avec une piscine par exemple). Le tracé de dépendance partielle est une méthode globale : cette méthode prend en compte toutes les instances et donne une indication sur la relation globale d'une variable avec le résultat prévu.

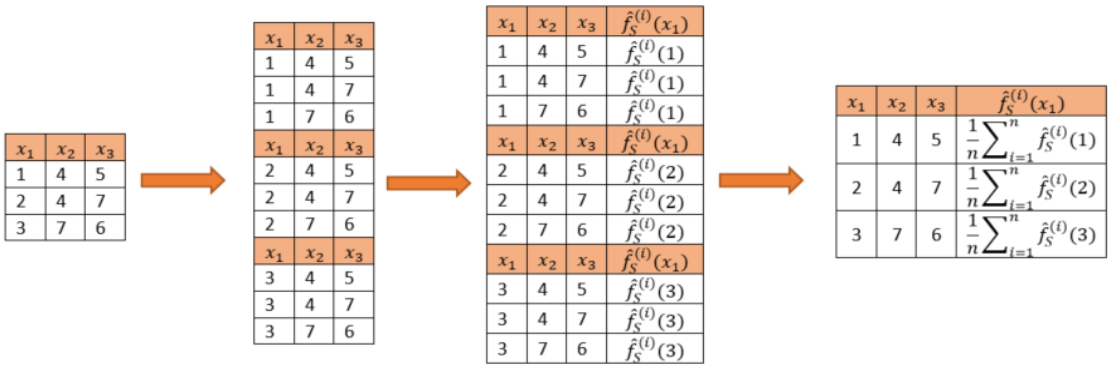


FIGURE F.1 – Exemple simple du calcul des PDP : S = {1}, C = {2,3}

Cas des variables catégorielles :
Pour les variables catégorielles, la dépendance partielle est facile à calculer. Pour chacune des modalités, on obtient une estimation PDP en forçant toutes les instances de données à avoir la même modalité. Par exemple, si nous examinons l'ensemble des données sur la fréquence de sinistre et que nous nous intéressons à la courbe de dépendance partielle pour la variable "Nombre de pièces", nous obtenons 4 chiffres (une fois le preprocessing effectué). Pour calculer la valeur "inf à 20m2" ie 0, on remplace le nombre de pièces de toutes les instances de données par 0 et on fait la moyenne des prévisions (voir § G.1).
Avantages et inconvénients :
La méthode PDP est intuitive et facile à mettre en place, mais elle présente l'inconvénient d'intégrer deux hypothèses qui sont rarement vérifiées. La première étant que toutes les variables doivent être indépendantes et uniformément distribuées sur leur support. Le deuxième inconvénient de cette méthode est que les effets hétérogènes peuvent être masqués car les PDP ne montrent que les effets marginaux moyens. Supposons que pour une variable, la moitié des points de données aient une association positive avec la prédiction - plus la valeur de la variable est grande, plus la prédiction est grande - et l'autre moitié a une association négative - plus

la valeur de la variable est petite, plus la prédiction est grande. La PDP pourrait être une ligne horizontale, car les effets des deux moitiés de l'ensemble de données pourraient s'annuler. On conclut ensuite que la fonction n'a aucun effet sur la prédiction. En traçant les courbes d'espérance conditionnelle individuelles (ICE) au lieu de la ligne agrégée, on peut découvrir des effets hétérogènes. Enfin certains PDP n'indiquent pas la distribution des variables. Omettre la distribution peut être trompeur, car on risque de sur-interpréter des régions avec presque aucune donnée. Ce problème est facilement résolu en montrant un histogramme.

F.2 Individual Conditional Expectation (ICE)

Il s'agit d'une méthode globale comme les PDP. Un graphique ICE visualise la dépendance de la prédiction sur une variable pour chaque observation séparément, ce qui donne une ligne par observation, par rapport à une ligne globale avec les PDP. Une PDP est la moyenne des lignes d'un graphique ICE. Cette méthode consiste à tracer sur le même graphique l'ensemble des $\hat{f}(x_S, x_C^{(i)})$ pour chaque $i \in \{1, ..., n\}$ avec x_S qui varie dans l'ensemble des modalités de S et $x_C^{(i)}$ qui reste fixé.

Avantages :

Les courbes d'espérance conditionnelle individuelles sont encore plus intuitives à comprendre que les graphiques de dépendance partielle. Une ligne représente les prédictions pour un individu si l'on fait varier les modalités de la variable d'intérêt. Contrairement aux graphiques de dépendance partielle, les courbes ICE peuvent révéler des relations hétérogènes.

