

Rapport de stage

Constance Bau

2024-02-08

1 Introduction

De nos jours, la diversification des usages en mer ne cesse de s'amplifier (éolien en mer, extraction de granulats, conchyliculture, etc.) et l'espace maritime fait plus que jamais l'objet de conflits sociaux, économiques et environnementaux. Dans ce contexte, la gestion spatialisée des activités humaines proposée par la Commission Européenne ([directive établissant un cadre pour la planification spatiale de l'espace maritime, directive cadre stratégie pour le milieu marin, politique commune des pêches](#)) prend tout son sens. La [modélisation devient dès lors un outil](#) incontournable pour aider à la prise de décision en apportant de la connaissance sur les conséquences possibles de réglementations spatiales complémentaires des mesures actuelles. Pour appréhender une gestion écosystémique spatialisée des pêches, nous utilisons le modèle de simulation ISIS-Fish ([isis-fish.org](#)). Ce modèle spatialisé permet de décrire la dynamique spatio-temporelle des flottilles de pêche et son incidence sur les principales espèces capturées. Pour permettre le choix d'une réglementation parmi plusieurs il est indispensable d'évaluer la robustesse des [sorties du modèle](#).

Toutes les incertitudes en entrée du modèle n'influent pas dans les mêmes ordres de grandeur sur les sorties du modèle. [L'analyse de sensibilité est une méthode qui permet de quantifier l'influence des paramètres d'entrée sur les variables de sorties](#). En plus de permettre de cibler les incertitudes importantes à prendre en compte dans une évaluation de la robustesse des sorties du modèle, l'analyse de sensibilité permet de trouver les covariables les plus influentes dans le but de ([Genuer, Poggi, and Tuleau-Malot \(2010\)](#)) : (i) trouver un petit nombre de covariables maximisant la précision ou (ii) classer les covariables par ordre d'influence pour l'interprétation du mécanisme de prédiction du modèle. Les stratégies adoptées varient selon l'objectif suivi. Par exemple, si deux variables influentes sont fortement corrélées, l'une sera écartée dans le cas (i) tandis que les deux seraont gardées dans le cas (ii). Des méthodes d'analyse de sensibilité supposant l'indépendance entre les entrées ont d'abord été développées, puis, comme dans de nombreuses applications il est courant que les variables d'entrée aient une structure de dépendance statistique (connue ou supposée), la recherche s'est alors portée sur des méthodes plus complexes prenant en compte la dépendance entre les entrées.

Dans le cadre de ce stage, nous étudierons la sensibilité d'une pêcherie langoustinière paramétrée avec ISIS-Fish et inspirée de la pêcherie de la grande vasière dans le golfe de Gascogne. Nous étudierons l'influence sur la sortie du modèle de cinq paramètres incertains. Parmi ces paramètres certains sont supposés dépendants, nous prendrons donc cette dépendance en compte. [citer les 5 paramètres dès l'introduction ?](#) Pour faire cela, nous avons d'abord recherché quelle méthode d'analyse de sensibilité nous paraissait la plus appropriée.

Ainsi, après avoir étudié les indices de Sobol ([Da Veiga et al. \(2021\)](#)), une méthode permettant, grâce une décomposition exacte de la variance, de calculer des indices de sensibilité dans le cadre d'entrées indépendantes (voir annexe), nous nous sommes intéressés à des méthodes prenant en compte la dépendance entre les entrées du modèle. Les effets de Shapley ([Da Veiga et al. \(2021\)](#)), initialement formulés pour mesurer la contribution au jeu de chaque joueur, dans le cadre de la théorie des jeux coopératifs, résultent, dans le cas de l'analyse de sensibilité, d'une allocation directe d'une part de la variance de la sortie à chaque entrée (voir annexe). Cette méthode est très intéressante dans le sens où elle prend en compte la corrélation entre les entrées et qu'elle repose sur une formule exacte faisant intervenir les indices de Sobol. Néanmoins, l'estimation des indices de Shapley peut s'avérer compliquée en grande dimension car elle nécessite de considérer toutes les permutations possibles [de](#) variables d'entrée. Par ailleurs, pour remédier au fait que dans

certains cas la variance ne représente pas très fidèlement la variabilité de la distribution de la sortie, des mesures d'importance du moment indépendant ont été introduites. Ces mesures visent à comparer la loi de probabilité de la sortie avec celles des différentes entrées pour voir à quel point celles-ci se ressemblent. Parmi ces mesures, on peut citer l'indice de Shapley-HSIC (Da Veiga (2021)), formulé en utilisant les indices Hsic (voir annexe) qui calculent la dépendance entre deux variables Y et X comme la distance entre leur distribution jointe et le produit de leurs distributions marginales. Nous nous sommes finalment intéressé à une dernière méthode utilisant le random forest, un algorithme d'apprentissage automatique, pour construire un méta-modèle et ensuite estimer les indices de sensibilité de ce méta-modèle.

Les random forests (Breiman (2001)) sont des algorithmes d'apprentissage statistique capables de résoudre des problèmes de régression et de classification. Ils peuvent s'appliquer à des données de grande dimension et aux sorties multivariées. Pour permettre l'interprétation des random forests, l'analyse d'importance des variables est principalement utilisée. Les covariables sont alors classées par ordre décroissant de leur importance dans le processus de prédiction de l'algorithme. L'algorithme Sobol-MDA (Bénard, Da Veiga, and Scornet (2022)), basé sur ranger, une implémentation rapide de random forest, permet la mesure d'importance des covariables dans le cadre dépendant. De plus il a été prouvé que cet algorithme converge vers l'indice de Sobol total (Bénard, Da Veiga, and Scornet (2022)) et des expérimentations sur des fonctions théoriques ont montré que le classement des variables par ordre d'importance obtenu avec cet algorithme est le même que celui obtenu avec les calcul théorique des indices de Sobol totaux. Pour toutes les raisons qui précédent, nous avons donc choisi d'utiliser l'algorithme Sobol-MDA afin d'obtenir un métamodèle ainsi que les mesures d'importance de chacune des covariables, et tout ceci à partir d'un échantillon de simulations de notre modèle.

Tout d'abord, nous expliquerons la construction des algorithmes random forests et de quelle manière Sobol-MDA calcule les indices de sensibilité. Dans un deuxième temps nous présenterons le plan de simulations avec ISIS-Fish permettant d'obtenir un échantillon de données de pêcherie langoustinière à partir duquel nous construirons le méta-modèle et calculerons les indices de sensibilité. Finalement nous présenterons nos résultats et nous en discuterons.

2 Méthode

Les random forests sont des algorithmes qui, après avoir été entraînés sur des données d'entraînement (un échantillon d'entrées associées à leur sorties), sont capables de calculer une valeur de sortie prédictive pour n'importe quelle valeur d'entrée. Les random forests sont constitués d'arbres de régression ou de classification, présentés dans Breiman (2017). Dans notre cas, le random forest sera composé d'arbres de régression car la sortie de notre modèle est continue. Nous nous baserons sur le livre de Breiman (2017) pour expliquer en détails la construction de ces arbres puis nous nous baserons sur les articles de Breiman (2001) et de Scornet (2023) pour comprendre la construction d'une forêt aléatoire et l'estimation de son erreur généralisée. Ensuite nous présenterons ranger (détailé dans Wright and Ziegler (2015)), une implémentation rapide et adaptée aux grandes dimensions de random forest utilisée dans Sobol-MDA pour la construction du métamodèle. Finalement, nous détaillerons la manière dont Sobol-MDA calcule les mesures d'importance de chaque covariables ainsi que les hypothèses et résultats de convergence de cet algorithme.

2.1 Arbres de régression

Les CART (Classification And Regression Tree décrit dans Breiman (2017)) sont les composants élémentaires d'une forêt aléatoire. Nous allons voir dans la suite comment se déroule leur construction. L'algorithme de construction effectue un partitionnement récursif des données, puis estime un modèle très simple dans chaque élément de la partition. Un élément de la partition est appelé feuille de l'arbre. Un exemple de partitions et de l'arbre obtenu est présenté sur la figure 1 (provenant de Scornet (2023)). En résumé, le but est de choisir intelligemment une caractéristiques puis de couper astucieusement les données selon cette caractéristique de sorte que la prévision \bar{y}_i soit la moyenne des observations dans la feuille correspondant

à x_i . On recommence ensuite ce processus sur les sous-arbres obtenus jusqu'à notre critère d'arrêt. Voyons cela en détails.

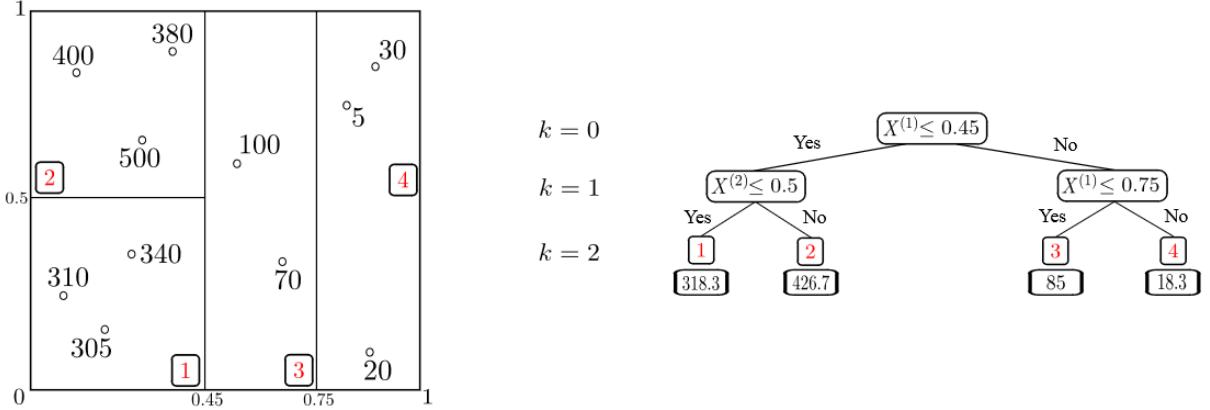


Figure 1: A decision tree of depth $k = 2$ in dimension $d = 2$ (right) and the corresponding partition (left).

On considère qu'on a n individus et p variables. On les note $X_{i,k}$ avec $i \in \{1, \dots, n\}$ et $k \in \{1, \dots, p\}$. L'objectif est de découper l'espace en J régions R_1, \dots, R_J (les feuilles de l'arbre) qui minimisent :

$$RSS = \sum_{j=1}^J \sum_{i \in R_j} (y_i - \hat{y}_j)^2$$

où \hat{y}_j est la moyenne des sorties obtenues avec les entrées situées dans la région R_j c'est à dire $\hat{y}_j = \frac{1}{n_j} \sum_{i \in R_j} y_i$ avec n_j le nombre d'observations dans la feuille R_j . Dans le CART de base on suppose que $\hat{y}_j = a$ où a est une constante mais il existe des variantes de type $\hat{y}_j = x\beta$.

Pour faire cela, à chaque noeud il faut choisir une variable (i.e une composante du vecteur X) selon laquelle on divise les données. On considère les zones $R_-(k, s) = \{x_i \text{ tel que } x_{i,k} < s\}$ et $R_+(k, s) = \{x_i \text{ tel que } x_{i,k} \geq s\}$. A chaque étape, CART choisit la variable j et le seuil s (on a $(k,s) \in C$ l'ensemble des coupes possibles dans les données d'un noeud) minimisant la variance intra-groupe définie comme suit :

$$VI = \frac{1}{n_{noeud}} \sum_{i \text{ tel que } X_i \in R_-(j,s)} (y_i - \hat{y}_{R_-})^2 + \frac{1}{n_{noeud}} \sum_{i \text{ tel que } X_i \in R_+(j,s)} (y_i - \hat{y}_{R_+})^2$$

où n_{noeud} est le nombre de données dans le noeud que l'on cherche à diviser (pour le noeud initial de l'arbre on a donc $n_{noeud} = n$). Minimiser la somme des carré des résidus revient donc à minimiser

$$RSS = \sum_{j=1}^J n_j V_j$$

où $V_j = \frac{1}{n_j} \sum_{i \in R_j} (y_i - \bar{y}_j)^2$ est la variance intra-groupe et n_j le nombre d'observations dans chaque feuille de l'arbre.

A chaque noeud de l'arbre, on recherche quelle est la meilleure division non sur toute les covariables possibles mais sur un échantillon de m covariables tirées aléatoirement et sans remplacement. La meilleure division est sélectionnée uniquement parmi ce m covariables (ce qui présente un gain de temps considérable pour les problèmes de grandes dimensions). Par défaut, on prend $m = \frac{p}{3}$ en régression mais on peut aussi prendre $m = \sqrt{p}$ dans les cas de très grandes dimensions.

Concernant l'arrêt de l'algorithme de construction, deux critères sont principalement utilisés :

- un nombre d'observation q suffisamment petit dans les feuilles
- un critère d'erreur (ici la variance inter-groupe) inférieur à un seuil δ

On peut choisir d'arrêter la récursion lorsque les deux critères sont vérifiés ou seulement lorsque l'un des deux est vérifié.

