

Mémoire présenté le :

**pour l'obtention du Diplôme Universitaire d'actuariat de l'ISFA
et l'admission à l'Institut des Actuaires**

Par : Romain Picabea

Titre Construction d'un métamodèle d'efficience de réassurance par différentes méthodes d'interpolation spatiale

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

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

*Membre présents du jury de l'Institut
des Actuaires*

signature

Entreprise

ADDACTIS France

Espace WeWork

Jules Lefebvre - 75009 Paris

Nom : ADDACTIS France

Tel. 01 56 89 07 70

Signature :

Siren 413 611 344 - NAF 7022-Z

Directeur de mémoire en entreprise
SAS au capital de 100 000 €

Nom : JOUVET Isabelle

Signature :

Invité :

Nom :

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



Signature du candidat





Construction d'un métamodèle d'efficience de réassurance par différentes méthodes d'interpolation spatiale

Mémoire d'actuariat
Institut de Science Financière et d'Assurances
Addactis France
2019
Romain Picabea

Résumé

Ce mémoire propose de répondre aux difficultés qui peuvent être rencontrées lors de l'utilisation et l'analyse de modèles actuariels coûteux en ressources temporelles et informatiques. La solution proposée dans ces travaux se base sur l'interpolation spatiale, pour construire des modèles simplifiés plus rapides à l'exécution, appelés « métamodèles », à partir d'un nombre de lancements limité d'un modèle initial chronophage. Les résultats obtenus sur ces quelques exécutions du modèle de base permettent d'obtenir un plan d'expérience, servant de base à la construction des métamodèles. Dans ce cadre, les principaux sujets à étudier concernent le choix des valeurs de paramètres sur-lesquelles lancer le modèle initial, et le choix des méthodes d'interpolation à appliquer.

Nos travaux se sont basés sur un modèle de tarification en réassurance, possédant un module dédié à la tarification par simulations des traités en excédent de sinistre, coûteuse en temps de calcul. Nous nous sommes concentrés sur cette partie du modèle, avec pour objectif d'en donner une version simplifiée, centré sur le calcul de l'efficience d'un traité du point de vue d'une cédante.

La première partie de ce mémoire présente différentes manières de concevoir un plan d'expérience, à travers plusieurs techniques d'échantillonnage et une technique de construction dite « séquentielle ». Elle détaille également les méthodes d'interpolation spatiale utilisées, à savoir l'interpolation bicubiques par splines et le krigeage, en prenant en compte les problématiques d'optimisation numérique et d'évaluation des modèles.

Les deux parties suivantes sont directement liées au modèle de tarification en réassurance. L'une présente les données utilisées, le modèle et le fonctionnement du module de tarification par simulations, l'autre les hypothèses préalables à la construction d'un métamodèle, comme la définition d'un critère de convergence du modèle initial ou d'un budget en temps de calcul pour la constitution du plan d'expérience.

La quatrième partie de ce mémoire traite de l'application pratique des méthodes vues précédemment au modèle de tarification en réassurance. Plusieurs métamodèles sont d'abord construits via une approche par plan d'expérience fixe, en faisant appel au krigeage et à l'interpolation par splines. Plusieurs avantages du krigeage nous conduisent à retenir cette méthode pour appliquer l'approche séquentielle. Nous construisons alors deux modèles de krigeage dans une optique d'amélioration globale de la qualité de prédiction, puis un troisième dédié à l'optimisation de l'efficience de réassurance. Les métamodèles précédents ayant été construits avec deux variables d'entrée, nous incluons deux variables supplémentaires pour concevoir un dernier métamodèle d'optimisation de l'efficience.

Dans une cinquième et dernière partie, nous revenons sur les limites des travaux effectués, et notamment les limites inhérentes aux méthodes utilisées, en proposant des pistes d'améliorations et de nouveaux axes d'étude.

Abstract

The aim of this thesis is to tackle some issues related to the use of actuarial models, in some instances time-consuming with high informatic resources costs. A possible solution is to set up spatial interpolation methods, in order to build simpler and faster models, also called metamodels. These metamodels are built from a limited number of runs of an initial time-consuming model. The outcomes of these few runs are used to create a design of experiments (DoE), from which the metamodel can be built. In this framework, the choice of the initial parameters values on which the model will be run and the choice of interpolation methods to apply are the main matters of concern. In this thesis, several approaches are developed and applied.

We chose to focus our work on a reinsurance pricing model, including a sub-model dedicated to the pricing of excess-of-loss treaties using simulations, a time-consuming process. Thus, this sub-model is at the heart of this thesis. The goal is to build a simpler version, especially computing the efficiency of such treaties for a ceding company.

The first part of this thesis expose several ways to shape a design of experiments, more specifically a technique called one-shot design using various sampling methods, and a technique called sequential design. The interpolation methods applied in the practical implementation, bicubic splines and kriging, are also described, taking into account numerical optimization issues and model assessment processes.

The next two parts are directly linked to the reinsurance pricing model. One of them presents the data used in the practical implementation, the global model architecture and the sub-model operation, whereas the other one lists the assumptions required before building a metamodel, such as the definition of a convergence criteria for the initial model, or the definition of a time budget so as to construct the design of experiments.

The fourth part of this thesis deals with the practical implementation of previous methods on the reinsurance pricing model. Various metamodels are first built with a one-shot design, applying kriging and splines interpolation. Numerous advantages of kriging lead us to keep this method in a sequential design approach. This way, we build two kriging models in order to improve the overall prediction quality, and then, we build a third one dedicated to reinsurance efficiency optimization. Previous models having been built with only two input variables, we choose to include two additional variables in order to design a last efficiency optimization model.

The fifth and last part is meant to give an overview of the limits of our work, sometimes related to the methods applied, also giving some lines of improvement as well as new focal areas.

Remerciements

Je tiens tout d'abord à remercier mon tuteur au sein d'Addactis France, Youness BENALI, pour ses nombreux conseils et sa disponibilité tout au long de mon alternance.

Je remercie particulièrement Nabil RACHDI, responsable de la practice *Data Science*, qui a contribué à l'élaboration de ce mémoire par son aide précieuse, son suivi régulier et sa pédagogie.

Mes remerciements s'adressent également à Benjamin POUDRET, responsable de la Practice *Modeling & Risk* au sein du pôle *P&C*, qui s'est toujours montré à l'écoute pendant la réalisation de ce mémoire. Je souhaite aussi remercier Isabelle JOULIN et Eve ZURFLUH pour leurs relectures et conseils, ainsi que l'ensemble des consultants du pôle *P&C* et plus largement d'Addactis pour leur accueil, leur disponibilité et l'aide qu'ils ont pu m'apporter au cours de ces travaux.

Pour terminer, je tiens à remercier Stéphane LOISEL, mon tuteur académique, pour ses conseils et l'apport de son expérience sur ce mémoire.

Note de Synthèse

Les méthodes utilisées pour la construction d'un métamodèle

Dans ce mémoire, deux techniques d'interpolation spatiale sont utilisées : l'interpolation par splines bicubiques et le krigeage. La première est l'extension à deux dimensions de l'interpolation par splines cubiques, qui consiste à relier les points d'observations $(x_i, f(x_i))_{i \in [1, N]}$ du plan d'expérience par des polynômes de degré trois, constituant ainsi une interpolation polynomiale par morceaux, où l'estimation des paramètres passe par la résolution d'un système à N inconnues. Le krigeage est une méthode d'interpolation spatiale linéaire, basée sur l'estimateur linéaire en les observations qui garantit une variance d'estimation minimale. Cette méthode suppose que la variable d'étude peut s'écrire sous la forme d'un processus gaussien $Z(x) = \mu(x) + \delta(x)$, $x \in D$, où μ est une fonction déterministe représentant l'espérance de Z , δ est un processus aléatoire et stationnaire, et D est l'espace d'observation. Cette méthode a plusieurs intérêts, elle permet notamment de prendre en compte une dépendance spatiale des données, et, à travers la modélisation stochastique sous-jacente, offre une mesure d'incertitude autour des prédictions effectuées par le modèle.

Appliquer ces techniques d'interpolation nécessite la définition d'un plan d'expérience, i.e. répartir dans l'espace les points initiaux sur lesquels lancer le modèle de base, afin d'obtenir les résultats qui serviront de base à l'interpolation.

Il existe deux grandes catégories de méthodes pour construire un plan d'expérience. La première est une approche de construction par plan « fixe » (ou *one-shot design*) : dans cette approche, un nombre de points et une méthode d'échantillonnage sont définis, et les points sont positionnés en conséquence dans l'espace. Le modèle est ensuite exécuté sur chacun de ces points. Dans ce mémoire, nous avons utilisé trois méthodes d'échantillonnage : l'échantillonnage aléatoire, factoriel, et par hypercube latin. La deuxième grande catégorie est la construction d'un plan d'expérience séquentiel (ou *sequential design*). Dans cette approche, les points ne sont pas tous définis simultanément : quelques points sont échantillonnés selon une certaine méthode, le modèle est exécuté sur ces points, puis, en se basant sur les résultats obtenus et sur un certain critère de sélection, les points suivants sont choisis au fur et à mesure des itérations. Une condition d'arrêt est ensuite définie, comme l'atteinte d'une précision suffisante du métamodèle ou d'un nombre de points maximal. En fonction du critère utilisé pour la sélection des points, la construction séquentielle peut être dépendante de la technique d'interpolation utilisée.

Le krigeage possède, pour la construction séquentielle, plusieurs critères prédéfinis de sélection des points, basés sur la modélisation stochastique par processus gaussien. Certains sont dédiés à l'amélioration globale de la qualité prédictive du modèle de krigeage en question, mais il est aussi possible de définir un critère de sélection dédié à l'optimisation, permettant d'améliorer la qualité du modèle de krigeage dans les zones susceptibles de contenir un optimum. La qualité de ces critères et d'autres avantages du krigeage nous conduirons, dans l'application pratique, à le préférer

aux splines pour la construction séquentielle.

Pour évaluer la qualité de prédiction d'un métamodèle, nous utiliserons une base de validation (ou *test set*) pour mesurer l'erreur quadratique moyenne de prédiction (EQMP) et le coefficient Q^2 , défini par analogie au coefficient de détermination R^2 comme :

$$Q^2 = 1 - \frac{\sum_{i=1}^{n_{test}} (\hat{f}(x_i) - f(x_i))^2}{\sum_{i=1}^{n_{test}} (f(x_i) - m_f)^2} = 1 - \frac{EQMP}{\frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (f(x_i) - m_f)^2}$$

avec $m_f = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} f(x_i)$ la moyenne des observations, et donc $\frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (f(x_i) - m_f)^2$ la variance empirique du modèle f estimée sur la base de validation. \hat{f} représente le métamodèle. Ainsi, plus le coefficient Q^2 est proche de 1, plus la qualité prédictive du métamodèle est bonne.

La tarification par simulations des traités en excédent de sinistre

Notre objectif est de construire un métamodèle de tarification des traités en excédent de sinistre, initialement réalisée par simulations, et de réduire ainsi considérablement les temps de calcul tout en facilitant l'analyse et l'exploration des résultats. Il est donc essentiel, avant de construire un tel métamodèle, de s'attarder d'une part sur le fonctionnement du module de tarification par simulations, et d'autre part sur les données utilisées dans l'application pratique.

Le schéma ci-dessous synthétise les différentes étapes de la tarification par simulations d'un traité en excédent de sinistre (ou XS) :

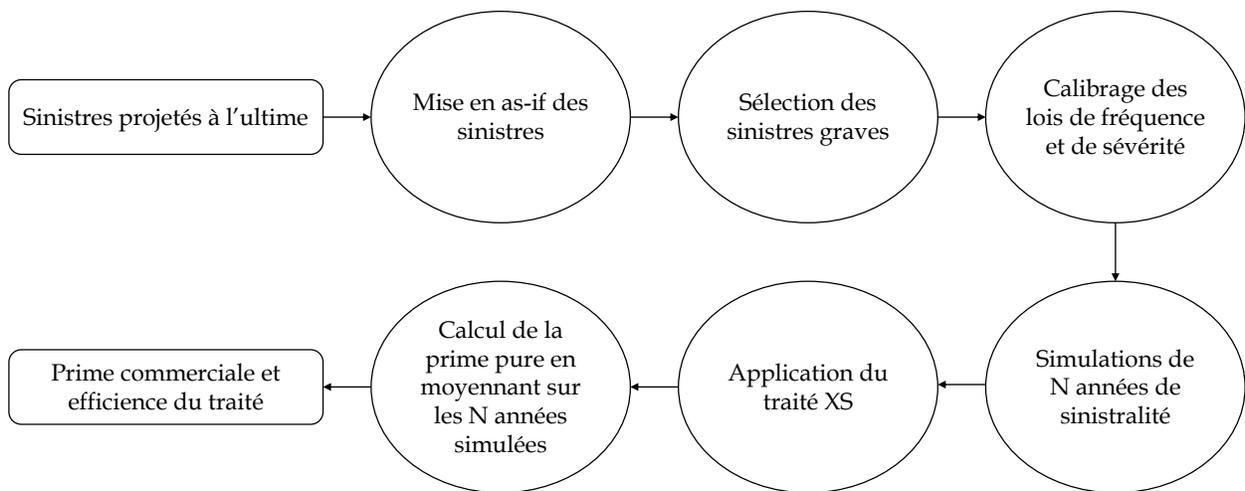


FIGURE 1 – Fonctionnement du module de tarification par simulations

Dans l'application pratique, nous avons utilisé les données de sinistralité d'une cédante sur la branche incendie, sur une période de six ans, de 2012 à 2018. Les sinistres ont été redressés via l'indice FFB (Fédération Française du Bâtiment) des coûts à la construction. Nous avons ensuite extrait les sinistres graves, dans notre cas les sinistres supérieurs à un seuil $s = 1\,150\,000$ €. En effet, notre étude ne concernant que les traités XS, seuls les sinistres graves nous intéressent, les sinistres attritionnels n'entrant pas dans le champ d'application de ces traités. A partir de cette liste composée de 114 sinistres graves, une loi de fréquence et une loi de sévérité ont été déterminées, via un module de calibrage de lois basé sur des tests statistiques d'adéquation et une analyse graphique. Pour la fréquence, nous avons retenu une loi binomiale négative, et pour la sévérité, une loi de Pareto généralisée. A cette étape, tous les paramètres du modèle sont définis pour permettre les simulations de N années de sinistralité et tarifier un traité XS quelconque.

Les hypothèses nécessaires à la construction d'un métamodèle de tarification

Avant de se lancer dans la conception du plan d'expérience, et d'exécuter le modèle initial sur ce plan dans le but construire un métamodèle, il reste quelques étapes à effectuer.

La première étape consiste à choisir les variables d'entrée et de sortie du métamodèle. Le modèle de tarification prenant en entrée un nombre conséquent de variables et calculant plusieurs indicateurs en sortie, nous considérerons un nombre restreint de paramètres pour la construction du métamodèle. Ainsi, notre choix s'est porté sur l'efficacité d'un traité de réassurance comme indicateur de sortie, définie comme $E = -(1 - \rho)R + \alpha G$, avec R le résultat du traité, G le gain en capital associé au traité, ρ le taux d'imposition et α le taux associé au coût du capital. Concernant les variables d'entrée, nous avons choisi de retenir pour commencer deux variables intrinsèques à un traité XS et ayant un impact direct sur la sinistralité : la priorité et la portée.

Le modèle initial évoluant dans un cadre stochastique, si le nombre de simulations n'est pas correctement calibré, les résultats obtenus d'un lancement à l'autre du modèle ne donneront pas nécessairement les mêmes résultats. La deuxième étape consiste donc à s'assurer de la convergence des indicateurs du modèle en définissant un nombre de simulations suffisant, qui nous permettra d'obtenir des résultats quasi-déterministes. Ainsi, nous avons utilisé l'inégalité de Bienaymé-Tchebychev pour déterminer un intervalle de confiance autour de la prime pure du traité, défini de sorte à garantir un écart relatif entre l'estimateur et la vraie prime pure inférieur à 1%, avec un niveau de confiance de 95%. Le nombre de simulations nécessaires pour rester dans cet intervalle dépend des caractéristiques du traité considéré. Il nous faut donc borner les espaces de définition des variables, de sorte à définir un nombre maximal de simulations qui permettra le respect de l'intervalle de confiance quelque soit le traité. En considérant $priority \in [1\,150\,000, 15\,000\,000]$ et $portée \in [500\,000, 30\,000\,000]$, nous avons obtenu un nombre de simulations requis égal à 1 500 000. La détermination des bornes des espaces de définition est détaillée à la fin de la section 3.2.1.

La troisième et dernière étape traite de la définition d'un budget en temps de calcul, qui va nous permettre de déterminer le nombre de points (i.e. les traités) qui constitueront le plan d'expérience.

Une fois ce « budget » fixé, il ne reste en effet qu'à estimer les temps de calcul du modèle pour 1 500 000 simulations en fonction du nombre de points en entrée. Cette estimation nous a conduit à retenir un nombre total de 135 points pour respecter un budget temporel défini à une heure. Etant donné qu'il nous faut constituer une base de validation pour évaluer la qualité d'un métamodèle, nous ne pouvons utiliser la totalité de ces 135 points pour concevoir le plan d'expérience. De plus, l'objectif est ici de montrer qu'un métamodèle de qualité peut être construit à partir d'un nombre minimal de points de départ. Ainsi, nous avons choisi de conserver seulement 25 points pour concevoir le plan d'expérience, utilisé dans la construction du métamodèle, et 110 points pour constituer la base de validation. Cette dernière doit être assez fournie pour permettre d'obtenir des valeurs d'efficience dans l'ensemble des zones de l'espace, afin de ne pas biaiser l'évaluation de la qualité d'un métamodèle.

Les résultats obtenus après application des différentes méthodes de construction

L'ensemble des hypothèses préalables à la construction d'un métamodèle étant réunies, nous avons pu nous concentrer sur l'application pratique de ces méthodes au modèle de tarification par simulations.

Dans un premier temps, nous avons construit plusieurs métamodèles via un plan d'expérience fixe. Ainsi, trois modèles de krigeage ont été conçus en utilisant les méthodes d'échantillonnage aléatoire, factoriel et par hypercube latin. Une modélisation par splines bicubiques a également été effectuée à partir d'un échantillonnage factoriel, les splines étant plus restrictives que le krigeage quant à la forme du plan d'expérience. Ces métamodèles ont donné les résultats suivants sur la base de validation, conformément aux mesures d'erreur utilisées dans ce mémoire :

Métamodèles construits - 25 points				
Interpolation	Krigeage			Splines bicubiques
Echantillonnage	Aleatoire	Latin	Factoriel	Factoriel
EQMP	309 707 763	455 336 327	4 550 935 460	1 033 081 498
EQMP ^{1/2}	17 599	21 339	67 461	32 142
Q ²	0,987	0,981	0,815	0,958

FIGURE 2 – EQMP et Q² selon le type de métamodèle

Dans cette configuration initiale à 25 points, le modèle de krigeage construit par échantillonnage aléatoire a produit les meilleurs résultats. Néanmoins, afin de montrer l'impact de la volatilité de cette méthode d'échantillonnage, nous avons réduit le nombre de points du plan d'expérience à 16 et obtenu les résultats suivants :

Métamodèles construits - 16 points				
Interpolation	Krigage			Splines bicubiques
Echantillonnage	Aleatoire	Latin	Factoriel	Factoriel
EQMP	4 858 264 163	1 300 673 745	9 015 740 796	3 670 319 585
EQMP ^{1/2}	69 701	36 065	94 951	60 583
Q ²	0,802	0,947	0,633	0,851

FIGURE 3 – EQMP et Q² selon le type de métamodèle

Cette fois, le modèle de krigage construit via un hypercube latin s’est avéré nettement plus performant que les autres métamodèles, mettant notamment en avant la stabilité de cette méthode d’échantillonnage par rapport à l’échantillonnage aléatoire. Cette réduction du nombre de points nous a également permis d’observer la souplesse du krigage, méthode non-paramétrique, moins rigide que la modélisation par splines.

Dans un deuxième temps, nous avons utilisé la construction séquentielle pour concevoir de nouveaux modèles de krigage. Le krigage a été privilégié pour la construction séquentielle en raison de l’existence de critères prédéfinis efficaces pour la sélection de nouveaux points, de sa robustesse quant au nombre de points initiaux, de sa qualité de prédiction, mais aussi en raison de son aspect non-paramétrique. Ce dernier aspect permet d’étendre aisément ce type de modèle à un problème de dimension supérieure, là où la modélisation paramétrique par splines impliquerait d’être complètement redéfinie.

Pour commencer, nous avons utilisé la construction séquentielle pour tenter d’améliorer la qualité de prédiction globale des modèles de krigage, via un critère de réduction ponctuelle de la variance. Le plan d’expérience de 25 points a été scindé en deux parties, l’une contenant 10 points générés via un échantillonnage hypercube latin, méthode ayant donné les meilleurs résultats pour un nombre de points réduit, l’autre contenant 15 points qui seront sélectionnés séquentiellement, au fur et à mesure des itérations. Cependant, nous avons pu observer des résultats moins satisfaisants que ceux espérés, en raison du compromis effectué entre la méthode d’échantillonnage, le nombre de points fixes et le nombre de points séquentiels. En utilisant un autre critère de sélection, basé sur la réduction de la variance du modèle de krigage au sens d’une intégrale, nous avons pu contourner cette difficulté et améliorer sensiblement les résultats, jusqu’à obtenir le métamodèle le plus stable et précis construit jusqu’alors dans l’application pratique.

Métamodèles - 25 points				
Interpolation	Krigage par plan fixe			Krigage séquentiel
Echantillonnage	Aleatoire	Latin	Factoriel	Latin (10 points)
EQMP	309 707 763	455 336 327	4 550 935 460	294 523 688
EQMP ^{1/2}	17 599	21 339	67 461	17 162
Q ²	0,987	0,981	0,815	0,988

FIGURE 4 – EQMP et Q^2 selon la méthode de construction

Ensuite, nous nous sommes penchés sur la construction séquentielle d'un modèle de krigage dédié à la recherche d'optima d'efficacité de réassurance. Pour plusieurs raisons, notamment celles ayant mis en défaut le premier critère de sélection utilisé dans la partie précédente, basé sur une réduction ponctuelle de la variance, nous avons choisi d'utiliser ici un échantillonnage factoriel pour la génération des points de départ plutôt qu'un échantillonnage par hypercube latin. Nous sommes donc partis d'un échantillon factoriel à $9 = 3^2$ points, en changeant simplement le critère de sélection utilisé : ici, nous nous sommes servis du critère d'amélioration espérée, ou *expected improvement*. Le modèle de krigage ainsi construit nous a permis d'estimer un optimum d'efficacité, obtenu pour un traité de réassurance de priorité égale 2 729 380 € et de portée égale à 30 000 000 €. La prédiction d'efficacité pour ce traité s'est avérée très proche de la vraie valeur, l'écart relatif étant de 0,016%. Pour terminer, nous avons comparé cet optimum prédit avec ceux prédits par plusieurs modèles de krigage construits par plan d'expérience fixe à 25 points via un échantillonnage aléatoire ou latin. Quelque soit le modèle de krigage aléatoire ou latin, le krigage séquentiel offrait d'une part une erreur plus faible entre la prédiction et la valeur réelle d'efficacité, et d'autre part trouvait un traité réellement plus efficace.

Dans un troisième temps, nous avons construit un dernier métamodèle, un modèle de krigage séquentiel dédié à l'optimisation, comme précédemment, mais en ajoutant cette fois-ci deux nouvelles variables d'entrée : l'AAD (Annual Aggregate Deductible) et l'AAL (Annual Aggregate Limit). Ce dernier modèle nous a permis de prédire un nouvel optimum d'efficacité, toujours avec précision, l'écart relatif avec la vraie valeur restant inférieur à 0,5%. L'interprétation du résultat numérique, c'est à dire les caractéristiques traités obtenus, s'est avérée délicate car ce traité est situé sur les bornes des espaces de définition de trois des quatre variables d'entrée (la portée, l'AAD et l'AAL). Néanmoins, nous avons pu expliquer ce résultat, propre à la structure du modèle de tarification et la définition de l'efficacité de réassurance. De plus, d'un point de vue méthodologique, nous avons pu prouver la validité et l'efficacité de la démarche employée.

Les limites de la modélisation utilisée et les perspectives d'évolution

Pour terminer, nous avons consacré une partie aux limites des travaux effectués, qui ont pu être identifiées lors de l'application des différentes méthodes de modélisation. Les principales concernent la sensibilité de la construction séquentielle au critère de sélection utilisé, la sensibilité des résultats à l'architecture du modèle sous-jacent, l'aspect stochastique du modèle de tarification, ou encore

la définition du critère d'efficience de réassurance. De nouveaux axes d'études sont également envisagés, comme l'application d'un nouveau critère de sélection séquentielle.

Table des matières

Introduction	13
1 Métamodèles et interpolation	14
1.1 Le plan d'expérience	14
1.1.1 Construction par plan d'expérience fixe	14
1.1.2 Construction séquentielle	19
1.2 Interpolation bicubique par splines	23
1.2.1 Splines cubiques	23
1.2.2 Généralisation en dimension deux	26
1.3 Interpolation par krigeage	26
1.3.1 Prédiction par krigeage	27
1.3.2 Estimation des paramètres par maximum de vraisemblance	28
1.3.3 Modélisation du variogramme	28
1.4 Evaluation de la qualité d'un métamodèle	30
1.4.1 Base de validation	30
1.4.2 Mesures d'erreur	31
1.5 Optimisation numérique par algorithme génétique	32
2 Le modèle de tarification en réassurance	35
2.1 Architecture du modèle	35
2.2 Données attendues en entrée du modèle	37
2.3 Préparation des données pour l'application pratique	38
2.3.1 Retraitements effectués	38
2.3.2 Hypothèses retenues	38
2.4 Fonctionnement du modèle	39
2.5 Lexique de réassurance	46
3 Hypothèses préalables à l'application de méthodes d'interpolation	48
3.1 Choix des variables à étudier	48
3.2 Nombre de simulations à effectuer	50
3.2.1 Détermination théorique du nombre de simulations	50
3.2.2 Validation empirique	54
3.3 Nombre de points initiaux	54
4 Construction d'un métamodèle de tarification en réassurance	58
4.1 Construction via un plan d'expérience fixe	58
4.1.1 Krigeage	59
4.1.2 Splines bicubiques	67
4.1.3 Comparaison des méthodes	69

4.1.4	Réduction du nombre de points de départ	70
4.1.5	Avantages du krigeage	73
4.2	Construction séquentielle par krigeage & Optimisation	73
4.2.1	Critères usuels pour la construction séquentielle par krigeage	74
4.2.2	Critère pour l'optimisation des traités	74
4.2.3	Nombre de points pour la construction séquentielle	75
4.2.4	Schéma du processus de construction séquentielle	75
4.2.5	Construction séquentielle du modèle de krigeage par réduction ponctuelle de la variance	76
4.2.6	Construction séquentielle du modèle de krigeage par réduction intégrale de la variance	79
4.2.7	Construction séquentielle du modèle de krigeage pour l'optimisation	84
4.3	Ajout des variables <i>AAD</i> et <i>AAL</i> dans la conception du métamodèle	89
4.3.1	Modèle de krigeage séquentiel pour l'optimisation en dimension quatre	89
4.4	Conclusion sur les résultats obtenus	92
4.5	Cas d'usage simplifié pour la construction d'un métamodèle d'optimisation	94
5	Limites et perspectives liées à la démarche et la modélisation utilisées	95
	Conclusion	98
	Table des figures	99
	Bibliographie	101

Introduction

Aujourd'hui, de nombreux modèles actuariels sont coûteux en ressources informatiques, notamment en raison de temps d'exécution conséquents et du besoin de l'utilisation de serveurs de calcul. Il peut donc s'avérer intéressant de disposer de versions simplifiées de certains modèles, visant à reproduire au mieux les résultats de ces derniers, avec des temps de calcul beaucoup plus courts.

Ce mémoire se concentre autour d'un modèle de tarification en réassurance, spécifiquement sur le module dédié à la tarification par simulations des traités en excédent de sinistres, coûteuse en temps d'exécution. L'idée sera donc de se donner un « budget » limité en temps de calcul, qui correspondra à un certain nombre d'exécutions du modèle, de sélectionner ensuite les points i.e. les traités de réassurance sur lesquels lancer le modèle, pour finalement effectuer une interpolation à partir des résultats obtenus sur ces points. De cette interpolation, nous tirerons une version simplifiée du modèle initial, plus rapide à exécuter et nous offrant la possibilité d'étudier facilement l'impact des paramètres d'entrée du modèle sur les résultats. La variable résultat considérée dans ce mémoire sera l'efficacité d'un traité de réassurance. Nous définirons cette grandeur dans la partie consacrée à la présentation du modèle de réassurance.

Dans cette approche par construction d'un « métamodèle » ou « modèle de substitution », la complexité réside dans le choix des points de départ, tout comme dans le choix de la méthode d'interpolation. Nous présenterons et comparerons différentes démarches relatives à ces deux sujets. Concernant les méthodes de sélection des points de départ, nous étudierons différentes configurations possibles des points dans l'espace, visant à explorer ce dernier de la façon la plus complète possible. Comme techniques d'interpolation, nous utiliserons et mettrons en regard deux méthodes, l'interpolation par splines bicubiques et l'interpolation par krigeage.

Nous élargirons également la réflexion en présentant une méthode permettant la recherche d'optima dans un modèle, qui pourra notamment être appliquée, dans le cas du modèle de tarification en réassurance, à la recherche de traités optimaux en termes d'efficacité. Cette méthode se base sur une sélection « séquentielle » des points de départ.

1 Métamodèles et interpolation

La conception d'un métamodèle, appelé *surrogate modelling* en anglais, consiste, à partir d'un modèle initial et d'un certain nombre de couples entrées-sorties associés, en la construction d'un nouveau modèle simplifié, dont les résultats doivent approcher au mieux ceux du modèle initial. La construction d'un tel modèle est notamment utile lorsque les temps de calcul du modèle initial sont particulièrement longs, permettant ainsi d'obtenir un nouveau modèle beaucoup plus rapide à l'exécution. Dans le cas d'un modèle initial au fonctionnement complexe, l'application de cette méthode procure également une expression analytique plus simple, offrant ainsi une interprétation plus aisée des liens entre variables d'entrée et de sortie, ce qui facilite l'étude de la structure du modèle.

Pour concevoir un métamodèle, il est nécessaire d'avoir à disposition plusieurs couples entrées-sorties du modèle initial, ce qui implique d'avoir au préalable exécuté ce dernier sur les points choisis en entrée. L'idée est d'obtenir le meilleur métamodèle possible, calibré à partir d'un nombre minimal de couples entrées-sorties du modèle initial, étant donné que chaque exécution est coûteuse en temps de calcul. Ces couples serviront de base à la construction du nouveau modèle. Plus précisément, la démarche est constituée des deux étapes ci-dessous :

- Choisir les paramètres du modèle à faire varier, i.e. les variables d'entrées à partir desquelles construire le métamodèle : lorsqu'un modèle possède un grand nombre de paramètres en entrée, souvent, nous ne souhaitons observer l'impact que de certaines variables sur les résultats. Sélectionner un nombre restreint de paramètres à faire varier permettra aussi une analyse plus commode des résultats, notamment de manière graphique. De plus, certaines structures de métamodèles possèdent des contraintes sur le nombre de variables d'entrée.
- Construire le plan d'expérience, ou *design of experiments* : une fois la première étape effectuée, il faut alors définir les points initiaux (i.e. les valeurs de paramètres d'entrée) sur lesquels évaluer le modèle. Pour ce faire, il existe deux méthodologies :
 - Une approche dite « par plan d'expérience fixe » (*one-shot design*).
 - Une approche dite « séquentielle » (*sequential design*).

Ces deux méthodologies vont être détaillées dans les parties suivantes.

1.1 Le plan d'expérience

1.1.1 Construction par plan d'expérience fixe

Dans l'approche par *one-shot design*, l'idée est de choisir un échantillon de points répartis de manière à remplir au mieux l'espace, au sens d'un certain critère de complétion. L'objectif est de pouvoir observer les sorties du modèle sur l'ensemble du domaine de définition des variables d'entrée. Nous allons maintenant présenter trois des principales méthodes d'échantillonnage généralement utilisées, en prenant comme espace de départ un hypercube de la forme $[0, 1]^d$, avec d le

nombre de dimensions, i.e. le nombre de variables d'entrée retenues pour la modélisation.

- **Echantillonnage aléatoire** : cette méthode consiste à générer aléatoirement N points dans l'hypercube considéré, de manière uniforme. Nous pouvons rapidement observer que l'efficacité de cette méthode dépend du nombre de points générés. Plus ce nombre est important, plus la méthode produit de bons résultats. Ainsi, l'échantillonnage aléatoire peut s'avérer intéressant lorsqu'il est possible de générer de larges échantillons de points dans l'espace de définition.