Inconvénients :

- Avec les courbes ICE une seule variable à la fois peut être analysée (on ne peut pas regarder les effets croisés).
- Les courbes ICE souffrent du même problème que les PDP : si la variable d'intérêt est corrélée avec les autres variables, certains points dans les lignes peuvent être des points de données non valides selon la distribution des variables jointes.
- Si de nombreuses courbes ICE sont dessinées, le tracé peut devenir surchargé et donc n'est plus lisible. Pour contourner ce problème, on peut ajouter de la transparence aux lignes ou ne dessiner qu'un échantillon des lignes.

ANNEXE

G

VÉRIFICATION DES MÉTHODES DE TRANSPARENCE POUR LE GLM

L'objectif de cette annexe est de confirmer le fait que les méthodes de transparences sont cohérentes avec les coefficients observés pour les GLM en annexe C. Pour cette annexe, seul les résultats de transparences pour le GLM de fréquence seront analysés étant donné que celle fonctionne de la même manière pour le GLM de coûts.
Regardons les résultats de transparence pour deux assurés équivalents dont seule la variable type d'occupant changera. Prenons un individus ayant les modalités suivantes : Formule : Confort, Type d'habitation : Domicile, Nature d'habitation : Appartement, Nombre de pièces : inf 40m2, Type d'occupant complémentaire : Sans infos, Valeur assurée :]3 500, 6800], Objets de valeur : Faible valeur, Zonier : Zonier R, Standing : Sans infos, Dépendance : moins de 200 m2, Nombre de sinistre en historique : 0 (pour les trois variables). D'après le tableau de coefficients du GLM pour la fréquence, être "Propriétaire" n'impacte pas la prédiction du fait qu'il s'agit de la modalité de référence, et être "Locataire" multiplie la prédiction par $e^{-0.329} = 0.719$. L'assurée Propriétaire aura donc une prédiction de 0.0486 contre une prédiction de 0.0349 pour l'assurée Locataire.
Vérifions que les méthodes d'ALE plot (méthode globale) et des valeurs Shapley (méthode locale) sont cohérentes avec l'interprétation de l'impact de la variable Type d'occupant sur la prédiction de la fréquence.
Cette analyse sert à prouver la cohérence des résultats trouvés grâce aux méthodes de transparence pour une variable en particulier, ici le type d'occupant. Par ailleurs, on pourrait généraliser cette méthode pour chaque variable, mais aussi en combinant plusieurs variables sans difficultés particulières.

G.1 ALE plot

L'ALE plot de la variable type d'occupant ci-dessous explique pourquoi la prédiction d'un assurée étant propriétaire est plus élevée que pour un assurée étant locataire. En effet, il montre que la différence de prédiction moyenne entre les modalités "Propriétaire" et "Locataire" est d'environ 0.015, ce qui correspond environ à la différence observée avec les calculs de prédictions grâce aux coefficients du GLM.

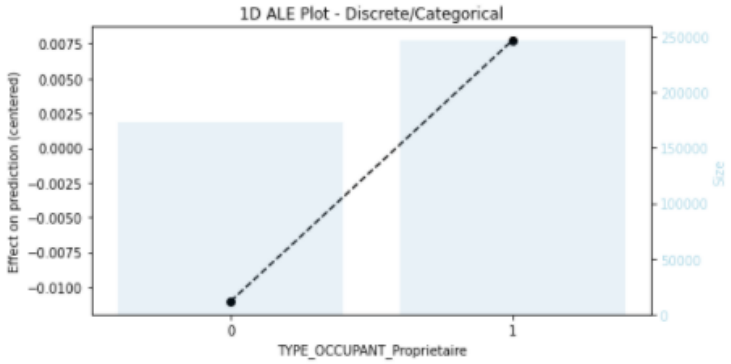
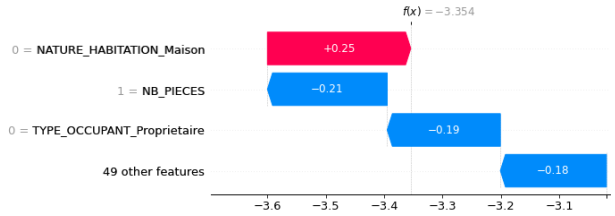