2.2 Construction d'une forêt aléatoire

Une forêt aléatoire n'est rien d'autre qu'une ensemble d'arbres de décision (dans notre cas arbre de régression). Ainsi, avant la construction de chaque arbre k on tire un échantillon aléatoire bootstrap D_k de taille n dans l'échantillon initial D de taille n lui aussi, c'est-à-dire que l'on tire aléatoirement, uniformément et avec remise n vecteurs x_i dans D. Cet échantillon D_k servira alors de données d'entraînement pour l'arbre k. Les vecteurs x_i qui n'ont pas été tirés (il y en a sûrement grâce au tirage avec remise) sont gardés de côté et constituent l'échantillon Out Of Bag de cet arbre. La prédiction par la forêt aléatoire est alors la moyenne des prédictions de chaque arbre de la forêt.

2.3 Calcul de l'erreur généralisée d'une forêt

Lorsque l'on construit une forêt aléatoire il est essentiel de savoir à quel point les prédictions qu'elle réalise sont précises. Pour cela il est possible d'estimer son erreur généralisée, ce qui revient en pratique à calculer le carré de la différence entre les valeurs de sorties des échantillons OOB et les valeurs de sorties des échantillons OOB prédites par la forêt et de moyenner le tout. Pour faire cela, l'algorithme suivant (trouvé dans Cutler, Cutler, and Stevens (2012) et) peut être utilisé :

Soit D_k le kème échantillon bootstrap et $\hat{h}_j(x)$ la prédiction de x à partir du jème arbre, pour $j = 1, \dots, J$. Pour $i = 1$ à n :

1. Soit $J_i = \{j : (x_i, y_i) \notin D_j\}$ et $|J_i|$ le cardinal de J_i .
2. Définissons la prédiction hors sac à x_i comme suit :
 - $\hat{f}_{oob}(x_i) = \frac{1}{|J_i|} \sum_{j \in J_i} \hat{h}_j(x_i)$ pour la régression

L'erreur de généralisation est généralement estimée en utilisant l'erreur quadratique moyenne (MSE) hors sac suivante :

$$\text{MSE}_{oob} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{f}_{oob}(x_i))^2$$

En résumé, pour chaque vecteur x_i de l'ensemble d'entraînement initial D, on calcule sa prédiction par chacun des arbres k dont il n'a pas fait partie de l'ensemble d'entraînement D_k et on moyenne ces prédictions pour avoir sa prédiction "par la forêt" (même si dans ce cas on ne l'a pas fait passer par tous les arbres). Ensuite, pour obtenir l'estimation généralisée de la forêt on fait la moyenne des différences au carré de la prévision de chaque x_i et de sa valeur de sortie présente dans l'échantillon d'entraînement D.

2.4 Mesure d'importance des variables grâce à l'algorithme Sobol-MDA

Un des problèmes majeurs des random forests réside dans le fait que leurs propriétés mathématiques restent toujours un peu "magiques", ce qui rend leur interprétabilité plus compliquée. Les personnes dérivateuses de se pencher d'avantage sur les forces à l'oeuvre derrière le processus de prédiction se sont alors principalement

tournées vers l'importance des variables, une mesure de l'influence de chaque variable d'entrée dans la prédiction la sortie. Dans l'article initial de Breiman (2001) sur les random forests, deux mesures d'importance sont présentées : la diminution moyenne de l'impureté (MDI, ou Gini importance, voir Breiman (2002)) qui somme les diminutions pondérées de l'impureté sur tous les noeuds qui se divisent selon une covariable donnée, et la diminution moyenne de la précision (MDA, voir Breiman (2001)) qui permuttent les valeurs d'entrée d'une certaine variable dans les données du test et calcule la différence entre l'erreur sur l'ensemble test permuted et l'ensemble test original. L'un des grands avantages du MDI et du MDA réside dans leur capacité à prendre en compte l'interaction entre les covariables mais d'un autre côté ils sont incapables de déterminer la partie de l'effet marginal d'une covariable donnée ((voir Wright, Ziegler, and König (2016)). Par ailleurs, MDA et MDI présentent tous les deux des biais non négligeables dans le cas de variables corrélées. Un version modifiée du MDA, nommée Sobol-MDA et expliquée dans Bénard, Da Veiga, and Scornet (2022), a ainsi été développée pour être capable de donner des résultats pertinents même dans des cas de corrélations entre les variables. Nous avons donc choisi d'utiliser Sobol-MDA pour évaluer la sensibilité de notre modèle.

Après avoir brièvement expliqué le fonctionnement du MDA, nous nous intéressons plus en détails à l'algorithme Sobol-MDA et nous verrons certains résultats de convergence de celui-ci.

2.4.1 Présentation et limites du MDA

Le MDA, mesure de la diminution de la précision en français, a été initialement introduit par Breiman dans son article Breiman (2001). Il repose sur le fonctionnement suivant : les valeurs d'une covariable spécifique sont permutees pour casser sa relation avec la variable de sortie. La précision prédictive est alors calculée pour cet ensemble de données permutes. La différence entre la précision de l'ensemble dégradé et la précision de l'ensemble initial est alors calculée, cette différence donne alors la mesure d'importance de la covariable pour laquelle on a permute les valeurs. Une grande diminution de la précision signifie que la variable considérée a une grande influence dans le mécanisme de prédiction. Bien que cette mesure soit très utilisée en pratique, on ne sait que très peu de choses sur ces propriétés statistiques. Sa convergence vers les indices de Sobol totaux n'a pas pu être prouvée et de nombreuses études empiriques ont montré que, lorsque les covariables sont dépendantes, le MDA ne détecte pas certaines variables influentes. Pour tenter d'y remédier, Williamson et al. (2023) proposent de mesurer la diminution de la précision entre la forêt originale et une forêt entraînée sans l'une des covariables. Néanmoins, comme il faut réentraîner la forêt et calculer sa précision autant de fois qu'il y a de covariables, cette méthode a un coût de calcul très élevé et n'est pas adaptée aux grandes dimensions.

2.4.2 Sobol-MDA

La mesure d'importance Sobol-MDA propose de mimer l'entraînement d'une forêt sans l'une des covariables sans avoir besoin de réentraîner une forêt. Expliquons cela plus clairement.

Le Sobol-MDA (Bénard, Da Veiga, and Scornet (2022)), une amélioration de la procédure MDA, a été introduit pour estimer les indices de Sobol totaux même lorsque les variables sont dépendantes. Les indices de Sobol totaux correspondent à la proportion de la variance expliquée de la réponse perdue lorsqu'une des covariables est retirée du modèle.

2.4.3 Sobol-MDA algorithme

Pour retirer une variable j du processus de prédiction de l'arbre on procède comme suit. La partition de l'espace de covariables obtenu avec les feuilles terminales de l'arbre d'origine est projeté selon la j -ième direction et les sorties des cellules de cette nouvelle partition projetée sont recalculées avec les données d'entraînement. Ce procédé permet de retirer la variable j du processus de prédiction de l'arbre. Ensuite, il est possible de calculer la précision de l'estimation de la forêt projetée grâce aux échantillons OOB, de

soustraire cette précision de la précision initiale et de normaliser la différence obtenue par $V[Y]$ pour obtenir le Sobol-MDA pour X_j .

En pratique, les données d'entraînement et les échantillons OOB sont mis dans l'arbre et envoyés à droite et gauche du noeud si celui-ci fait une division sur la covariable j . A la fin, chaque donnée peut donc se retrouver dans plusieurs feuilles terminales. Pour chaque donnée OOB, la prédiction associée est donc la moyenne des sorties des données d'entraînement qui sont tombées sur les mêmes feuilles terminales que la donnée OOB. En d'autres termes on calcule l'intersection entre les feuilles terminales où est tombée la donnée OOB et les feuilles terminales où sont tombées les données d'entraînement. Cette intersection donne la cellule projetée. Ce mécanisme est équivalent à projeter la partition de l'arbre sur le sous-espace engendré par $X^{(-j)}$.

Ecrivons cela plus formellement.

On note Θ le vecteur qui contient les indices des vecteurs de l'échantillon initial utilisé pour l'entraînement de l'arbre auquel il est associé et on note $m_n(x, \Theta)$ l'estimation de la valeur de la sortie de x par un arbre entraîner avec un échantillon indiqué par Θ , Sobol-MDA prédit donc $m^{(-j)}(X^{(-j)}) = E[m(X)|X^{(-j)}]$.

On note $A_n(X, \Theta)$ la cellule de la partition de l'arbre d'origine où X tombe et on note $A_n^{(-j)}(X^{(-j)}, \Theta)$ la cellule associé de la partition projetée. On note l'estimation par l'arbre projeté associé $m_n^{(-j)}(X^{(-j)}, \Theta)$ et l'estimation par la forêt projeté associée $m_{M,n}^{(-j, OOB)}(X_i^{(-j)}, \Theta_{(M)})$. Ces estimations sont définies de la manière suivante :

$$m_n^{(-j)}(X^{(-j)}, \Theta) = \frac{\sum_{i=1}^{a_n} Y_i 1_{X_i \in A_n^{(-j)}(X^{(-j)}, \Theta)}}{\sum_{i=1}^{a_n} 1_{X_i \in A_n^{(-j)}(X^{(-j)}, \Theta)}}$$

$$m_{M,n}^{(-j, OOB)}(X_i^{(-j)}, \Theta_{(M)}) = \frac{1}{|\Lambda_{n,i}|} \sum_{l \in \Lambda_{n,i}} m_n^{(-j)}(X_i^{(-j)}, \Theta_l) 1_{|\Lambda_{n,i}| > 0}$$

L'indice de Sobol-MDA est donné par la différence normalisée de l'erreur carrée (calculée grâce aux OOB) de la forêt projetée et l'erreur carrée (calculée grâce aux OOB) de la forêt initial, Sobol-MDA est donc défini de la manière suivante :

$$\widehat{S - MDA}_{M,n}(X^{(j)}) = \frac{1}{\hat{\sigma}_Y^2} \frac{1}{n} \sum_{i=1}^n \{Y_i - m_{M,n}^{(-j, OOB)}(X_i^{(-j)}, \Theta_{(M)})\}^2$$

$$- \{Y_i - m_{M,n}^{(OOB)}(X_i, \Theta_{(M)})\}^2$$

où $\hat{\sigma}_Y^2 = \frac{1}{n-1} \sum_{i=1}^n (Y_i - \bar{Y})^2$ est la variance standard estimée de la réponse Y .

2.4.4 ~~Sobol-MDA~~ - convergence

Dans l'article Bénard, Da Veiga, and Scornet (2022), la convergence, sous certaines hypothèse sur la construction du random-forest faciles à mettre en place, des indices Sobol-MDA vers les indices de Sobol totaux lorsque le nombre d'échantillons augmente a été prouvé. Nous présentons ici l'idée générale de la preuve.

L'indice de Sobol total $S_j^T = \frac{E[Var(G(X)|X^{(-j)})]}{VarG(X)}$, s'écrit, dans le cas du random forest, en remplaçant $G(X)$ par l'estimation obtenue par la forêt, l'indice de Sobol total s'écrit donc $S_j^T = \frac{E[Var(m(X)|X^{(-j)})]}{Var[m(X)]} = \frac{E[(m(X) - E[m(X)|X^{(-j)}])^2]}{Var(m(X))}$. On cherche donc à montrer que

$$\widehat{S - MDA}_{M,n}(X^j) \xrightarrow{P} S_j^T$$

Pour cela, on majore $E[\widehat{S - MDA}_{M,n}(X^j) \sigma_Y^2 - E[(m(X) - E[m(X)|X^{(-j)}])^2]]$ par une expression dépendant de n en faisant une décomposition sur laquelle on applique l'inégalité triangulaire. On obtient alors

$$\widehat{S - MDA}_{M,n}(X^j) \sigma_Y^2 \xrightarrow{P} E[(m(X) - E[m(X)|X^{(-j)}])^2]$$

Ensuite l'indice de Sobol-MDA est normalisé par la variance standard estimée σ_Y^2 de la sortie Y qui est convergente par la loi des grands nombres. Grâce au “mapping theorem” on a

$$\frac{1}{\hat{\sigma}_Y^2} \xrightarrow{p} \frac{1}{V[Y]}$$

Sobol-MDA est donc le produit de deux quantités aléatoires qui converge en probabilité donc on obtient bien

$$S - \widehat{MDA}_{M,n}(X^j) \xrightarrow{p} \frac{E[(m(X) - E[m(X)|X^{(-j)}])^2]}{V[Y]} = S_j^T$$

2.4.5 Sobol-MDA - avantages

L'approche de Williamson et al. (2023), consistant à réentraîner une forêt sans l'une des covariables, permet elle aussi d'approximer les indices de Sobol totaux. Néanmoins, l'avantage de l'approche Sobol-MDA est qu'elle nécessite seulement de faire des prédition supplémentaires avec la forêt, ce qui est plus rapide que le réentraînement d'une forêt. L'algorithme de Williamson et al. (2023) a une complexité en $O\{Mp^2nlog^2(n)\}$ qui est quadratique avec les dimensions et l'algorithme Sobol-MDA a une complexité en $O\{Mnlog^3(n)\}$ ce qui est un grand avantage lorsque l'on travaille avec un grand nombre de variables d'entrée.

3 Données

L'objectif de ce stage étant d'évaluer la sensibilité des sorties du modèle ISIS-Fish à certains paramètres d'entrée, nous allons dans ce chapitre présenter plus en détails le simulateur ISIS-Fish (voir [Mahévas and Pelletier \(2004\)](#) ~~le premier article écrit sur ce modèle~~). Ensuite nous nous intéressons à la pêcherie langoustinière et aux paramètres d'entrée que nous allons faire varier pour évaluer la sensibilité. Finalement nous expliquerons le plan de simulations mis en place.