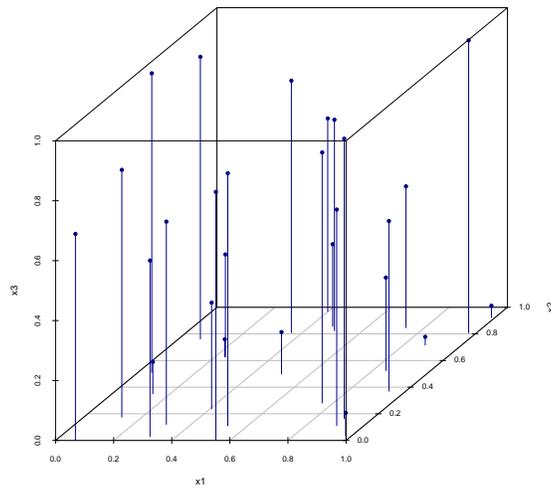


FIGURE 5 – Hypercube généré par échantillonnage aléatoire, pour $d = 3$ et $N = 27$

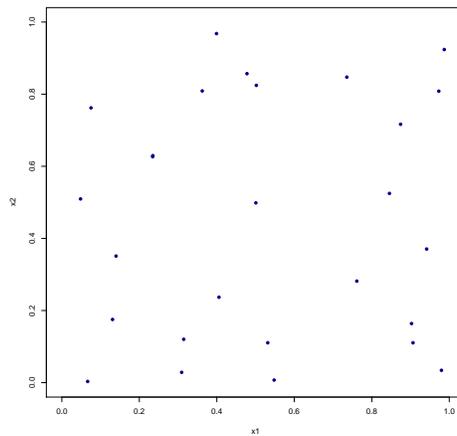


FIGURE 6 – Projection sur le plan (x_1, x_2)

- **Echantillonnage par plan factoriel** : cette méthode consiste à répartir $N = k^d$ points dans une configuration optimale au sens du critère dit « maximin » appliqué à la distance euclidienne entre deux points. Posons $d_2(x, y) = \sqrt{\sum_{i=1}^d (x^i - y^i)^2}$, la distance euclidienne entre deux points x et y . La vérification de ce critère consiste en la maximisation de la distance minimale d_2 sur l'ensemble des N points définis. Autrement dit, le plan factoriel est défini comme la configuration P (i.e. l'ensemble de N points) telle que

$$P = \arg \max_{P \in \mathcal{C}} \left(\min_{x, y \in P} d_2(x, y) \right)$$

où \mathcal{C} représente l'ensemble des configurations possibles.

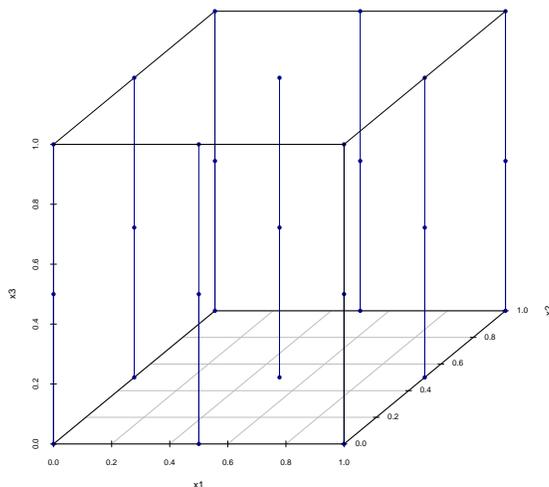


FIGURE 7 – Hypercube généré par plan factoriel, pour $d = 3$ et $N = 27$

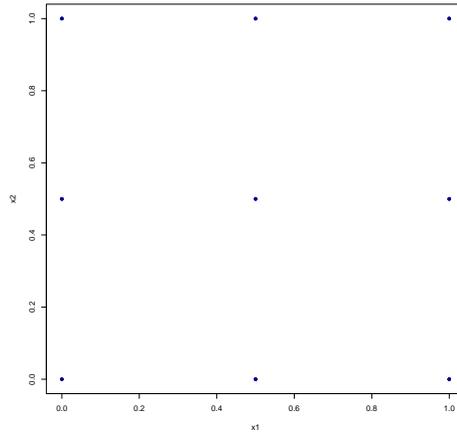


FIGURE 8 – Projection sur le plan (x_1, x_2)

Comme il est possible de le déduire d'après la configuration visible sur la figure 7, cette méthode positionne les points de la même manière pour les trois variables considérées, permettant d'éviter l'introduction d'un biais dans la configuration des points selon les variables.

- **Echantillonnage par hypercube latin** : cette méthode consiste à partitionner l'hypercube de départ en n^d sous-hypercubes identiques. Chaque dimension est donc divisée en n intervalles de même longueur. L'hypercube latin est alors construit de telle sorte que pour chacune des d dimensions, chacun des n intervalles soit associé à exactement un point. Autrement dit, chaque intervalle doit contenir une et une seule coordonnée : il ne doit pas y avoir de superposition des points lors de la projection sur un espace de dimension $d - 1$.

Cette méthode d'échantillonnage offre ainsi plusieurs configurations possibles. Elle est notamment intéressante du fait de la propriété d'unicité des coordonnées (par dimension) qu'elle garantit. La complexité théorique réside dans l'optimisation de l'hypercube latin afin d'obtenir un bon remplissage de l'espace. Par exemple, l'hypercube latin ci-dessous n'est pas optimal au sens de l'exploration du domaine de définition des variables :

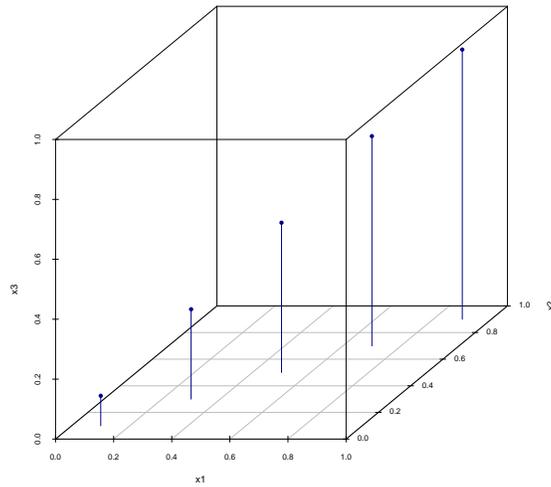


FIGURE 9 – Exemple d’hypercube latin non-optimal, pour $d = 3$ et $N = 5$

Dans ce mémoire, pour l’application pratique, la conception des hypercubes latins sera effectuée sous R via la fonction `maximinLHS` du package `lhs`, qui intègre l’optimisation de la configuration de l’hypercube. Cette optimisation est effectuée en positionnant les points dans l’hypercube selon le critère maximin décrit précédemment, venant s’ajouter aux contraintes posées par la méthode de construction. Voici un exemple d’hypercube latin construit avec cette fonction :

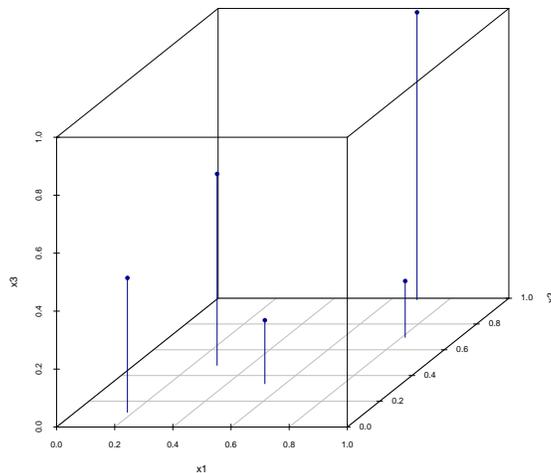


FIGURE 10 – Exemple d’hypercube latin optimal avec `maximinLHS`, pour $d = 3$ et $N = 5$

A titre de comparaison avec les hypercubes précédemment construits par les méthodes aléa-

toire et factorielle, voici un hypercube latin construit pour $d = 3$ et $N = 27$, avec la fonction maximinLHS :

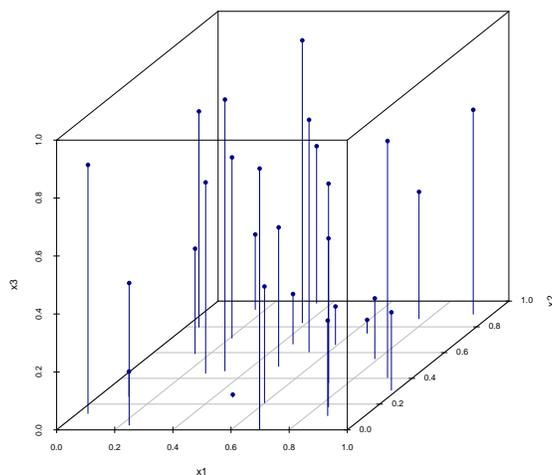


FIGURE 11 – Hypercube latin optimal avec maximinLHS, pour $d = 3$ et $N = 27$

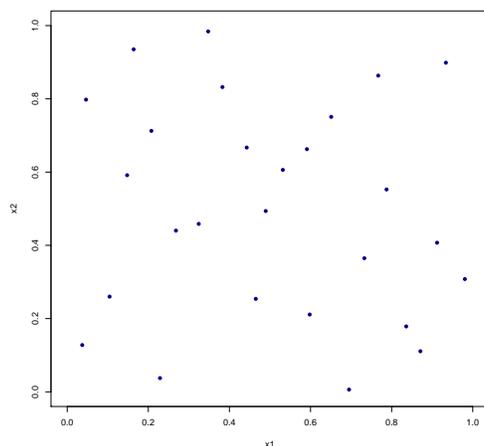


FIGURE 12 – Projection sur le plan (x_1, x_2)

1.1.2 Construction séquentielle

Dans l'approche séquentielle, l'idée première est de définir un nombre très réduit de points initiaux sur lesquels exécuter le modèle, puis de se baser sur les résultats obtenus pour construire un premier métamodèle et ajouter ensuite un nouveau point au plan d'expérience. Le modèle est alors réévalué sur ce point et le métamodèle est ajusté en conséquence. Ce schéma est répété

jusqu'à ce qu'un certain critère soit vérifié, par exemple l'atteinte d'un nombre de points maximum dans le plan d'expérience ou d'une précision suffisante du métamodèle. Généralement, les points déterminés au fur et à mesure des itérations du modèle sont choisis dans les zones où la variabilité du modèle initial est importante, afin d'améliorer la précision du nouveau modèle dans ces zones. Cependant, il est possible de définir un critère différent pour la sélection des nouveaux points, afin d'obtenir un métamodèle (ou modèle de substitution) plus précis dans des zones d'intérêt bien choisies.

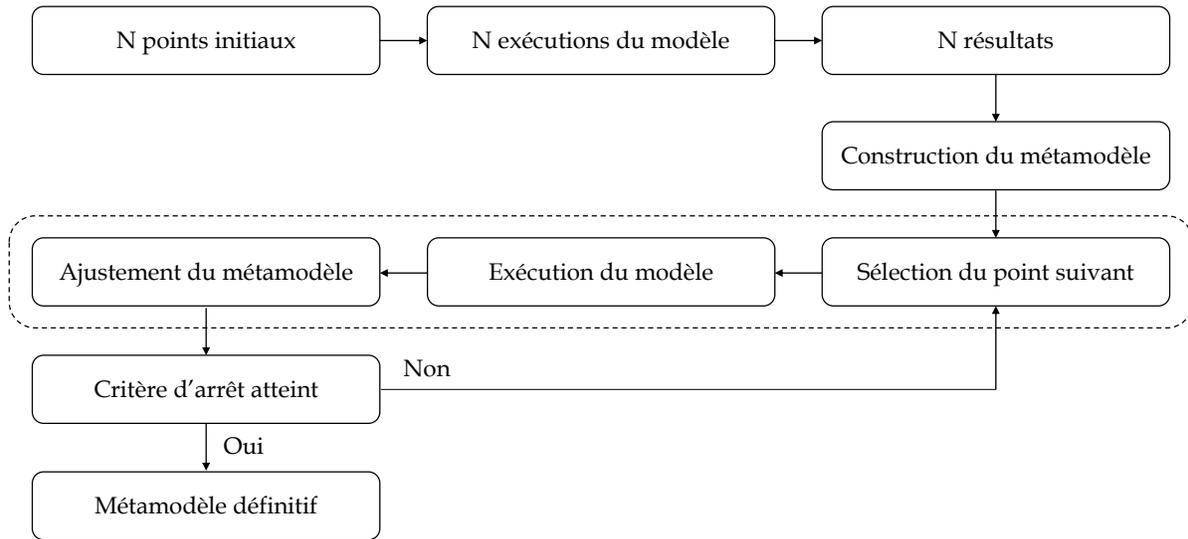


FIGURE 13 – Construction séquentielle d'un métamodèle

Par rapport à la construction par plan d'expérience fixe, la construction séquentielle offre la possibilité de prendre en compte de nouvelles sorties du modèle initial dans la construction du métamodèle, et d'affiner au fur et à mesure des itérations ce modèle de substitution en fonction de critères définis selon l'usage.

- **Premier exemple** : exécution d'un modèle noté f sur dix points initiaux avec $d = 1$, puis sélection d'un nouveau point dans une zone où les résultats du modèle présente de grandes variations.

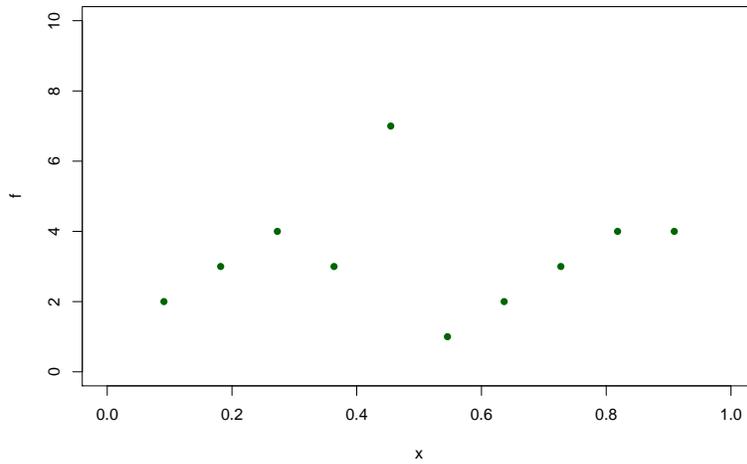


FIGURE 14 – Résultats du modèle f sur les dix points initiaux

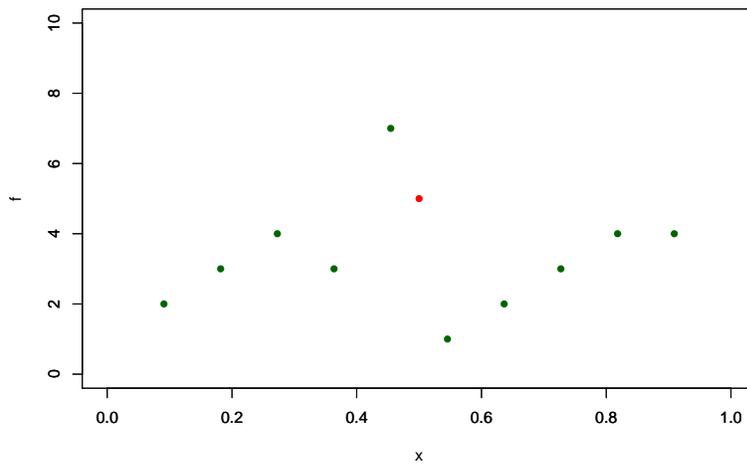


FIGURE 15 – Ajout du résultat obtenu sur le nouveau point

Il est souvent judicieux de sélectionner de nouveaux points à des endroits où la variabilité du modèle est importante. En effet, dans ces zones, l'incertitude sur les prédictions du métamodèle sera potentiellement plus grande que dans des zones où le modèle initial a un comportement plus linéaire, donc plus facile à interpoler.

- **Deuxième exemple** : exécution d'un modèle noté g sur dix points initiaux avec $d = 1$, puis sélection d'un nouveau point dans une zone d'intérêt où les résultats du modèle présente de grandes variations, en prenant en compte un domaine d'intérêt défini par $g \geq 4$.

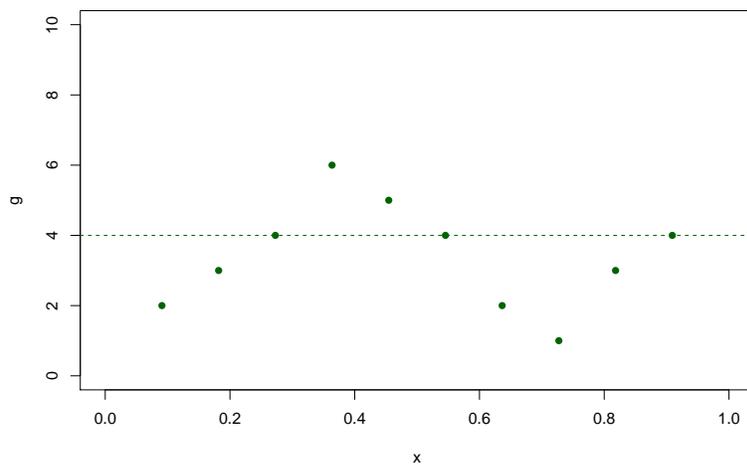


FIGURE 16 – Résultats du modèle g sur les dix points initiaux

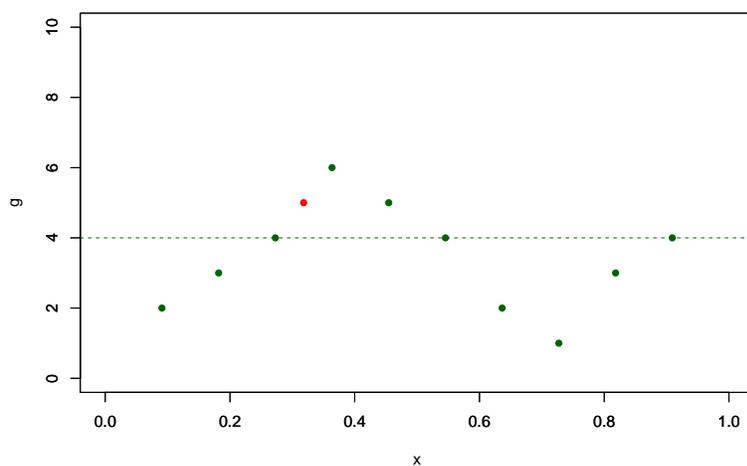


FIGURE 17 – Ajout du résultat obtenu sur le nouveau point

- **Troisième exemple** : il est également possible de choisir q nouveaux points à la fois. Ci-dessous, sélection de deux nouveaux points dans la situation décrite au deuxième exemple (figure 16).

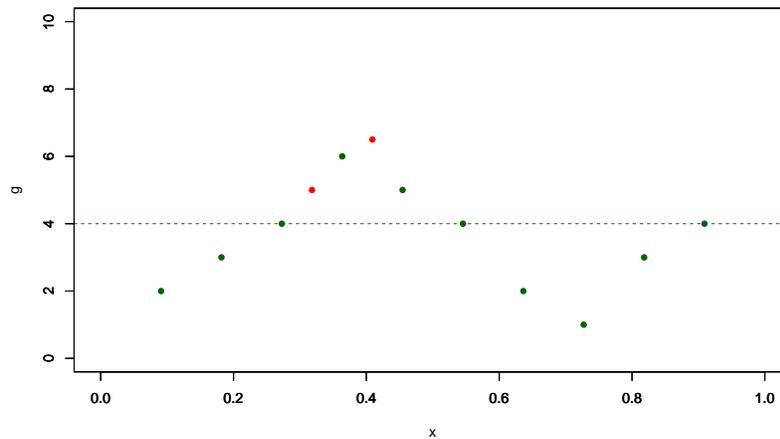


FIGURE 18 – Résultats après sélection de deux nouveaux points à la fois

Dans la partie 4.2, nous détaillerons les différentes méthodes utilisées pour la sélection des nouveaux points, qui peuvent différer selon les techniques d’interpolation.

Une fois choisie la méthode de construction du plan d’expérience, il reste à définir la méthode d’interpolation. Celle-ci déterminera la structure du métamodèle. Nous allons présenter dans les parties suivantes deux techniques d’interpolation, qui seront ensuite utilisées dans l’application pratique.

1.2 Interpolation bicubique par splines

1.2.1 Splines cubiques

De manière générale, l’interpolation par splines consiste à relier les points obtenus en sortie du modèle par des polynômes. C’est donc une interpolation polynomiale par morceaux. Nous allons ici présenter le cas où l’interpolation est effectuée par un polynôme cubique par morceaux. En se donnant un échantillon $(x_i)_{i \in \llbracket 1, N \rrbracket}$, de points initiaux sur lesquels exécuter le modèle initial noté f , nous obtenons après exécution les couples $(x_i, f(x_i))_{i \in \llbracket 1, N \rrbracket}$, sur lesquels va être basée l’interpolation.

Ci-dessous un exemple d’interpolation par une spline cubique sur les dix couples représentés en figure 16 :

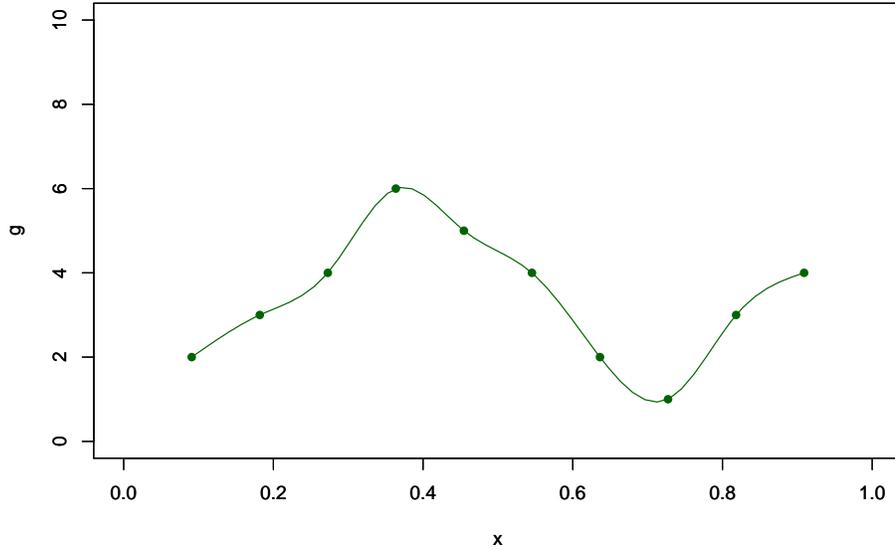


FIGURE 19 – Exemple de spline cubique

L'interpolation par splines cubiques, s'effectuant via un polynôme de degré trois P_i au sein de chaque intervalle $[x_i, x_{i+1}]$ avec $i \in \llbracket 1, N - 1 \rrbracket$, impose plusieurs conditions :

- La spline doit passer par les couples $(x_i, f(x_i))$, ce qui impose la condition :

$$P_i(x_i) = f(x_i) \text{ et } P_i(x_{i+1}) = f(x_{i+1}), \forall i \in \llbracket 1, N - 1 \rrbracket. \quad (1)$$

- La dérivée première de la spline doit être continue, ce qui impose aux points initiaux $(x_i)_i$ la condition :

$$P'_i(x_{i+1}) = P'_{i+1}(x_{i+1}), \forall i \in \llbracket 1, N - 2 \rrbracket. \quad (2)$$

- La dérivée seconde de la spline doit être continue, ce qui impose aux points initiaux $(x_i)_i$ la condition :

$$P''_i(x_{i+1}) = P''_{i+1}(x_{i+1}), \forall i \in \llbracket 1, N - 2 \rrbracket. \quad (3)$$

P_i étant un polynôme cubique $\forall i \in \llbracket 1, N - 1 \rrbracket$, il est alors possible de l'exprimer sous la forme :

$$P_i(x) = f(x_i) + a_{i1}(x - x_i) + a_{i2}(x - x_i)^2 + a_{i3}(x - x_i)^3, \quad i \in \llbracket 1, N - 1 \rrbracket.$$

En dérivant P_i , les relations suivantes s'obtiennent $\forall i \in \llbracket 1, N - 1 \rrbracket$:

$$\begin{cases} P'_i(x_i) = a_{i1} \\ P''_i(x_i) = 2 \times a_{i2} \\ P'''_i(x_i) = 6 \times a_{i3} \end{cases} \iff \begin{cases} a_{i1} = P'_i(x_i) \\ a_{i2} = P''_i(x_i)/2 \\ a_{i3} = P'''_i(x_i)/6 \end{cases}$$

Posons S la spline cubique considérée, qui peut donc s'exprimer de la sorte :

$$S(x) = P_i(x) \text{ pour tout } x \in [x_i, x_{i+1}], \text{ où } i \in \llbracket 1, N-1 \rrbracket.$$

Ou encore :

$$S(x) = \sum_{i=1}^{N-1} P_i(x) \mathbf{1}_{[x_i, x_{i+1}]}(x).$$

Ainsi, le système précédent devient, $\forall i \in \llbracket 1, N-1 \rrbracket$:

$$\begin{cases} a_{i1} = S'(x_i) \\ a_{i2} = \frac{S''(x_i)}{2} \\ a_{i3} = \frac{S'''(x_i)}{6} \end{cases}$$

D'où, en remplaçant dans l'expression de P_i :

$$P_i(x) = S(x_i) + S'(x_i)(x - x_i) + \frac{S''(x_i)}{2}(x - x_i)^2 + \frac{S'''(x_i)}{6}(x - x_i)^3, \quad i \in \llbracket 1, N-1 \rrbracket.$$

En notant $u_i = x_{i+1} - x_i$ et en évaluant P_i en x_{i+1} , nous obtenons par la condition 1 :

$$f(x_{i+1}) = f(x_i) + S'(x_i)u_i + \frac{S''(x_i)}{2}u_i^2 + \frac{S'''(x_i)}{6}u_i^3, \quad i \in \llbracket 1, N-1 \rrbracket,$$

d'où :

$$S'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{u_i} - \frac{S''(x_i)}{2}u_i - \frac{S'''(x_i)}{6}u_i^2, \quad i \in \llbracket 1, N-1 \rrbracket. \quad (4)$$

Puis, en évaluant P_i'' en x_{i+1} , nous obtenons par la condition 3 :

$$S''(x_{i+1}) = S''(x_i) + S'''(x_i)u_i, \quad i \in \llbracket 1, N-2 \rrbracket,$$

et donc :

$$S'''(x_i) = \frac{S''(x_{i+1}) - S''(x_i)}{u_i}, \quad i \in \llbracket 1, N-2 \rrbracket.$$

En remplaçant dans l'expression 4, il vient :

$$S'(x_i) = \frac{f(x_{i+1}) - f(x_i)}{u_i} - \frac{S''(x_i)}{3}u_i - \frac{S''(x_{i+1})}{6}u_i, \quad i \in \llbracket 1, N-2 \rrbracket. \quad (5)$$

Enfin, en utilisant la condition 2, nous trouvons :

$$S'(x_{i+1}) = S'(x_i) + S''(x_i)u_i + \frac{S'''(x_i)}{2}u_i^2, \quad i \in \llbracket 1, N-2 \rrbracket,$$

soit, grâce à l'expression de $S'''(x_i)$:

$$S'(x_{i+1}) = S'(x_i) + \frac{S''(x_i)}{2}u_i + \frac{S''(x_{i+1})}{2}u_i, \quad i \in \llbracket 1, N-2 \rrbracket.$$

Finalement, en utilisant l'expression 5 pour exprimer $S'(x_i)$ et $S'(x_{i+1})$, nous obtenons une équation ne dépendant plus que S'' , pour $i \in \llbracket 1, N-2 \rrbracket$:

$$S''(x_{i+2})u_{i+1} + 2S''(x_{i+1})(u_i + u_{i+1}) + S''(x_i)u_i = 6 \left(\frac{f(x_{i+2}) - f(x_{i+1})}{u_{i+1}} - \frac{f(x_{i+1}) - f(x_i)}{u_i} \right)$$

Nous avons donc N inconnues $(S''(x_i))_{i \in \llbracket 1, N \rrbracket}$ pour $N-2$ équations. Afin de résoudre le système d'équations, il est nécessaire d'avoir deux équations supplémentaires. Elles peuvent être obtenues en fixant les valeurs de S'' aux points extrêmes : $S''(x_1) = v_1$ et $S''(x_N) = v_N$. Nous pouvons alors remarquer que le système d'équations peut se mettre sous une forme matricielle $AX = B$, où A est une matrice tridiagonale. Les coefficients diagonaux de cette matrice étant tous non nuls, A est inversible et la résolution du système est donc possible (non détaillée dans ce mémoire, se référer à [4], Chapitre 11 Section 5). Les valeurs obtenues pour les $(S''(x_i))$ permettent alors, en reprenant les équations précédentes, d'obtenir les coefficients des polynômes P_i et donc l'expression de la spline S .

1.2.2 Généralisation en dimension deux

Nous venons de voir la démarche d'interpolation par splines cubiques, applicable dans le cadre d'un problème à une dimension. Cette méthode peut s'étendre pour un problème à deux dimensions, i.e. lorsque le modèle comporte deux variables en entrée. Son extension se nomme alors interpolation bicubique par splines. Cette technique sera utilisée dans l'application pratique de ce mémoire pour observer l'impact simultané de deux variables sur les sorties du modèle.

1.3 Interpolation par krigeage

Le krigeage est une méthode d'interpolation spatiale linéaire, basée sur l'estimateur linéaire qui garantit une variance d'estimation minimale. Un des principaux intérêts de cette méthode est qu'elle prend en compte la dépendance spatiale des données. Elle a été développée par l'ingénieur minier sud-africain G.D. Krige en 1951, puis formalisée par le mathématicien français G. Matheron. Le modèle est le suivant :

$$Z(x) = \mu(x) + \delta(x), \quad x \in D$$

- Z est la variable d'étude, un processus gaussien.
- μ est une fonction déterministe représentant l'espérance de Z .
- δ est un processus aléatoire et stationnaire, tel que :
 - $\forall x \in D, E[\delta(x)] = 0$.

- $\forall x, \tilde{x} \in D, Cov(\delta(x), \delta(\tilde{x})) = k(x, \tilde{x}) = \sigma^2 r(x, \tilde{x})$, où r modélisant la fonction de corrélation ne doit en fait dépendre que de la norme $\|x - \tilde{x}\|$ et non de la position de x et \tilde{x} .

— D est l'espace d'observation, $D \subset \mathbb{R}^d$.

Selon l'hypothèse que l'on pose sur μ , on peut définir trois types de krigeage :

- Simple : $\mu = m$ est une constante connue.
- Ordinaire : μ est une constante inconnue.
- Universel : μ est une combinaison linéaire de fonctions de x , i.e. $\mu(x) = \mathbf{f}(x)^t \boldsymbol{\beta} = \sum_{j=1}^k f_j(x) \beta_j$.

1.3.1 Prédiction par krigeage

La méthode est basée sur la connaissance de n valeurs de sorties du modèle $\mathbf{z}_n = (z_1, \dots, z_n)^t$ en n points de l'espace $\mathbf{x}_n = (x_1, \dots, x_n)^t$, représentant l'événement $\{Z(x_1) = z_1, \dots, Z(x_n) = z_n\}$.

$\mathbf{Z}_n = (Z(x_1), \dots, Z(x_n))^t$ est un vecteur gaussien, tel que $\mathbf{Z}_n \sim \mathcal{N}(\boldsymbol{\mu}_n, \mathbf{K}_n)$, où $\boldsymbol{\mu}_n = (\mu(x_1), \dots, \mu(x_n))^t$ et $\mathbf{K}_n = (k(x_i, x_j))_{i,j \in [1,n]}$.

De plus, en ajoutant un nouveau point \tilde{x} , $(\mathbf{Z}_n, Z(\tilde{x}))^t$ est un vecteur gaussien tel que :

$$\begin{pmatrix} \mathbf{Z}_n \\ Z(\tilde{x}) \end{pmatrix} \sim \mathcal{N} \left(\begin{pmatrix} \boldsymbol{\mu}_n \\ \mu(\tilde{x}) \end{pmatrix}, \begin{pmatrix} \mathbf{K}_n & \mathbf{k}(\mathbf{x}_n, \tilde{x}) \\ \mathbf{k}(\mathbf{x}_n, \tilde{x})^t & \sigma^2 \end{pmatrix} \right)$$

où $\mathbf{k}(\mathbf{x}_n, \tilde{x}) = (k(x_1, \tilde{x}), \dots, k(x_n, \tilde{x}))^t$.

Grâce à la stabilité des vecteurs gaussiens par conditionnement, il vient :

$$\forall \tilde{x} \in D, Z(\tilde{x}) | \{\mathbf{Z}_n = \mathbf{z}_n\} \sim \mathcal{N}(m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}), \theta_{\mathbf{x}_n}^2(\tilde{x}))$$

avec :

$$\begin{cases} m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}) = \mu(\tilde{x}) + \mathbf{k}(\mathbf{x}_n, \tilde{x})^t \mathbf{K}_n^{-1} (\mathbf{z}_n - \boldsymbol{\mu}_n) \\ \theta_{\mathbf{x}_n}^2(\tilde{x}) = \sigma^2 - \mathbf{k}(\mathbf{x}_n, \tilde{x})^t \mathbf{K}_n^{-1} \mathbf{k}(\mathbf{x}_n, \tilde{x}) \end{cases}$$

$Z(\tilde{x})$ sera alors estimé par sa moyenne $m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x})$, qui constitue bien un estimateur linéaire en les observations $(z_1, \dots, z_n)^t = \mathbf{z}_n$. Une mesure de la dispersion autour de cette moyenne est également disponible via la variance $\theta_{\mathbf{x}_n}^2(\tilde{x})$.