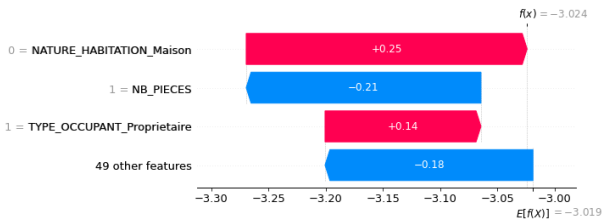
FIGURE G.1 – ALE plot : Type d'occupant

G.2 Valeurs Shapley

On peut voir sur les décomposition additives ci-dessous réalisées grâce aux valeurs Shapley que la modalité Propriétaire impacte à la hausse la prédiction de la fréquence, alors que la modalité Locataire impacte à la baisse la prédiction de la fréquence. Ceci explique la prédiction de la fréquence plus élevée de l'assurée propriétaire par rapport à l'assurée locataire. Les valeurs affichées sur les graphiques ci-dessous sont des valeurs avant avoir appliqué le lien logarithmique du GLM. On retrouve bien les valeurs de prédiction énoncées précédemment : $0.04860 = e^{-3.024}$ pour le propriétaire et $0.03494 = e^{-3.354}$ pour le locataire.



(a) Impact des variables : Assurée Locataire



(b) Impact des variables : Assurée Propriétaire

ANNEXE

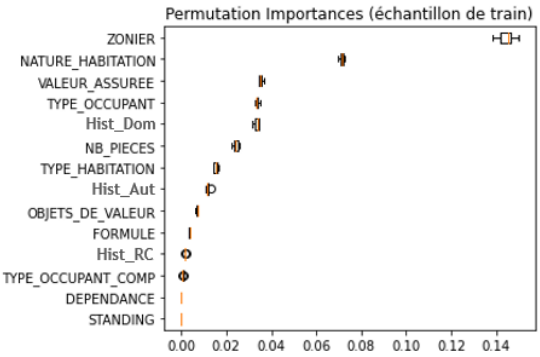
H

APPLICATION DES MÉTHODES DE TRANSPARENCE AUX MODÈLES DE FORÊTS ALÉATOIRES

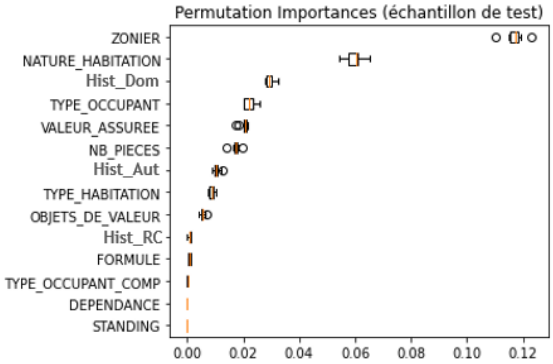
Dans cette section, l'ensemble des graphiques de sorties des méthodes de transparence pour les modèles de forêts aléatoires seront affichés. Ce modèle n'ayant pas été retenu comme modèle final, il est moins intéressant de la mettre en avant que le modèle de gradient boosting. Toutefois observer la transparence d'un autre type de modèle reste intéressant.

H.1 PFI

Importance des variables pour la fréquence :



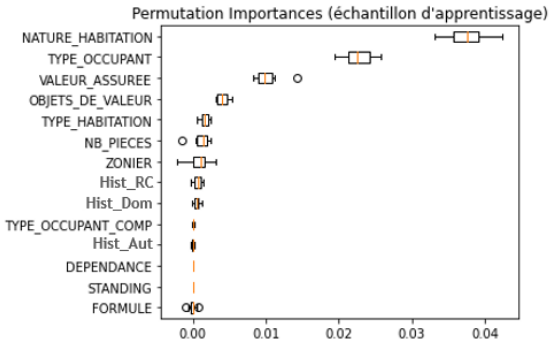
(a) Modèle de fréquence : Importance des variables sur les données d'apprentissage



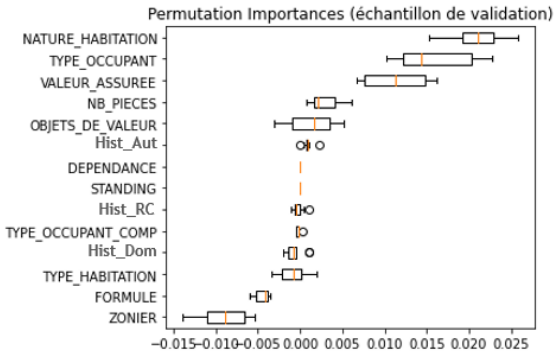
(b) Modèle de fréquence : Importance des variables sur les données de test

Les PFI pour le modèle de forêts aléatoires sur la fréquence indiquent que les trois variables les plus importantes sont la zone géographique, la nature d'habitation et le nombre de pièces ou Hist_Dom ou le type d'occupant.