3.1 Présentation d'ISIS-Fish

ISIS-Fish (Integration of Spatial Information and Simulation for Fisheries) est un simulateur des dynamiques de pêche. Il peut aider les pêcheurs, les managers, les entreprises et les scientifiques dans leur recherche de la meilleure gestion écosystémique spatialisée des pêches. Mais comment marche ce modèle mécanistique ? Tout d'abord, il faut avoir en tête que ce modèle est composé de trois sous-modèles qui interagissent à travers le temps et l'espace. Précisons que le pas de temps est le mois et qu'un maillage de l'espace est réalisé pour obtenir des zones. Parmi les sous-modèles on trouve la modélisation du management, de l'activité de pêche et des dynamiques des populations de poissons. Leurs interactions sont représentées dans la Figure 1.

La composante “management” simule les règles qui s'appliquent sur la pêche (mesures de sélectivité, TCA, zones protégées, saisons interdites, etc). La composante “populations de poissons” simule le cycle de vie des poissons. Chaque mois les poissons grandissent, migrent, se reproduisent, et certains meurent de causes naturelles (voir Figure 2). Pour chaque mois, une carte de l'abondance des poissons par zone est produite par ISIS.

La composante “dynamique des pêches” simule l'exploitation des ressources halieutiques. Celle-ci est quantifiée par l'effort de pêche, qui dépend des engins de pêche utilisés et de la durée passée à pêcher. Les navires de pêche ne sont pas représentés individuellement mais ils sont organisés en flottes rattachées à des zones de pêche selon la durée passée en mer et selon leurs caractéristiques techniques. Pour chaque flotte, des stratégies décrivent la distribution de l'effort de pêche en fonction des métiers et des mois. Les stratégies peuvent changer en fonction des mesures de management, de l'abondance des populations, du prix de l'essence et du prix de vente des poissons. Chaque mois, la carte (toujours à l'échelle des zones) de l'effort de pêche est alors mise à jour.

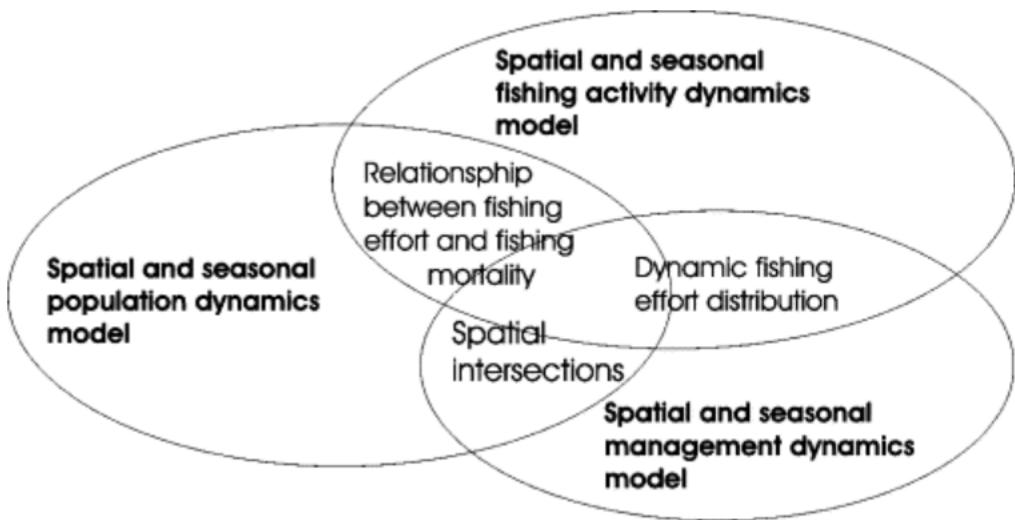


Figure 1: Une vue générale du modèle de pêcherie spatiale mixte décomposé en trois sous-modèles interagissant à travers le temps et l'espace. Les sous-modèles interagissent via une intersection spatiale et temporelle. (trouvé dans Mahévas and Pelletier (2004))

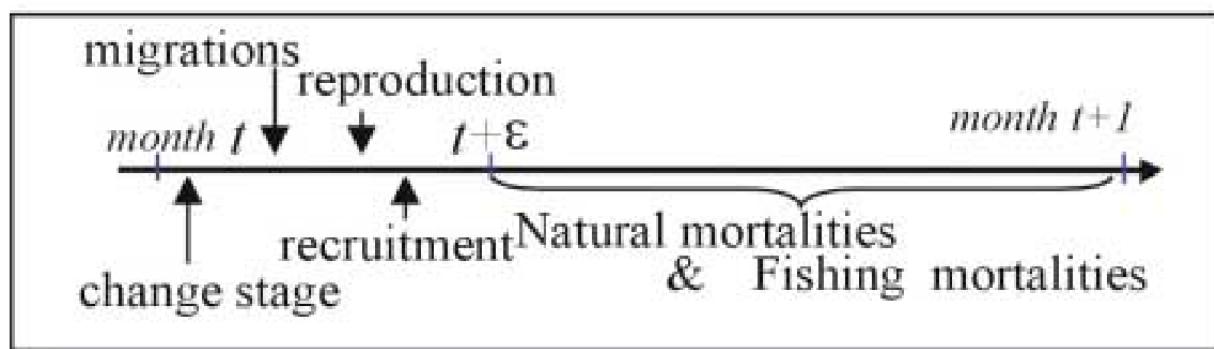


Figure 2: Chronologie des processus considérés dans le modèle de population, fishing mortalities proviennent de l'interaction entre la composante “dynamique des populations” et “dynamique des pêches”

A la fin de chaque mois, ISIS-Fish superpose la carte de l'effort de pêche sur celle de l'abondance des populations. Pour les espèces que l'on considère "immobiles", les captures par la flotte sont alors calculées sur les zones d'intersection (car on a donc sur ces zones des poissons et des pêcheurs). Pour les espèces qui se déplacent, les captures ne sont pas seulement calculées sur le nombre d'animaux marins présents dans la zone d'intersection mais aussi sur ceux des zones adjacentes qui passeront probablement par les zones d'intersection durant la durée de la pêche. En fonction des stratégies mises en place et de la réglementation, les pêcheurs décident des poissons à garder et de ceux à remettre à l'eau, on représente cela dans le modèle avec la probabilité de survie après relâchement.

Avant d'utiliser ISIS-Fish comme un outil d'aide à la prise de décision, les valeurs des paramètres doivent être sélectionnées pour représenter au mieux la pêcherie étudiée. Pour cela, des données de pêche et des données d'études sont utilisées et des hypothèses sont faites dans le cas d'informations manquantes. Pour valider ou invalider les paramètres, on compare les résultats du simulateur avec les données sur la situation passée de la pêcherie étudiée.

Après validation des paramètres on peut simuler différents scénarios de management pour voir lequel conviendrait le mieux. Par exemple, on peut estimer le poids des captures de pêche dans 5 ans dans le cadre de mise en place d'une zone de pêche protégée. Des simulations sont alors faites avec des valeurs de paramètres prises aléatoirement dans leur intervalle de confiance. On obtient alors non pas une valeur de sortie mais un intervalle de valeurs de sortie. On prend ainsi en compte la propagation de l'incertitude d'entrée dans la sortie. Un des objectif est de réduire les incertitudes en entrée pour ainsi réduire l'incertitude en sortie. L'analyse d'incertitude est très utiles dans le sens où, en nous fournissant les paramètres d'entrée qui augmente le plus la variance de la sortie, elle nous permet de connaître les paramètres pour lesquels on doit chercher en priorité à réduire l'incertitude.

3.2 Précisions sur la pêcherie langoustinière

Dans le cadre de ce stage, nous avons décidé de calculé les indices de sensibilité pour une pêcherie langoustinière inspirée de la pêcherie de la grande vasière dans le golfe de Gascogne paramétrée avec ISIS-Fish. La paramétrisation a été simplifiée pour faciliter l'utilisation et la compréhension de ce modèle. Présentons brièvement la modélisation de cette pêcherie langoustinière.

Le cycle de vie de la langoustine est décrit dans le modèle au travers de 10 classes d'âge qui se distribuent spatialement sur 9 rectangles de taille 1 degré en longitude et 0.5 degré en latitude. Chaque classe d'âge se caractérise par une largeur moyenne de la carapace de la tête (longueur céphalothoracique) qui permet de décrire l'interdiction de débarquer des langoustines dont la carapace de la tête est plus petite que 20 mm (réglementation de la taille minimale de débarquement). Si une langoustine trop petite est capturée, elle est rejetée dans la mer avec une chance de survie (proportion de survie). Le renouvellement annuel des langoustines juvéniles est dépendant de la quantité de langoustines en âge de se reproduire (relation stock-recruement de Beverton et Holt). Dans cette description, on fait l'hypothèse qu'il n'y a pas de dispersion des larves à l'extérieur des 9 rectangles ni entre les rectangles. A chaque mois de l'année, le modèle décrit la carte du nombre de langoustines par classe d'âge (abondance ou biomasse en poids). Les langoustines sont capturées par plusieurs groupes de bateaux de pêche (chalutiers) qui diffèrent par leur port d'attache, leur longueur, les espèces ciblées (métiers : langoustiniers, poissons benthiques, poissons démersaux) et leurs pratiques annuelles des métiers (stratégies). Le temps passé à pêcher (effort de pêche) est spatialisé et se distribue différemment selon les métiers et les saisons dans les 9 rectangles. Selon le métier, cet effort de pêche est plus ou moins efficace pour capturer des langoustines et l'efficacité de pêche change au cours du temps (dérive d'efficacité de pêche). A chaque mois de l'année, en multipliant l'effort de pêche par un facteur appelé capturabilité, le modèle calcule une carte de la mortalité par pêche des langoustines pour chaque métier. Chaque mois, sur la période de 5 ans simulée, le modèle prédit les captures, la biomasse et la biomasse féconde de langoustine par rectangle en superposant les cartes de mortalité par pêche des métiers et la carte d'abondance des langoustines. Pour simplifier les exports des résultats du modèles, on a choisi de sommer les résultats sur les zones et sur les mois. On obtient ainsi la biomasse, la biomasse féconde et les captures de pêche (en tonnes) sur la zone de présence totale des langoustines au début du mois de janvier pour les 5 années suivantes.

3.3 Paramètres incertains

Pour calculer les indices de sensibilité de 5 paramètres particulièrement incertains, il nous a d'abord fallu faire des simulations en prenant en compte l'incertitude de ces paramètres. Nous avons donc simuler la pêcherie pour un ensemble de valeurs probables des paramètres incertains suivants :

- la proportion de survie : pourcentage de survie des captures non débarquées pour des raisons diverses (taille illégale, poisson endommagé, absence de marché ou dépassement des quotas).
- la dérive d'efficacité de pêche : évolution moyenne de la capacité à capturer le poisson accessible.
- le facteur de standardisation des engins : capacité des chaluts (simple ou jumeau) à capturer les langoustines en fonction de la taille de leurs mailles.
- la mortalité naturelle : taux de mortalité naturelle des langoustines selon la classe d'âge
- la fécondité : taux de reproduction des langoustines selon la classe d'âge

Chacun de ces paramètres a une valeur de référence autour de laquelle ils peuvent osciller. La valeur de référence de la proportion de survie est de 0.5 et on considère que ce paramètre suit une loi uniforme sur l'intervalle [0.2,0.7]. La valeur de référence de la dérive d'efficacité de pêche est de 0 et on considère que ce paramètre suit une loi uniforme sur [-0.1,0.2]. La valeur de référence de la standardisation est de 1 et on considère que ce paramètre suit une loi uniforme sur [0.8,1.2]. Concernant la mortalité et la fécondité, c'est un peu plus compliqué car le taux diffère selon la classe d'âge. Chaque classe d'âge a donc une valeur de référence (comprise entre 0 et 0.2 pour la mortalité et entre 0.0 et 6.01 pour la fécondité) et pour chaque classe d'âge nous multiplierons la valeur de référence par un valeur tirée aléatoirement sur [0.8,1.2] pour ainsi faire varier le taux de plus ou moins 20 pour cent.