Dans l'application pratique, nous utiliserons le krigeage ordinaire, où μ est supposée être une constante inconnue. Cette constante μ , le paramètre σ et les éventuels autres paramètres de la fonction $k(x, \tilde{x})$ seront alors estimés par maximum de vraisemblance sur les n points initiaux.

1.3.2 Estimation des paramètres par maximum de vraisemblance

Pour un paramètre donné, par exemple μ (considéré comme constante inconnue), la vraisemblance associée à ce paramètre s'écrit : $L(\mu) = f_\mu(\mathbf{z}_n; \mu)$, avec f_μ la densité de \mathbf{Z}_n connaissant μ . L'estimateur du maximum de vraisemblance associé à μ est alors défini par :

$$\hat{\mu} = \arg \max_{\mu} L(\mu)$$

Dans l'application pratique, cette maximisation de la vraisemblance sera effectuée sous R, via l'algorithme génétique d'optimisation `genoud` de la librairie `rgenoud`. En outre, les différents problèmes d'optimisation rencontrés dans ce mémoire seront résolus dans l'application pratique par des algorithmes génétiques. Cette technique d'optimisation numérique sera ainsi développée dans la section 1.5.

1.3.3 Modélisation du variogramme

Il reste à définir la fonction de covariance $k(x, \tilde{x}) = \sigma^2 r(x, \tilde{x})$. Dans un espace de dimension d , cette fonction peut être modélisée de la manière suivante :

$$k(x, \tilde{x}) = \prod_{j=1}^d k_j(x^j, \tilde{x}^j)$$

avec $x = (x^1, \dots, x^d)^t$ et $\tilde{x} = (\tilde{x}^1, \dots, \tilde{x}^d)^t$, où les fonctions (k_j) (aussi appelées noyaux de covariance) sont usuellement de l'un des cinq types suivants :

- Gaussien : $k_j(a, b) = \sigma^2 e^{-\frac{(a-b)^2}{2\alpha^2}}$
- Exponentiel : $k_j(a, b) = \sigma^2 e^{-\frac{|a-b|}{\alpha}}$
- Exponentiel-Puissance : $k_j(a, b) = \sigma^2 e^{-\frac{|a-b|^m}{\alpha^m}}$, avec $m \in]0, 2[$
- Matern 3/2 : $k_j(a, b) = \sigma^2 \left(1 + \frac{\sqrt{3}|a-b|}{\alpha}\right) e^{-\sqrt{3}\frac{|a-b|}{\alpha}}$
- Matern 5/2 : $k_j(a, b) = \sigma^2 \left(1 + \frac{\sqrt{5}|a-b|}{\alpha} + \frac{5(a-b)^2}{3\alpha^2}\right) e^{-\sqrt{5}\frac{|a-b|}{\alpha}}$

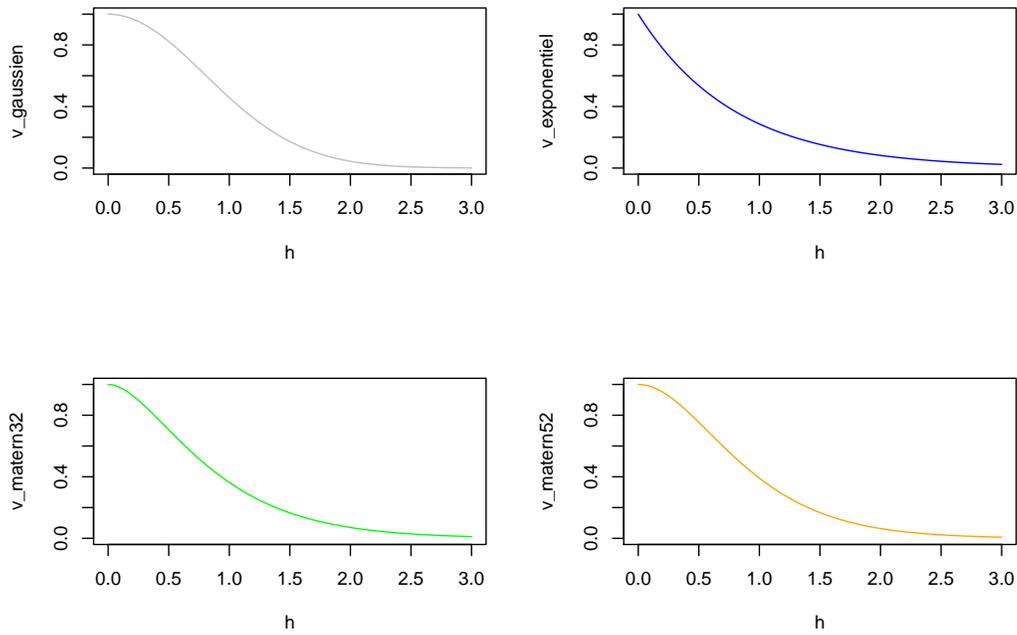


FIGURE 20 – Covariance en fonction de la distance selon le type de fonction ($\alpha = 0,8$)

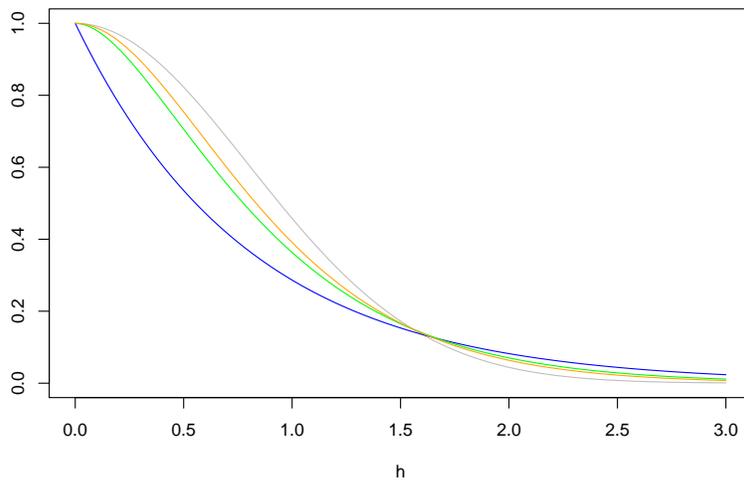


FIGURE 21 – Superposition des courbes de la figure 20

La figure 21 permet de comparer les structures de corrélation selon les types de fonctions, car elle représente, pour chaque fonction, le niveau de corrélation entre deux points séparés d'une

distance h . Il est important de noter les différences de régularité entre ces types de fonctions. Les types exponentiel et exponentiel-puissance sont de classe \mathcal{C}^0 , les types matern 3/2 et matern 5/2 sont respectivement \mathcal{C}^1 et \mathcal{C}^2 , et le type gaussien est \mathcal{C}^∞ . La classe de la fonction de covariance conditionne en effet la régularité du processus gaussien associé.

Pour choisir le type de fonction de covariance à utiliser, une approche classique est d'appliquer une validation croisée (« cross-validation »), en se basant sur la base initiale constituée des n sorties et n points de l'échantillon associé. Cette méthode permet d'évaluer la qualité de prédiction d'un modèle à partir d'une base de données initiale.

Dans le cadre de ce mémoire, pour choisir au mieux cette fonction parmi les cinq présentées, nous appliquerons une validation croisée de type « leave one out » (LOO), qui consiste à effectuer, pour chaque point de l'échantillon initial, les étapes suivantes :

1. Retirer ce point de l'échantillon.
2. Construire un modèle de krigeage à partir des points restants.
3. Prédire via ce modèle de krigeage la valeur de la variable de sortie pour le point retiré.
4. Calculer l'erreur de prédiction associée.

Une fois ces étapes effectuées pour tous les points de l'échantillon, il est alors possible de calculer, pour ce processus de validation croisée, une erreur quadratique moyenne de prédiction sur l'ensemble de l'échantillon. Pour sélectionner la forme du noyau de covariance, nous allons donc répéter ce processus de validation croisée pour chacune des fonctions de covariance présentée, et nous retiendrons la fonction de covariance ayant engendré la plus petite erreur quadratique moyenne de prédiction.

Une fois la forme du noyau de covariance déterminée, il reste à estimer le(s) paramètre(s) associé(s) à la fonction de covariance. Comme expliqué en 1.3.1, cette estimation est faite via la méthode du maximum de vraisemblance.

1.4 Evaluation de la qualité d'un métamodèle

Une fois un métamodèle construit, il est nécessaire d'évaluer sa qualité de prédiction. Pour ce faire, l'approche adoptée dans ce mémoire, consiste en la construction d'une base de validation sur laquelle le métamodèle construit sera évalué.

1.4.1 Base de validation

La base de validation va nous permettre de vérifier la précision du métamodèle, construit à partir d'un nombre limité de points, en estimant au mieux sur cette base les écarts entre l'espace des résultats du métamodèle et l'espace des résultats du modèle initial.

En pratique, la construction de la base de validation est ici analogue à celle de la base utilisée pour la conception du métamodèle. Des points sont sélectionnés et le modèle est exécuté sur chacun de ces points pour obtenir les résultats associés. Le nombre de points n_{test} constituant cette

base de validation sera conséquent, car il doit permettre d'évaluer le métamodèle sur l'ensemble de l'espace de définition des variables. Les points $(x_i)_{i \in \llbracket 1, n_{test} \rrbracket}$ constituant l'échantillon de validation seront générés via un hypercube latin.

Ici, la notion de base de validation, connexe au machine learning, n'est pas à considérer dans la même optique. En effet, dans une approche classique de machine learning, l'utilité de la base de validation est de permettre une mesure de la capacité d'extrapolation d'un modèle en-dehors de la base de construction. L'approche adoptée dans ce mémoire est différente : l'objectif n'est pas d'optimiser la capacité d'extrapolation d'un point de vue opérationnel, mais de valider la démarche théorique de construction, de s'assurer du bon fonctionnement du métamodèle, conçu via un nombre de points limité, sur l'ensemble du périmètre d'étude.

Dans ce mémoire, la base de validation doit être vue comme un support de discrétisation d'une métrique, le Q^2 , sur un périmètre défini, l'hypercube étudié. L'idée sous-jacente est de mesurer un écart global entre deux espaces, l'espace de prédiction du métamodèle et l'espace des résultats du modèle initial. Nous devons pour cela nous donner une quantité d'information importante sur les résultats du modèle initial dans l'ensemble de l'hypercube.

Nous souhaitons donc démontrer la validité de la démarche théorique, qui est la construction d'un métamodèle efficient via un plan d'expérience fixe ou séquentiel composé d'un faible nombre de points. Cette démonstration faite, la démarche pourra ainsi être répliquée d'un point de vue opérationnel sur des modèles initiaux de structure proche, sans nécessiter à nouveau la création d'une base de validation si importante. Il sera par exemple possible de procéder par validation croisée (« cross-validation »).

Pour trouver des démarches analogues dans le processus de mesure de précision d'un métamodèle, il est possible de se référer aux travaux de M. Nabil Rachdi, membre du groupement de recherche MASCOT-NUM [1] du CNRS (Centre National de la Recherche Scientifique). Nous pouvons par exemple citer le développement logiciel du module « otpod » [2], auquel il a participé, où des méthodes similaires sont utilisées dans le cadre de la construction de modèles de krigeage, afin d'estimer ici des probabilités de détection de défaut. Ces modèles de krigeage requièrent un grand nombre de simulations pour évaluer leur qualité de prédiction. D'un point de vue applicatif, ces différents projets ont notamment contribué à l'intégration du krigeage dans certains processus industriels en aéronautique.

1.4.2 Mesures d'erreur

Notons f le modèle initial à estimer et \hat{f} le métamodèle construit, visant à approcher f au mieux. Pour évaluer la qualité de prédiction du métamodèle, plusieurs mesures d'erreur usuelles peuvent être calculées :

- L'erreur quadratique moyenne de prédiction : $EQMP = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (\hat{f}(x_i) - f(x_i))^2$. Dans cette mesure, le poids de l'erreur pour une prédiction augmente quadratiquement avec l'augmentation de l'écart entre la prédiction et la vraie valeur.

- L'erreur absolue moyenne de prédiction : $EAMP = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} |\hat{f}(x_i) - f(x_i)|$. Dans cette mesure, au contraire de la précédente, la relation entre le poids et l'écart est linéaire.
- L'erreur relative moyenne de prédiction : $ERMP = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} \frac{|\hat{f}(x_i) - f(x_i)|}{|f(x_i)|}$. Les erreurs de prédiction sont ici exprimées en pourcentage des vraies valeurs.

Dans l'application pratique de ce mémoire, nous utiliserons essentiellement l'erreur quadratique moyenne de prédiction, afin de pénaliser les erreurs trop conséquentes du métamodèle.

Nous utiliserons aussi une autre mesure de la qualité prédictive du métamodèle, le coefficient Q^2 , défini, par analogie au coefficient de détermination R^2 , comme :

$$Q^2 = 1 - \frac{\sum_{i=1}^{n_{test}} (\hat{f}(x_i) - f(x_i))^2}{\sum_{i=1}^{n_{test}} (f(x_i) - m_f)^2} = 1 - \frac{EQMP}{\frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (f(x_i) - m_f)^2}$$

avec $m_f = \frac{1}{n_{test}} \sum_{i=1}^{n_{test}} f(x_i)$ la moyenne des observations, et donc $\frac{1}{n_{test}} \sum_{i=1}^{n_{test}} (f(x_i) - m_f)^2$ la variance empirique du modèle f estimée sur la base de validation. Ainsi, plus le coefficient Q^2 est proche de 1, plus la qualité prédictive du métamodèle est bonne.

1.5 Optimisation numérique par algorithme génétique

Ce mémoire traitant plusieurs problèmes d'optimisation dans différents cadres, il nous a paru pertinent de présenter la méthode de résolution numérique utilisée dans l'application pratique. Notre choix s'est porté sur l'algorithme génétique, qui appartient à la famille des algorithmes évolutionnistes. Cette dernière regroupe différents algorithmes inspirés de la théorie de l'évolution. L'idée principale dans ce type d'algorithme consiste en la génération d'un ensemble de solutions, que l'algorithme va peu à peu faire évoluer en une solution optimale.

L'algorithme génétique présente l'avantage d'être composé de mécanismes simples et généraux, lui permettant de s'adapter à des problèmes variés. Sa mise en oeuvre ne requiert en effet que peu de connaissance sur la fonction à optimiser, seuls le domaine de définition des variables et la capacité d'évaluer la fonction cible sont nécessaires. Il permet donc de s'affranchir de certaines contraintes inhérentes à d'autres types d'algorithmes, comme la différentiabilité de la fonction étudiée. En ce sens, l'algorithme génétique s'associe bien à la trame générale de notre sujet, où l'objectif est de présenter et démontrer l'efficacité de méthodes d'interpolation très génériques, utilisables à partir d'une quantité réduite d'information sur le modèle sous-jacent, sans présupposer de structure initiale, mais exploitant seulement des couples entrées-sorties.

Le vocabulaire de l'algorithme génétique est emprunté à la biologie et la génétique. Nous ferons l'analogie avec la situation qui consisterait à optimiser l'efficacité de réassurance en fonction du traité.

Le principe est de partir d'une population de départ appelée première génération, composée de n individus $(p^i)_{i \in [1, n]}$ (les traités de réassurance) chacun constitué de différents chromosomes

$(p_j^i)_{j \in \llbracket 1, h \rrbracket}$ (les caractéristiques des traités), puis d'évaluer la fonction objectif f (l'efficacité de réassurance) sur ces individus. A partir de là, le but est de constituer une nouvelle génération plus performante au sens de f . Les étapes suivantes sont alors réalisées (nous nous plaçons dans le cas où nous cherchons à maximiser f) :

1. **Sélection** : parmi la population de départ, deux individus p^l et p^k sont tirés au sort. L'individu pour lequel la valeur de f est la plus élevée est retenu. Il sera le premier individu de la nouvelle génération. Cette étape est ensuite répétée $n - 1$ fois de sorte à définir les n individus de la nouvelle génération. (Pour ne pas alourdir l'écriture, nous conservons les mêmes notations $(p^i)_{i \in \llbracket 1, n \rrbracket}$ pour les individus de la nouvelle génération, en ayant bien conscience que ce ne sont plus les mêmes).
2. **Croisement** : pour chaque individu de la nouvelle génération, un croisement est effectué avec une probabilité c définie au préalable. Ce croisement est effectué entre p^1 et p^2 , p^3 et p^4 , et ainsi de suite. Chacun des h chromosomes de p^k et p^{k+1} est croisé avec la probabilité c . Pour le chromosome d'indice $j \in \llbracket 1, h \rrbracket$, un nombre b_j^k est tiré selon une loi uniforme $\mathcal{U}[0, 1]$; si $b_j^k \geq c$, aucun croisement n'a lieu, mais si $b_j^k < c$, p_j^k est changé en $p_j^k(1 - b_j^k) + p_j^{k+1}b_j^k$, et p_j^{k+1} en $p_j^{k+1}(1 - b_j^k) + p_j^k b_j^k$. Si n est impair, le dernier individu p^n ne subit pas de croisement.
3. **Mutation** : chacun des chromosomes composant les n individus subit une mutation avec une probabilité m définie au préalable. Considérons l'individu p^k . Pour le chromosome d'indice $j \in \llbracket 1, h \rrbracket$, un nombre u_j^k est tiré selon une loi uniforme $\mathcal{U}[0, 1]$; si $u_j^k \geq m$, aucune mutation n'a lieu, mais si $u_j^k < m$, p_j^k est changé en :

$$\max(I_j; \min(S_j; p_j^k + \alpha_j^k \frac{S_j - I_j}{2})),$$

où α_j^k est tiré selon $\mathcal{U}[-1, 1]$, et $[I_j, S_j]$ est le domaine de définition du chromosome d'indice j .

La deuxième génération est ainsi constituée. Ensuite, les étapes 1, 2 et 3 sont répétées jusqu'à la vérification d'un critère d'arrêt de l'algorithme, défini comme l'atteinte d'une convergence suffisante des individus ou d'un nombre maximal de générations. Une convergence pour un tel type d'algorithme sera remarquée par une similarité suffisamment forte des individus d'une même génération. Finalement, l'individu (le traité) retenu sera celui maximisant f (l'efficacité) sur l'ensemble des individus de la dernière génération constituée.

Le schéma ci-dessous synthétise le fonctionnement de l'algorithme :

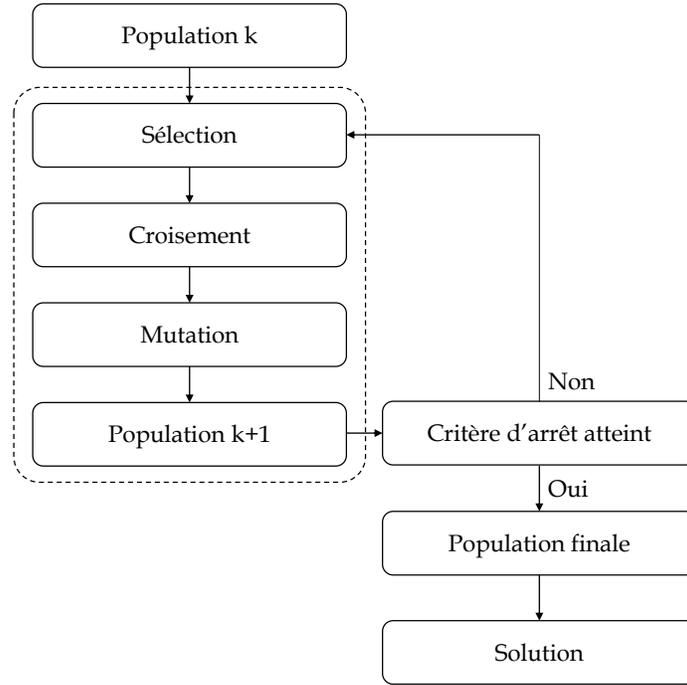


FIGURE 22 – Fonctionnement de l'algorithme génétique

2 Le modèle de tarification en réassurance

Ce modèle, implémenté sous le logiciel ADDACTIS[®] Modeling, est dédié à la tarification de traités de réassurance. La conception de ce modèle a fait l'objet d'une partie de mon alternance. Dans le cadre de ce mémoire, nous n'utiliserons que la partie réalisant la tarification par simulations des traités en excédent de sinistre (ou XS). ADDACTIS[®] Modeling est la solution de modélisation du cabinet Addaactis, permettant la conception de tout type de modèle via une plateforme de codage.

Cette partie vise à donner une vue d'ensemble du modèle de réassurance, en décrivant sa structure, les données attendues, ainsi que son fonctionnement sur la partie utilisée dans le cadre du mémoire.

Un lexique est présent en fin de partie pour rappeler quelques notions de réassurance, concernant notamment les traités en excédent de sinistre.

2.1 Architecture du modèle

Le schéma suivant synthétise l'architecture du modèle, en exposant les différents modules qui le composent :

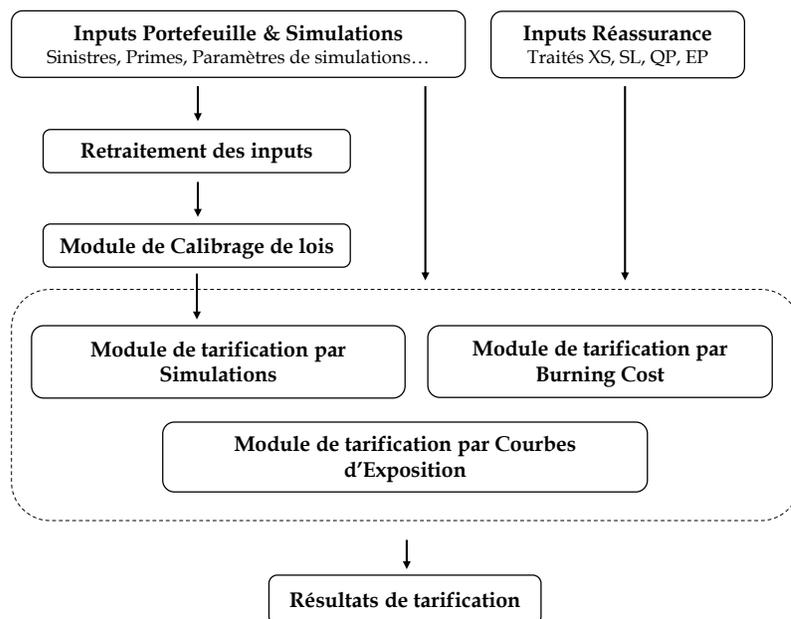


FIGURE 23 – Architecture du modèle de tarification en réassurance

Le modèle permet de tarifier des traités de réassurance en excédent de sinistre (XS), excédent de perte ou « stop-loss » (SL), quote-part (QP) et excédent de plein (EP). La tarification utilise des données relatives au portefeuille : historique de sinistres, historique de primes, type de branche étudiée, cadences de règlement des sinistres (notamment pour les branches longues), sommes assurées ; l'objectif étant de coter, i.e. tarifer, les traités de la manière la plus précise possible. Le modèle utilise trois méthodes de cotations usuellement appliquées en réassurance, brièvement décrites ci-dessous :

- **Tarification par Burning Cost** : cette méthode, usuellement appliquée pour tarifier les tranches travaillantes des traités XS, se base sur un historique de sinistres et de primes, redressés au préalable pour se mettre dans les conditions de l'année de cotation. Pour chaque année de l'historique, un taux burning cost est calculé. Ce taux est obtenu en déterminant le montant qui aurait été cédé en réassurance sur l'année considérée, puis en divisant ce montant par l'assiette de prime de cette même année. Les taux obtenus sur l'ensemble des années sont ensuite agrégés par le biais d'une moyenne pondérée par les assiettes de primes annuelles, pour donner le taux burning cost final. Afin d'obtenir la prime pure du traité, il suffit alors d'appliquer ce dernier taux à l'assiette de primes estimée pour l'année de cotation.
- **Tarification par Simulations** : cette méthode se base sur un modèle « fréquence \times sévérité », qui consiste à calibrer deux lois de probabilités, une loi de fréquence modélisant le nombre annuel d'occurrences de sinistres et une loi de sévérité modélisant le montant de chaque sinistre. Ces deux lois sont généralement supposées indépendantes dans cette méthode, ce qui en l'absence de réassurance permet d'obtenir facilement la prime pure par multiplication des espérances. Dans le cas de la réassurance non-proportionnelle et notamment de la tarification de traités XS, l'obtention de la prime est plus complexe et nécessite souvent de passer par la simulation d'années de sinistralité, afin de pouvoir prendre en compte les différentes clauses (restitutions, AAD, AAL...) ou encore les tranches hautes des traités. La prime pure finale est alors obtenue en effectuant la moyenne des primes pures obtenues sur les années simulées. L'avantage de cette méthode est qu'elle reproduit en détail le déroulement d'une année de sinistralité. Son caractère stochastique permet également d'obtenir différents indicateurs statistiques intéressants (écart-type, Value-at-Risk, ...). Cependant, les simulations peuvent s'avérer coûteuses en temps de calcul, notamment sur les tranches où la volatilité est élevée et qui nécessitent donc un nombre important de simulations pour obtenir une convergence satisfaisante.
- **Tarification par Courbes d'Exposition** : cette méthode est également utilisée lors de la tarification de traités XS. Elle n'utilise pas d'historique de sinistres, mais un profil de portefeuille donnant l'agrégation des sommes assurées et des primes associées par bande de risque (i.e. par niveau de somme assurée), ainsi qu'une courbe d'exposition dont la forme dépend de la branche ou du type de risque sous-jacent. Cette courbe d'exposition est en fait la fonction de répartition du taux de destruction des sommes assurées pour le type de risque

étudié. Il en découle que pour un risque individuel du portefeuille, avec une prime et une somme assurée associées, il est possible de calculer la part de la prime revenant au traité de réassurance : c'est la portion de risque entrant dans le traité. Pour la déterminer, il suffit de savoir, pour ce risque, à partir de quel niveau de taux de destruction le traité s'applique, et jusqu'à quel niveau. Cet intervalle est donc défini par les bornes du traité normalisées par la somme assurée considérée. Une fois cet intervalle déterminé, la proportion de risque correspondante s'obtient directement via la courbe d'exposition (image de cet intervalle par la courbe) : c'est donc la part de la prime revenant au traité de réassurance. Cette démarche n'est pas appliquée risque par risque sur l'ensemble du portefeuille, mais par bande de risque via le profil de portefeuille. Une somme assurée moyenne par bande de risque est calculée, permettant d'obtenir la proportion de prime revenant au réassureur, puis la prime totale de réassurance est déterminée pour chaque bande. Il suffit alors de sommer ces primes pour avoir la prime pure du traité.

2.2 Données attendues en entrée du modèle

Nous allons dans cette partie décrire les types de données attendus par le modèle (les inputs), et les données effectivement utilisées dans l'application pratique de ce mémoire.

- Données du portefeuille :
 - Historique de primes, indice de mise en « as-if » des primes, primes estimées pour l'année de cotation, profil de portefeuille par bande de risque, courbes d'exposition, branche étudiée.
 - Historique de sinistres individuels (paiements et suspens), cadence de développement des paiements et de la charge, méthode retenue pour la projection de la charge, historique des ratios sinistres sur primes, indice de mise en « as-if » des sinistres, seuil retenu pour les sinistres graves, pondération des années d'étude.
 - Hypothèses retenues pour la tarification par simulations : nombre de simulations à effectuer, calibrage des lois automatique (lois calibrées par le modèle à partir de l'historique de sinistres) ou manuel (lois calibrées par l'utilisateur qui définit lui-même les lois à utiliser et leurs paramètres), tests d'adéquation appliqués pour le calibrage automatique des lois.

- Données de réassurance :
 - Caractéristiques des traités XS : priorité, portée, AAD, AAL, reconstitutions, coût du capital, taux de frais et de courtage, clause de stabilité.
 - Caractéristiques des autres traités : SL, QP, EP.

Dans le cadre de ce mémoire, nous avons utilisé les données d'une cédante sur la branche incendie, sur une période de six ans, de 2012 à 2018. Le seuil retenu pour classifier les sinistres

comme graves ou attritionnels, ayant fait l'objet d'une étude préalable (basée sur la théorie des valeurs extrêmes), est fixé à 1 150 000 €. Nous avons également à disposition un historique de primes, ainsi qu'un profil de portefeuille par bande de sommes assurées. Concernant les traités en excédent de sinistre, ceux sur lesquels sera centrée notre étude, nous générerons plusieurs traités via les méthodes d'échantillonnage décrites dans la partie 1.1.

Les données utilisées seront étudiées plus en détail dans la section 2.3.

2.3 Préparation des données pour l'application pratique

Dans cette partie, nous allons recenser les retraitements qui ont dû être effectués sur les données effectivement utilisées dans l'application pratique de ce mémoire ainsi que les hypothèses qui ont dû être posées, en amont de l'utilisation du modèle.

2.3.1 Retraitements effectués

Avant d'intégrer les données au modèle, quelques étapes intermédiaires ont dû être suivies concernant la base de sinistres transmise. Il y avait initialement 87 209 sinistres individuels dans la base, pour lesquels nous disposons de la charge vue à l'ultime. Afin de correspondre avec les données attendues en entrée du modèle, il a donc fallu :

- sélectionner les sinistres de type incendie : ils sont au nombre de 10 503,
- déterminer ensuite les loss-ratios annuels, grâce aux assiettes de primes correspondantes,
- pour finir, extraire les sinistres graves de la base, afin d'alimenter la liste de sinistres graves attendue par le modèle. Le seuil de sinistralité grave retenu étant de 1 150 000 €, nous avons conservé les sinistres supérieurs à 500 000 €. Ce seuil nettement inférieur a été choisi afin qu'aucun sinistre grave ne soit oublié après la mise en « as-if ». Les sinistres vont en effet être redressés via un indice de sorte à se mettre dans les conditions de l'année de cotation. Nous vérifierons après avoir choisi l'indice en question que ce seuil est suffisant. Les sinistres incendie supérieurs à 500 000 € sont au nombre de 200.

2.3.2 Hypothèses retenues

Des hypothèses supplémentaires, non disponibles dans les données transmises, sont nécessaires à la cotation des traités. Il s'agit notamment des indices utilisés pour la mise en « as-if » des sinistres et des primes, et de la courbe d'exposition utilisée pour la tarification par exposition. Les hypothèses suivantes ont été retenues :

- Les sinistres ont été mis en « as-if » via l'indice FFB (Fédération Française du Bâtiment) des coûts à la construction.
- Les primes ont été mises en « as-if » via l'indice des prix à la consommation (ou IPC).
- La courbe d'exposition retenue est une courbe calibrée sur des données marché, relative à une branche de type « Commercial lines - Low risks ».

2.4 Fonctionnement du modèle

Nous allons ici détailler le fonctionnement du modèle concernant la partie tarification par simulations. Une fois les données importées, le modèle fonctionne de la façon suivante :

1. **Calculs intermédiaires** : cette première étape est dédiée au redressement des sinistres et des primes par le biais des indices de mise en « as-if », à la projection des charges à l'ultime lorsque les sinistres sont renseignés sous forme de triangles de paiements et de suspens, ainsi qu'à la stabilisation des sinistres si une clause de stabilité est présente dans un traité XS. Dans notre cas, seul le redressement des sinistres et des primes est effectué car nous disposons des sinistres vus à l'ultime et les traités considérés ne possèdent pas de clause de stabilité (clause spécifique aux branches longues).

Tableau des indices pour la mise en « as-if » :

Année	Indice Sinistres	Indice Primes
2019	1010,9	104,9
2018	989,9	103,5
2017	974,8	101,9
2016	942,0	100,7
2015	929,5	100,0
2014	930,8	99,9
2013	920,8	99,8
2012	903,1	99,1

- Redressement des sinistres : les sinistres sont redressés via les indices correspondants $(I_j^s)_{j \in \llbracket 2012, 2019 \rrbracket}$ par la formule $X = \frac{I_{2019}^s}{I_j^s} X_i^j$, $j \in \llbracket 2012, 2018 \rrbracket$, avec X_i^j le $i^{\text{ème}}$ sinistre de l'année j . Le seuil précédemment choisi, égal à 500 000 €, est amplement suffisant car $\frac{I_{2019}^s}{I_{2012}^s} \times 500\,000 = 559\,683 \text{ €} < 1\,150\,000 \text{ €}$.

I_{2019}^s a été estimé en multipliant I_{2018}^s par la moyenne des rapports $\left(\frac{I_j^s}{I_{j-1}^s} \right)_{j \in \llbracket 2016, 2018 \rrbracket}$.