Importance des variables pour les coûts :



(a) Modèle de coûts : Importance des variables sur les données d'apprentissage

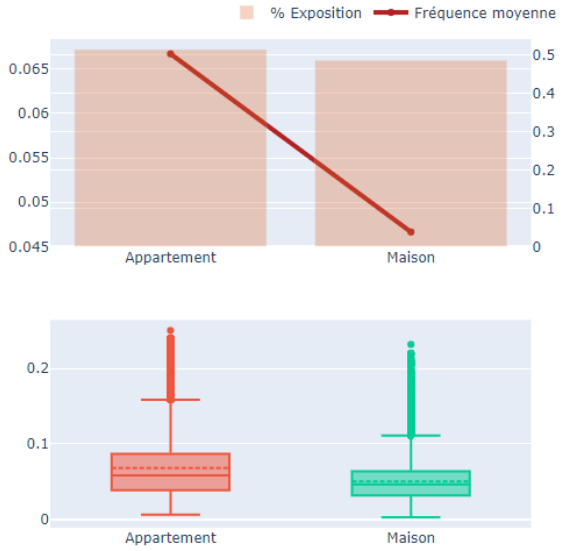


(b) Modèle de coûts : Importance des variables sur les données de test

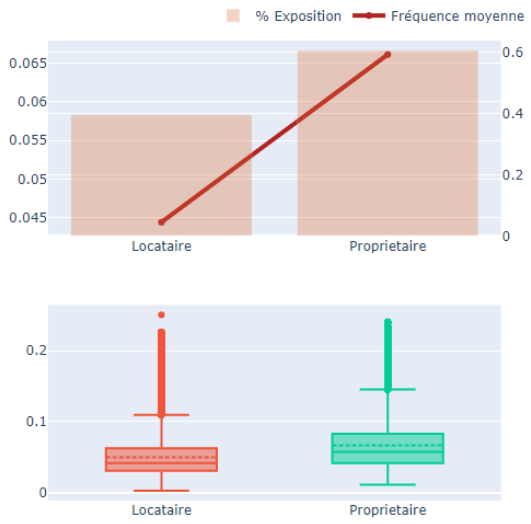
Les PFI pour le modèle de forêts aléatoires sur les coûts indiquent que les trois variables les plus importantes sont la Nature d'habitation, le type d'occupant et le montant en valeur assurée pour la base d'apprentissage et la base de validation.

H.2 Tracé des prédictions

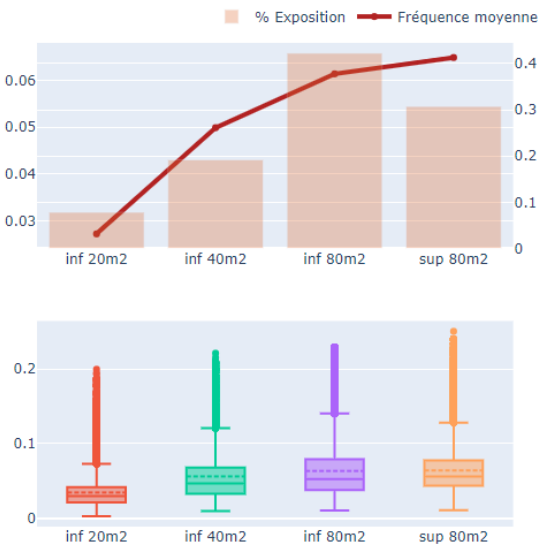
Tracé des prédictions pour le modèle de fréquence :



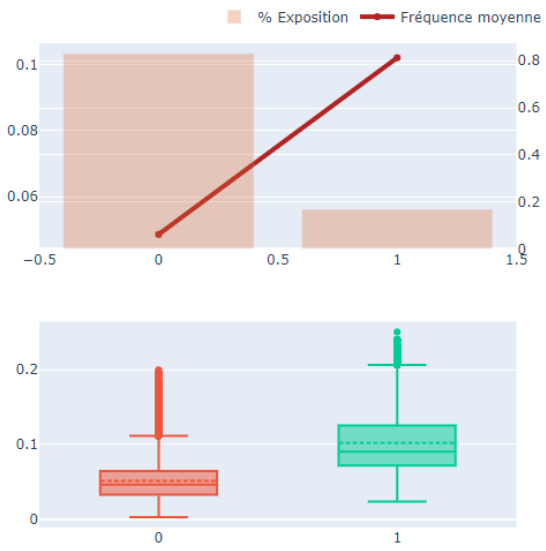
(a) Nature d'habitation : Prédictions vs valeurs réels