3.4 Plan de simulation

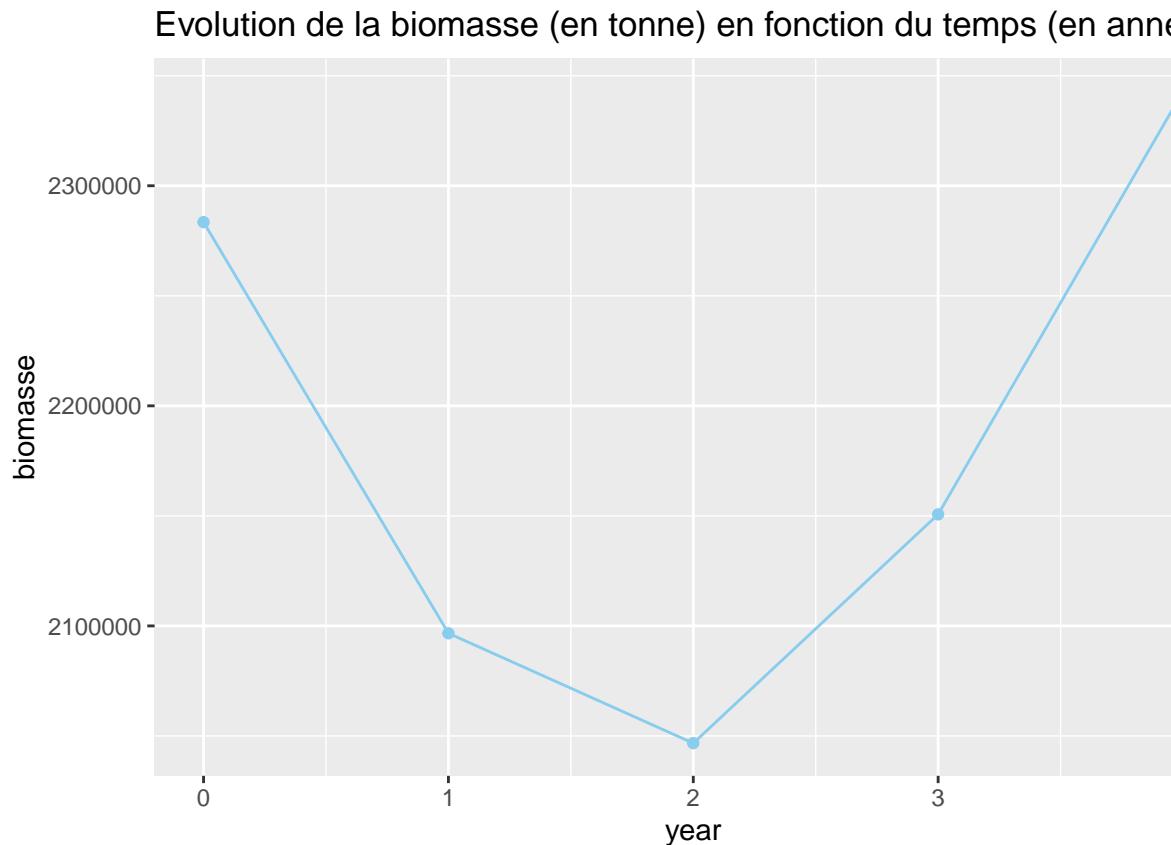
Pour commencer, nous chercherons à avoir un aperçu de l'influence des différents paramètres sur les sorties. Pour cela, nous ferons 50 simulations en faisant varier un seul paramètre (les autres étant fixés à leur valeur de référence) et nous répéterons l'opération pour chaque paramètre.

Ensuite, pour obtenir les données qui serviront à entraîner le random forest et estimer les indices de Sobol-MDA, nous feront 1000 simulations pour lesquelles les valeurs des paramètres d'entrées sont tirées aléatoirement dans leur intervalle respectif. Pour cela nous créerons une matrice d'entrée composée de 5 colonnes (une pour chaque paramètre) et 1000 lignes, les valeurs des paramètres sont tirées aléatoirement selon une loi uniforme (runif dans R) dans leur intervalle respectif et nous choisissons de mettre une corrélation entre certains paramètres. On suppose alors que la probabilité de survie après relâchement est corrélée négativement avec la mortalité naturelle et avec la dérive d'efficacité. Ainsi, la corrélation entre mortalité naturelle et proportion de survie après relâchement est établie à -0.3 et la corrélation entre la proportion de survie après relâchement et la dérive d'efficacité de pêche est établie à -0.1.

4 Résutats

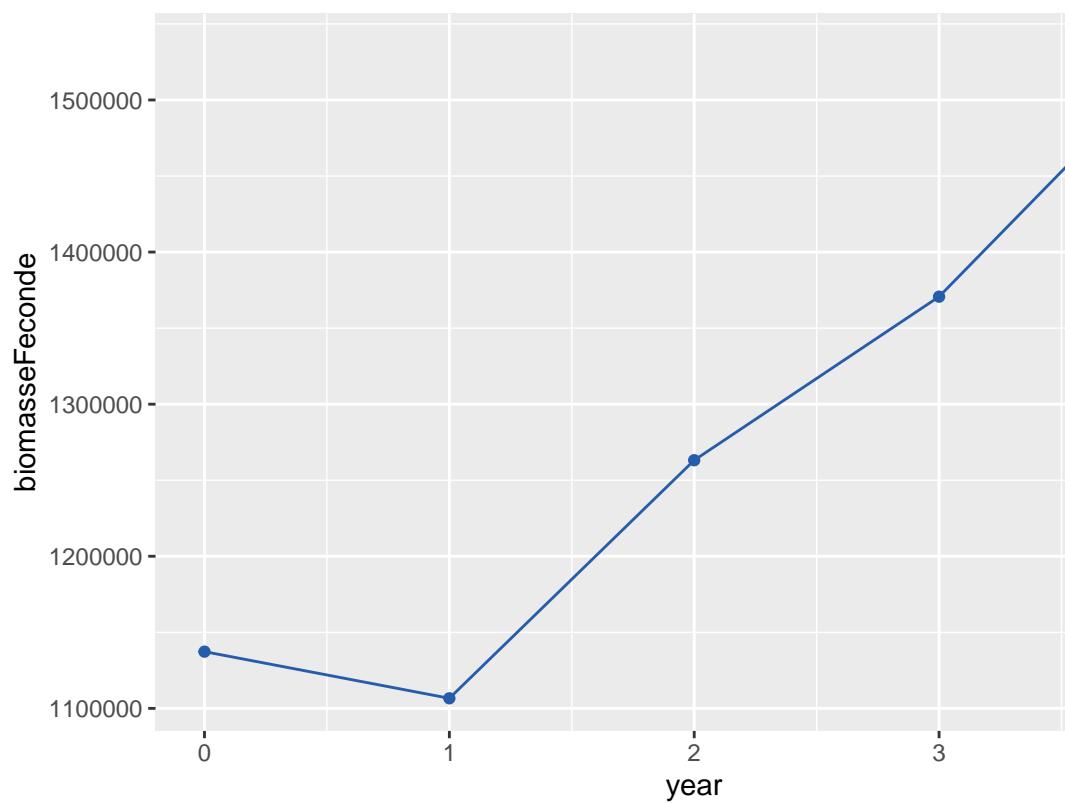
4.1 Exploration des sorties ISIS

4.1.1 Evolution au cours des années lorsque tous les paramètres sont à leur valeur de référence



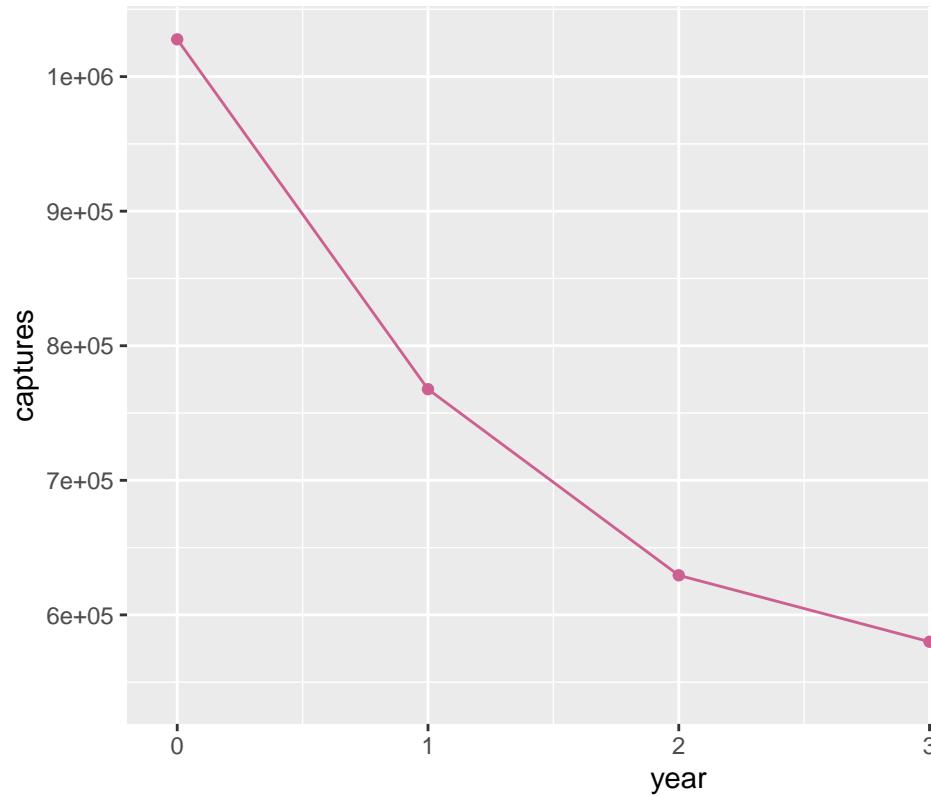
4.1.1.1 Biomasse

Evolution de la biomasse féconde (en tonne) en fonction du



4.1.1.2 Biomasse féconde

Evolution du poids des captures (en tonne) en fonction de l'année

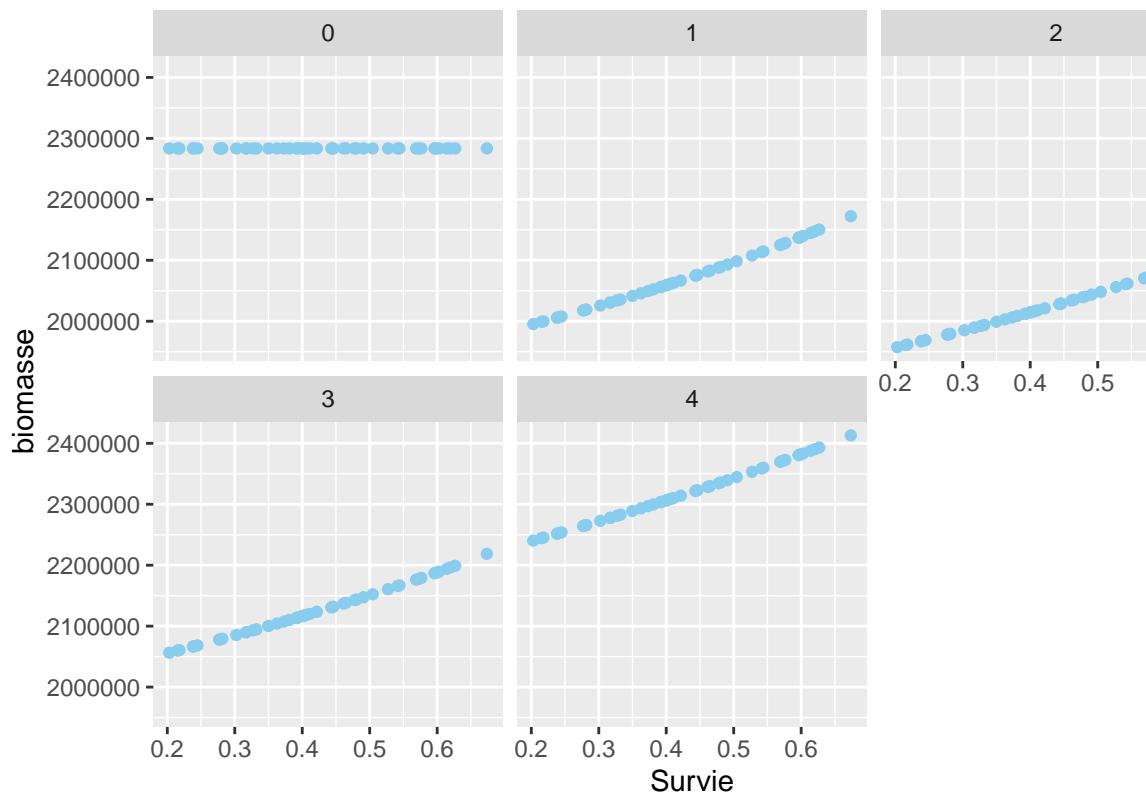


4.1.1.3 Poids des captures de pêche

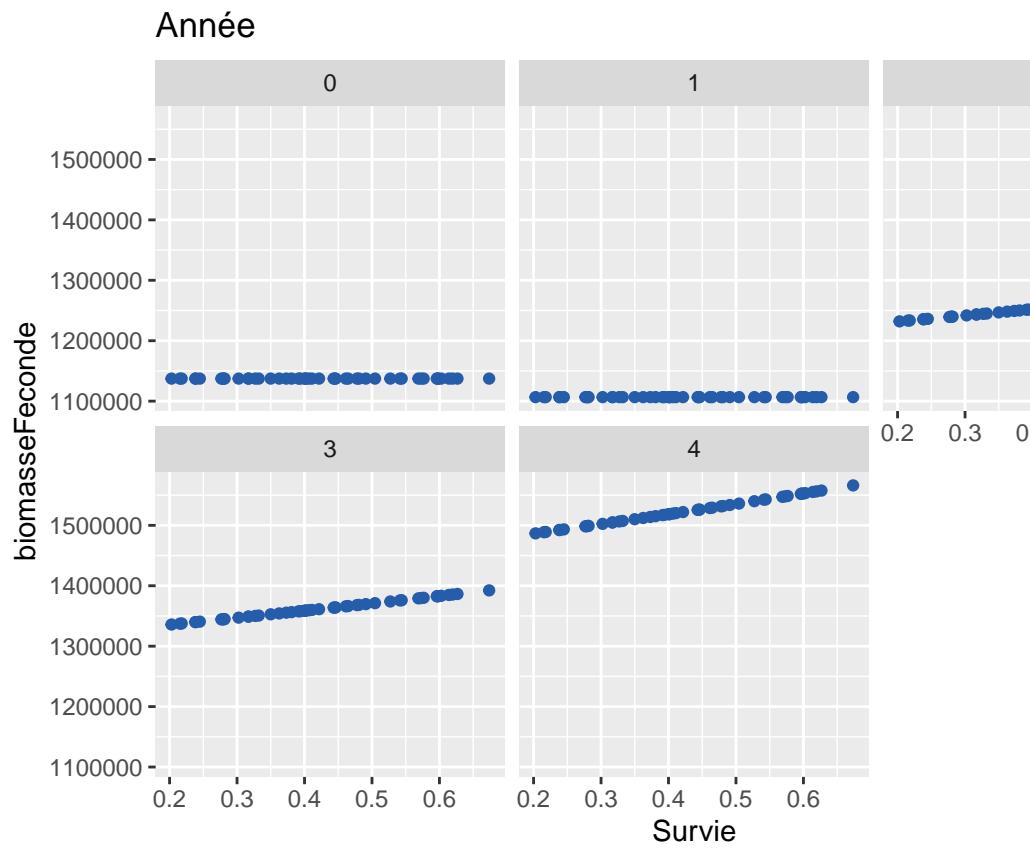
4.1.2 Exploration des résultats dans le cas où un seul paramètre varie