- Redressement des primes : ne disposant pas de mesure d'exposition du portefeuille, les primes sont redressées en utilisant l'indice IPC $(I_j^p)_{j \in \llbracket 2012, 2019 \rrbracket}$ (valeurs dans le tableau précédent) par la formule $V = \frac{I_{2019}^p}{I_j^p} V_j$, $j \in \llbracket 2012, 2018 \rrbracket$, avec V_j le volume de primes pour l'année j .

I_{2019}^p a été estimé en multipliant I_{2018}^p par la moyenne des rapports $\left(\frac{I_j^p}{I_{j-1}^p} \right)_{j \in \llbracket 2017, 2018 \rrbracket}$.

2. **Classification des sinistres** : après avoir été redressés, les sinistres renseignés sont ensuite séparés entre attritionnels et graves. Les loss-ratios annuels historiques pour les sinistres attritionnels sont alors calculés.

Initialement, dans la base de données, 106 sinistres étaient supérieurs à 1 150 000 € et donc considérés comme sinistres graves. Après mise en « as-if », 8 sinistres supplémentaires dépassent ce seuil, ce qui porte le nombre de sinistres graves à 114.

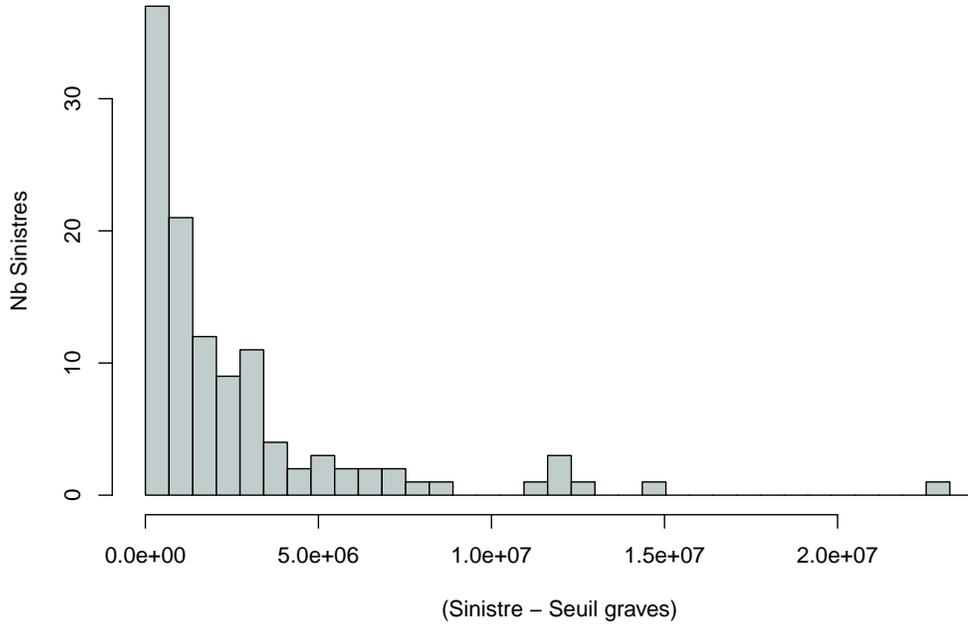


FIGURE 24 – Histogramme des sinistres graves

3. **Calibrage des lois de probabilité** : à partir de l'historique de sinistres graves déterminé (après mise en « as-if »), sont calibrées une loi de fréquence et loi de sévérité. Pour la fréquence, la loi est choisie parmi la loi de Poisson et la loi binomiale négative. Les paramètres de ces deux lois sont estimés par maximum de vraisemblance et par la méthode des moments :

- Loi de Poisson : N suit une loi de Poisson de paramètre $\lambda > 0$, notée $\mathcal{P}(\lambda)$, lorsque $\forall k \in \mathbb{N}$, $P(N = k) = e^{-\lambda} \frac{\lambda^k}{k!}$. N représente dans notre cas le nombre annuel de sinistres.

L'estimateur du maximum de vraisemblance pour la loi de Poisson est défini par $\hat{\lambda} = \bar{n}$ où $\bar{n} = \frac{1}{m} \sum_{j=1}^m n_j$, avec m le nombre d'années d'observation et n_j le nombre de sinistres pour l'année j .

La loi de Poisson est généralement appelée « loi des événements rares », elle est souvent adaptée pour modéliser la fréquence d'événements extrêmes, et donc usuellement utilisée en réassurance. Néanmoins, la propriété $E[N] = V[N] = \lambda$ vérifiée lorsque $N \sim \mathcal{P}(\lambda)$, indique que la loi de Poisson n'est pas adaptée pour modéliser un phénomène présentant de la surdispersion.

- Loi binomiale négative : la loi binomiale négative peut être considérée comme alternative à la loi de Poisson lorsque les données présentent de la surdispersion. N suit une loi binomiale négative de paramètres $r > 0$ et $p \in]0, 1[$, notée $\mathcal{BN}(r, p)$, lorsque $\forall k \in \mathbb{N}$, $P(N = k) = C_{k+r-1}^k p^r (1-p)^k$, où $C_{k+r-1}^k = \binom{k+r-1}{k}$.

Par la méthode des moments, les estimateurs suivants s'obtiennent :

$$\hat{p} = \frac{\bar{n}}{\hat{s}^2} \text{ et } \hat{r} = \bar{n} \frac{\hat{p}}{1 - \hat{p}}, \text{ avec } \hat{s}^2 = \frac{1}{m-1} \sum_{j=1}^m (n_j - \bar{n})^2.$$

Pour choisir la loi de fréquence, un test de surdispersion est effectué sur la distribution empirique : si $\frac{V[N]}{E[N]} > 1$, la loi binomiale négative est retenue, dans le cas contraire la loi de Poisson est privilégiée.

Données empiriques :

Année	Nombre de sinistres graves
2018	25
2017	21
2016	11
2015	18
2014	7
2013	14
2012	18

La moyenne empirique est égale à 16,29 et l'écart-type à 6,10.

Sur nos données, la surdispersion obtenue est $\frac{V[N]}{E[N]} = \frac{37,24}{16,29} = 2,29 > 1$. La loi binomiale négative est donc retenue, avec les paramètres estimés $\hat{r} = 12,66$ et $\hat{p} = 0,44$.

Concernant la sévérité des sinistres graves, cinq lois sont disponibles : la loi de Pareto, la loi de Pareto généralisée, la loi Log-Normale, la loi de Weibull et la loi Gamma. Les paramètres de ces lois sont estimés par la méthode du maximum de vraisemblance, et le test d'adéquation utilisé permettant de choisir une loi parmi ces cinq est le test d'Anderson-Darling. Il permet de tester l'hypothèse H_0 : « la sévérité suit la loi de fonction de répartition F », contre H_1 : « la sévérité ne suit pas la loi de fonction de répartition F ». La statistique de ce test, basée sur une notion de distance, est définie par :

$$A_n^2 = n \int_{-\infty}^{+\infty} \frac{(\hat{F}(x) - F(x))^2}{F(x)(1 - F(x))} dx,$$

et donc

$$A_n^2 = -n - \frac{1}{n} \sum_{i=1}^n (2i - 1) \ln(F(x_{(i)})) + (2n + 1 - 2i) \ln(1 - F(x_{(i)})) dx,$$

où les $(x_{(i)})$ sont les (x_i) classés par ordre croissant. Ainsi, le test d'Anderson-Darling accorde un poids important aux observations en queue de distribution.

Lors du lancement du modèle sur les données, la loi Gamma est retenue par ce test. Les distances d'Anderson-Darling pour chaque loi sont affichées dans le tableau suivant :

Loi de sévérité	A_n^2
Gamma	0,34
Pareto généralisée	1,41
Pareto	1,44
Log-Normale	1,49
Weibull	4,19

Nous effectuons cependant une vérification grâce à trois graphiques en sortie du modèle, le premier représentant les fonctions de répartition empirique et théoriques, le deuxième étant de type Quantile-Quantile et le troisième étant de type Probabilité-Probabilité.

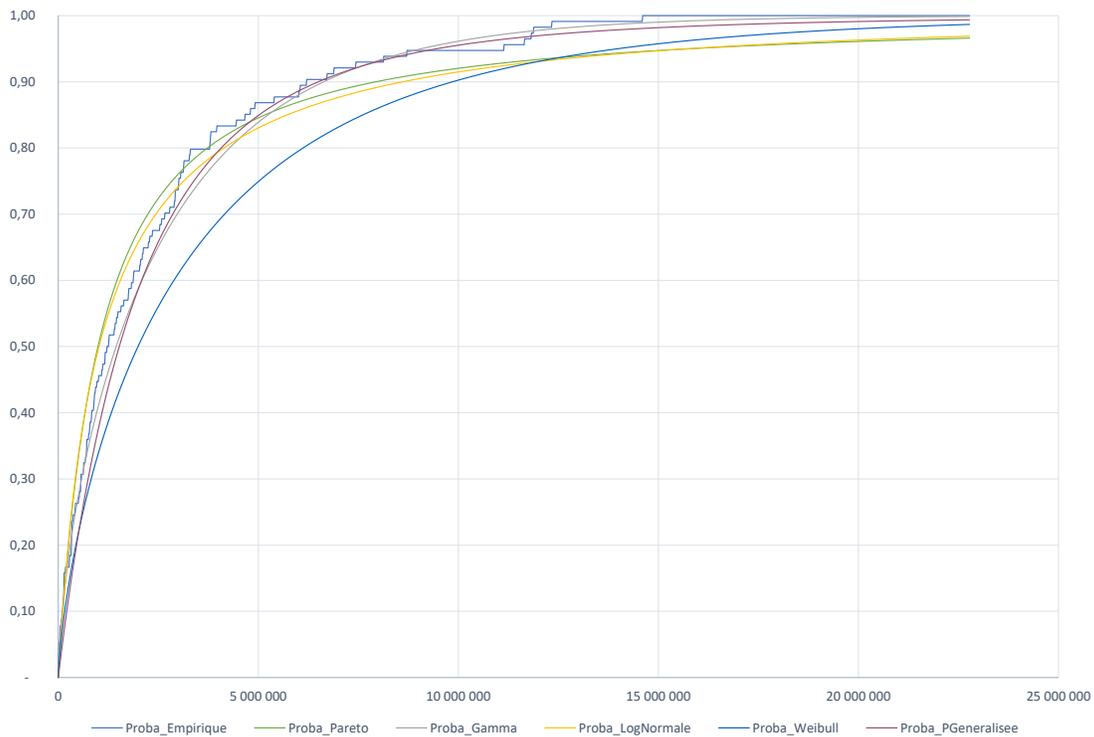


FIGURE 25 – Fonctions de répartition empirique et théoriques pour la sévérité des graves

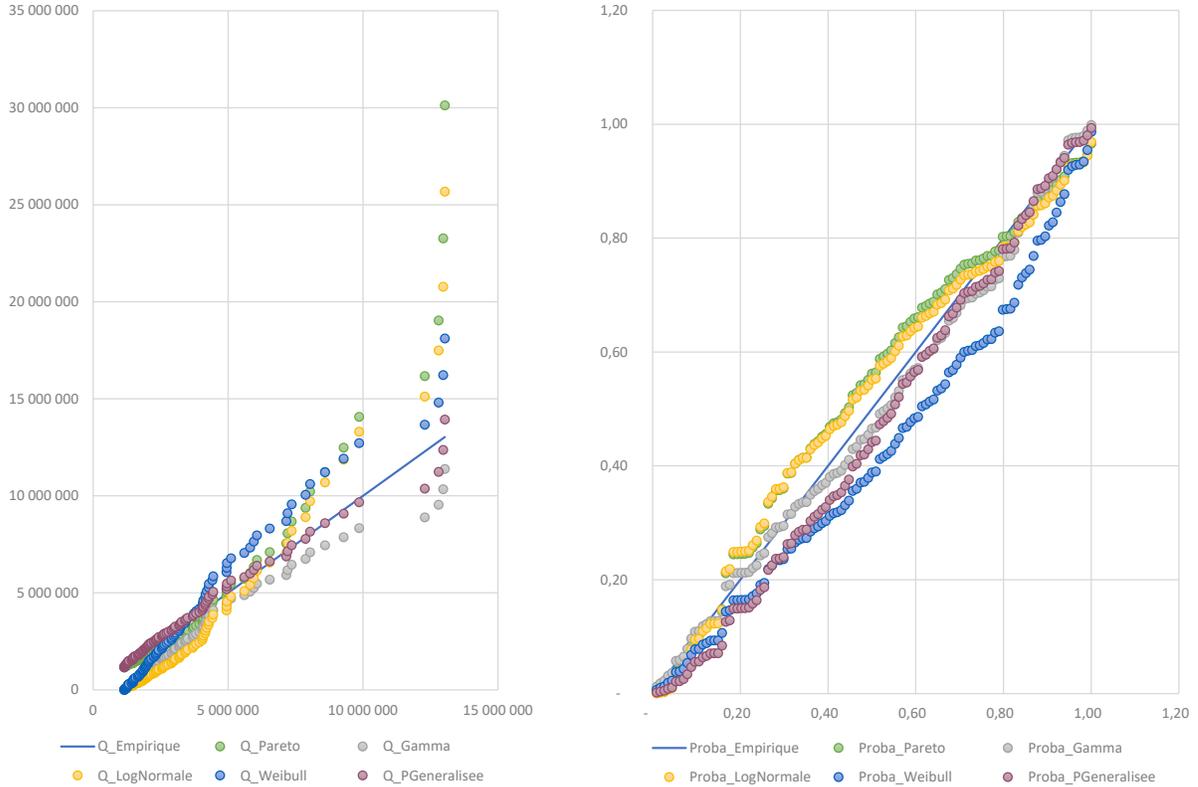


FIGURE 26 – Graphe Q-Q et graphe P-P pour la sévérité des graves

De l'analyse visuelle de ces trois graphiques, et notamment de celle du graphique Quantile-Quantile, nous concluons que la loi de Pareto généralisée est probablement plus adaptée pour modéliser la sinistralité empirique au-delà du seuil de graves. De plus, elle offre une alternative plus prudente en termes de gestion de risque par rapport à la loi Gamma, de par sa queue de distribution plus épaisse.

Nous allons maintenant présenter plus en détail la loi de Pareto généralisée (Generalized Pareto Distribution), qui sera donc utilisée lors de la simulation des sinistres.

$$X \sim GPD(\sigma, \xi), \text{ lorsque } F_X(x) = \begin{cases} 1 - \left(1 + \xi \frac{x}{\sigma}\right)^{-\frac{1}{\xi}} & \text{si } \xi \neq 0 \\ 1 - e^{-\frac{x}{\sigma}} & \text{si } \xi = 0 \end{cases}$$

Cette distribution modélise ainsi la loi des dépassements de seuil (où le seuil en question est le seuil de sinistres graves fixé à 1 150 000 €), c'est-à-dire $X = Y - s | Y > s$ avec $s = 1\,150\,000$ et Y modélisant la sévérité initiale des sinistres sur le portefeuille. Dans ce cas, le montant

d'un sinistre grave vaudra donc $(1\,150\,000 + X)$ euros.

L'estimation des paramètres par le modèle, par maximum de vraisemblance, conduit à $\sigma = 1\,987\,590$ et $\xi = 0,29$. Ainsi, $F_X(x) = 1 - \left(1 + \xi \frac{x}{\sigma}\right)^{-\frac{1}{\xi}}$. L'espérance et la variance de cette loi s'expriment de la sorte :

$$E[X] = \frac{\sigma}{1 - \xi} \text{ si } \xi < 1, \text{ et } Var[X] = \frac{\sigma^2}{(1 - \xi)^2(1 - 2\xi)} \text{ si } \xi < \frac{1}{2}.$$

Dans notre cas, $\xi = 0,29 < \frac{1}{2}$ donc l'espérance et la variance sont bien définies.

Il est également à noter que si $X \sim GPD(\sigma, \xi)$, alors $X - a | X > a \sim GPD(\sigma + \xi a, \xi)$. De plus, la loi de Pareto généralisée peut être simulée via la propriété : si $U \sim U[0, 1]$, alors $X \sim \sigma \left(\frac{U^{-\xi} - 1}{\xi}\right)$ suit une loi $GPD(\sigma, \xi)$.

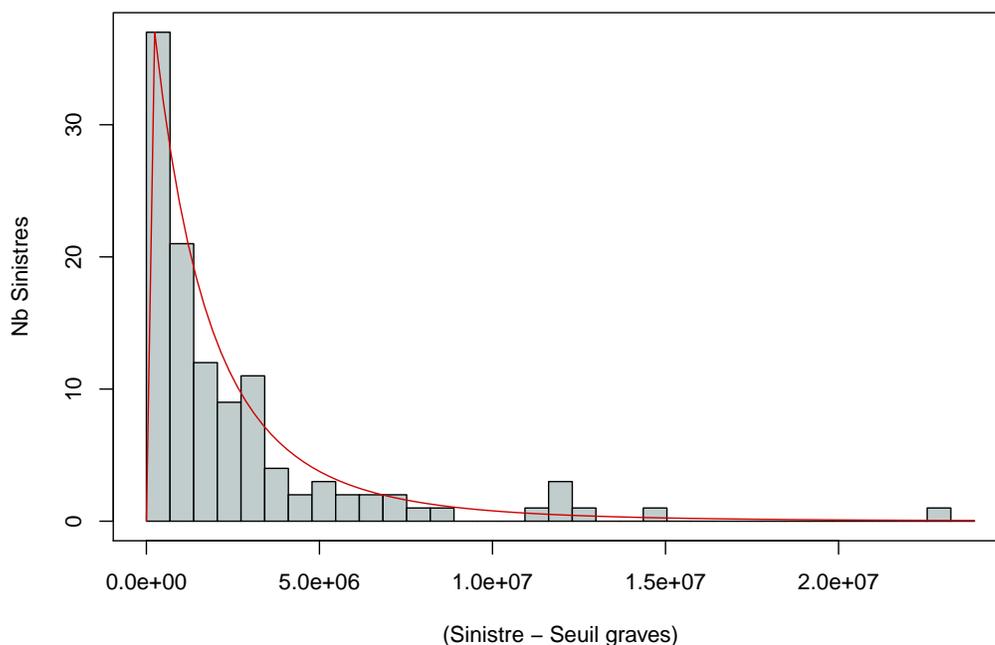


FIGURE 27 – Histogramme des sinistres graves et densité de la GPD retenue

Concernant les sinistres attritionnels, la variable aléatoire considérée est le loss-ratio annuel des sinistres attritionnels. Une loi de probabilité est donc calibrée à partir de l'historique des loss-ratios attritionnels, parmi les lois Exponentielle, Gamma, de Weibull et Log-Normale. Le processus n'est pas détaillé ici car nous nous intéressons essentiellement aux traités XS,

qui ne sont pas impactés par les sinistres attritionnels.

4. **Simulations des sinistres** : pour la phase de simulations, l'utilisateur à le choix entre deux options, soit utiliser les lois et paramètres calibrés automatiquement dans l'étape précédente, soit utiliser les lois et paramètres qu'il aura définis manuellement dans le fichier d'initialisation du modèle. p années de sinistralité supposées indépendantes sont ensuite simulées, où p est laissé au choix de l'utilisateur et doit être suffisamment grand pour obtenir une convergence de la prime pure. Pour une année k simulée donnée, N_k sinistres sont générés aléatoirement selon la loi de sévérité retenue, où N_k suit la loi de fréquence retenue.
5. **Estimation de la prime pure** : Pour chaque année k simulée, une prime pure est calculée pour chacun des traités XS renseignés en entrée du modèle. Pour un sinistre d'indice i de montant X_i donné, la formule pour calculer la cession est :

$$R_k^i = \min(\max(0, X_i - \text{Priorité}), \text{Portée}).$$

Sur l'année k , la cession théorique (i.e. avec un nombre infini de reconstitutions du traité) est alors :

$$R_k = \sum_{i=1}^{N_k} R_k^i.$$

Avec une *AAD* (Annual Aggregate Deductible) et une *AAL* (Annual Aggregate Limit), en considérant des reconstitutions gratuites, la cession réelle i.e. la prime pure du traité est donc égale à

$$P_k = \min(\max(0, R_k - AAD), AAL).$$

Les p années de simulations nous donnent donc une distribution empirique via les p réalisations des variables aléatoires $(P_k)_{k \in [1, p]}$. Nous pouvons ainsi estimer la prime pure $P^{pure} = E[P_k]$ via l'estimateur $\widehat{P^{pure}} = \frac{1}{p} \sum_{k=1}^p P_k$, qui vérifie bien $E[\widehat{P^{pure}}] = E[P_k]$ et est donc non-biaisé.

6. **Calcul de la prime commerciale** : A partir de la prime pure obtenue et des caractéristiques du traité XS considéré, nous pouvons calculer la prime commerciale.

Un premier chargement est introduit, appelé chargement pour risque. En raison du caractère stochastique de la prime pure, un capital doit être immobilisé pour contrer l'incertitude sur la prime pure engendrée par cet aspect stochastique. L'immobilisation de ce capital a un coût, fonction des exigences de rentabilité attendue. Le capital considéré peut être modélisé de plusieurs manières, basées sur différentes appréciations du risque sous-jacent. Le modèle propose trois mesures de risques $\mu(P^{pure})$ associées à la prime pure : l'écart-type, la Value-at-Risk à 99,5% et la Tail Value-at-Risk à 99%. un taux de chargement $taux_{capital}$ défini en entrée du modèle est ensuite appliqué à cette mesure de risque pour obtenir le chargement pour risque. Ainsi :

$$P^{risque} = P^{pure} + taux_{capital} \times \mu(P^{pure}).$$

Un chargement pour frais est ensuite appliqué, relatif aux frais internes et externes et notamment au courtage. Il est défini à partir d'un taux de frais $taux_{frais}$ défini en entrée du modèle, exprimé en pourcentage de la prime commerciale. Finalement :

$$P^{commerciale} = \frac{P^{risque}}{1 - taux_{frais}}.$$

D'autres indicateurs sont également calculés, comme le Rate On Line et le Pay Back :

$$ROL = \frac{P^{commerciale}}{Portée}, \text{ et } Pay\ Back = \frac{1}{ROL}.$$

2.5 Lexique de réassurance

Cette section présente brièvement différentes notions de réassurance citées dans les paragraphes précédents.

- **Réassurance** : opération par laquelle un assureur cède une partie de ses risques à une compagnie tierce, le réassureur, moyennant le versement d'une portion de la prime afférente à ces risques. Elle a plusieurs utilités pour l'assureur, comme la protection contre les sinistres extrêmes, le lissage des résultats ou l'augmentation de la capacité de souscription.
- **Mise en « as-if »** : la mise en as-if consiste à redresser des données historiques de sinistres et de primes, afin de se replacer dans les conditions économiques de l'année de cotation. Elle est généralement effectuée par le biais d'un indice adapté.
- **Traité en excédent de sinistre (XS)** : traité de réassurance non-proportionnel, pour lequel on fixe un seuil d'intervention du réassureur, la priorité, et une limite de prise en charge par le réassureur, la portée. Ainsi, pour un sinistre concerné par le traité, le réassureur paiera la partie du sinistre supérieure à la priorité, dans la limite de la portée.
- **Reconstitution** : permet de reconstituer la garantie relative à un traité XS, lorsqu'elle a été consommée. Le nombre de reconstitutions associées à un traité est variable (aucune, une, deux, ..., un nombre illimité). Une reconstitution peut être gratuite ou payante.
- **Annual Aggregate Deductible (AAD)** : l'AAD, si elle est présente dans un traité XS, définit un seuil minimal annuel à partir duquel le réassureur interviendra pour les sinistres de l'année entrant dans le périmètre du traité. En-deçà de ce seuil, le réassureur ne paie rien.

- **Annual Aggregate Limit (AAL)** : à l'inverse de l'AAD, l'AAL agit comme une limite haute, c'est le montant maximal qui sera pris en charge par le réassureur pour les sinistres de l'année entrant dans le périmètre du traité.
- **Clause de stabilité** : clause relative à un traité XS, permettant de partager l'inflation entre le réassureur et l'assureur lors du développement d'un sinistre pris en charge par le réassureur. Cette clause est spécifique aux branches longues.

3 Hypothèses préalables à l'application de méthodes d'interpolation

La partie précédente était dédiée à la description du modèle de tarification en réassurance. Elle a notamment permis de présenter les différentes variables du modèle et le caractère stochastique de la tarification par simulations. Avant de construire un métamodèle de tarification à proprement parler, il reste à définir les variables d'entrée et de sortie du métamodèle, étudier la convergence de la partie stochastique du modèle initial, et définir un budget en temps de calcul pour la construction du métamodèle.

Nous procéderons donc en trois temps :

- Premièrement, nous définirons les paramètres à considérer comme variables d'entrée pour la conception des métamodèles, c'est à dire les caractéristiques des traités XS dont nous souhaitons observer l'impact, lorsqu'elles varient, sur les résultats de tarification du modèle. Ce sont donc ces variables qui seront déterminantes dans l'optimisation des traités. Nous déterminerons également la variable de sortie à retenir dans le métamodèle.
- Deuxièmement, le modèle étant stochastique, nous déterminerons un nombre de simulations à effectuer afin d'avoir une convergence satisfaisante avant l'application des méthodes d'interpolation.
- Troisièmement, nous définirons un nombre total de points à utiliser dans les différentes méthodes d'échantillonnage et de sélection séquentielle, points sur lesquels sera basée la construction des métamodèles. Cela nous permettra de limiter les temps de calcul en limitant le nombre d'exécutions du modèle.

3.1 Choix des variables à étudier

Dans cette section, nous allons présenter les variables d'entrée et de sortie du modèle initial retenues pour la construction des métamodèles.

Il nous faut premièrement définir une variable de sortie du modèle initial, qui sera la variable de sortie de notre métamodèle. La sortie retenue sera la variable mesurant l'efficacité du traité de réassurance en excédent de sinistre considéré, définie par :

$$E = -(1 - \rho)R + \alpha G$$

avec :

- $R = P^{\text{commerciale}} - S^{\text{cédés}}$, le résultat du traité XS de réassurance, défini comme la prime commerciale moins les sinistres cédés.
- $\rho = 33, 33\%$ le taux d'imposition sur le résultat.

- $G = q_{99,5\%}(S^{bruts}) - q_{99,5\%}(S^{nets})$, le gain en capital, défini comme la différence entre les quantiles à 99,5% de la sinistralité brute (de réassurance) et de la sinistralité nette (de réassurance).
- $\alpha = ROE - Taux_{rendement\ placements}$ le taux associé au coût du capital. Ce taux n'est pas disponible dans les données que nous avons reçues de la part de la cédante. Nous construirons nos métamodèles en choisissant $\alpha = 7\% - 2\% = 5\%$, en se basant sur les taux usuels observés sur le marché de l'assurance.

Il reste maintenant à définir les paramètres d'entrée du métamodèle. Une fois définis, les autres paramètres d'entrée du modèle initial seront considérés constants. Notre objectif est d'étudier l'efficacité E du traité en fonction de ses caractéristiques. Nous choisirons ainsi comme variables d'entrée du métamodèle, quatre variables relatives au traité XS :

- La priorité du traité, que nous noterons a .
- La portée du traité, que nous noterons b .
- La clause de franchise annuelle du traité, l' AAD , et la clause de limite annuelle du traité, l' AAL , qui seront traitées dans un deuxième temps.

La conception de métamodèles à partir de ces variables va ainsi nous permettre d'étudier l'efficacité d'un traité en fonction de ses paramètres intrinsèques.

Les variables d'entrée et de sortie vont donc servir de base à la construction des différents métamodèles, et notamment permettre d'estimer et d'ajuster les paramètres/hyper-paramètres de ces nouveaux modèles. Nous utiliserons en premier lieu la priorité et la portée comme variables d'entrée des métamodèles.

Les deux graphiques ci-dessous donnent un premier aperçu de l'évolution de l'efficacité en fonction de la priorité, puis de la portée d'un traité :

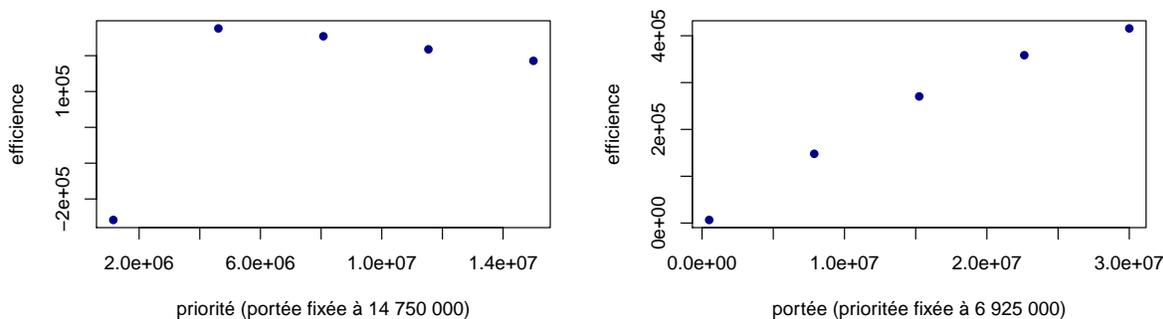


FIGURE 28 – Efficacité d'un traité XS selon la portée et la priorité

3.2 Nombre de simulations à effectuer

Les sorties du modèle de réassurance sur lesquelles seront construits les métamodèles et donc appliquées les méthodes d'interpolations, sont basées sur le calcul stochastique de la prime pure. Ainsi, d'un run à l'autre, en changeant la « graine » utilisée pour la génération des nombres pseudo-aléatoires, les sorties obtenues seront différentes. La loi faible des grands nombres nous indique cependant que l'estimateur $\widehat{P^{pure}}$ défini en 2.4 converge en probabilité vers la prime pure P^{pure} . Les méthodes d'interpolation décrites dans ce mémoire sont initialement vouées à s'appliquer à des modèles aux sorties déterministes. Nous allons ainsi déterminer un nombre suffisant de simulations pour obtenir un degré de convergence convenable de l'estimateur vers la prime pure. En fixant ensuite la graine de génération des nombres pseudo-aléatoires du modèle, nous obtiendrons un modèle aux sorties déterministes d'un run à l'autre, dont nous saurons qu'elles respectent un certain critère de convergence vers la prime pure. Il sera alors possible d'appliquer les méthodes d'interpolation en question sur ce modèle.

Pour choisir le nombre de simulations à effectuer, nous allons adopter une approche théorique que nous validerons empiriquement.

3.2.1 Détermination théorique du nombre de simulations

Dans cette partie, nous allons déterminer un nombre de simulations à effectuer de manière théorique, en utilisant l'inégalité de Bienaymé-Tchebychev :

$$\forall \eta > 0, P(|Y - E[Y]| \geq \eta) \leq \frac{Var[Y]}{\eta^2},$$

pour Y variable aléatoire admettant un moment d'ordre deux.

Ici, $Y = \widehat{P^{pure}} = \frac{1}{p} \sum_{k=1}^p P_k$. Les (P_k) étant i.i.d., $Var[\widehat{P^{pure}}] = \frac{pVar[P_k]}{p^2} = \frac{Var[P_k]}{p}$, et $E[\widehat{P^{pure}}] = \frac{pE[P_k]}{p} = E[P_k] = P^{pure}$. Pour un traité de priorité $a > s = 1\,150\,000$ et de portée $b > 0$ (nombre de reconstitutions illimitées et gratuites), les (P_k) ont la loi de $S = \sum_{i=1}^{N_a} X_i^{a,b}$, avec $N_a = \sum_{i=1}^N \mathbf{1}_{\{X_i > a\}}$ et $X_i^{a,b} = \min((X_i - a | X_i > a), b)$. D'où $P^{pure} = E[P_k] = E[S]$ et $Var[P_k] = Var[S]$.

Remarque : les (X_i) sont i.i.d., donc les $(X_i^{a,b})$ le sont aussi.

L'inégalité devient donc :

$$P\left(\left|\widehat{P^{pure}} - P^{pure}\right| \geq \eta\right) = P\left(\left|\frac{1}{p} \sum_{k=1}^p P_k - P^{pure}\right| \geq \eta\right) \leq \frac{Var[S]}{p\eta^2}.$$

Ainsi, par l'inégalité de Bienaymé-Tchebychev, il est possible d'obtenir un intervalle de confiance pour $\widehat{P^{pure}}$ autour de $P^{pure} = E[\widehat{P^{pure}}]$, selon η que nous allons définir comme un pourcentage ϵ

de la prime pure : $\eta = \epsilon P^{pure} = \epsilon E[S]$.

Finalement, l'inégalité s'écrit :

$$P\left(\left|\widehat{P^{pure}} - P^{pure}\right| \geq \epsilon P^{pure}\right) \leq \frac{Var[S]}{p\epsilon^2(P^{pure})^2} = \frac{Var[S]}{p\epsilon^2 E[S]^2} = \alpha.$$

En se donnant un pourcentage ϵ et une probabilité α , nous pouvons donc déterminer p afin que $\widehat{P^{pure}}$ soit dans l'intervalle $[(1 - \epsilon)P^{pure}, (1 + \epsilon)P^{pure}]$ avec une probabilité égale à $1 - \alpha$.