(b) Type d'occupant : Prédictions vs valeurs réelles

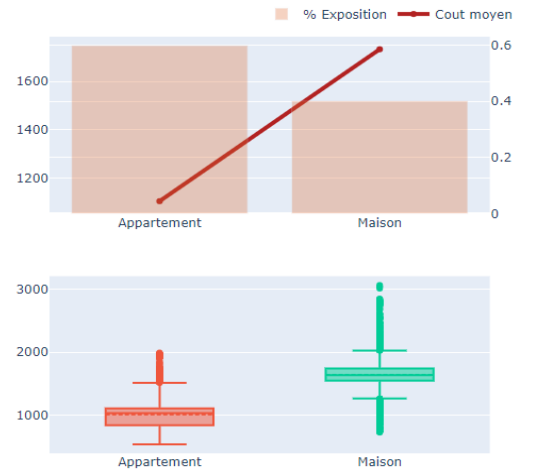


(a) Nombre de pièces : Prédictions vs valeurs réelles

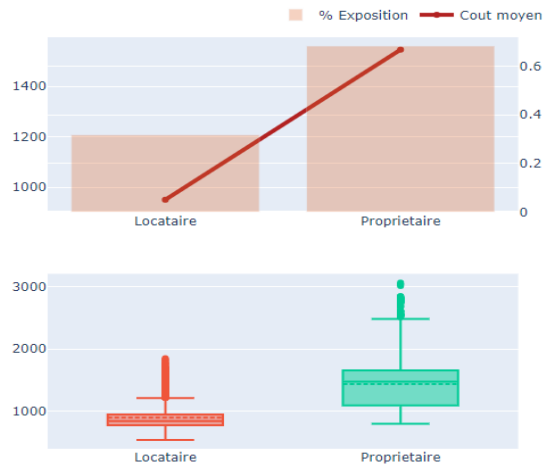


(b) Hist_Dom : Prédictions vs valeurs réelles

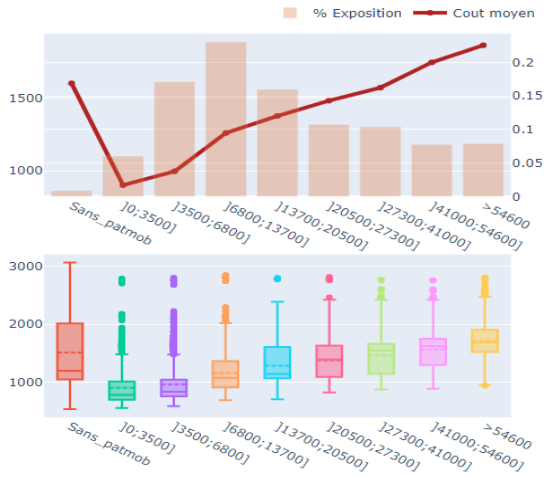
Tracé des prédictions pour le modèle de coûts :



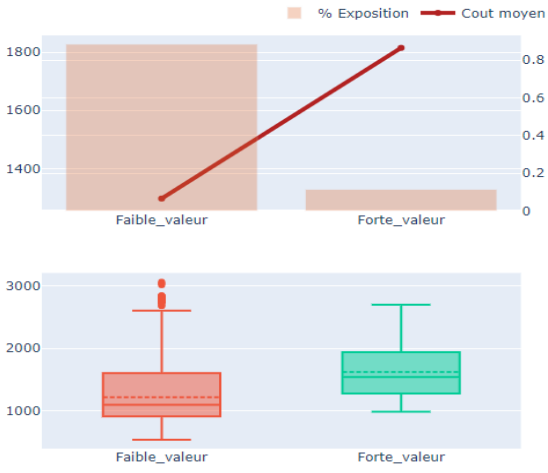
(a) Nature d'habitation : Prédictions vs valeurs réelles



(b) Type d'occupant : Prédictions vs valeurs réelles



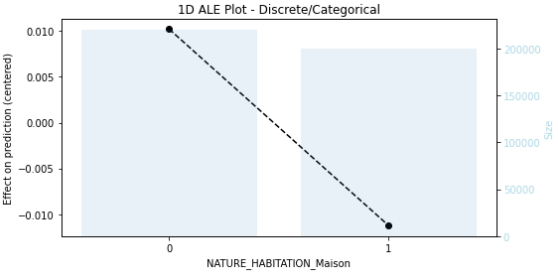
(c) Valeurs assurées : Prédictions vs valeurs réelles



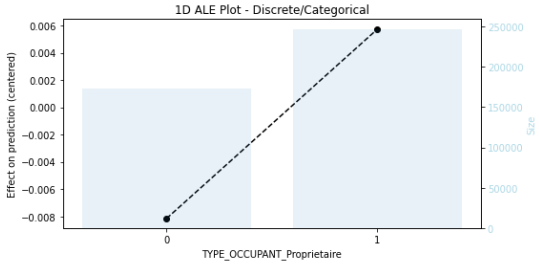
(d) Montant en objets de valeur : Prédictions vs valeurs réelles