4.1.2.1 Survie

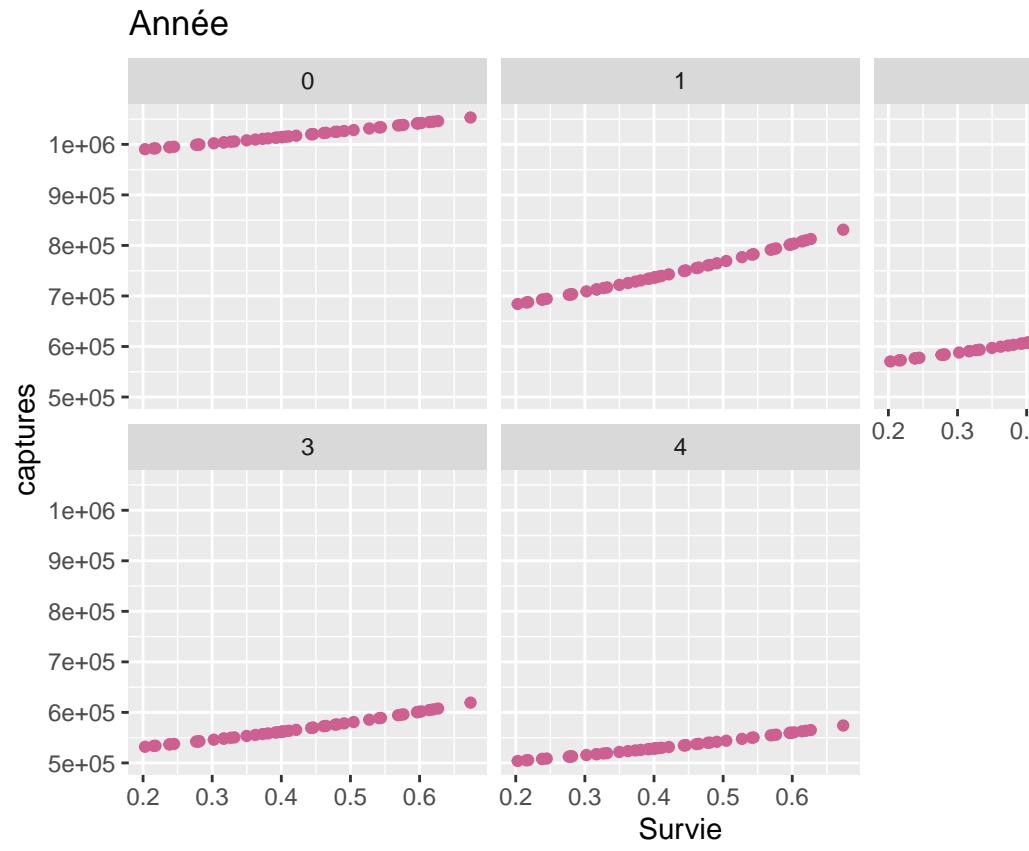
Variations de la biomasse en fonction de la survie pour chaque a



4.1.2.1.1 Biomasse



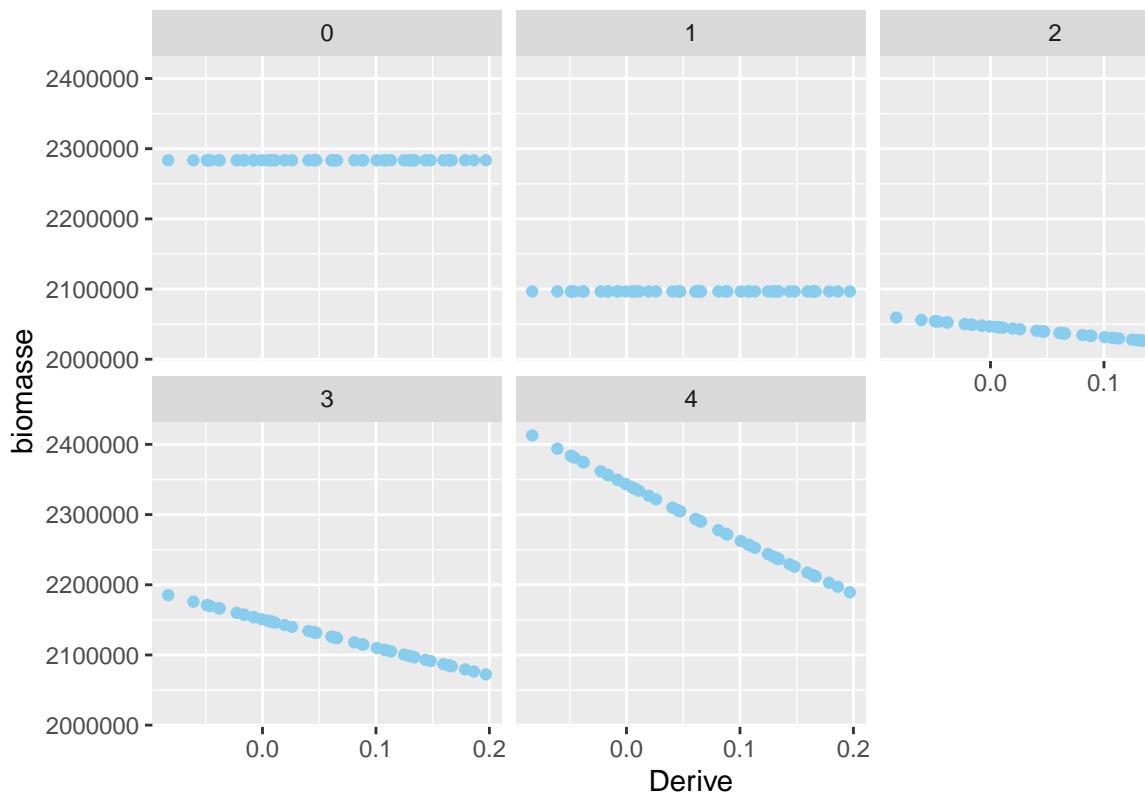
4.1.2.1.2 Biomasse Feconde



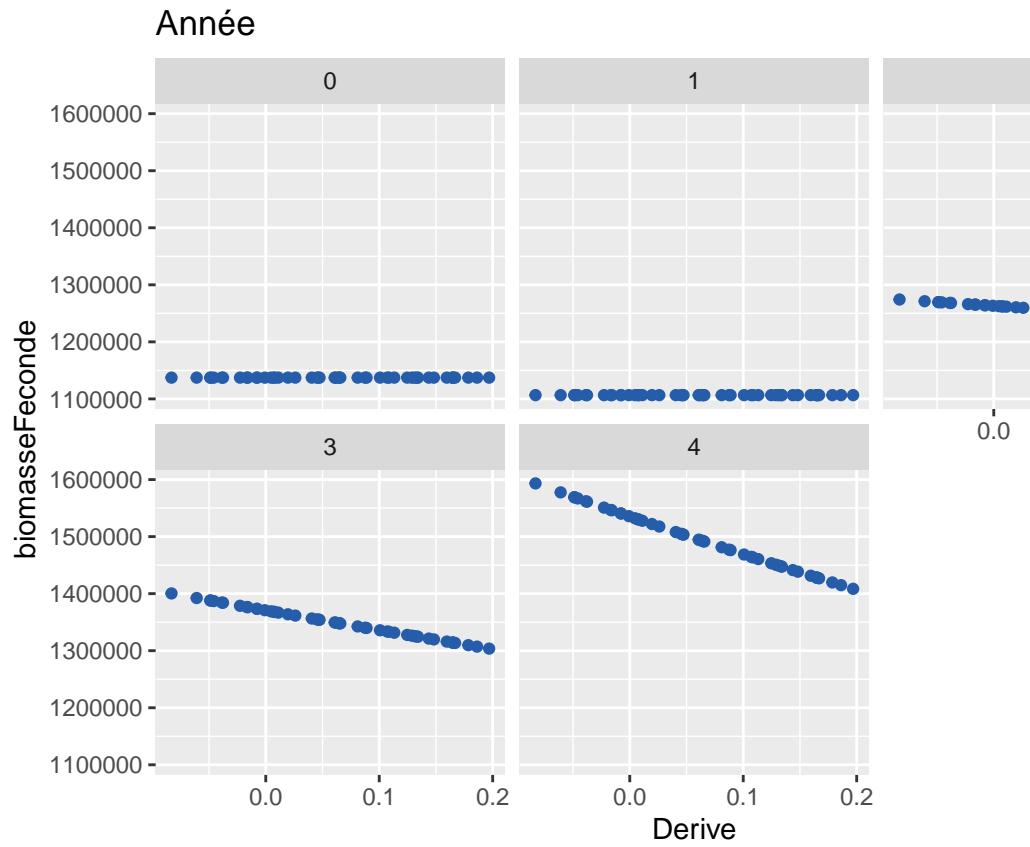
4.1.2.1.3 Poids des captures

4.1.2.2 Dérive

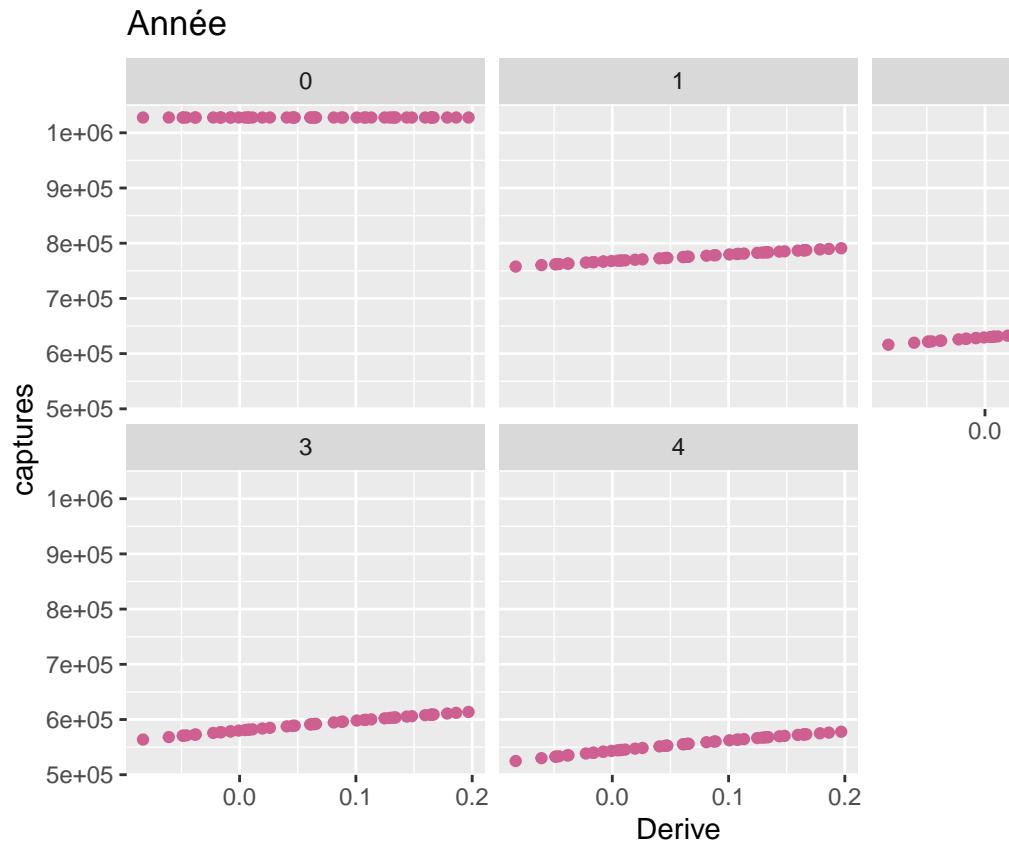
Variations de la biomasse en fonction de la dérive pour chaque a