Le nombre de simulations retenu sera ainsi $p = \frac{Var[S]}{\alpha\epsilon^2 E[S]^2}$. Concernant les paramètres ϵ et α , nous prendrons $\epsilon = 0,01$ et $\alpha = 0,05$. Autrement dit, p défini de la sorte garantit un écart relatif entre l'estimateur et la vraie prime pure inférieur à 1%, avec un niveau de confiance de 95%. Il reste maintenant à calculer l'espérance et la variance de S .

- Espérance de S :

$$E[S] = E\left[\sum_{i=1}^{N_a} X_i^{a,b}\right] = E[N_a]E[X_1^{a,b}]$$

Avec

$$E[N_a] = E\left[\sum_{i=1}^N \mathbb{1}_{\{X_i > a\}}\right] = E[N]E[\mathbb{1}_{\{X_1 > a\}}] = E[N]P(X_1 > a) = E[N](1 - F_{X_1}(a)).$$

D'autre part, nous savons que $\forall i, X_i - s | X_i > s \sim GPD(\sigma, \xi)$, avec $s = 1\,150\,000$. Ainsi, $\forall a > s, X_i - a | X_i > a \sim GPD(\sigma_a, \xi)$, où $\sigma_a = \sigma + \xi(a - s)$.

$$X_i^{a,b} = \begin{cases} (X_i - a | X_i > a) & \text{lorsque } (X_i - a | X_i > a) < b \\ b & \text{lorsque } (X_i - a | X_i > a) \geq b \end{cases}$$

Ainsi,

$$\begin{aligned} E[X_1^{a,b}] &= \int_0^b P((X_1 - a | X_1 > a) > x) dx \\ &= \int_0^b \left(1 + \xi \frac{x}{\sigma_a}\right)^{-\frac{1}{\xi}} dx \\ &= \frac{\sigma_a}{1 - \xi} \left[1 - \left(1 + \xi \frac{b}{\sigma_a}\right)^{1 - \frac{1}{\xi}}\right] \end{aligned}$$

- Variance de S :

$$\begin{aligned}
Var[S] &= E[Var[S|N_a]] + Var[E[S|N_a]] \\
&= E[N_a Var[X_1^{a,b}]] + Var[N_a E[X_1^{a,b}]] \\
&= E[N_a] Var[X_1^{a,b}] + E[X_1^{a,b}]^2 Var[N_a]
\end{aligned}$$

$$\begin{aligned}
\text{Avec } Var[N_a] &= E[Var[N_a|N]] + Var[E[N_a|N]] \\
&= E[N Var[\mathbf{1}_{\{X_1 > a\}}]] + Var[NE[\mathbf{1}_{\{X_1 > a\}}]] \\
&= E[N] Var[\mathbf{1}_{\{X_1 > a\}}] + E[\mathbf{1}_{\{X_1 > a\}}]^2 Var[N] \\
&= E[N] P(X_1 > a)(1 - P(X_1 > a)) + P(X_1 > a)^2 Var[N] \\
&= E[N](1 - F_{X_1}(a))F_{X_1}(a) + (1 - F_{X_1}(a))^2 Var[N] \\
&= (1 - F_{X_1}(a))(E[N]F_{X_1}(a) + (1 - F_{X_1}(a))Var[N])
\end{aligned}$$

$$\begin{aligned}
\text{Et } Var[X_1^{a,b}] &= E[(X_1^{a,b})^2] - E[X_1^{a,b}]^2 \\
\text{où } E[(X_1^{a,b})^2] &= 2 \int_0^b x P((X_1 - a | X_1 > a) > x) dx \\
&= 2 \int_0^b x \left(1 + \xi \frac{x}{\sigma_a}\right)^{-\frac{1}{\xi}} dx \\
&= \frac{2\sigma_a}{1 - \xi} \left[\frac{\sigma_a}{1 - 2\xi} \left(1 - \left(1 + \xi \frac{b}{\sigma_a}\right)^{2 - \frac{1}{\xi}}\right) - b \left(1 + \xi \frac{b}{\sigma_a}\right)^{1 - \frac{1}{\xi}} \right]
\end{aligned}$$

Ainsi,

$$Var[S] = (1 - F_{X_1}(a))(E[N]Var[X_1^{a,b}] + E[X_1^{a,b}]^2(E[N]F_{X_1}(a) + (1 - F_{X_1}(a))Var[N]))$$

Nous connaissons à présent l'expression de l'espérance et de la variance de S , qui est fonction de la priorité a et de la portée b du traité considéré. Nous noterons $Var[S] = H(a, b)$. Le nombre p de simulations requises pour être dans l'intervalle de confiance fixé s'écrit donc $p(a, b) = \frac{H(a,b)}{\alpha \epsilon^2 E[S]^2}$. Nous devons donc définir un nombre de simulations qui assurera le respect du critère de convergence pour tous les traités en entrée du modèle. Il faut donc choisir $p_{max} = \max_{a,b} p(a, b)$. Cette valeur est dépendante du domaine de définition de a , noté $A = [1 \ 150 \ 000, a_{max}]$, et du domaine de définition de b , noté $B = [b_{min}, b_{max}]$. Il faut donc choisir une valeur de priorité maximale et des valeurs de portée minimale et maximale. La priorité et la portée d'un traité sont calibrées en fonction de plusieurs facteurs, notamment le profil de risque sur le portefeuille en question, l'appétence au risque de la cédante, des contraintes de budget...etc.

Dans notre cas, l'idéal serait d'avoir les plus larges espaces de définition possibles afin de pouvoir analyser une grande variété de traités. Cependant, nous observons que $p(a, b)$ croît avec a et b :

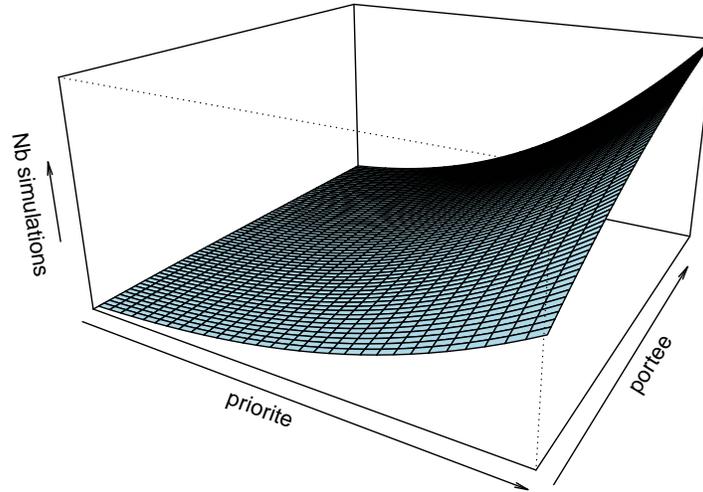


FIGURE 29 – $p(a, b)$ en fonction de la priorité a et de la portée b

Un nombre trop élevé de simulations alourdira fortement l'exécution du modèle, en faisant augmenter les temps de calcul. Afin de borner les espaces A et B nous retiendrons les hypothèses suivantes :

- Pour A , la valeur $a_{max} = 15\,000\,000\text{€}$ est choisie comme valeur maximale de la priorité d'un traité. Nous observons que ce montant donne $a_{max} - s = q_{97,8\%}(X_i - s | X_i > s)$, quantile à 97,8% de la sévérité d'un sinistre grave brut de réassurance en excès du seuil s . En se mettant à la place de la cédante, il semble acceptable de retenir une telle valeur, signifiant qu'en théorie, même pour le traité avec la priorité la plus haute, 2,2% des sinistres graves seront tout de même concernés par une cession en réassurance.
- Pour B , la valeur b_{max} est choisie telle que $a_{max} + b_{max} - s = q_{99,9\%}(X_i - s | X_i > s)$, quantile à 99,9% de la sévérité d'un sinistre grave brut de réassurance. Cela signifie qu'en théorie, pour le traité au plafond le plus élevé, seulement 0,1% des sinistres graves dépassera ce plafond. Ici, $q_{99,9\%}(X_i - s | X_i > s) = 43\,676\,302\text{€}$, d'où $b_{max} = 29\,826\,302\text{€}$. Une valeur quelque peu supérieure sera finalement retenue pour être plus cohérent en termes de définition de traité : $b_{max} = 30\,000\,000\text{€}$.

La valeur $b_{min} = 500\,000\text{€}$ est choisie volontairement basse pour élargir le spectre des traités. Il n'y a en effet pas de contrainte sur le choix de cette valeur concernant le nombre de simulations, ce dernier étant conditionné par les valeurs a_{max} et b_{max} .

De ces choix découle la valeur $p_{max} = 1\,440\,848$. Nous prendrons une légère marge en fixant, en définitive, le nombre de simulations à 1 500 000 pour chaque exécution de notre modèle.

3.2.2 Validation empirique

Afin de vérifier empiriquement le nombre p_{max} de simulations déterminé théoriquement, le graphique suivant présente les résultats obtenus pour la prime pure d'un traité avec les caractéristiques $a_{max} = 15\,000\,000\text{ €}$ et $b_{max} = 30\,000\,000\text{ €}$, selon le nombre de simulations effectuées :

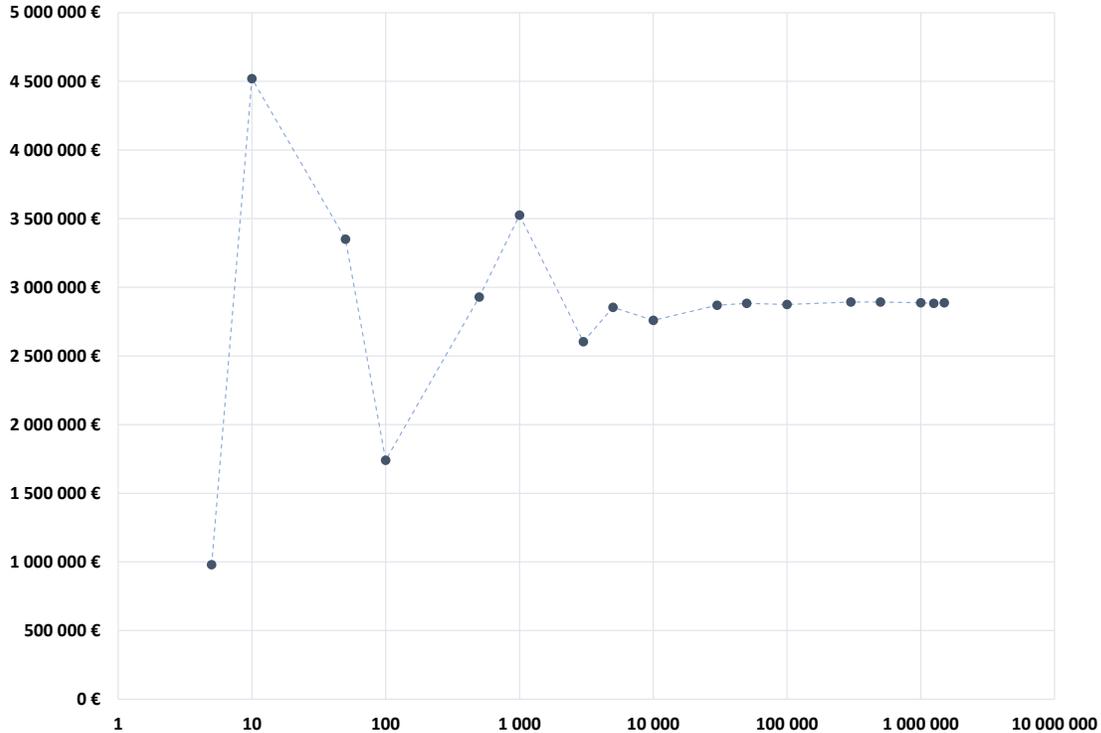


FIGURE 30 – Prime pure estimée par le modèle en fonction du nombre de simulations

3.3 Nombre de points initiaux

Dans l'approche de sélection simultanée des points (*one-shot design*), il est nécessaire de définir un nombre fixe de points pour définir l'hypercube de départ, i.e. le plan d'expérience. Ce nombre de points dépend de plusieurs paramètres, comme la dimension de l'espace des entrées du métamodèle, la méthode utilisée pour construire l'hypercube (la construction par échantillonnage aléatoire nécessite par exemple un nombre de points conséquent pour produire de bons résultats), la précision souhaitée du métamodèle, le temps d'exécution du modèle initial, le budget accordé en temps de calcul...etc.

Dans l'approche de sélection séquentielle des points, il est également nécessaire de définir un nombre de points de départ, sensible aux mêmes paramètres que dans l'approche simultanée, mais

généralement inférieur, car d'autres points seront ajoutés au fur et à mesure des itérations de l'algorithme de construction séquentielle du métamodèle. Un critère d'arrêt doit être défini pour cet algorithme. Cela peut être, par exemple, l'atteinte d'un nombre total de points sur lesquels exécuter le modèle, mais aussi un critère de précision du métamodèle. Il n'y a donc pas nécessairement un nombre total de points fixé au préalable, en amont de la construction du métamodèle.

Dans ce mémoire, nous fixerons cependant un nombre total de points n_c à utiliser pour la construction du métamodèle, aussi bien pour l'approche simultanée que séquentielle. Le critère d'arrêt pour la construction séquentielle ne sera donc pas un critère de précision. Il est à noter qu'il faudra également prendre en compte l'effet de la dimension de l'espace d'entrée du métamodèle dans la détermination du nombre de points.

En parallèle, comme expliqué dans la sous-section 1.4.1, il est nécessaire de constituer une base de validation afin d'évaluer la qualité de prédiction du métamodèle construit. Le nombre de points constituant cette base doit être suffisamment élevé pour permettre de d'évaluer convenablement le métamodèle sur l'ensemble de l'espace. Le nombre de points contenu dans la base de validation sera noté n_v .

Nous déterminerons le nombre total de points $n = n_c + n_v$ à partir d'une contrainte de budget, exprimée en temps de calcul. Cette contrainte sera fixée à une heure. Autrement dit, il faudra qu'en une heure, le modèle ait produit les n résultats d'efficacité E pour les n points correspondants (les traités XS). Il est donc nécessaire de connaître le temps d'exécution du modèle en fonction du nombre de traités (i.e. du nombre de points) renseignés en entrée.

Le graphique suivant montre l'évolution de ce temps d'exécution en secondes après ajouts successifs de traités de paramètres $(a, b) \in A \times B$ choisi aléatoirement (de façon uniforme). Le nombre de simulations est fixé à $p_{max} = 1\,500\,000$.

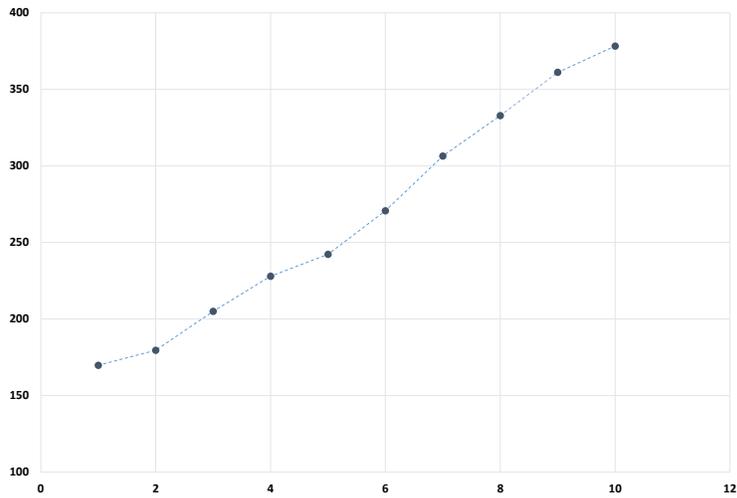


FIGURE 31 – Evolution du temps d'exécution du modèle (en s) en fonction du nombre de traités

A partir de deux traités, la tendance est presque linéaire, ce qui nous conduit à prédire le temps d'exécution en fonction du nombre de traités par une équation de la forme $y = ax + b$, avec $a = 25,62$ et $b = 124,54$ déterminés par une rapide régression linéaire sur les 9 points (de 2 à 10 traités). Pour $y = 3600$, nous obtenons $x = 135,7$. Nous prévoyons donc que 135 traités au maximum permettront de respecter le budget retenu en temps de calcul, ou du moins de s'en approcher convenablement.

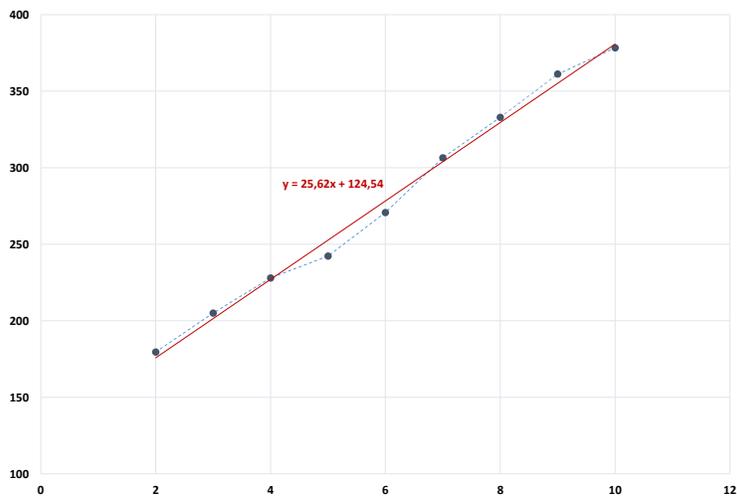


FIGURE 32 – Régression linéaire sur les temps d'exécution (en s) en fonction du nombre de traités

Il faut ensuite répartir ces $n = 135$ points entre les points dédiés à la construction du métamodèle et les points dédiés à sa validation. Notre objectif est de construire un métamodèle à partir d'un nombre restreint de points, et de pouvoir ensuite correctement mesurer sa qualité de prédiction sur un nombre conséquent de points répartis sur l'ensemble du domaine de définition, comme expliqué en 1.4.1. Dans cette optique, nous choisissons en première approche un nombre de points $n_c = \frac{1}{5}n = 27$. Cependant, la génération de points par échantillonnage factoriel nous impose $n_c = k^d$ avec k entier et $d = 2$ la dimension de l'espace d'entrée. Ainsi, nous retenons finalement $n_c = 5^2 = 25$. Il n'existe pas de règle précise permettant de définir le nombre de points n_c nécessaire à la construction d'un métamodèle, ce nombre étant fonction de nombreux paramètres. Cependant, une heuristique répandue est de choisir environ 10 points par dimension. Notre choix final est donc dans le même ordre de grandeur que cette heuristique. Il reste ensuite $n_v = 110$ points pour constituer la base de validation, qui seront échantillonnés dans les différentes zones du domaine de définition via un hypercube latin.

4 Construction d'un métamodèle de tarification en réassurance

Dans cette partie, nous allons appliquer les méthodes présentées dans la première partie au modèle de tarification en réassurance précédemment décrit. Plus précisément, ces méthodes seront appliquées à la partie du modèle dédiée à la tarification par simulations des traités de réassurance non-proportionnels en excédent de sinistre. La tarification par simulations étant coûteuse en temps de calcul, l'objectif sera donc, à partir d'un nombre limité de lancements du modèle, d'approcher au mieux les résultats de ce dernier via les techniques d'interpolation précédemment décrites. L'idée générale est d'obtenir un métamodèle, plus rapide à l'exécution, et permettant une exploration des résultats sur l'ensemble de l'espace de définition des paramètres du modèle, notamment de manière graphique. La construction d'un métamodèle offrira également la possibilité de déterminer des traités de réassurance « optimaux », c'est à dire les traités les plus efficaces. Pour rappel, la variable de sortie à étudier est l'efficacité $E = -(1 - \rho)R + \alpha G$, avec $\rho = 33,33\%$ et $\alpha = 5\%$.

Nous appliquerons d'abord les méthodes de construction de métamodèle par génération simultanée de points initiaux, puis les méthodes de construction séquentielle.

4.1 Construction via un plan d'expérience fixe

Cette partie est dédiée à la construction de plusieurs métamodèles à partir d'un échantillon de points générés simultanément. Nous utiliserons différentes méthodes d'échantillonnage et d'interpolation afin de concevoir et comparer plusieurs métamodèles. La taille de l'échantillon de départ, soit le nombre de traités à générer avec le modèle pour la construction du métamodèle, est $n_c = 25$. La base de validation est composée de $n_v = 110$ points, générés via un hypercube latin.

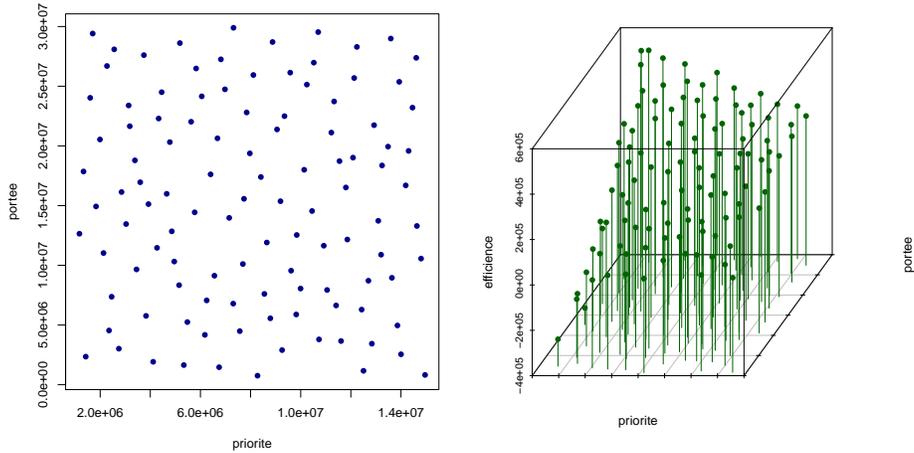


FIGURE 33 – Traités constituant la base de validation (hypercube latin)

4.1.1 Krigeage

Dans cette section, trois modèles de krigeage seront construits, basés sur les trois méthodes d'échantillonnage présentées auparavant : l'échantillonnage aléatoire, factoriel, et par hypercube latin.

— **Modèle issu d'un échantillonnage aléatoire :**

Ci-dessous, l'échantillon aléatoire généré pour les variables *priorité* et *portée*, ainsi que les résultats d'efficacité calculés par le modèle sur ces points.

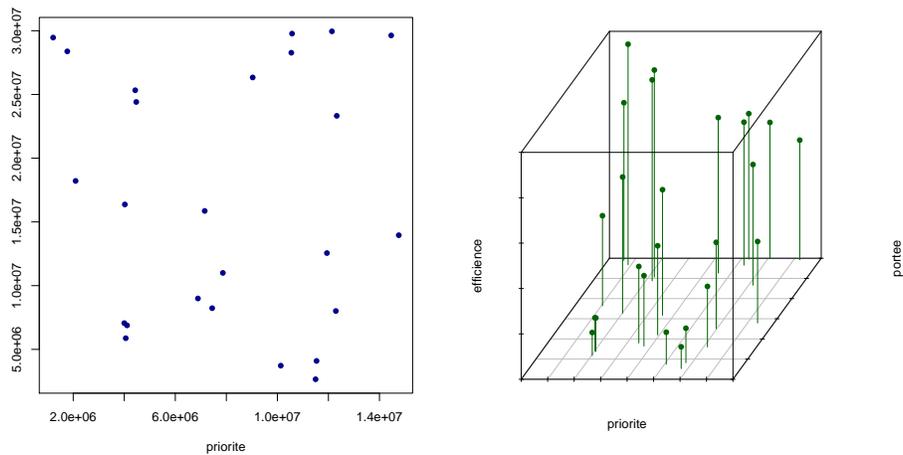


FIGURE 34 – Echantillon aléatoire de couples (*priorité*, *portée*) et efficacités associées

La première étape dans la construction du modèle de krigeage consiste en la spécification de la fonction de covariance à utiliser. Comme vu précédemment, nous utilisons une validation croisée de type LOO (LOO-CV ou *leave-one-out cross-validation*) pour déterminer le noyau de covariance le plus adapté, effectuée via la fonction `leaveOneOut.km` du package `DiceKriging`. Nous retiendrons le noyau avec l'EQMP (erreur quadratique moyenne de prédiction) la plus basse suite à la validation croisée.

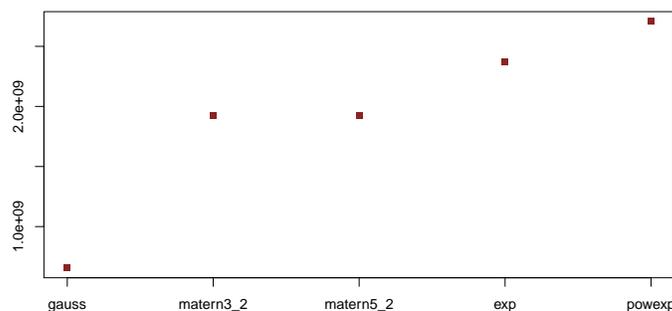


FIGURE 35 – EQMP sur la LOO cross-validation, selon le noyau de covariance

D'après les EQMP représentées à la figure 35, le noyau retenu sera donc de type gaussien.

Le modèle de krigeage est construit sous R via la fonction `km` du package `DiceKriging`. Les paramètres μ (moyenne du modèle de krigeage ordinaire considérée comme une constante inconnue), σ et α (paramètres de la fonction de covariance k) sont estimés par maximum de vraisemblance, via un algorithme génétique. La matrice \mathbf{K}_n est alors calculée et inversée. A cette étape, les différents éléments composant le modèle de krigeage sont tous définis, et le modèle peut donc être utilisé dans un cadre de prédiction.

La surface ci-dessous représente les valeurs prédites par le modèle de krigeage avec un noyau de type gaussien, sur l'ensemble de la surface de définition des variables *priorité* et *portée* :

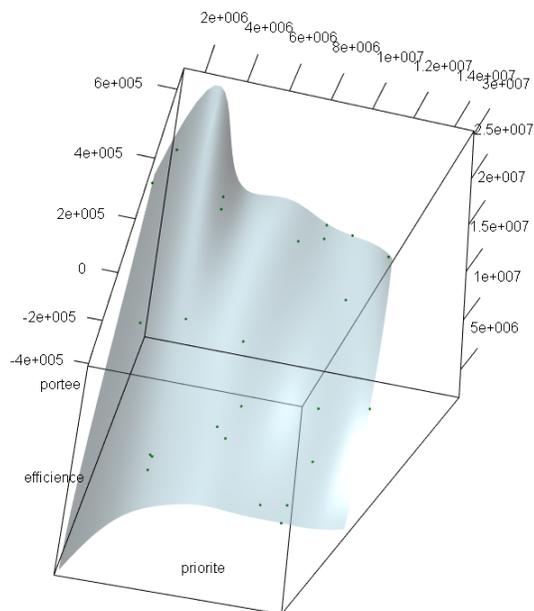


FIGURE 36 – Interpolation par krigeage à partir d'un échantillon aléatoire

Afin d'estimer la qualité de prédiction du modèle de krigeage ainsi construit, nous allons comme prévu évaluer sa précision sur les données composant la base de validation. Les deux graphiques ci-dessous présentent les résultats obtenus par krigeage sur les données composant la base de validation par rapport aux valeurs réelles d'efficacité sur cette même base.

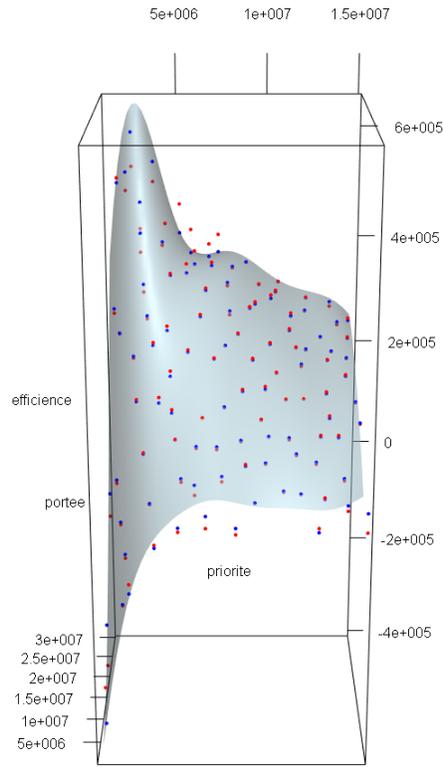


FIGURE 37 – Prédications d’efficience par krigeage (en bleu) et valeurs réelles (en rouge)

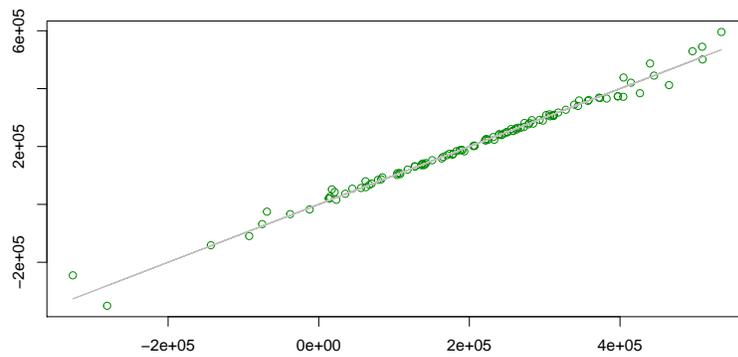


FIGURE 38 – Prédications d’efficience par krigeage (ordonnée) vs Valeurs réelles (abscisse)

Sur les valeurs extrêmes d’efficience, le modèle de krigeage affiche une certaine volatilité. La précision est meilleure pour des valeurs situées entre 0 et 400 000. Cette perte de qualité de prédiction aux extrêmes est liée au faible nombre d’observations dans ces zones qui ne

représentent que de petites parties de l'espace.

— **Modèle issu d'un échantillonnage factoriel :**

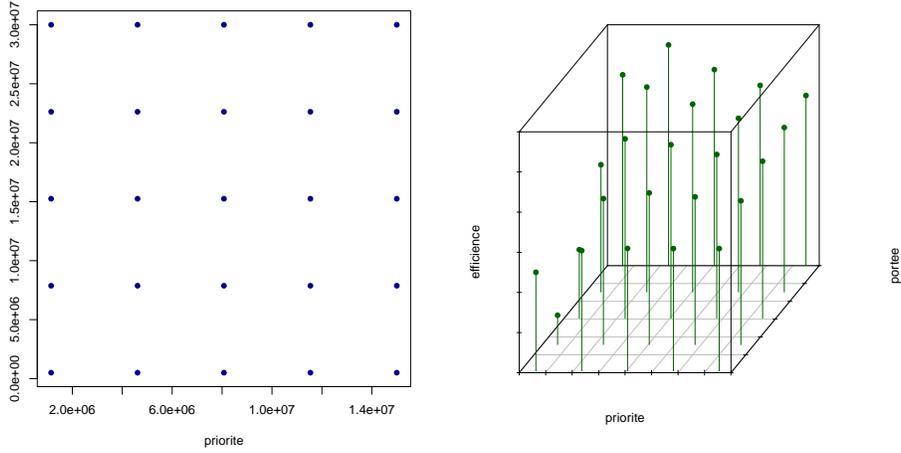


FIGURE 39 – Echantillon factoriel de couples (*priorité*, *portée*) et efficacités associées

En utilisant la même méthode que pour l'échantillonnage aléatoire, nous obtenons un noyau de covariance de type exponentiel-puissance pour le modèle de krigeage issu de l'échantillonnage factoriel. Les prédictions obtenues sur la base de validation sont les suivantes :

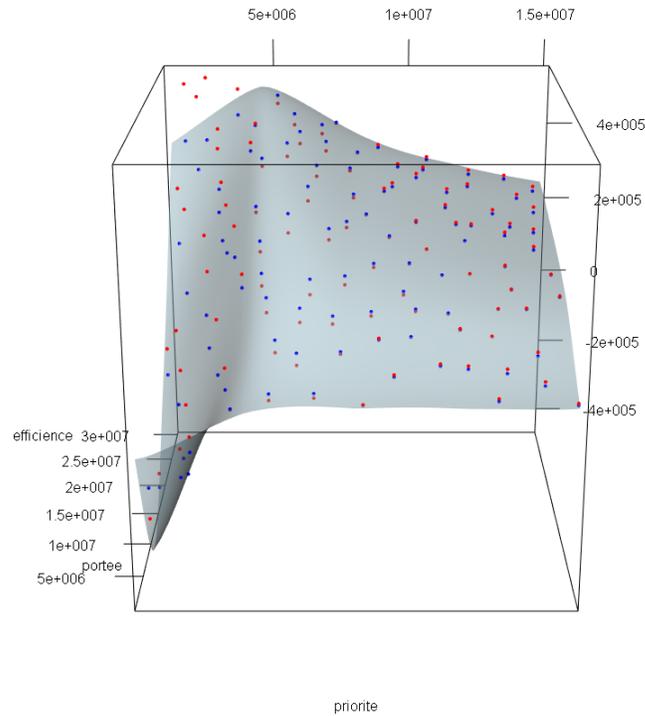


FIGURE 40 – Prédications d’efficience par krigeage (en bleu) et valeurs réelles (en rouge)

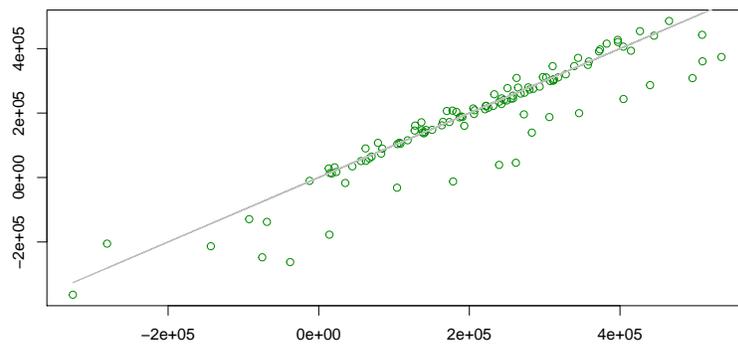


FIGURE 41 – Prédications d’efficience par krigeage (ordonnée) vs Valeurs réelles (abscisse)

Ce modèle de krigeage montre une nette tendance à la sous-estimation, quasi-systématique pour les valeurs extrêmes d’efficience. Cela est dû au fait que la configuration des points via l’échantillonnage factoriel n’a pas permis de détecter les zones d’efficience extrêmes du

modèle initial. D'une première approche graphique, ce métamodèle semble moins performant que le précédent construit par échantillonnage aléatoire.