H.3 ALE plot

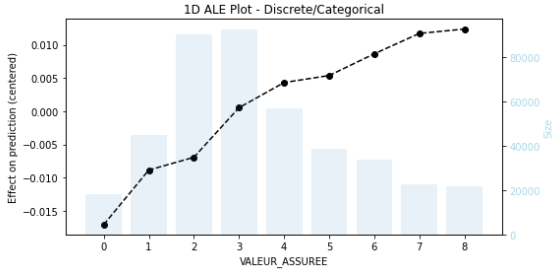
Tracé des prédictions pour le modèle de fréquence :



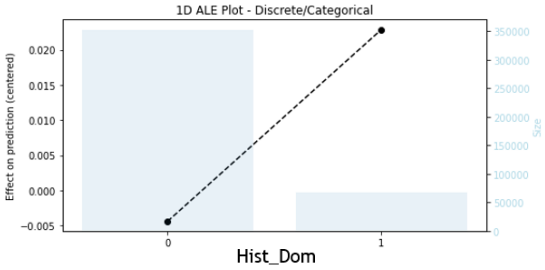
(a) Nature d'habitation



(b) Type d'occupant



(c) Valeurs assurée



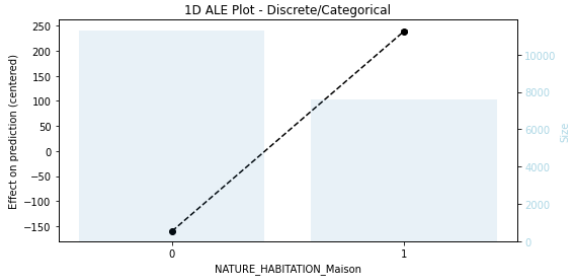
(d) Hist_Dom

FIGURE H.6 – Modèle de fréquence : ALE plot des forêts aléatoires

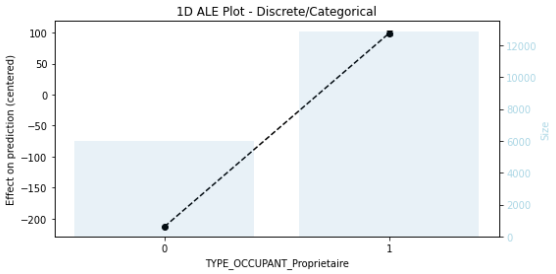
Les ALE plot pour le modèle de fréquence nous informent des effets suivants :

- Nature d'habitation : l'effet moyen d'être en Maison sur la prédiction des fréquences sera de -0.010, contre +0.010 pour un individu étant en appartement.
- Type occupant : l'effet moyen d'être propriétaire sur la prédiction de la fréquence sera de +0.006, contre -0.008 pour un individu étant en location.
- Hist_Dom : l'effet moyen sur la prédiction d'avoir eu un sinistre les années passées est de +0.020, contre -0.005 pour un individu n'ayant pas eu de sinistre récemment.
- Valeur assurée : l'effet moyen de la valeur assurée sur la prédiction de la fréquence varie de -0.015 pour une déclaration *Sans patrimoine* à +0.010 pour la plus haute tranche (>54 600).

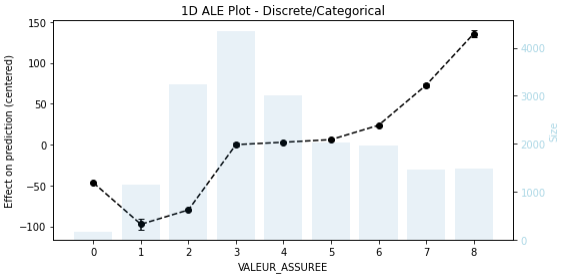
Tracé des prédictions pour le modèle de coûts :



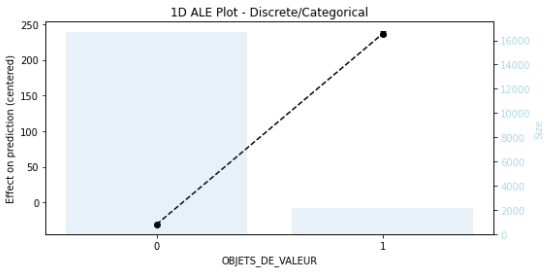
(a) Nature d'habitation



(b) Type d'occupant



(c) Valeurs assurée



(d) Montant en objets de valeur

FIGURE H.7 – Modèle de coûts : ALE plot des forêts aléatoires

Les ALE plot pour le modèle de coûts nous informent des effets suivants :