4.1.2.2.1 Biomasse



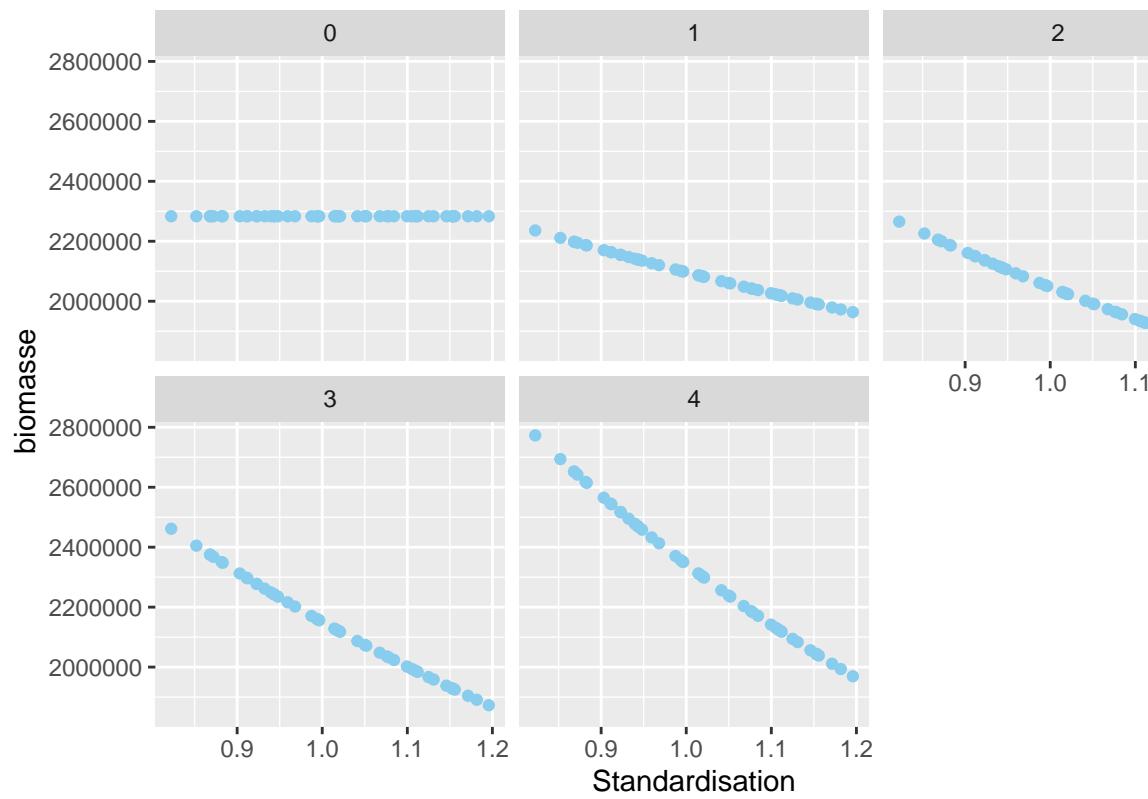
4.1.2.2.2 Biomasse Feconde



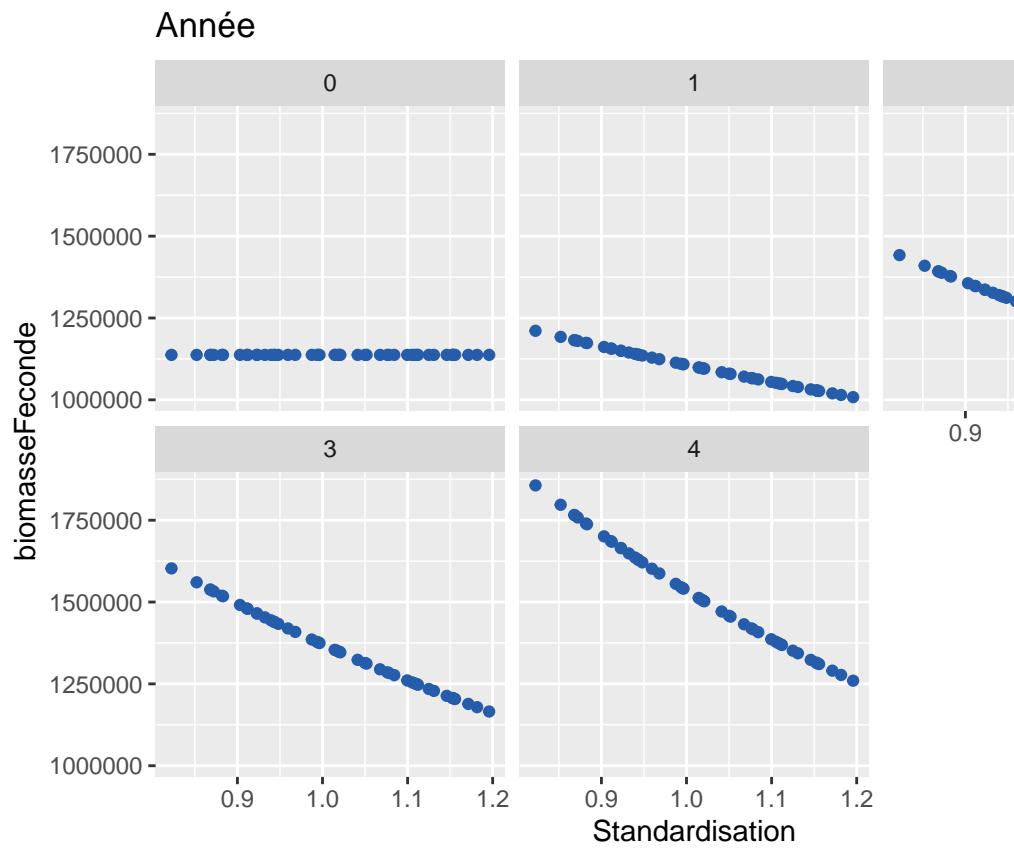
4.1.2.2.3 Poids des captures

4.1.2.3 Standardisation

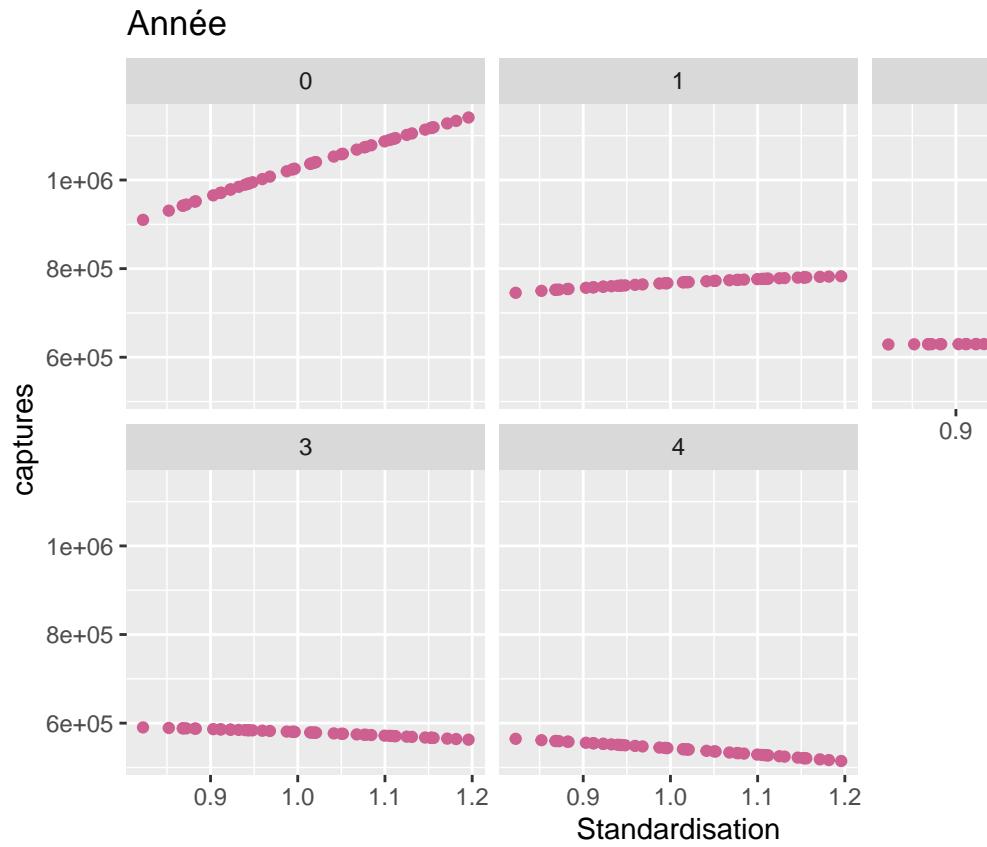
Variations de la biomasse en fonction de la standardisation pour