— **Modèle issu d'un échantillonnage par hypercube latin :**

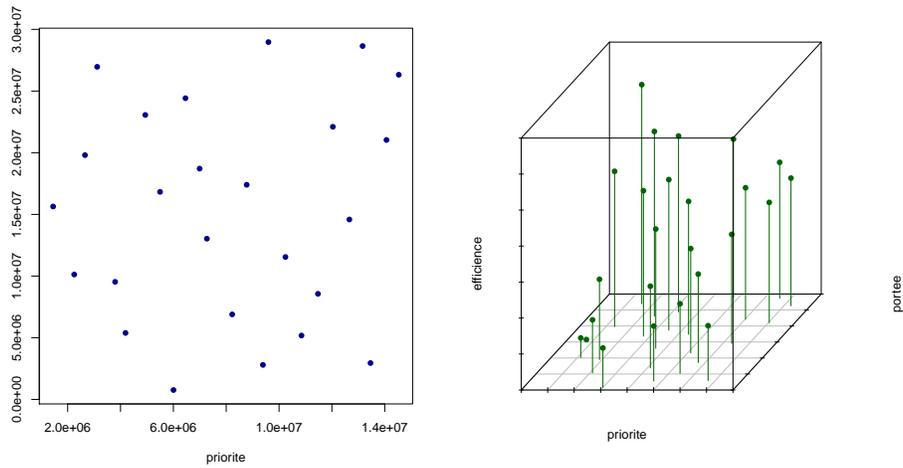


FIGURE 42 – Echantillon latin de couples (*priorité*, *portée*) et résultats d'efficience associés

Ici, le noyau retenu est de type gaussien. Les prédictions obtenues sur la base de validation sont les suivantes :

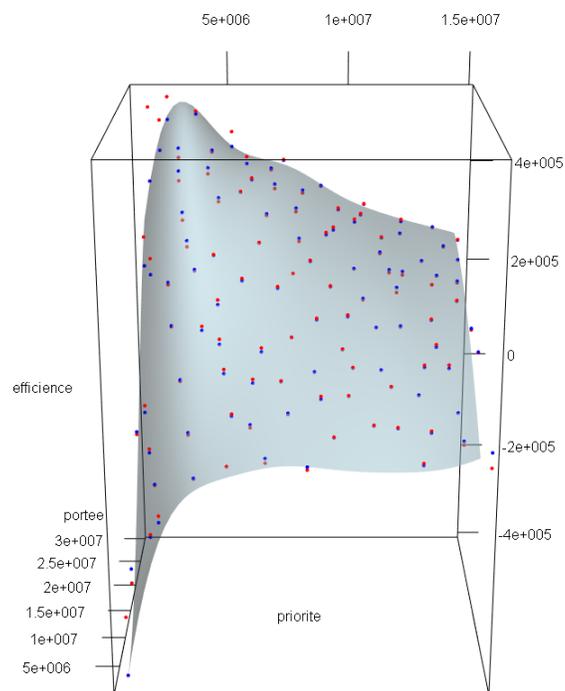


FIGURE 43 – Prédications d’efficience par krigeage (en bleu) et valeurs réelles (en rouge)

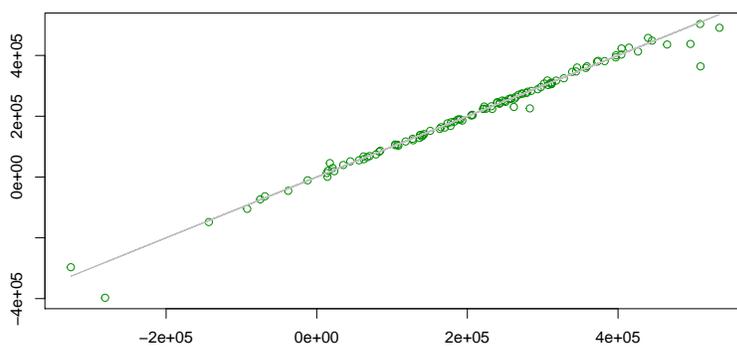


FIGURE 44 – Prédications d’efficience par krigeage (ordonnée) vs Valeurs réelles (abscisse)

Ici, les résultats de prédiction paraissent satisfaisants et plutôt précis, même s’il subsiste de la volatilité pour les valeurs extrêmes d’efficience.

Nous avons mis en place trois modèles de krigeage basés sur trois méthodes d'échantillonnage différentes. Des valeurs d'efficience ont ensuite été prédites pour la base de validation via ces trois métamodèles. En plus des analyses graphiques effectuées précédemment, il est possible de mesurer la performance de ces métamodèles en calculant l'erreur quadratique moyenne de prédiction pour chacun d'entre eux, ou encore le coefficient Q^2 :

Modèles de krigeage			
Echantillonnage	Aleatoire	Latin	Factoriel
EQMP	309 707 763	455 336 327	4 550 935 460
EQMP ^{1/2}	17 599	21 339	67 461
Q ²	0,987	0,981	0,815

FIGURE 45 – $EQMP$ et Q^2 selon le modèle de krigeage

Le modèle de krigeage le plus performant (i.e à l' $EQMP$ minimale et au Q^2 maximal), est celui construit en utilisant un échantillonnage aléatoire. Celui construit par hypercube latin présente une qualité de prédiction presque identique. Dans notre cas, une configuration de départ par plan factoriel engendre un modèle de krigeage nettement moins performant que pour les deux autres types de configurations. Les résultats restent cependant satisfaisants dans les trois cas.

4.1.2 Splines bicubiques

Ici, nous n'utiliserons que l'échantillonnage factoriel. Les splines bicubiques réalisent en effet une interpolation polynomiale par morceaux, chaque polynôme étant défini entre deux points pour lesquels les valeurs du modèle sont connues. Étendre l'interpolation polynomiale au-delà de l'hyper-polygone formé par les points de résultats connus posent donc de grandes difficultés de modélisation. Seul l'échantillonnage factoriel assure d'obtenir des points dans tous les coins de l'hypercube. L'interpolation par splines peut alors s'effectuer sans problème dans tout l'espace, et permettre ainsi la construction d'un métamodèle sur l'ensemble de l'hypercube.

Ci-dessous, les résultats d'efficience calculés par le modèle sur l'échantillon factoriel (le même qu'en figure 39) et l'interpolation effectuée par splines bicubiques entre les points obtenus.

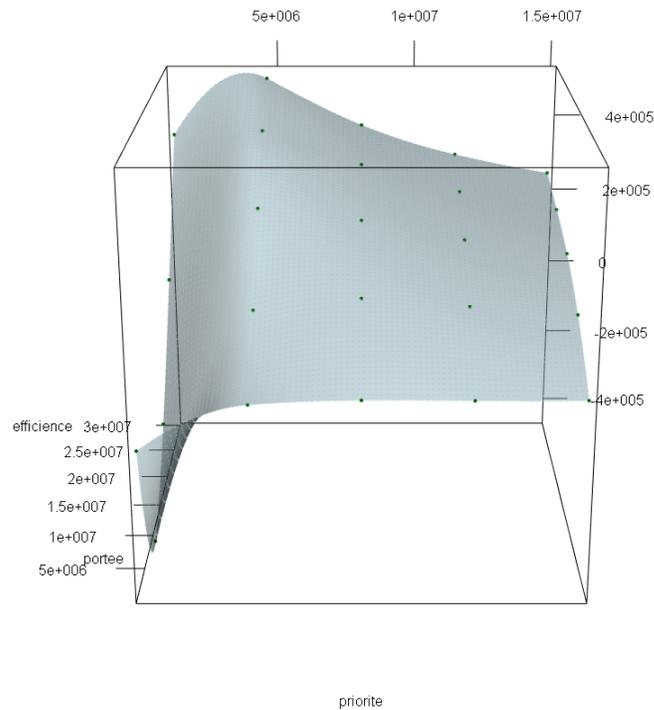


FIGURE 46 – Interpolation par splines bicubiques à partir d'un échantillon factoriel

Graphiquement, en comparaison avec les modèles de krigeage construits, la surface obtenue ici via l'interpolation par splines est légèrement plus lisse. Comme pour les modèles construits par krigeage, nous mesurons la qualité de prédiction de l'interpolation par splines sur la base de validation.

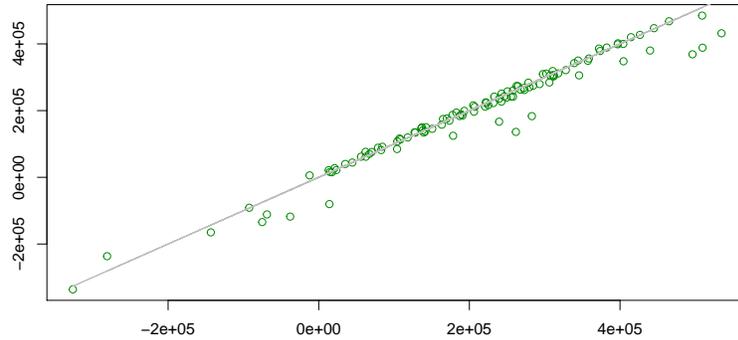


FIGURE 47 – Prédictions d’efficience par splines (ordonnée) vs Valeurs réelles (abscisse)

Visuellement, les prédictions obtenues avec les splines affichent une légère tendance à la sous-estimation. Nous pouvons ensuite calculer l’erreur quadratique moyenne de prédiction, $EQMP = 1\,033\,081\,498$. Il vient donc $EQMP^{1/2} = 32\,142$, et d’autre part $Q^2 = 0,958$. Pour l’échantillonnage factoriel, l’interpolation par splines s’avère ainsi plus performante (au sens du coefficient Q^2) que l’interpolation par krigeage. Cependant, les modèles de krigeage construits par échantillonnage aléatoire et latin offrent quant à eux une meilleure qualité de prédiction que les splines.

4.1.3 Comparaison des méthodes

Le tableau suivant récapitule les résultats obtenus pour les différents métamodèles de tarification construits :

Métamodèles construits - 25 points				
Interpolation	Krigeage			Splines bicubiques
Echantillonnage	Aleatoire	Latin	Factoriel	Factoriel
EQMP	309 707 763	455 336 327	4 550 935 460	1 033 081 498
$EQMP^{1/2}$	17 599	21 339	67 461	32 142
Q^2	0,987	0,981	0,815	0,958

FIGURE 48 – $EQMP$ et Q^2 selon le type de métamodèle

Avec un Q^2 de 0,987, l’interpolation par krigeage à échantillonnage aléatoire offre une très bonne qualité de prédiction, suivie par le modèle de krigeage à échantillonnage par hypercube latin où $Q^2 = 0,981$.

Il faut néanmoins prendre ces résultats avec précaution. En effet, nous allons voir que ces résultats sont sensibles à plusieurs facteurs, et notamment au nombre de points utilisés. De plus, les

échantillonnages aléatoire et latin ne sont pas déterministes, et introduisent donc une variabilité des configurations lors de l'échantillonnage, pouvant être conséquente pour l'approche aléatoire. Nous verrons aussi que le krigeage possède plusieurs avantages sur les splines.

Pour conclure sur cette partie, **les quatre métamodèles construits offrent des résultats satisfaisants**, et permettent ainsi de prédire de manière quasi-instantanée une valeur d'efficience pour un traité XS donné.

4.1.4 Réduction du nombre de points de départ

Ici, nous allons illustrer la robustesse du krigeage par rapport au splines, en réduisant le nombre de points utilisés pour la construction des métamodèles. Nous choisirons un nombre de points de départ égal à $n_c = 16 = 4^2$. A partir de ces points, nous construirons quatre métamodèles, trois modèles de krigeage via les méthodes d'échantillonnage utilisées précédemment et un modèle de splines bicubiques basé sur un échantillonnage factoriel.

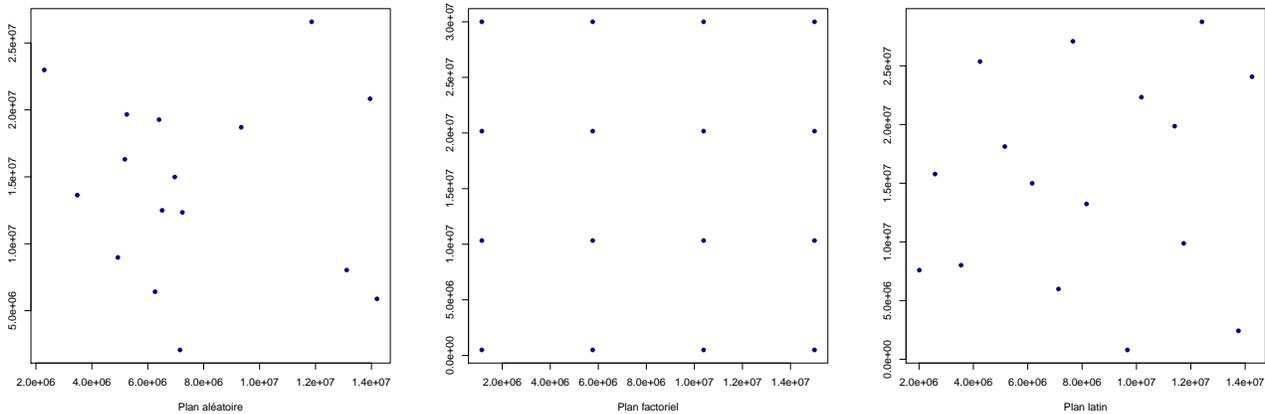


FIGURE 49 – Plans d'expérience avec 16 points, selon la méthode d'échantillonnage

Les modèles seront construits suivant la même méthodologie que dans la section précédente.

- **Modèles de krigeage :** voici les résultats de prédiction obtenus sur la base de validation pour les trois modèles de krigeage :

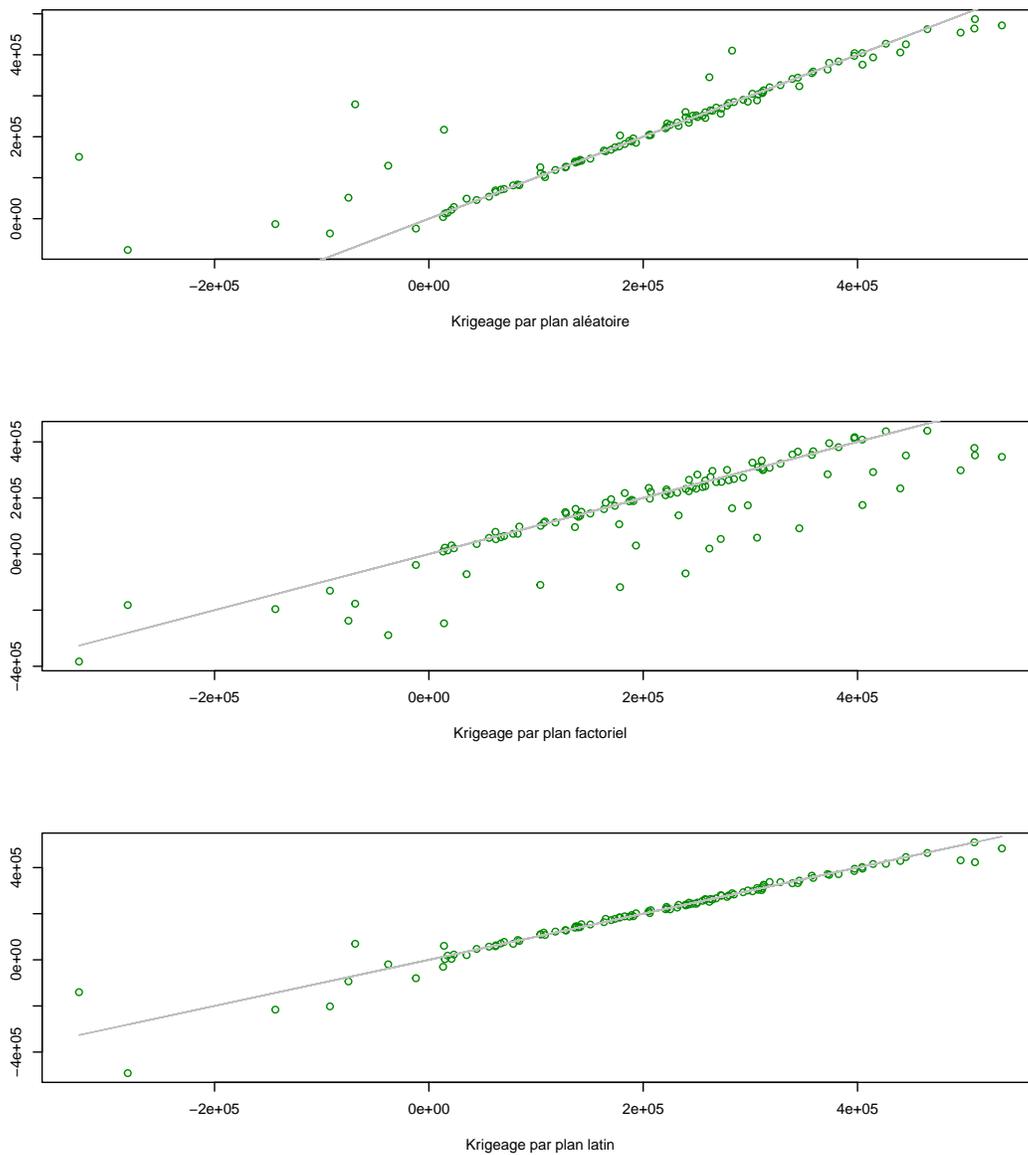


FIGURE 50 – Prédications d’efficacité par krigage (ordonnée) vs Valeurs réelles (abscisse), selon le plan d’expérience

Graphiquement, le modèle construit via l’échantillonnage par hypercube latin semble le plus performant.

— Splines bicubiques :

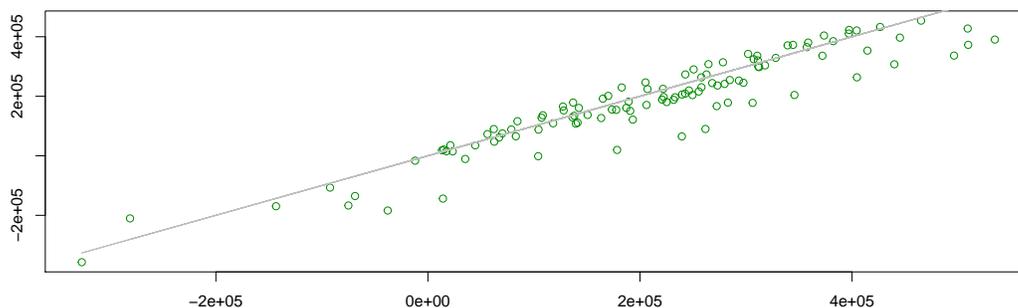


FIGURE 51 – Prédications d’efficience par splines (ordonnée) vs Valeurs réelles (abscisse)

Graphiquement, nous pouvons constater une dégradation de la qualité de prédiction comparé au modèle par splines basé sur échantillon de 25 points.

Le tableau suivant résume les résultats pour les quatre modèles obtenus avec 16 points de départ :

Métamodèles construits - 16 points				
Interpolation	Krigage			Splines bicubiques
Echantillonnage	Aleatoire	Latin	Factoriel	Factoriel
EQMP	4 858 264 163	1 300 673 745	9 015 740 796	3 670 319 585
EQMP ^{1/2}	69 701	36 065	94 951	60 583
Q ²	0,802	0,947	0,633	0,851

FIGURE 52 – $EQMP$ et Q^2 selon le type de métamodèle

Le krigage par échantillonnage latin donne clairement les meilleurs résultats au sens du coefficient Q^2 . La méthode d’échantillonnage par hypercube latin a ainsi permis de conserver de très bons résultats malgré la réduction du nombre de points, indiquant une certaine stabilité de ce type de configuration.

Par rapport aux modèles à 25 points de départ, l’ $EQMP^{1/2}$ a été multipliée par 4 pour le krigage par échantillonnage aléatoire. Cela met en exergue un défaut inhérent à l’échantillonnage aléatoire, qui est son instabilité lorsque le nombre de points générés est insuffisant par rapport à l’espace considéré. Ici, la configuration spatiale des points obtenue par ce type d’échantillonnage a complètement délaissé des zones de l’espace, dégradant ainsi fortement la qualité de prédiction

du modèle de krigeage associé. Dans ce cas, l'échantillonnage par hypercube latin, qui impose des contraintes sur le positionnement des points et la complétion de l'espace, est à privilégier.

Pour l'échantillonnage factoriel, l' $EQMP^{1/2}$ a été multipliée par 1,4 pour le krigeage, alors qu'elle a été multipliée par 1,9 pour les splines bicubiques. Même si les résultats restent meilleurs pour les splines, cela permet de montrer la forte sensibilité des splines à la réduction du nombre de points.

De ces observations, nous pouvons conclure que le krigeage par hypercube latin se montre comme la méthode la plus robuste en termes de qualité de prédiction.

4.1.5 Avantages du krigeage

Outre le fait que la modélisation par krigeage s'avère plus robuste quant au nombre points de départ par rapport aux splines, elle possède également d'autres avantages :

- Méthodes d'échantillonnage : si les splines sont assez rigides quant aux méthodes d'échantillonnage possibles pour les points de départ, nous avons vu que le krigeage est beaucoup plus souple sur ce sujet, et permet donc d'utiliser facilement plusieurs méthodes d'échantillonnage, pouvant mener à une meilleure qualité de prédiction.
- Modèle non-paramétrique : la spécification d'un modèle de krigeage, basée sur les processus gaussiens, est relativement simple et identique quelle que soit la dimension de l'espace des variables en entrée. La modélisation par splines, quant à elle, implique l'estimation d'un nombre conséquent de paramètres, les coefficients des polynômes utilisés, croissant avec la dimension de l'espace. La forte dépendance de la structure de cette modélisation paramétrique à la dimension de l'espace est donc un désavantage des splines par rapport au krigeage, basé sur une modélisation non-paramétrique.
- Construction séquentielle d'un métamodèle : la méthode dispose de plusieurs critères prédéfinis pour la sélection séquentielle, là où l'approche par splines bicubiques nécessiterait la définition « à la main » d'un critère, tâche pouvant s'avérer délicate.

4.2 Construction séquentielle par krigeage & Optimisation

Dans cette partie, nous allons nous placer dans une situation où l'objectif sera d'identifier le ou les meilleurs traités au sens de l'efficience, c'est à dire ceux la maximisant. Pour ce faire, nous allons construire un métamodèle de manière séquentielle, en prenant en compte ce critère d'optimisation pour la sélection des nouveaux points. Le métamodèle en question sera construit par krigeage. Cependant, avant de construire ce modèle de krigeage pour l'optimisation de traités, nous effectuerons une première modélisation par krigeage de manière séquentielle dans une optique d'amélioration globale du métamodèle, en utilisant différents critères de sélection. Cela nous permettra de comparer, au global, la qualité de prédiction entre l'approche séquentielle et l'approche par plan d'expérience fixe.

Dans une approche séquentielle de construction d'un métamodèle, l'avantage du krigeage est que la modélisation stochastique sous-jacente via un processus gaussien permet de définir facilement plusieurs critères de sélection des nouveaux points. En effet, dans une optique d'amélioration globale du métamodèle, avoir à disposition une mesure d'incertitude sur la prédiction à travers la variance permet de choisir de nouveaux points en visant à réduire cette incertitude sur les prédictions effectuées par le modèle de krigeage. Avant de passer à la définition du critère d'optimisation séquentielle, nous allons rapidement présenter différents critères usuels destinés à l'amélioration globale d'un modèle de krigeage en approche séquentielle.

4.2.1 Critères usuels pour la construction séquentielle par krigeage

En krigeage, les deux critères ci-dessous sont souvent utilisés pour la construction séquentielle. Les notations sont celles de la partie 1.3.1.

- Le nouveau point choisi est celui pour lequel la variance de krigeage est maximale :

$$x_{n+1} = \arg \max_{\tilde{x} \in D} \theta_{\mathbf{x}_n}^2(\tilde{x}).$$

C'est donc un critère de réduction ponctuelle de la variance.

- Le nouveau point choisi est celui qui permet de réduire au maximum la variance de krigeage sur l'ensemble de l'espace, c'est à dire au sens d'une intégrale :

$$x_{n+1} = \arg \max_{\tilde{x} \in D} \int_{x \in D} (\theta_{\mathbf{x}_n}^2(x) - \theta_{(\mathbf{x}_n, \tilde{x})}^2(x)) dx, \text{ avec } (\mathbf{x}_n, \tilde{x}) = (x_1, \dots, x_n, \tilde{x})^t.$$

Ce critère peut en fait se formuler plus simplement par :

$$x_{n+1} = \arg \min_{\tilde{x} \in D} \int_{x \in D} \theta_{(\mathbf{x}_n, \tilde{x})}^2(x) dx.$$

Il est appelé *imse* (pour *integral mean squared error*) en anglais. Son utilisation est plus coûteuse en temps de calcul que celle du critère de réduction ponctuelle de la variance, en raison du calcul intégral nécessaire. Il peut cependant s'avérer plus performant.

Il est clair que ces deux critères visent à améliorer globalement la qualité de prédiction du modèle de krigeage, aucune zone de l'espace n'est privilégiée a priori. Afin d'améliorer la précision du modèle dans les zones où l'efficacité est élevée, il faut définir un autre critère de sélection des points.

4.2.2 Critère pour l'optimisation des traités

Afin de cibler le choix des points dans une optique d'optimisation de l'efficacité, nous utiliserons le critère dit d'amélioration espérée ou « expected improvement » en anglais, défini comme suit pour la maximisation :

$$\forall \tilde{x} \in D, EI(\tilde{x}) = E[(Z(\tilde{x}) - \max \mathbf{Z}_n)_+ | \{\mathbf{Z}_n = \mathbf{z}_n\}], \text{ où } \max \mathbf{Z}_n = \max(Z(x_1), \dots, Z(x_n)).$$

Cette expression peut se développer sous la forme :

$$EI(\tilde{x}) = (m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}) - \max \mathbf{z}_n) \Phi \left(\frac{m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}) - \max \mathbf{z}_n}{\theta_{\mathbf{x}_n}(\tilde{x})} \right) + \theta_{\mathbf{x}_n}(\tilde{x}) \phi \left(\frac{m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}) - \max \mathbf{z}_n}{\theta_{\mathbf{x}_n}(\tilde{x})} \right),$$

avec Φ fonction de répartition et ϕ densité de la loi $\mathcal{N}(0, 1)$. Cela nous permet donc de calculer facilement $EI(\tilde{x})$.

Le nouveau point est alors choisi de façon à maximiser l'amélioration espérée :

$$x_{n+1} = \arg \max_{\tilde{x} \in D} EI(\tilde{x})$$

Le package `R DiceOptim` contient une fonction `max_EI` déterminant via un algorithme génétique le point $\tilde{x} \in D$ pour lequel l'amélioration espérée EI est maximale.

4.2.3 Nombre de points pour la construction séquentielle

Dans la partie 3.3, nous avons fixé un budget en temps de calcul correspondant à un nombre de points total égal à 25 pour la construction d'un métamodèle. Ici, nous devons donc couper en deux ce nombre n_c de points : un certain nombre n_1 de ces points sera dédié à la constitution d'un échantillon de départ (i.e. un plan d'expérience fixe) qui servira à la construction du premier modèle de krigeage, et le nombre de points disponibles restant n_2 servira pour la construction séquentielle des modèles de krigeage suivants, à travers la sélection d'un nouveau point dans l'espace D à chaque itération.

Nous définissons en première approche une répartition des points de la manière suivante : $n_1 = \frac{2}{5}n_c = 10$ et $n_2 = \frac{3}{5}n_c = 15$. Pour commencer, les n_1 points initiaux seront générés via un échantillonnage par hypercube latin, cette méthode ayant engendré les meilleurs résultats selon l'arbitrage « qualité de prédiction & robustesse pour un nombre réduit de points ». Il restera alors n_2 points à choisir de manière séquentielle, selon un critère d'optimisation de l'efficacité ou d'amélioration globale du modèle.

4.2.4 Schéma du processus de construction séquentielle

Une fois le premier modèle de krigeage construit à partir des 10 points initiaux et le nouveau point sélectionné, le schéma de fonctionnement de la construction séquentielle du modèle de krigeage est le suivant :

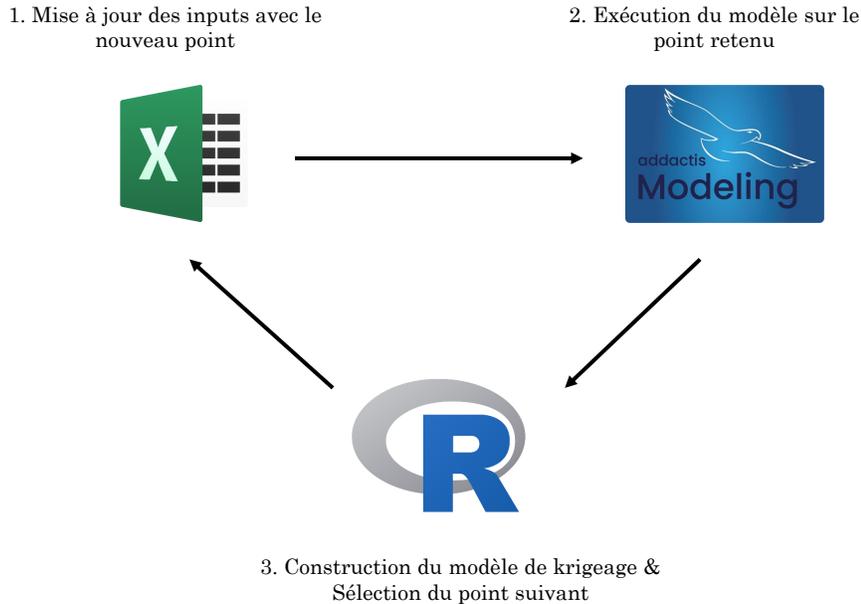


FIGURE 53 – Schéma de fonctionnement de la construction séquentielle

La boucle algorithmique est implémentée sous VBA (Visual Basic Application), dans le fichier Excel d’initialisation du modèle de tarification en réassurance. Schématiquement, la macro va écrire les caractéristiques du nouveau point directement dans le fichier Excel d’initialisation, lance l’exécution du modèle sous ADDACTIS Modeling[®], récupère le résultat d’efficacité obtenu pour le traité en question dans un fichier au format .csv, puis lance un script R : ce script permet d’une part de mettre à jour le modèle de krigeage à partir du modèle de krigeage précédent et du .csv nouvellement créé, et d’autre part de sélectionner le point suivant, qu’il exporte dans un fichier Excel. C’est donc sur ce dernier fichier que la macro se base pour écrire les caractéristiques du nouveau point dans le fichier d’initialisation du modèle.

4.2.5 Construction séquentielle du modèle de krigeage par réduction ponctuelle de la variance

Dans cette section, nous allons construire un modèle de krigeage de manière séquentielle en essayant d’améliorer globalement sa qualité de prédiction. Nous utiliserons le critère visant à sélectionner le point de l’espace pour lequel la variance du modèle est maximale, commode pour l’implémentation et la visualisation des résultats : $x_{n+1} = \arg \max_{\tilde{x} \in D} \theta_{\mathbf{x}_n}^2(\tilde{x})$.