- Nature d'habitation : l'effet moyen d'être en Maison sur la prédiction des coûts sera de +250, contre -150 pour un individu étant en appartement.
- Type occupant : l'effet moyen d'être propriétaire sur la prédiction des coûts sera de +100, contre -200 pour un individu étant en location.
- Valeur assurée : l'effet de la valeur assurée sur la prédiction des coûts s'observe principalement pour les classes extrêmes. Appartenir à la classe *>54 600* de valeur assurée, va avoir un effet moyen de +250 sur la prédiction moyenne d'après le graphique. L'ALE plot nous montre par ailleurs la même tendance que sur les valeurs réelles.
- Objets de valeur : l'effet moyen d'avoir un fort montant en objet de valeur sur la prédiction des coûts sera de +250, contre un effet moyen presque nul pour un montant en objet de valeur faible déclaré.

H.4 Valeurs Shapley

SHAP pour le modèle de fréquence :

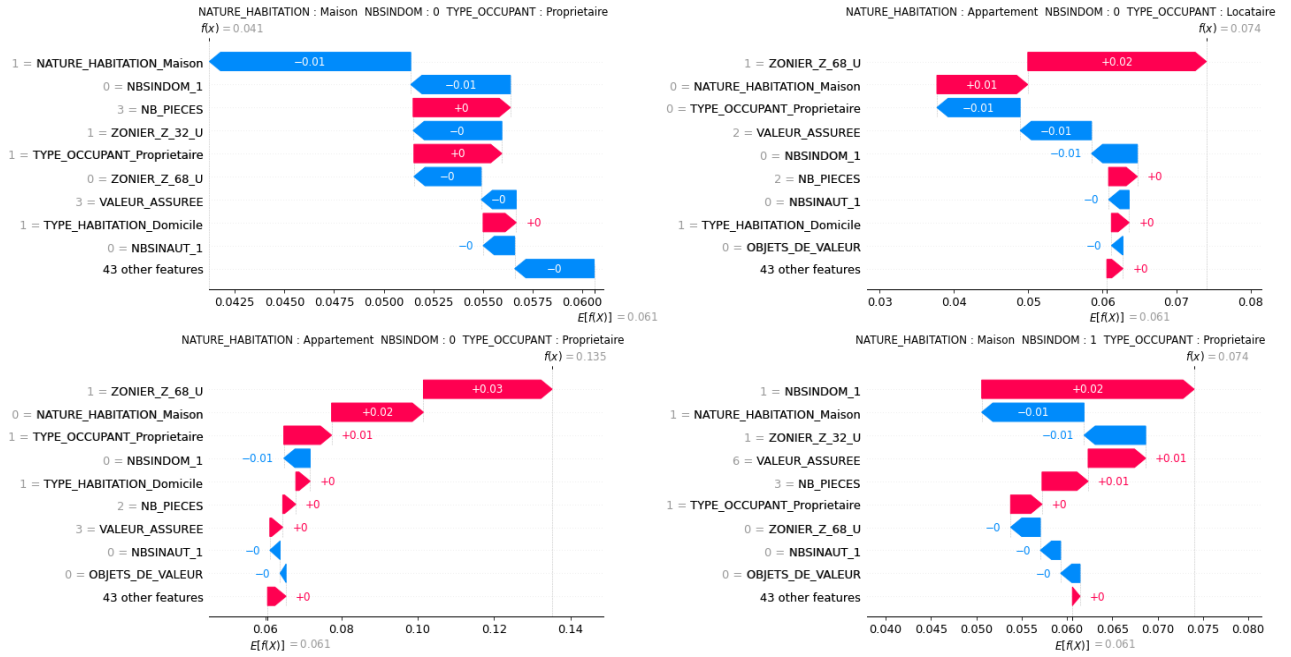


FIGURE H.8 – Impact des différentes variables sur la prédiction de la fréquence pour 4 individus représentatifs

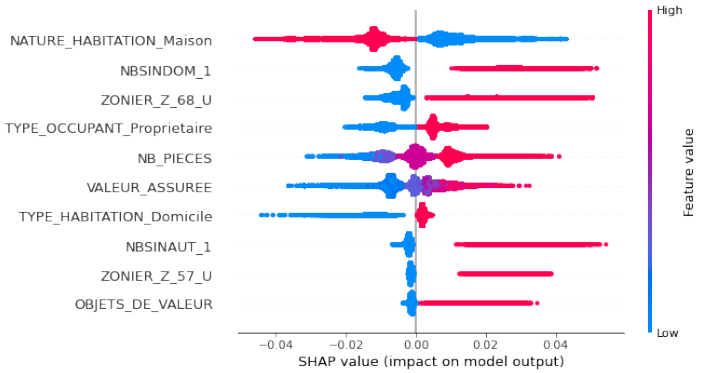


FIGURE H.9 – Summary plot : modèle de fréquence

Le summary plot du modèle de forêts aléatoires pour la fréquence est similaire à celui du modèle de gradient boosting (4.13), la même analyse peut donc être faite.