4.1.2.3.1 Biomasse



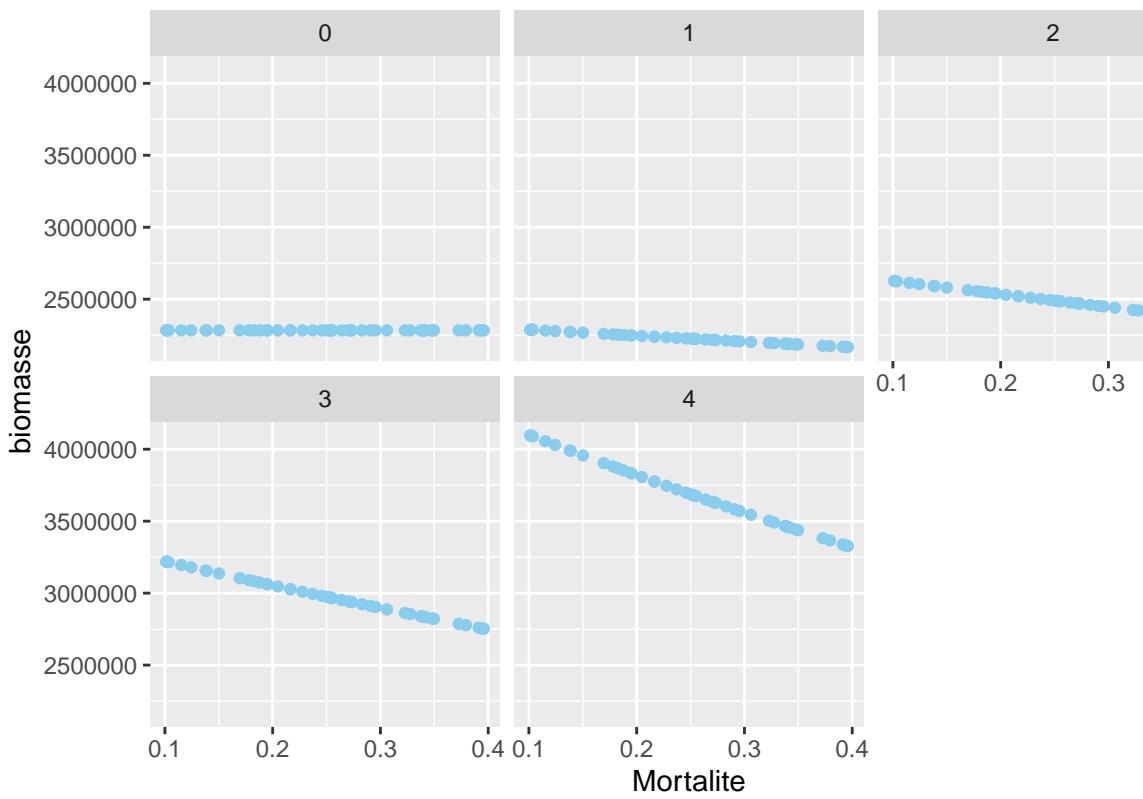
4.1.2.3.2 Biomasse Feconde



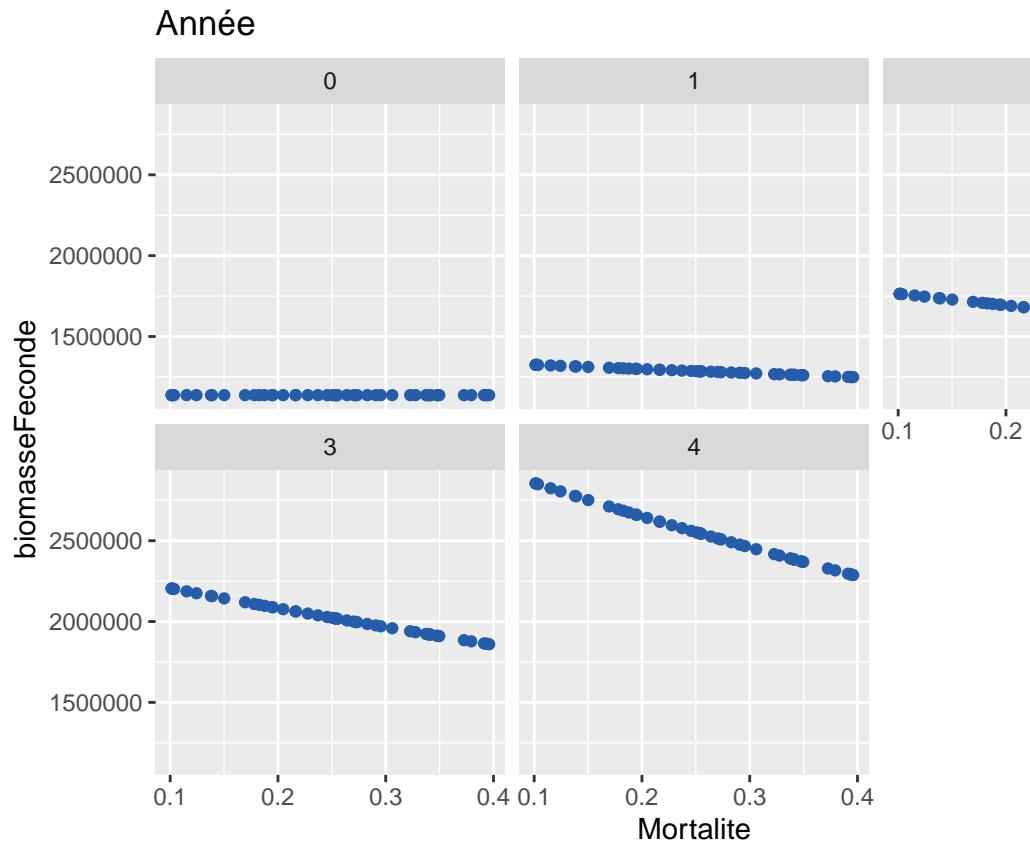
4.1.2.3.3 Poids des captures

4.1.2.4 Mortalite

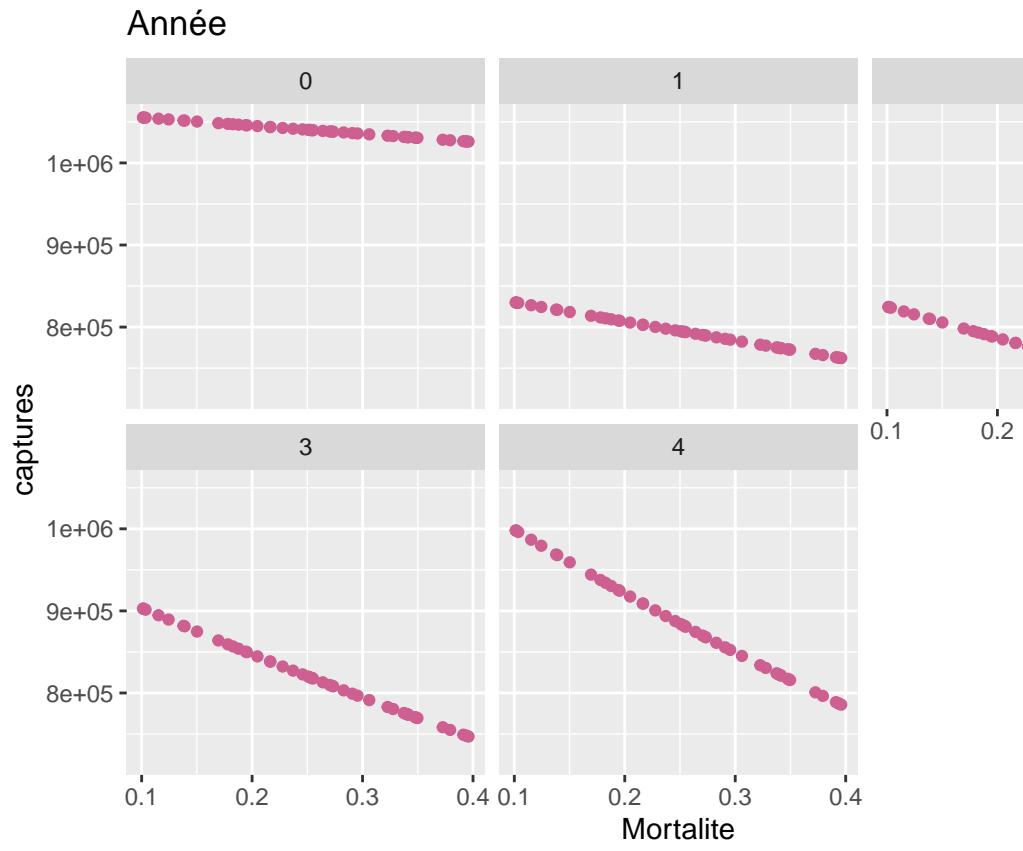
Variations de la biomasse en fonction de la mortalité pour chaque



4.1.2.4.1 Biomasse

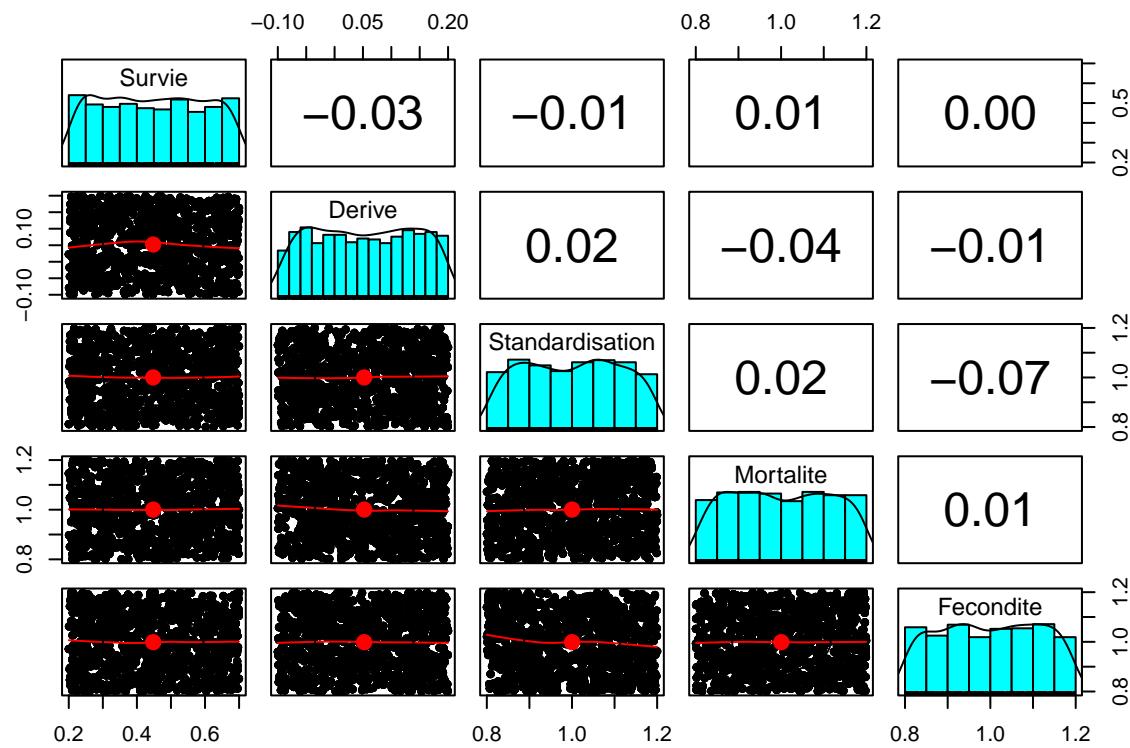


4.1.2.4.2 Biomasse Feconde

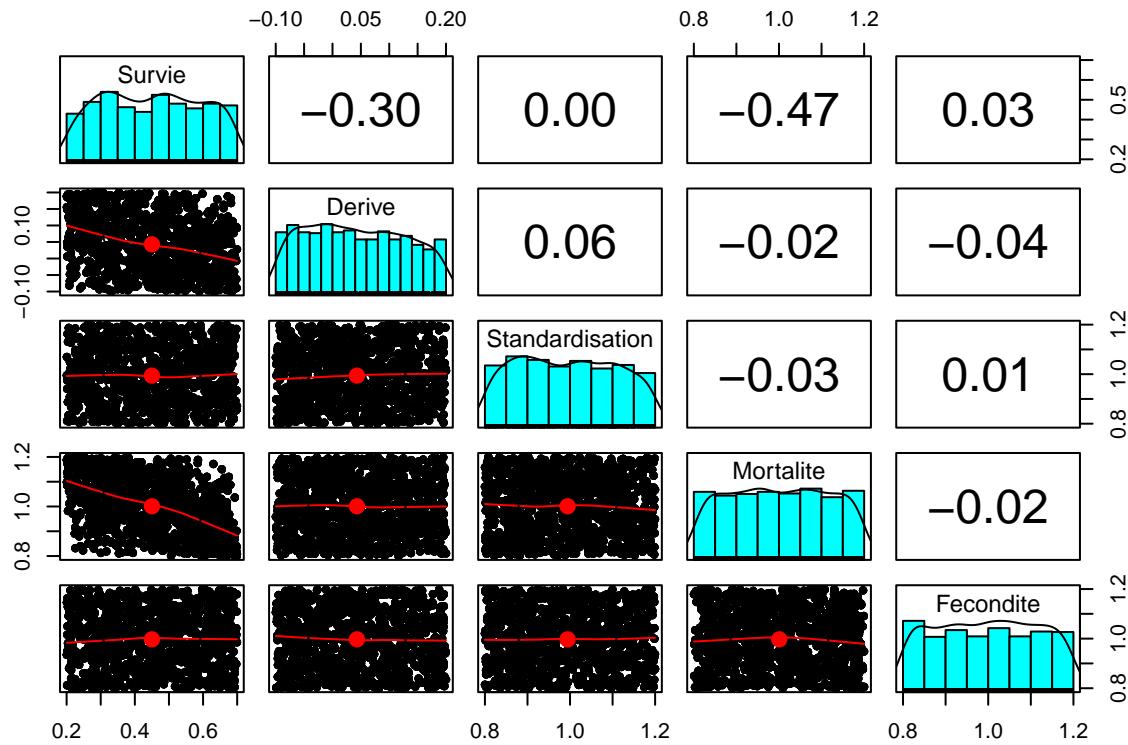


4.1.2.4.3 Poids des captures

4.1.3 Visualisation des données d'entrée - paramètres indépendants

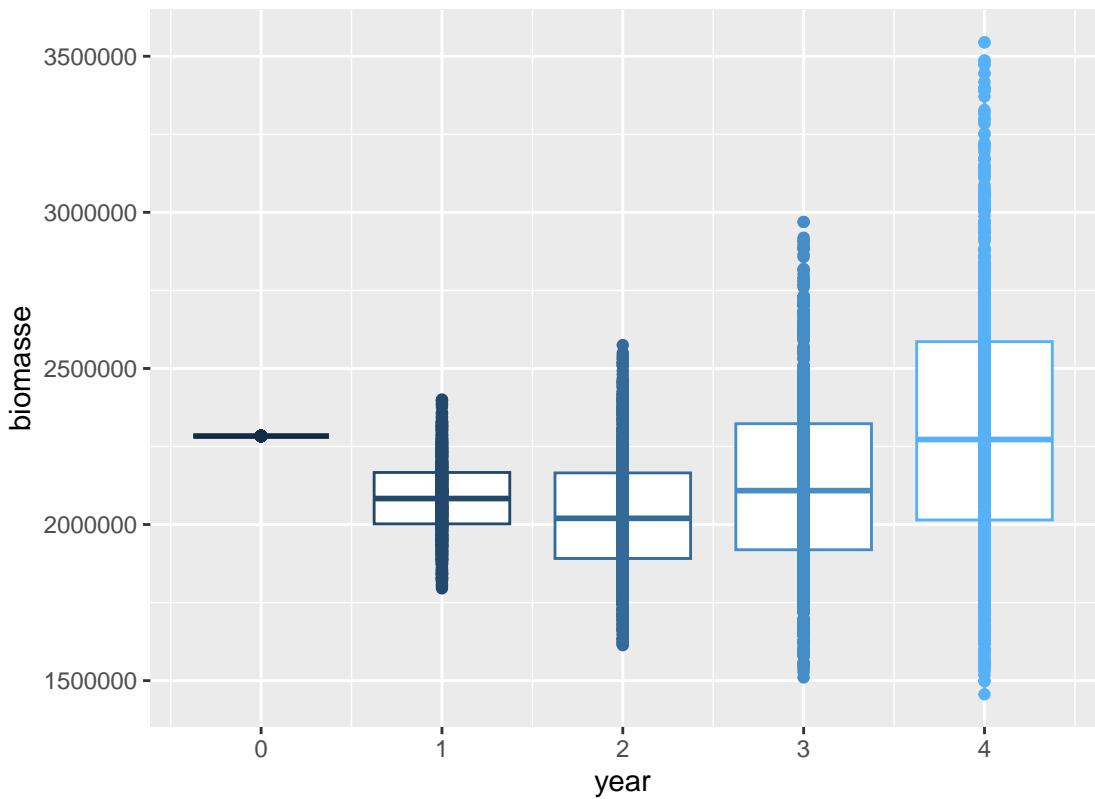


4.1.4 visualisation des données d'entrées utilisées pour les simulations - paramètres dépendants