Pour commencer, un premier modèle de krigeage est construit par plan d’expérience fixe selon la méthodologie habituelle, en utilisant $n_1 = 10$ points générés via un échantillonnage par hypercube latin. La surface d’interpolation a la forme suivante :

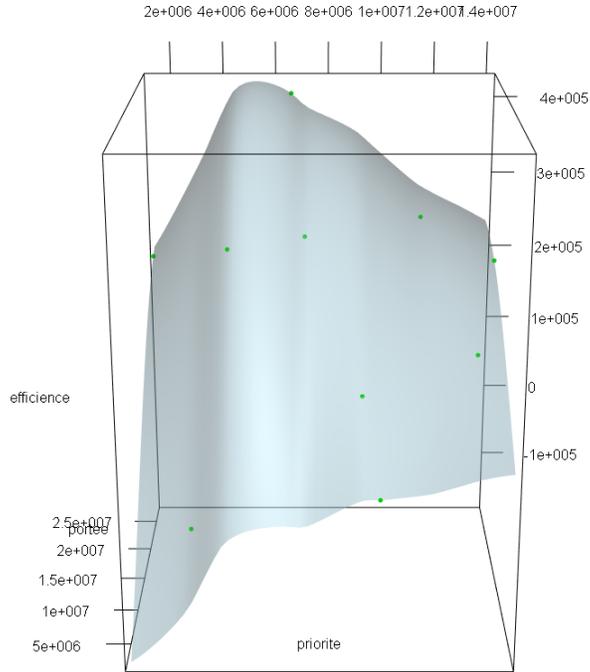


FIGURE 54 – Premier modèle de krigeage avec 10 points générés par hypercube latin

Il faut à présent déterminer le point suivant à ajouter au plan d'expérience pour construire le prochain modèle de krigeage. L'objectif est donc de maximiser $\theta_{\mathbf{x}_n}^2(\tilde{x})$ pour $\tilde{x} \in D$, avec $\theta_{\mathbf{x}_n}^2(\tilde{x}) = \sigma^2 - \mathbf{k}(\mathbf{x}_n, \tilde{x})^t \mathbf{K}_n^{-1} \mathbf{k}(\mathbf{x}_n, \tilde{x})$, comme spécifié dans la partie 1.3. Pour résoudre ce problème de maximisation, nous utilisons sous R un algorithme génétique. Le point sélectionné par l'algorithme est le traité avec les paramètres *priorité* = 15 000 000 € et *portée* = 500 000 €. Ce choix de point, dans un angle du plan d'expérience, paraît cohérent puisque le point en question est éloigné des 10 points initiaux, il est donc normal que la variance de la prédiction résultat du modèle de krigeage soit élevée en ce point, la corrélation avec les points alentours diminuant du fait de la distance.

En ajoutant ce point au plan d'expérience, un nouveau modèle de krigeage peut être construit, impliquant une ré-estimation des paramètres relatifs au modèle. A partir de ce deuxième modèle, un nouveau point est sélectionné de manière séquentielle, puis un troisième modèle est construit, et ainsi de suite jusqu'à ce que 15 points aient été choisis par cette méthode. Le modèle final de krigeage est donc construit sur la base de 25 points, 10 ayant été générés via un hypercube latin, 15 ayant été choisis de façon séquentielle.

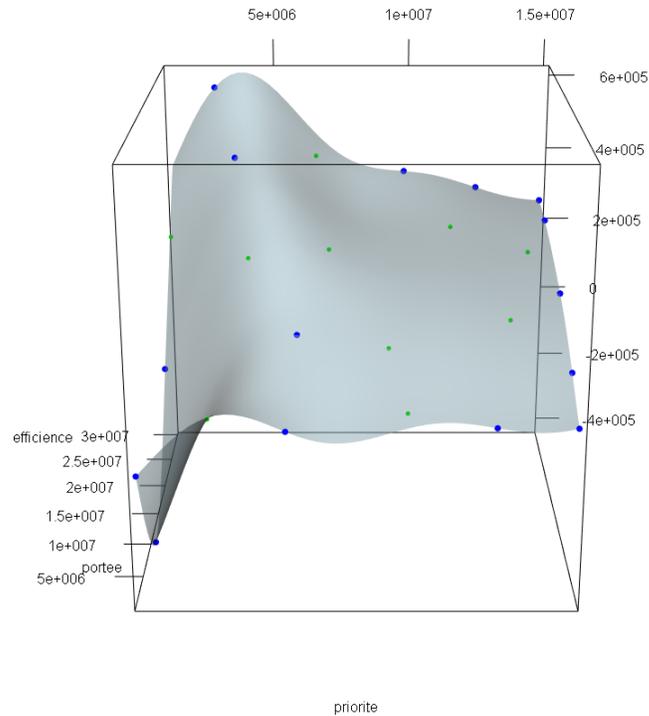


FIGURE 55 – Interpolation par krigeage basée sur les 10 points initiaux (en vert) et les 15 points séquentiels (en bleu), par réduction ponctuelle de la variance

Ce modèle de krigeage a un noyau de covariance de type exponentiel-puissance. Nous pouvons alors mesurer la qualité de prédiction de ce métamodèle sur la base de validation, comme précédemment.

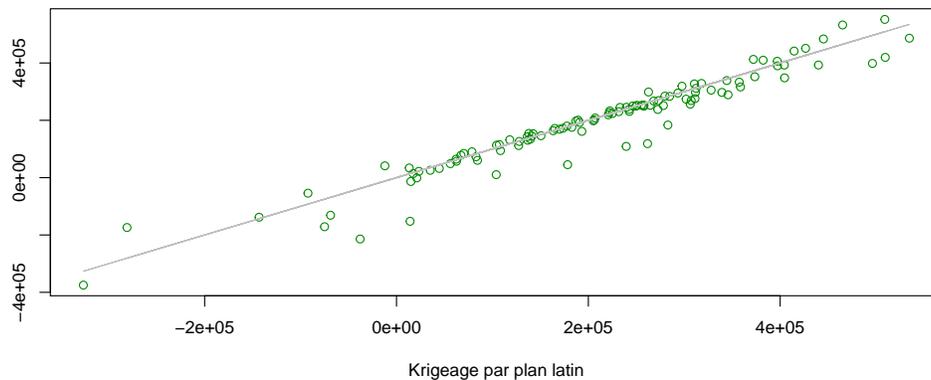


FIGURE 56 – Prédications d’efficacité par krigage (ordonnée) vs Valeurs réelles (abscisse)

Pour ce métamodèle, nous obtenons un coefficient Q^2 égal à 0,916. La qualité de prédiction s’avère inférieure à celle du métamodèle équivalent construit par plan d’expérience fixe avec 25 points affichant un Q^2 égal à 0,981 (voir figure 48), et inférieure à celle du métamodèle similaire avec 16 points. Ce résultat semble contraire à l’intuition, puisque le critère de sélection séquentielle est défini de sorte à choisir « au mieux » les points de construction, i.e. dans les zones où la prédiction est incertaine. En observant de plus près la figure 55, nous pouvons voir que presque tous les points choisis séquentiellement sont situés sur les bords de l’hypercube. Ces choix semblent cohérents au sens du critère retenu, car plus un point est éloigné des autres, plus l’incertitude autour de la prédiction est grande (cette propriété découle de la forme des noyaux de covariance utilisés, pour lesquels la covariance entre deux points décroît avec la distance). Un grand nombre de points séquentiels sont donc consommés sur les bornes de l’espace de définition des variables, et à l’opposé, peu de points sont placés dans le « coeur » de l’hypercube. De ce fait, dans cette dernière zone, les prédictions du métamodèle seront moins fiables, générant des erreurs plus importantes lors de la prédiction sur la base de validation. La meilleure qualité de prédiction du métamodèle sur les bords de l’hypercube ne permettant pas de contrebalancer ce phénomène, la qualité globale de prédiction s’en trouve dégradée. **Le critère de sélection séquentielle est en quelque sorte « mis en défaut » par la configuration initiale des dix points échantillonnés par hypercube latin.**

Afin de contourner cette difficulté, nous allons utiliser un autre critère de sélection, le critère *imse*, consistant à réduire la variance du modèle au sens d’une intégrale.

4.2.6 Construction séquentielle du modèle de krigage par réduction intégrale de la variance

L’utilisation du critère *imse*, prenant en compte la variance sur l’ensemble de l’espace lors de la sélection séquentielle, ne devrait pas nécessairement privilégier les points situés sur les bords de

l'hypercube. Pour rappel, la sélection d'un nouveau point par ce critère se fait de la façon suivante :

$$x_{n+1} = \arg \min_{\tilde{x} \in D} \int_{x \in D} \theta_{(\mathbf{x}_n, \tilde{x})}^2(x) dx, \text{ avec } (\mathbf{x}_n, \tilde{x}) = (x_1, \dots, x_n, \tilde{x})^t.$$

En reprenant les étapes précédentes, un nouveau modèle de krigeage est donc construit de manière séquentielle avec comme base le même plan d'expérience de 10 points qu'en 4.2.5, construit par hypercube latin. Seul le critère de sélection séquentielle des points diffère. La surface d'interpolation du modèle final de krigeage a la forme suivante :

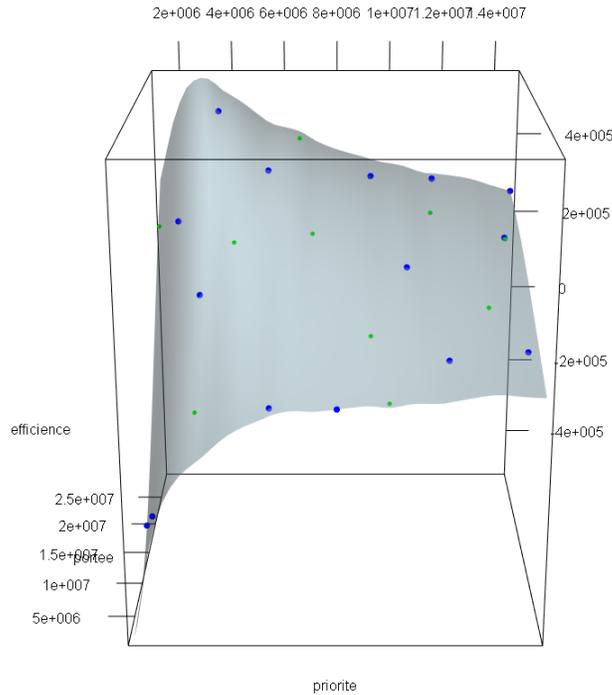


FIGURE 57 – Interpolation par krigeage basée sur les 10 points initiaux (en vert) et les 15 points séquentiels (en bleu), par réduction intégrale de la variance

Même si quelques points sont encore consommés sur les bords de l'hypercube, l'utilisation du critère *imse* nous a permis d'obtenir plusieurs points séquentiels moins éloignés du centre comparé au métamodèle précédent, et d'avoir une meilleure répartition des points dans l'espace. Ce modèle de krigeage a aussi un noyau de covariance de type exponentiel-puissance. Comme précédemment, nous mesurons la qualité de prédiction de ce métamodèle sur la base de validation.

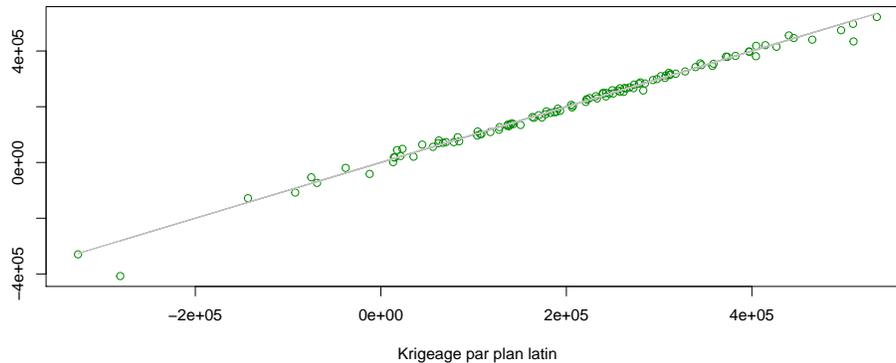


FIGURE 58 – Prédications d’efficacité par krigage (ordonnée) vs Valeurs réelles (abscisse)

Pour ce métamodèle, nous obtenons $Q^2 = 0,988$, c’est donc **le plus précis parmi l’ensemble des métamodèles construits**, au sens de cette mesure. Il offre ainsi une meilleure qualité de prédiction que le modèle de krigage construit avec le critère de réduction ponctuelle de la variance, pour lequel nous avons un Q^2 égal à 0,916.

Afin de mieux visualiser la construction séquentielle de ce dernier métamodèle, voici les différents Q^2 obtenus après ajout de chaque point séquentiel en utilisant l’*imse*, comparés avec les Q^2 de quatre modèles de krigage aléatoire, à nombre de points égal. (Le plan initial est constitué de dix points, le premier point sélectionné séquentiellement est donc le onzième point. Les points ajoutés au fur et à mesure aux modèles de krigage aléatoire sont échantillonnés aléatoirement, ce qui revient donc bien à effectuer une construction par plan fixe aléatoire, avec un point supplémentaire à chaque fois.)

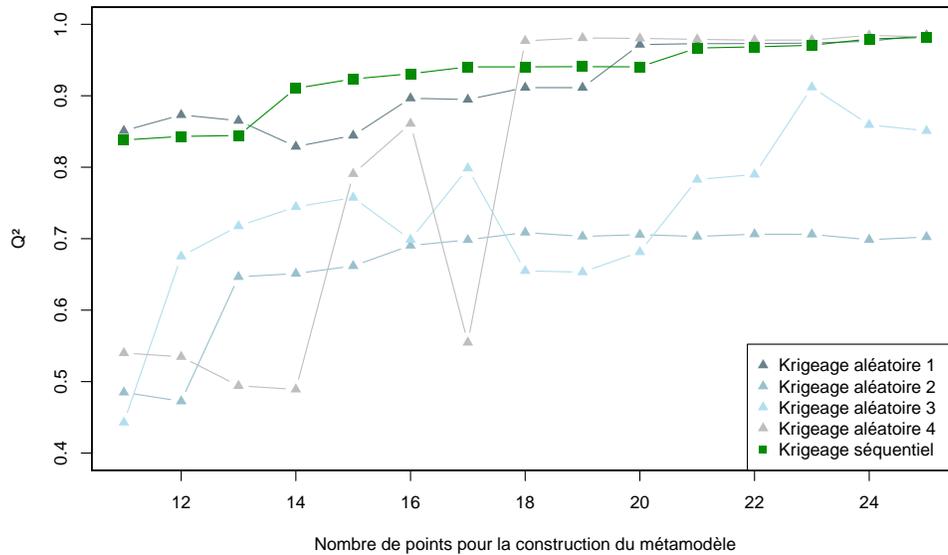


FIGURE 59 – Q^2 selon le modèle de krigeage et le nombre de points n_c

Ce graphique permet d’observer la stabilité des performances pour la construction séquentielle par critère *imse* en fonction du nombre de points. Il permet aussi d’illustrer, à travers le quatrième modèle de krigeage notamment, l’instabilité que peut présenter le krigeage par échantillonnage aléatoire selon les points générés. **Le krigeage séquentiel permet donc d’avoir une méthode de construction beaucoup plus fiable.**

Les modèles de krigeages construits par hypercube latin dans la partie 4.1 s’étant avérés les plus performants au sens de l’arbitrage « qualité de prédiction & robustesse », nous décidons de comparer notre dernier modèle de krigeage séquentiel par *imse* à cinq modèles de krigeage construits via cinq hypercubes latins de 25 points chacun.

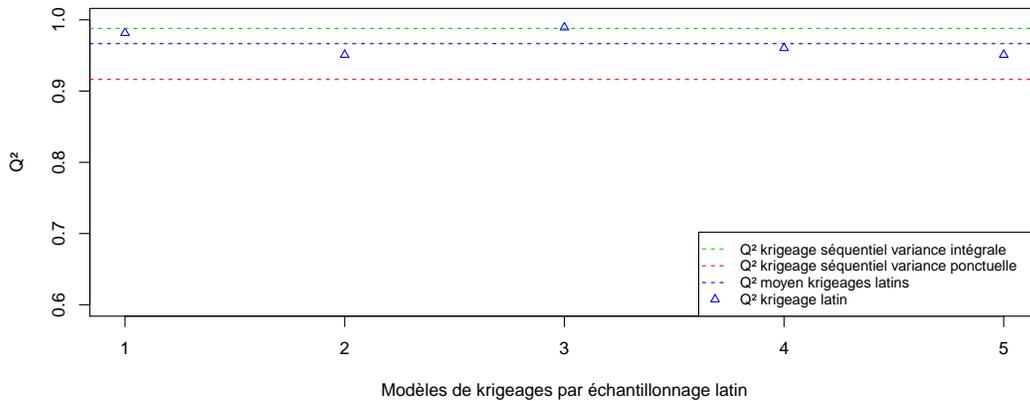


FIGURE 60 – Q^2 selon le type de krigeage

Nous observons qu'en moyenne, le krigeage par hypercube latin est moins performant que le krigeage séquentiel par *imse* (réduction intégrale de la variance). Il est revanche plus performant que le krigeage séquentiel par réduction ponctuelle de la variance.

En effectuant la même comparaison avec cinq modèles de krigeages construits par échantillonnage aléatoire, nous obtenons le graphique suivant :

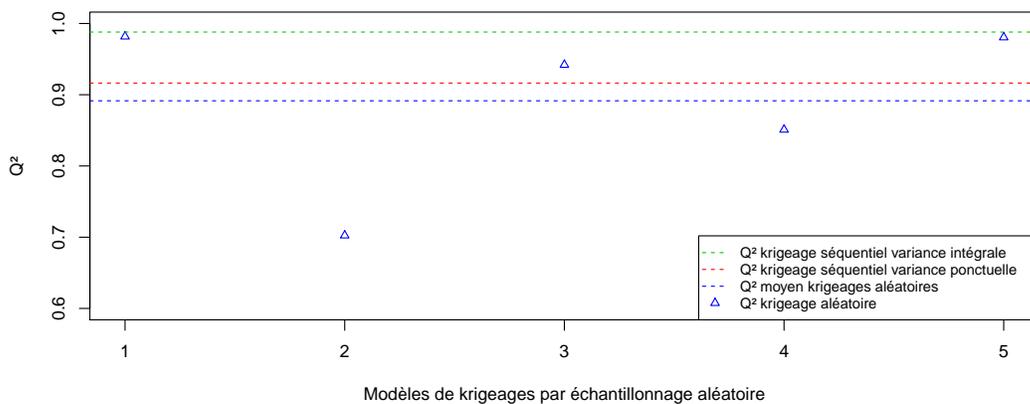


FIGURE 61 – Q^2 selon le type de krigeage

Ici, nous pouvons voir qu'en moyenne, le krigeage par échantillonnage aléatoire est moins performant que les deux types de krigeage séquentiel. De plus, le krigeage séquentiel permet d'éliminer l'aléa sur la qualité de prédiction par rapport au krigeage aléatoire.

Après avoir construit plusieurs métamodèles visant à offrir la meilleure qualité de prédiction possible de manière globale, sur l'ensemble de l'espace, nous allons nous pencher sur la construction séquentielle d'un modèle de krigeage dédié à l'optimisation de l'efficience.

4.2.7 Construction séquentielle du modèle de krigeage pour l'optimisation

Dans cette section, l'idée est de construire un métamodèle visant à être le plus fiable possible dans les zones où l'efficience des traités est maximale. Dans cette optique, nous allons, selon le même schéma que dans la section précédente, construire de manière séquentielle un modèle de krigeage. La seule différence se situera au niveau du critère utilisé pour la sélection des nouveaux points : ici, nous nous servirons du critère d'amélioration espérée ou « expected improvement » (EI).

Une fois le modèle de krigeage construit, nous chercherons alors l'optimum prédit par ce métamodèle pour l'efficience, puis nous évaluerons le point i.e. le traité obtenu avec le vrai modèle de tarification pour obtenir la valeur exacte d'efficience et mesurer l'erreur de prédiction. Enfin, nous comparerons l'optimum trouvé de manière séquentielle avec les optima obtenus via plusieurs modèle construits par plan d'expérience fixe, afin d'évaluer l'apport de la construction séquentielle dans ce problème d'optimisation.

Il faut pour commencer définir une méthode d'échantillonnage pour générer le plan fixe de départ. Dans les deux sections précédentes, nous avons utilisé un échantillonnage par hypercube latin pour construire le plan de départ. Cependant, nous avons pu observer en 4.2.5 la mise en défaut du critère de sélection par réduction ponctuelle de la variance, poussé à choisir de nombreux points sur les bords de l'hypercube. Ce fait est notamment lié à la configuration par hypercube latin, qui ne génère pas de points sur les bords de l'hypercube et engendre une variance élevée dans ces zones.

Comme le critère de réduction ponctuelle de la variance, le critère d'amélioration espérée défini par $x_{n+1} = \arg \max_{\tilde{x} \in D} EI(\tilde{x})$ avec

$$EI(\tilde{x}) = (m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}) - \max \mathbf{z}_n) \Phi \left(\frac{m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}) - \max \mathbf{z}_n}{\theta_{\mathbf{x}_n}(\tilde{x})} \right) + \theta_{\mathbf{x}_n}(\tilde{x}) \phi \left(\frac{m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}) - \max \mathbf{z}_n}{\theta_{\mathbf{x}_n}(\tilde{x})} \right)$$

fait directement intervenir la variance du modèle au point \tilde{x} considéré. Nous pouvons constater qu'à $m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x})$ fixé, $EI(\tilde{x})$ croît avec $\theta_{\mathbf{x}_n}(\tilde{x})$:

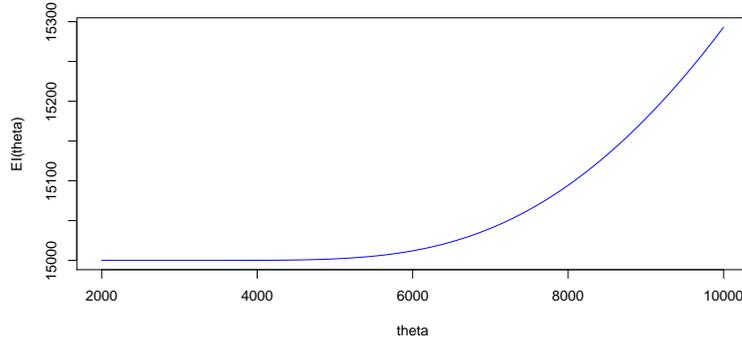


FIGURE 62 – Evolution de $EI(\tilde{x})$ selon $\theta_{\mathbf{x}_n}(\tilde{x})$, avec $m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}) - \max \mathbf{z}_n = c = 15\,000$

Ainsi, pour éviter que des points ne soient choisis par ce critère sur les bords de l’hypercube uniquement par un effet d’augmentation de la variance, il paraît judicieux de réduire a priori la variance du modèle de krigeage aux bords de l’hypercube, en positionnant dès le départ des points dans ces zones. De plus, nous sommes ici dans un cadre de recherche d’optimum. Il nous serait donc utile d’avoir a priori une connaissance du comportement du modèle sur les valeurs extrêmes de l’espace de définition, notamment pour faciliter la recherche si l’optimum du modèle se trouve dans une de ces zones.

Ces deux arguments nous conduisent à préférer l’utilisation d’un échantillonnage factoriel pour la constitution du plan fixe de départ, positionnant de fait des points sur les bords de l’hypercube. Ce type d’échantillonnage avait produit les résultats les moins concluants dans un cadre de maximisation globale de la qualité de prédiction ; cependant, ici, la démarche n’est plus la même puisqu’il s’agit de construire un métamodèle dédié à la recherche d’optima d’efficience. Le plan d’expérience fixe de départ sera ainsi constitué de $9 = 3^2$ points générés par échantillonnage factoriel. 16 points seront donc sélectionnés séquentiellement pour arriver à $n_c = 9 + 16 = 25$.

Un premier modèle de krigeage est donc construit à partir du plan factoriel de 9 points. Il possède la surface d’interpolation suivante :

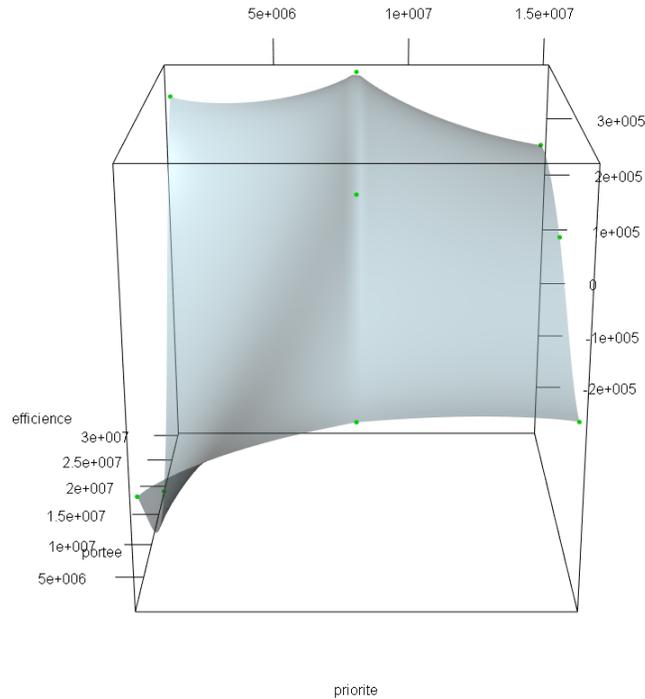


FIGURE 63 – Modèle de krigeage initial pour la recherche de traités optimaux

Pour sélectionner le nouveau point, nous devons donc ici maximiser $EI(\tilde{x}) = E[(Z(\tilde{x}) - \max \mathbf{Z}_n)_+ | \{\mathbf{Z}_n = \mathbf{z}_n\}]$ pour $\tilde{x} \in D$. Cette maximisation est effectuée par algorithme génétique. Le premier point sélectionné est le traité avec les paramètres *priorité* = 5 604 684 € et *portée* = 30 000 000 €. Il se situe en effet dans une zone élevée d'efficience selon le modèle de prédiction, comme le montre la figure 63. Un nouveau modèle de krigeage est ensuite construit, et ainsi de suite jusqu'à obtenir le métamodèle final, conçu à partir d'un nombre total de $n_c = n_1 + n_2 = 9 + 16 = 25$ points. La surface d'interpolation de ce métamodèle final a la forme suivante :

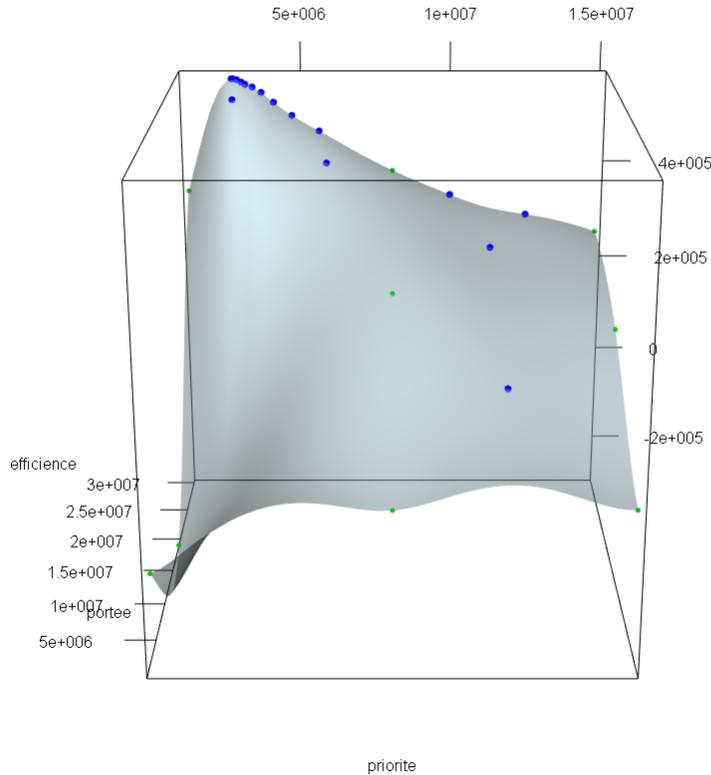


FIGURE 64 – Modèle de krigage final pour la recherche de traités optimaux

Sur la figure 64, nous pouvons observer une concentration de points retenus par l’algorithme de sélection séquentielle dans une zone d’efficience élevée du modèle. Ce résultat est conforme à l’intuition.

Le métamodèle final étant construit, nous allons déterminer le point qui maximise l’efficience sur la surface d’interpolation créée. L’algorithme de maximisation utilisé est toujours un algorithme génétique. Nous obtenons le point correspondant au traité de caractéristiques $priorité = 2\,729\,380\text{ €}$ et $portée = 30\,000\,000\text{ €}$, pour laquelle l’efficience prédite vaut $E = 570\,254\text{ €}$. Ensuite, nous déterminons grâce au modèle de tarification en réassurance la vraie valeur d’efficience pour ce point : elle vaut $E = 570\,162\text{ €}$. **L’erreur relative de prédiction est de 0,016%. La prédiction pour l’optimum est donc de très bonne qualité.** Il est à noter que ce modèle de krigage conserve une qualité de prédiction globale convenable : sur la base de validation, le coefficient Q^2 obtenu vaut 0,817.

Pour terminer, nous allons comparer ce résultat avec ceux qui seraient obtenus pour différents modèles de krigage construits en *one-shot design*, c’est-à-dire par plan fixe, via l’échantillonnage

aléatoire et l'échantillonnage par hypercube latin. Pour ce faire, nous utilisons les dix modèles de krigeage construits en 4.2.6 (voir figures 60 et 61). Comme pour la construction séquentielle, nous déterminons les optima d'efficacité prédits par ces dix modèles.

Les résultats obtenus pour les cinq modèles de krigeage aléatoire et les cinq modèles de krigeage latin concernant la prédiction d'optima sont résumés dans le tableau ci-dessous. Les résultats du krigeage séquentiel sont également repris :

Traité optimal selon le métamodèle - 25 points					
Traité optimal sur la surface d'interpolation					
Métamodèle	Priorité	Portée	Valeur d'efficacité prédite	Vraie valeur d'efficacité	Ecart relatif
Krigeage Latin 1	3 110 165	30 000 000	523 940	561 888	-6,8%
Krigeage Latin 2	3 522 879	30 000 000	542 437	548 739	-1,1%
Krigeage Latin 3	2 612 625	30 000 000	639 759	570 031	12,2%
Krigeage Latin 4	2 904 946	30 000 000	660 281	566 638	16,5%
Krigeage Latin 5	3 740 773	30 000 000	567 851	539 954	5,2%
Krigeage Aleatoire 1	2 815 556	30 000 000	678 416	569 233	19,2%
Krigeage Aleatoire 2	1 150 000	30 000 000	583 440	341 130	71,0%
Krigeage Aleatoire 3	3 217 918	30 000 000	552 565	557 673	-0,9%
Krigeage Aleatoire 4	7 270 073	30 000 000	522 536	406 342	28,6%
Krigeage Aleatoire 5	3 019 132	30 000 000	580 245	563 547	3,0%
Krigeage Séquentiel	2 729 380	30 000 000	570 254	570 162	0,016%

FIGURE 65 – Prédiction des traités optimaux selon le modèle de krigeage

Le deuxième modèle de krigeage aléatoire affiche les plus mauvais résultats. Les résultats obtenus pour les modèles de krigeage latin sont moins volatiles que ceux des modèles de krigeage aléatoire. Par rapport aux résultats obtenus avec le modèle de krigeage séquentiel, deux choses sont remarquables :

- Quelque soit le modèle de krigeage aléatoire ou latin, la vraie valeur d'efficacité obtenue pour le traité optimisé est inférieure à la vraie valeur d'efficacité obtenue pour le traité optimisé par krigeage séquentiel. Le krigeage séquentiel s'avère donc plus précis pour rechercher le véritable optimum du modèle.
- Quelque soit le modèle de krigeage aléatoire ou latin, l'écart relatif entre la valeur prédite et la vraie valeur d'efficacité est largement supérieur à l'écart obtenu dans le cas du krigeage séquentiel.

Ces observations nous permettent de conclure que **la construction séquentielle d'un modèle de krigeage se montre plus performante que la construction par plan d'expérience fixe pour la recherche et la prédiction d'un optimum d'efficacité de traité.**

4.3 Ajout des variables *AAD* et *AAL* dans la conception du métamodèle

Jusqu'à présent, les métamodèles ont été construits sur la base de deux variables caractéristiques des traités, la priorité et la portée. Avoir un espace de dimension deux en entrée des métamodèles s'avère très commode pour la visualisation et l'interprétation des résultats, facilement observables sous forme de surface. Cela permet aussi d'avoir une première validation des méthodes employées, tout en vérifiant si les résultats obtenus sont conformes à l'intuition. Cependant, ne considérer que deux variables en entrée limite l'analyse du comportement du modèle, et ne permet pas d'exploiter pleinement le potentiel des méthodes d'interpolation utilisées, notamment concernant le krigeage. Il peut donc être intéressant d'inclure de nouvelles variables, afin d'approfondir l'étude du comportement de l'efficacité des traités.