SHAP pour le modèle de coûts :

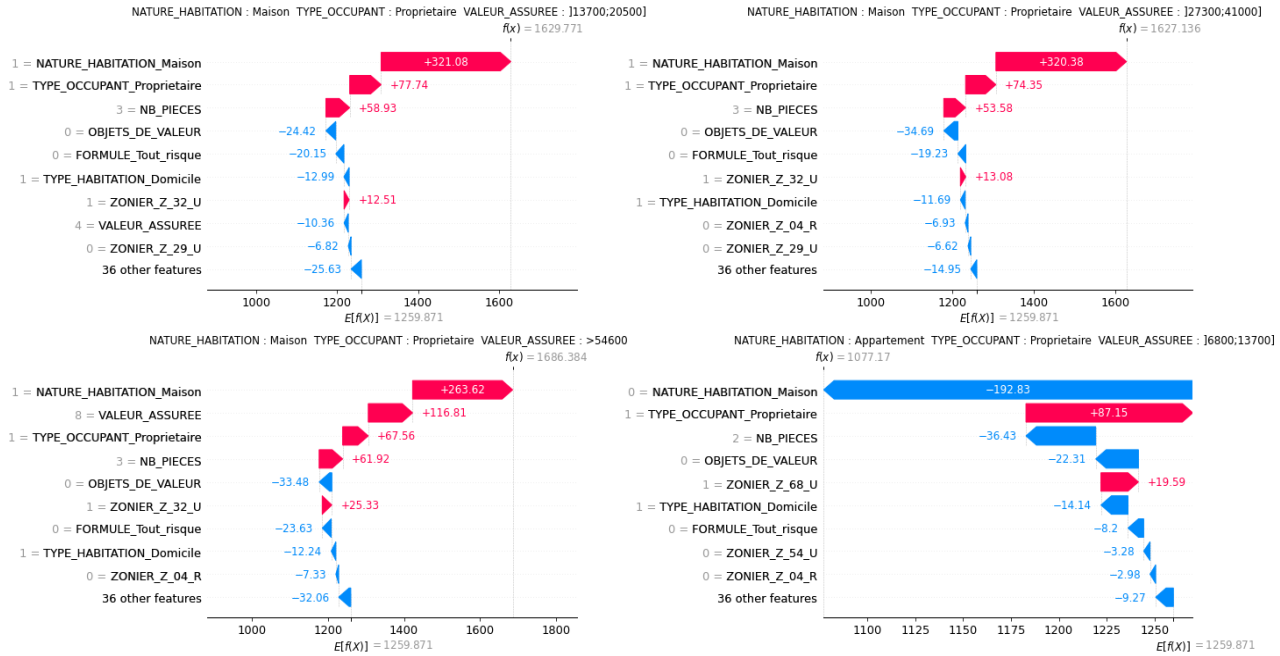


FIGURE H.10 – Impact des différentes variables sur la prédiction des coûts pour 4 individus représentatifs

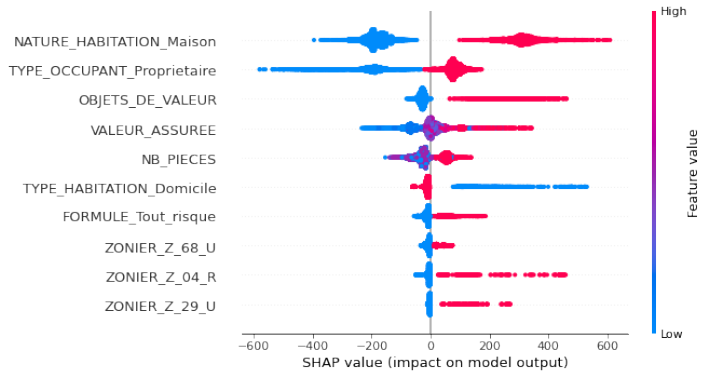


FIGURE H.11 – Summary plot : modèle de coûts

Le summary plot pour le modèle de forêts aléatoires, est sensiblement différent de celui du modèle de gradient boosting pour les coûts. Dans le cas des forêts aléatoire, l'impact d'une variable sera borné entre -600 et 600, contre -250 et 1250 pour le modèle de gradient boosting. Ces modèles ont notamment capté de manière différente l'effet du montant de valeur assurée et le type d'occupant pour la prédiction des coûts.