Nous allons ajouter l'*AAD* et l'*AAL* aux variables d'entrée du métamodèle. Nous ferons varier la première dans l'intervalle $C = [0, 30\,000\,000]$ et la deuxième dans l'intervalle $D = [30\,000\,000, 120\,000\,000]$. Ces espaces sont définis tels que $\max(C) = \min(D) = \max(B)$ et $\max(D) = 4 \times \max(B)$. L'objectif sera ici de déterminer un nouvel optimum d'efficacité via une construction séquentielle d'un modèle de krigeage, comme dans la partie précédente. Le métamodèle aura donc quatre variables en entrée. Nous allons augmenter en conséquence le nombre de points $n_c = n_1 + n_2$ nécessaire à sa construction, quitte à dépasser légèrement la contrainte de budget en temps de calculs initialement fixée. Nous retiendrons $n_c = 40$, respectant l'heuristique $n_c = 10 \times d = 10 \times 4$. Il reste à définir n_1 et n_2 . Ici, pour les n_1 points initiaux, utiliser un hypercube généré par échantillonnage factoriel impliquerait un nombre de points égal au minimum à $n_1 = 2^d = 2^4 = 64$, beaucoup trop élevé. Nous choisirons donc comme configuration initiale un hypercube latin, composé de $n_1 = \frac{2}{5}n_c = 16$ points, laissant ainsi $n_2 = \frac{3}{5}n_c = 24$ points à choisir séquentiellement.

4.3.1 Modèle de krigeage séquentiel pour l'optimisation en dimension quatre

Comme dans les parties précédentes, un premier modèle de krigeage est construit via l'hypercube initial de 16 points, puis ce modèle est affiné au fur et à mesure des itérations de l'algorithme de sélection séquentiel. Le modèle final est construit avec 40 points. Le noyau de covariance est de type matern $5/2$.

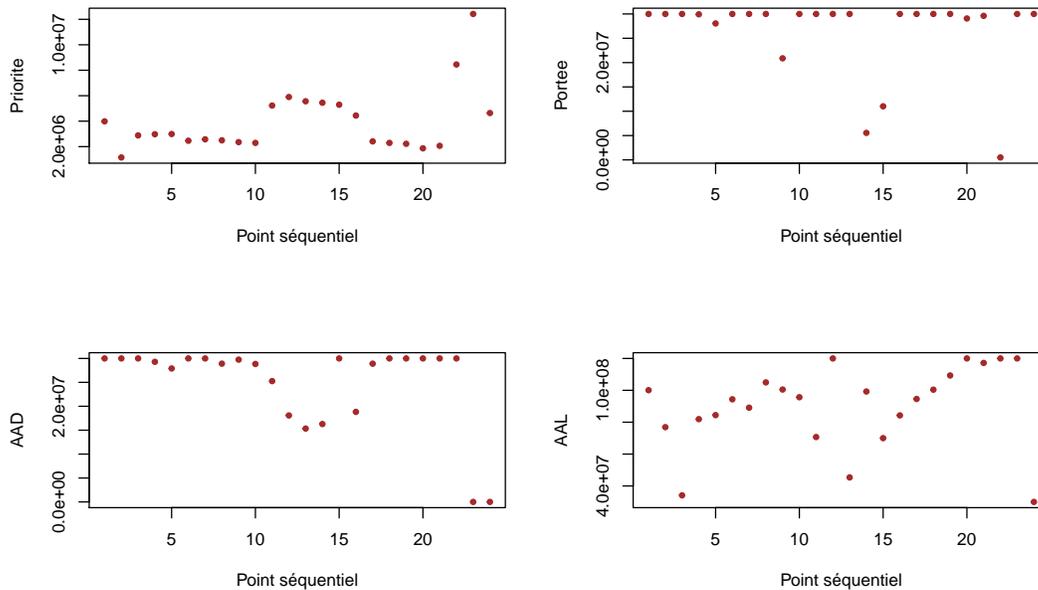


FIGURE 66 – Sélection séquentielle des 24 points par maximisation du critère d’amélioration espérée (l’axe des abscisses correspond au rang de chaque point séquentiel dans le processus de sélection)

Les quatre variables présentent différentes zones de stabilité par rapport aux points sélectionnés. Néanmoins, il y a une certaine stabilité globale des variables entre les points 5 et 8, et entre les points 19 et 21. Nous pouvons donc nous attendre à trouver l’optimum d’efficience dans une de ces deux zones.

Nous déterminons alors le maximum d’efficience prédit par le modèle de krigeage sur l’espace de définition des variables. Nous obtenons le point correspondant au traité de caractéristiques $(\text{priorité}, \text{portée}, \text{AAD}, \text{AAL}) = (2\,019\,370, 30\,000\,000, 30\,000\,000, 120\,000\,000)$, pour laquelle l’efficience prédite vaut $E = 1\,247\,501$. Comparé à l’optimum obtenu dans la section 4.2.7, le traité de caractéristiques $(\text{priorité}, \text{portée}) = (2\,729\,380, 30\,000\,000)$ et d’efficience prédite $E = 570\,254$, ici, la priorité a diminué, et l’efficience prédite du traité a augmenté. L’ajout de deux degrés de liberté supplémentaires au métamodèle aurait donc permis de trouver un meilleur traité de réassurance au sens de l’efficience. Cela reste à confirmer, en évaluant l’efficience du traité obtenu avec le modèle de tarification en réassurance. En exécutant ce dernier sur le traité considéré, nous obtenons une efficience égale à 1 253 476, ce qui valide le résultat obtenu. **L’écart relatif entre l’efficience prédite et la vraie valeur d’efficience vaut 0,48%, il est supérieur à l’écart relatif obtenu avec seulement deux variables, mais reste faible.** L’ajout de variables supplémentaires a donc dégradé la qualité de prédiction, qui reste cependant très satisfaisante.

La structure du modèle est telle que les optima d’efficience ont tendance à se trouver aux bornes de l’espace des traités, pour les variables *portée*, *AAD* et *AAL*. Ce fait est inhérent à la définition

du critère d'efficience $E = -(1 - \rho)R + \alpha G$ et à la distribution sous-jacente de sinistralité. Les quatre variables agissent toutes (de différentes manières) comme des bornes sur la distribution de sinistralité initiale, les impacts sont donc sur l'efficience sont donc observables : augmenter la portée d'un traité fera par exemple augmenter le gain en capital $G = q_{99,5\%}(S^{bruts}) - q_{99,5\%}(S^{nets})$ en faisant baisser $q_{99,5\%}(S^{nets})$, et augmenter le résultat

$$\begin{aligned} R &= P^{commerciale} - S^{cédés} \\ &= \frac{S^{cédés} + \text{taux}_{capital} \times \mu(S^{cédés})}{1 - \text{taux}_{frais}} - S^{cédés} \\ &= \frac{\text{taux}_{frais}}{1 - \text{taux}_{frais}} S^{cédés} + \frac{\text{taux}_{capital}}{1 - \text{taux}_{frais}} \mu(S^{cédés}) \end{aligned}$$

en faisant augmenter $S^{cédés}$. Ces deux effets agissent donc de manière opposée dans le calcul de l'efficience. Selon la distribution de la sinistralité et les différents paramètres (ρ , α , autres caractéristiques du traité), l'un de ces deux effets finira par l'emporter au-delà d'une certaine valeur de portée, dans notre cas au bénéfice du gain G , expliquant la valeur maximale d'efficience obtenue pour une portée égale à $30\,000\,000 = \max(B)$. La structure relativement « simple » du modèle de tarification est donc à l'origine de ces résultats optimaux d'efficience obtenus sur les bornes des espaces de définition.

D'un point de vue opérationnel, ces résultats sont cependant à exploiter au regard des contraintes qui peuvent être fixées par la cédante, en termes de budget ou de niveau de cession des risques par exemple. Si les résultats d'optimisation d'efficience en dimension quatre, sur le cas particulier de notre modèle de tarification en réassurance, ont pu s'avérer quelque peu décevants quant à leur interprétation, la méthodologie employée pour les obtenir, i.e. la construction séquentielle d'un modèle de krigeage par maximisation de l'amélioration espérée, a quant à elle été éprouvée et a montré sa qualité ainsi que sa fiabilité pour construire un métamodèle dédié à l'optimisation.

Le graphique suivant présente une coupe en trois dimensions du modèle de krigeage obtenu en fonction de la *priorité* et de l'*AAL*, la *portée* et l'*AAD* étant fixées à $30\,000\,000\text{€}$:

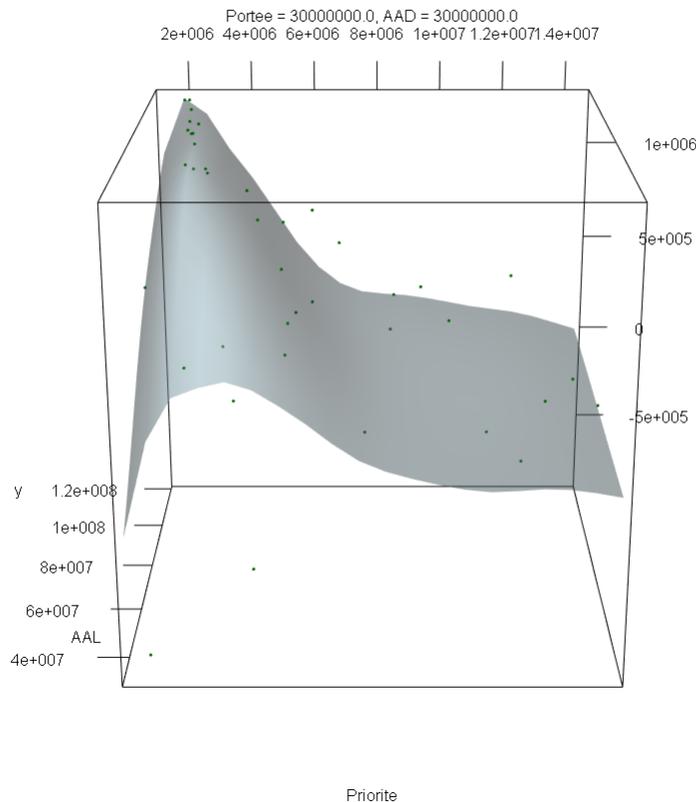


FIGURE 67 – Sélection séquentielle des 24 points par maximisation du critère d’amélioration espérée

Les 40 points de l’espace $A \times B \times C \times D$ ayant servi à la construction du modèle de krigeage ont été projetés sur l’espace $A \times D$, et sont affichés sur le graphique avec les valeurs d’efficacité associées, en vert. La surface grise représente les prédictions du métamodèle selon les valeurs des variables d’entrée restreintes à l’espace $A \times \{30\,000\,000\} \times \{30\,000\,000\} \times D$.

4.4 Conclusion sur les résultats obtenus

Dans cette section, nous allons récapituler les résultats obtenus dans la partie 4.

Dans un premier temps, nous avons construit, à partir de 25 points de départ, trois modèles de krigeage basés sur des méthodes d’échantillonnage aléatoire, factoriel et latin. Leur qualité de prédiction a ensuite été évaluée sur une base de validation composée de 110 points, en utilisant le coefficient Q^2 . L’échantillonnage aléatoire a donné les meilleurs résultats, suivi de très près par l’échantillonnage latin. L’échantillonnage factoriel a engendré un modèle un peu moins précis. En

outre, les résultats sont satisfaisants pour les trois métamodèles. Nous avons également construit un modèle d'interpolation par splines bicubiques à partir de l'échantillon factoriel. Ce métamodèle a offert des résultats satisfaisants, légèrement inférieurs à ceux des modèles de krigeage construits par échantillonnages aléatoire et latin.

Dans un deuxième temps, nous avons réduit le nombre de points de départ à 16 et reconstruit les quatre métamodèles à partir des nouveaux plans d'expérience obtenus. Cette fois, le modèle de krigeage basé sur un hypercube latin s'est avéré nettement plus performant que les autres modèles de krigeage et les splines bicubiques. La dégradation des performances du krigeage par échantillonnage aléatoire nous a notamment permis d'illustrer l'impact de la volatilité de ce type d'échantillonnage sur les résultats du métamodèle. Nous avons également pu observer une dégradation conséquente de la qualité de prédiction des splines. Cette deuxième étape nous a ainsi permis de mettre en lumière la robustesse du krigeage par rapport aux splines lors de la réduction du nombre de points de départ. D'autres avantages du krigeage, relatifs à la pluralité des méthodes d'échantillonnage applicables, à la spécification non-paramétrique de la modélisation ou encore à la disponibilité de critères de sélection pour la construction séquentielle, nous ont conduits à privilégier le krigeage aux splines pour la suite de notre étude.

Dans un troisième temps, nous avons mis en avant l'avantage de la construction séquentielle d'un métamodèle, en utilisant un critère de réduction intégrale de la variance d'un modèle de krigeage. Cela nous a permis d'obtenir un métamodèle plus précis pour la prédiction que l'ensemble des métamodèles construits jusqu'alors. Nous avons aussi observé certaines limites inhérentes à un autre critère de sélection, basé sur la réduction ponctuelle de la variance. Ensuite, nous avons exploité un deuxième type de critère de construction séquentielle, destiné à optimiser la qualité de prédiction dans les zones de l'espace où l'efficacité des traités est élevée. Le métamodèle construit via cette dernière méthode nous a permis d'approximer, sur notre espace de définition, l'optimum d'efficacité du modèle de tarification en réassurance. Nous avons pu illustrer la supériorité de cette approche pour la recherche et la prédiction d'optima sur l'approche de construction par plan fixe d'un métamodèle.

Enfin, une dernière phase a été consacrée à l'inclusion de deux nouvelles variables dans l'espace d'entrée du métamodèle, l'*AAD* et l'*AAL*, afin d'obtenir une analyse plus fine du comportement de l'efficacité de réassurance. Nous avons ainsi repris la démarche appliquée dans la partie précédente, relative à la construction séquentielle d'un modèle de krigeage dédié à l'optimisation. Les résultats obtenus se sont avérés satisfaisants concernant la qualité de prédiction du métamodèle, mais légèrement décevants quant à l'interprétation des valeurs numériques obtenues, l'optimum d'efficacité étant situé sur les bornes des espaces de définition de trois des quatre variables d'entrée. Ce résultat est lié à la structure relativement simple du modèle étudié, et notamment aux effets assez stables des variables d'entrée du métamodèle sur l'efficacité de réassurance. Cependant, cette dernière étape nous a permis de valider la méthodologie employée, soit la construction séquentielle d'un modèle de krigeage pour l'optimisation, qui, indépendamment de l'interprétation des résultats numériques obtenus, a produit d'excellents résultats de prédiction, d'écart relatif inférieur à 0,5%

par rapport à la valeur réelle, même en dimension quatre.

4.5 Cas d'usage simplifié pour la construction d'un métamodèle d'optimisation

La démarche mise en oeuvre et les résultats obtenus dans les parties précédentes, nous permettent, à titre d'exemple, de définir une méthodologie qui pourrait être employée pour une cédante cherchant à déterminer le traité XS le plus efficient sur une branche donnée de son portefeuille. Nous ne reprenons pas les étapes concernant le retraitement et l'exploitation des données renseignées en entrée du modèle, ni les étapes relatives à la sélection des variables d'entrée et la détermination d'un nombre de simulations adéquat.

1. **Définition d'un budget en temps de calcul** : dans un premier temps, il faut définir une durée maximale à accorder à la production des résultats d'efficience qui serviront de base à l'interpolation. Une fois cette durée T définie, connaissant le temps d'exécution du modèle en fonction du nombre de traités en entrée, il est possible de définir le nombre de traités pour lesquels calculer une valeur d'efficience via le modèle en un temps T . Une fois ce nombre n déterminé, il n'est pas nécessaire de réserver un certain nombre de ces traités pour constituer une base de validation, car pour cette problématique d'optimisation, la base de validation ne sera donc composée que d'un seul traité, le traité optimal prédit par le métamodèle.
2. **Choix du plan d'expérience de départ** : il faut néanmoins scinder en deux ce nombre $n = n_1 + n_2$. En pratique, n_1 traités viendront constituer le plan d'expérience de départ, et n_2 traités s'ajouteront au fur et à mesure de la construction séquentielle du métamodèle. Ce choix effectué, il est nécessaire, pour terminer, de définir une méthode d'échantillonnage dédiée à la génération des traités du plan d'expérience de départ.
3. **Construction séquentielle du modèle de krigeage** : les traités initiaux ayant été générés et les valeurs d'efficience associées calculées par le modèle de tarification, la construction du métamodèle peut commencer. Un premier modèle de krigeage est construit à partir des valeurs d'efficience obtenues sur le plan d'expérience de départ. Ensuite, un nouveau traité est sélectionné par le critère EI d'amélioration espérée, le modèle de krigeage est ajusté avec la valeur d'efficience calculée par le modèle de tarification sur ce nouveau traité, et ainsi de suite jusqu'à atteindre n_2 traités sélectionnés séquentiellement. A partir des n traités et n valeurs d'efficience qui composent le plan d'expérience complet, le modèle final de krigeage est construit.
4. **Prédiction de l'optimum d'efficience de réassurance** : via le métamodèle construit, le traité présentant la valeur d'efficience prédite maximale est recherché sur la surface d'interpolation. Une fois trouvé, le modèle initial peut être exécuté sur ce traité pour obtenir la valeur réelle d'efficience, et calculer l'erreur de prédiction.

5 Limites et perspectives liées à la démarche et la modélisation utilisées

Si les résultats obtenus dans la partie 4 se sont avérés plutôt satisfaisants, certaines limites peuvent être identifiées dans la démarche mise en oeuvre et la modélisation utilisée. Nous les détaillons ici, en proposant des pistes d'améliorations ou de nouveaux axes d'études.

- **Echantillonnage dans un hypercube** : une limite inhérente à la construction de méta-modèles basée sur des méthodes d'échantillonnage telles que l'échantillonnage par hypercube latin ou plan factoriel, est la nécessité d'avoir des espaces bornés pour les domaines de définition des variables. En effet, ces méthodes d'échantillonnage s'appliquent uniquement sur des hypercubes, qui sont des ensembles fermés. Cette contrainte ne permet donc pas d'étudier le comportement du modèle initial sur l'ensemble de son domaine de définition théorique.
- **Sensibilité de la construction séquentielle au critère de sélection utilisé** : comme nous avons pu le voir dans la section 4.2.5, la construction séquentielle d'un métamodèle, basée sur un critère de sélection, peut se montrer très sensible au plan d'expérience initial. Dans notre cas, un critère de réduction ponctuelle de la variance d'un modèle de krigeage a d'abord été choisi pour construire le métamodèle, en utilisant comme plan fixe de départ un hypercube latin. Cependant, nous avons pu observer que les points déterminés par l'algorithme de sélection séquentielle venaient se positionner sur les bords de l'hypercube. Si nous avons pu contourner cet inconvénient en se servant ensuite d'un critère de sélection plus adapté, cela montre qu'il faut toujours se poser la question de l'adéquation du critère de sélection au modèle étudié et à la méthode d'échantillonnage retenue.
- **Sensibilité des résultats à l'architecture du modèle** : selon le modèle initial étudié, les résultats d'un métamodèle pour une méthode de construction donnée peuvent s'avérer variables. En effet, en fonction du comportement de la variable de sortie du modèle, par exemple très linéaire ou au contraire présentant de fortes variations, les résultats pourront être très différents. Dans le deuxième cas, il faudra très certainement un plan d'expérience avec plus de points que dans le premier cas pour obtenir des résultats équivalents. D'autre part, la méthode d'échantillonnage utilisée peut avoir un fort impact sur les résultats selon le modèle. Ces deux sujets posent ainsi la question du choix de la méthode d'échantillonnage et du nombre de points de départ, pouvant faire l'objet d'une étude préalable.
- **Construction de la base de validation** : Afin de tester la qualité de prédiction des méta-modèles construits, nous avons dû construire une base de validation. Il a donc fallu affecter un nombre conséquent de points (traités de réassurance) à cette base, pour ensuite évaluer leur efficacité via le modèle de tarification. Dans l'idéal, la base de validation doit contenir des traités situés dans toutes les zones de l'espace afin de ne pas biaiser l'évaluation des

métamodèles. Il faut donc avoir conscience que la répartition dans l'espace des traités composant cette base aura un impact sur la mesure de la qualité de prédiction des métamodèles. De plus, la constitution de cette base de validation implique de consommer un grand nombre de points pour cette dernière, ce qui en laisse beaucoup moins pour la constitution du plan d'expérience, conformément à la contrainte de budget temporel. Il y a donc un compromis à faire sur l'affectation du nombre de points à la base de validation et au plan d'expérience, qui doit être effectué en fonction des attentes de l'utilisateur et de la connaissance du modèle.

- **Choix des critères de sélection pour la construction séquentielle** : dans ce mémoire, trois critères de sélection associés au krigeage ont été utilisés dans l'application pratique, deux visant à améliorer globalement le métamodèle en cherchant à réduire sa variance d'estimation, le troisième se destinant à la recherche d'optima. Ces critères, prédéfinis grâce aux propriétés du krigeage, ne sont bien évidemment pas exhaustifs ; il serait tout à fait possible de définir d'autres critères de sélection. Par exemple, un critère spécifique à une problématique ou à un besoin précis de l'utilisateur pourrait être défini, en intégrant des contraintes sur-mesure. Inversement, un critère très générique, non-spécifique au krigeage et applicable à tout type de métamodèle pourrait être mis en place.
- **Intervalle de confiance sur les résultats du modèle de tarification** : la partie du modèle de tarification en réassurance étudiée dans ce mémoire, soit le module de tarification par simulations des traités en excédent de sinistre, évolue dans un cadre stochastique. Nous avons donc défini un nombre de simulations suffisant pour avoir une convergence convenable des résultats du modèle vers la valeur théorique, modélisée par le biais d'un intervalle de confiance. Cela permet de s'approcher d'un modèle déterministe, et de pouvoir ainsi réaliser une interpolation spatiale. Néanmoins, d'un point de vue opérationnel, il faut garder à l'esprit que les résultats comportent toujours une petite part d'aléa.
- **Critère d'arrêt pour la construction séquentielle** : notre démarche a consisté en la définition d'un budget en temps de calcul, à partir duquel nous avons souhaité obtenir le meilleur métamodèle possible. Nous avons donc en fonction du budget déterminé un nombre de points maximal pour constituer le plan d'expérience, base de la construction des métamodèles. Ce nombre de points fixé dès le départ ne peut donc garantir une précision donnée des métamodèles construits. Lorsque la contrainte temporelle est moins forte, la construction séquentielle peut avoir comme critère d'arrêt une précision souhaitée du métamodèle sous-jacent, au lieu d'un nombre de points maximal. Dans notre cas, le nombre de points sélectionnés séquentiellement était suffisant pour obtenir des résultats de qualité. Dans d'autres situations, une approche utilisant un critère de précision pourrait être intéressante à exploiter.
- **Définition du critère d'efficience de réassurance** : comme nous avons pu le voir dans la section 4.3.1, la définition telle quelle du critère d'efficience a engendré des résultats numériques peu exploitables concernant le métamodèle d'optimisation à quatre variables. Une

redéfinition de ce critère, avec par exemple la prise en compte de contraintes supplémentaires, pourrait constituer une piste d'amélioration.

Conclusion

Le principal objectif de ce mémoire était de répondre à une problématique d'optimisation temporelle d'un modèle et d'exploration de l'espace des variables associées à ce modèle. Cette motivation a fait suite à la conception d'un modèle de tarification en réassurance, présentant une composante stochastique à travers la tarification par simulations. C'est l'aspect coûteux en temps de calcul de cette méthode de tarification qui a décidé le choix de ce sujet. Ces travaux ont également permis, sur un aspect « métier », de présenter et de mettre en pratique une démarche de tarification en réassurance.

Concevoir un métamodèle de tarification en réassurance, via l'application de techniques d'interpolation, a fait naître de nouvelles problématiques, inhérentes à l'architecture du modèle utilisé. Il a notamment fallu fixer un budget en temps de calcul pour cadrer la construction du métamodèle, choisir la configuration des points de départ dans l'espace de définition des variables, ou encore déterminer le nombre adéquat de simulations permettant d'obtenir une convergence du modèle suffisante pour la mise en oeuvre de l'interpolation. Ainsi, l'application des différentes méthodes présentées doit toujours prendre en compte les spécificités du modèle sous-jacent étudié. Dans notre cas, nous nous sommes intéressés à l'efficacité des traités de réassurance XS.

Nous avons ainsi utilisé différentes méthodes d'échantillonnage pour générer les points de départ, et comparé deux méthodes d'interpolation spatiale, le krigeage et les splines bicubiques. Après une première approche par plan d'expérience fixe ayant produit des résultats satisfaisants, nous avons choisi de poursuivre notre étude avec le krigeage, plus performant, souple et robuste, en adoptant une approche de construction séquentielle du plan d'expérience. Deux types de métamodèles ont été construits, le premier dans l'optique d'obtenir la meilleure qualité possible de prédiction sur l'ensemble de l'espace, le deuxième se destinant à la recherche d'optima d'efficacité de réassurance. La construction séquentielle a permis d'améliorer sensiblement la qualité des métamodèles. Pour élargir l'étude, nous avons construit un dernier métamodèle dédié à la recherche d'optima d'efficacité avec deux variables supplémentaires. Cette dernière étape a pu illustrer certaines limites de la définition du critère d'efficacité de réassurance.

Les travaux effectués nous ont donné l'occasion de tester plusieurs approches et techniques destinées à l'approximation des résultats d'un modèle coûteux en temps de calcul, en se basant sur des techniques d'interpolation spatiale. Cela nous a permis d'éprouver et de valider une méthodologie, qui pourra être répliquée dans le cadre de travaux futurs, en l'appliquant à d'autres types de modèles. Cette dernière présente bien sûr des limites, détaillées dans ce mémoire, comme la sensibilité des méthodes au modèle étudié. D'autres aspects des techniques employées, par exemple l'utilisation de critères différents pour la sélection séquentielle, pourraient faire l'objet de nouveaux axes d'étude.

Table des figures

1	Fonctionnement du module de tarification par simulations	5
2	<i>EQMP</i> et Q^2 selon le type de métamodèle	7
3	<i>EQMP</i> et Q^2 selon le type de métamodèle	8
4	<i>EQMP</i> et Q^2 selon la méthode de construction	9
5	Hypercube généré par échantillonnage aléatoire, pour $d = 3$ et $N = 27$	15
6	Projection sur le plan (x_1, x_2)	15
7	Hypercube généré par plan factoriel, pour $d = 3$ et $N = 27$	16
8	Projection sur le plan (x_1, x_2)	17
9	Exemple d'hypercube latin non-optimal, pour $d = 3$ et $N = 5$	18
10	Exemple d'hypercube latin optimal avec maximinLHS, pour $d = 3$ et $N = 5$	18
11	Hypercube latin optimal avec maximinLHS, pour $d = 3$ et $N = 27$	19
12	Projection sur le plan (x_1, x_2)	19
13	Construction séquentielle d'un métamodèle	20
14	Résultats du modèle f sur les dix points initiaux	21
15	Ajout du résultat obtenu sur le nouveau point	21
16	Résultats du modèle g sur les dix points initiaux	22
17	Ajout du résultat obtenu sur le nouveau point	22
18	Résultats après sélection de deux nouveaux points à la fois	23
19	Exemple de spline cubique	24
20	Covariance en fonction de la distance selon le type de fonction ($\alpha = 0, 8$)	29
21	Superposition des courbes de la figure 20	29
22	Fonctionnement de l'algorithme génétique	34
23	Architecture du modèle de tarification en réassurance	35
24	Histogramme des sinistres graves	40
25	Fonctions de répartition empirique et théoriques pour la sévérité des graves	42
26	Grappe Q-Q et graphe P-P pour la sévérité des graves	43
27	Histogramme des sinistres graves et densité de la GPD retenue	44
28	Efficience d'un traité XS selon la portée et la priorité	49
29	$p(a, b)$ en fonction de la priorité a et de la portée b	53
30	Prime pure estimée par le modèle en fonction du nombre de simulations	54
31	Evolution du temps d'exécution du modèle (en s) en fonction du nombre de traités	56
32	Régression linéaire sur les temps d'exécution (en s) en fonction du nombre de traités	56
33	Traités constituant la base de validation (hypercube latin)	59
34	Echantillon aléatoire de couples (<i>priorité, portée</i>) et efficacités associées	59
35	<i>EQMP</i> sur la LOO cross-validation, selon le noyau de covariance	60
36	Interpolation par krigeage à partir d'un échantillon aléatoire	61
37	Prédictions d'efficience par krigeage (en bleu) et valeurs réelles (en rouge)	62
38	Prédictions d'efficience par krigeage (ordonnée) vs Valeurs réelles (abscisse)	62
39	Echantillon factoriel de couples (<i>priorité, portée</i>) et efficacités associées	63

40	Prédictions d'efficience par krigeage (en bleu) et valeurs réelles (en rouge)	64
41	Prédictions d'efficience par krigeage (ordonnée) vs Valeurs réelles (abscisse)	64
42	Echantillon latin de couples (<i>priorité, portée</i>) et résultats d'efficience associés	65
43	Prédictions d'efficience par krigeage (en bleu) et valeurs réelles (en rouge)	66
44	Prédictions d'efficience par krigeage (ordonnée) vs Valeurs réelles (abscisse)	66
45	<i>EQMP</i> et Q^2 selon le modèle de krigeage	67
46	Interpolation par splines bicubiques à partir d'un échantillon factoriel	68
47	Prédictions d'efficience par splines (ordonnée) vs Valeurs réelles (abscisse)	69
48	<i>EQMP</i> et Q^2 selon le type de métamodèle	69
49	Plans d'expérience avec 16 points, selon la méthode d'échantillonnage	70
50	Prédictions d'efficience par krigeage (ordonnée) vs Valeurs réelles (abscisse), selon le plan d'expérience	71
51	Prédictions d'efficience par splines (ordonnée) vs Valeurs réelles (abscisse)	72
52	<i>EQMP</i> et Q^2 selon le type de métamodèle	72
53	Schéma de fonctionnement de la construction séquentielle	76
54	Premier modèle de krigeage avec 10 points générés par hypercube latin	77
55	Interpolation par krigeage basée sur les 10 points initiaux (en vert) et les 15 points séquentiels (en bleu), par réduction ponctuelle de la variance	78
56	Prédictions d'efficience par krigeage (ordonnée) vs Valeurs réelles (abscisse)	79
57	Interpolation par krigeage basée sur les 10 points initiaux (en vert) et les 15 points séquentiels (en bleu), par réduction intégrale de la variance	80
58	Prédictions d'efficience par krigeage (ordonnée) vs Valeurs réelles (abscisse)	81
59	Q^2 selon le modèle de krigeage et le nombre de points n_c	82
60	Q^2 selon le type de krigeage	83
61	Q^2 selon le type de krigeage	83
62	Evolution de $EI(\tilde{x})$ selon $\theta_{\mathbf{x}_n}(\tilde{x})$, avec $m_{\mathbf{x}_n, \mathbf{z}_n}(\tilde{x}) - \max_{\mathbf{z}_n} = c = 15\,000$	85
63	Modèle de krigeage initial pour la recherche de traités optimaux	86
64	Modèle de krigeage final pour la recherche de traités optimaux	87
65	Prédictions des traités optimaux selon le modèle de krigeage	88
66	Sélection séquentielle des 24 points par maximisation du critère d'amélioration espérée (l'axe des abscisses correspond au rang de chaque point séquentiel dans le processus de sélection)	90
67	Sélection séquentielle des 24 points par maximisation du critère d'amélioration espérée	92

Bibliographie

- [1] GDR MASCOT-NUM. <https://www.gdr-mascotnum.fr/committees.html>, 2016-2020.
- [2] Module OTPOD. <http://adumasphi.github.io/otpod/index.html>, 2017.
- [3] Peter Antal. Quantative methods in reinsurance. 2003.
- [4] Howard Anton and Chris Rorres. Elementary linear algebra, ninth edition. *New York : John Wiley and Sons*, 1984.
- [5] François Bachoc, Emile Contal, Hassan Maatouk, and Didier Rullière. Gaussian processes for computer experiments. *ESAIM : Proceedings and Surveys*, 2017.
- [6] Sophie Baillargeon. Le krigeage : revue de la théorie et application à l'interpolation spatiale de données de précipitations. *Faculté des études supérieures de l'Université Laval*, 2005.
- [7] Areski Cousin, Hassan Maatouk, and Didier Rullière. Kriging of financial term-structures. *European Journal of Operational Research*, 2016.
- [8] Karel Crombecq. Surrogate modelling of computer experiments with sequential experimental design. 2011.
- [9] David Ginsbourger. Multiples métamodèles pour l'approximation et l'optimisation de fonctions numériques multivariées. *Ecole Nationale Supérieure des Mines de Saint-Etienne*, 2009.
- [10] Thomas Guillod. Interpolations, courbes de Bézier et B-splines. 2008.
- [11] Akira Horiguchi. Efficient space-filling and non-collapsing sequential design strategies for simulation-based modeling. *The Ohio State University*, 2018.
- [12] Loic Le Gratiet and Claire Cannamela. Kriging-based sequential design strategies using fast cross-validation techniques with extensions to multi-fidelity computer codes. *Université Paris Diderot*, 2018.
- [13] Christian Y. Robert. Théorie des valeurs extrêmes. *ISFA - Université Lyon 1*.
- [14] Joachim van der Herten, Tom Van Steenkiste, Ivo Couckuyt, and Tom Dhaene. Surrogate modelling with sequential design for expensive simulation applications. 2017.