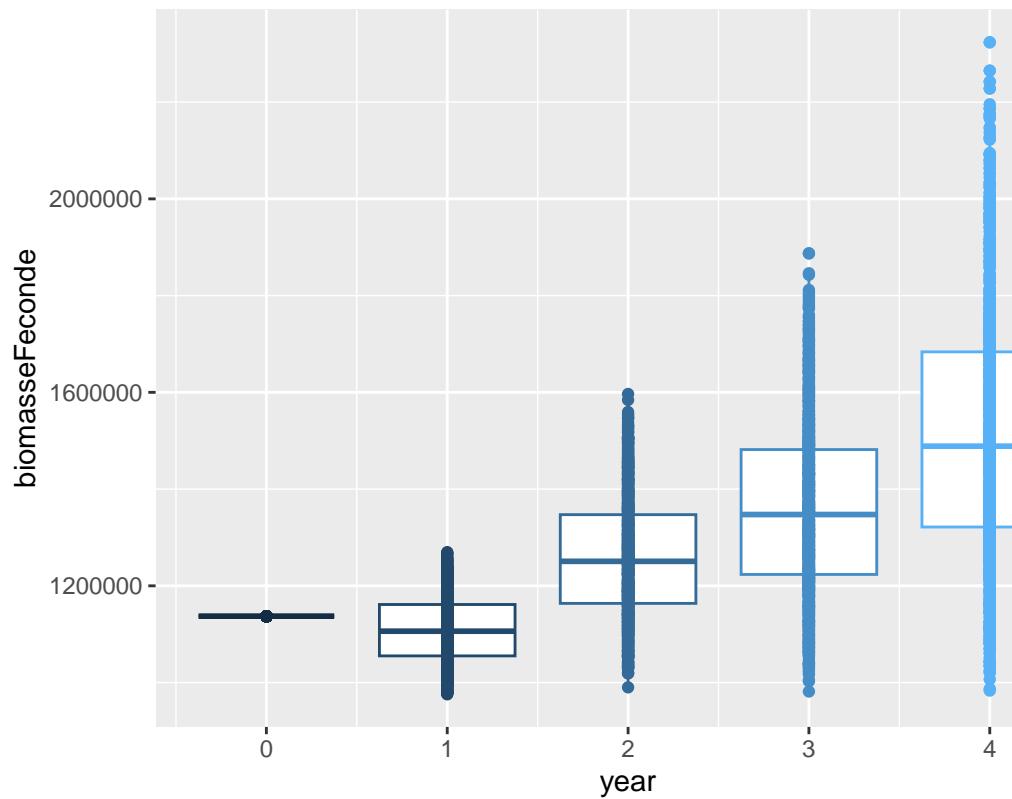
4.1.4.1 Evolution au fil des années - paramètres indépendants

boîtes à moustache de la biomasse par année



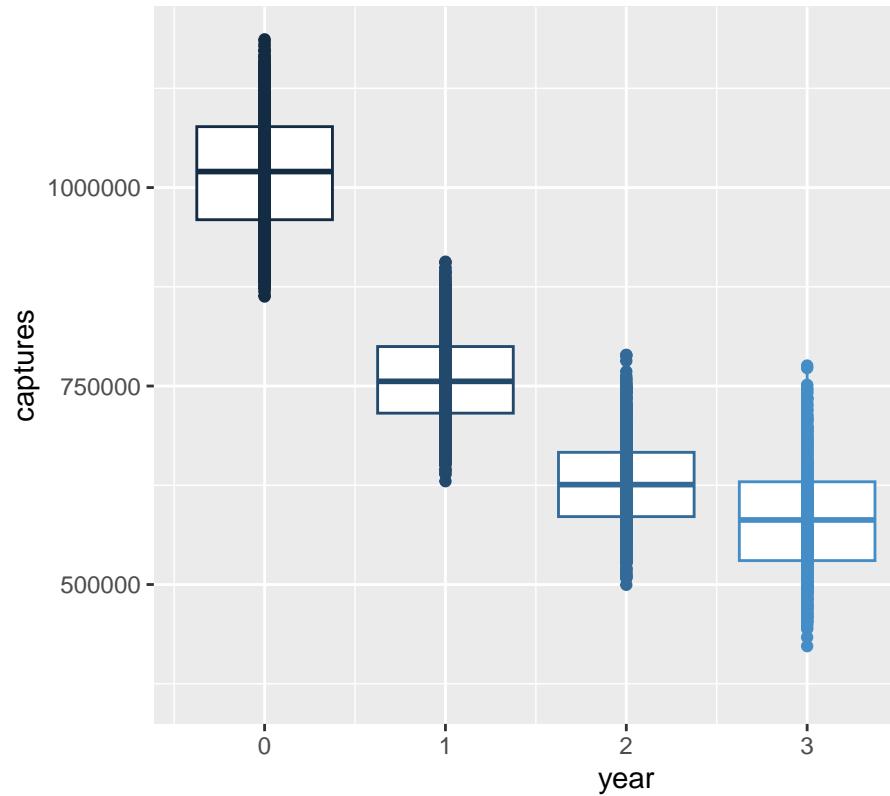
4.1.4.1.1 Biomasse

boîtes à moustache de la biomasse féconde par année



4.1.4.1.2 Biomasse Feconde

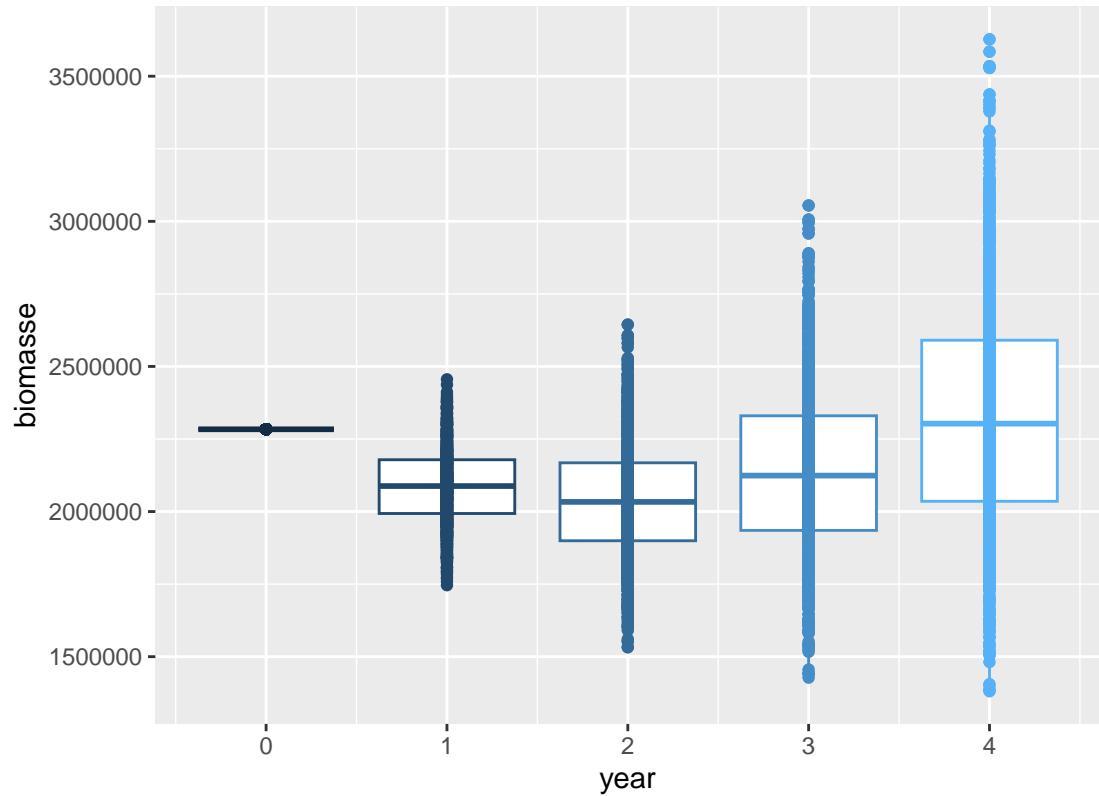
boîtes à moustache du poids des captures par an



4.1.4.1.3 Poids des captures de pêche

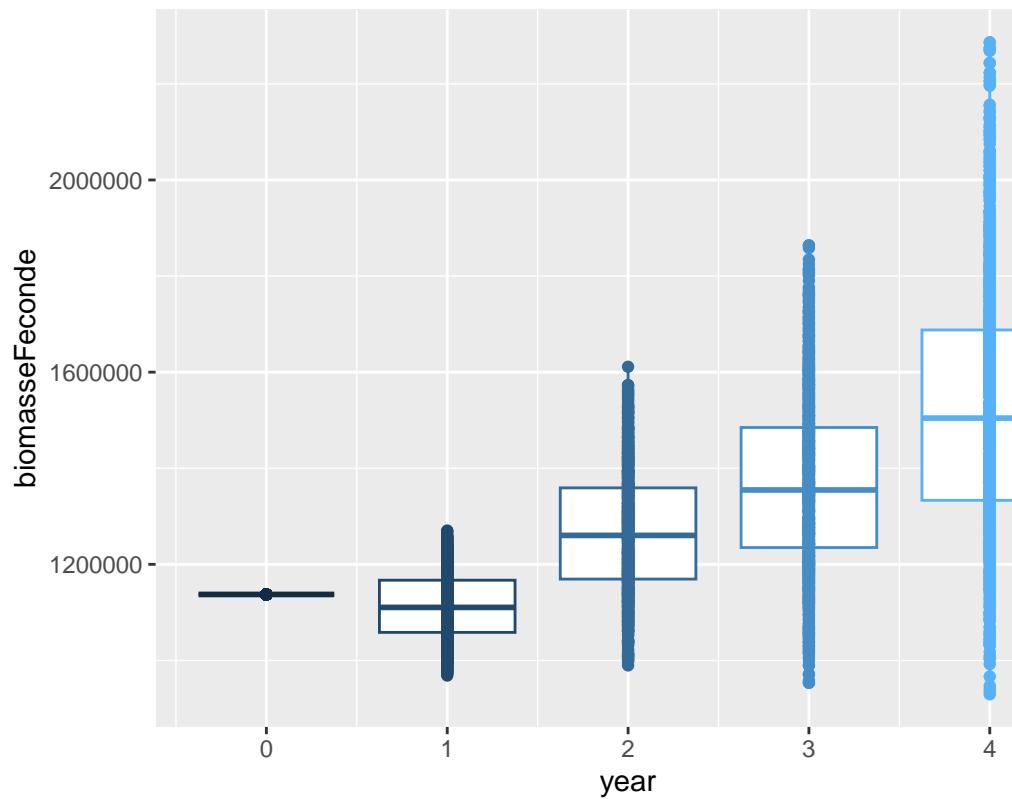
4.1.4.2 Evolution au fil des années - paramètres dépendants

boîtes à moustache de la biomasse par année



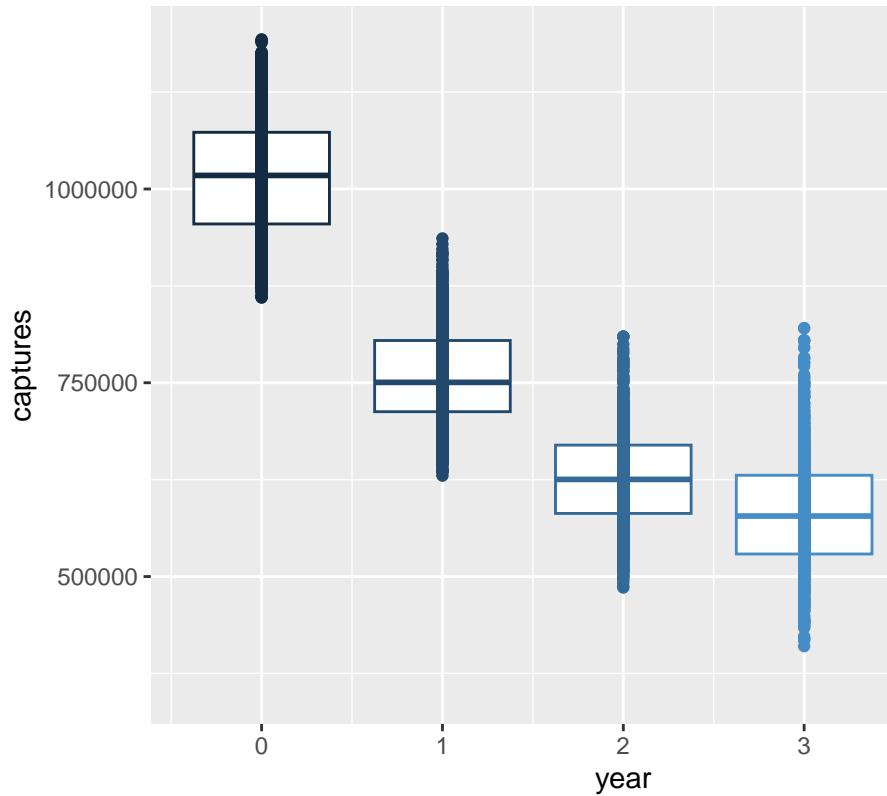
4.1.4.2.1 Biomasse

boîtes à moustache de la biomasse féconde par année



4.1.4.2.2 Biomasse Feconde

boîtes à moustache du poids des captures par an



4.1.4.2.3 Poids des captures de pêche

4.2 Construction du random forest

4.3 Calcul des indices de sensibilité

5 Conclusion

6 Annexe

6.1 Indices de Sobol

Les indices de Sobol (Da Veiga et al. (2021)) sont des indices de sensibilité obtenus grâce à une méthode de décomposition de la variance fonctionnelle. Ils sont compris entre 0 et 1 et leur somme vaut 1. Ils représentent le pourcentage de la variance de la réponse expliquée par la variable (ou le groupe de variables) pour laquelle ils sont calculés. Voyons cela plus en détails.

On note dans la suite X le vecteur incluant l'ensemble des variables d'entrée incertaines : $X = (X_1, \dots, X_K)$, où K est le nombre de facteurs incertains. On note finalement $G(X)$ le vecteur des variables de sortie du modèle associées à X . Si le modèle ne comporte qu'une seule variable de sortie ou si on ne s'intéresse qu'à une seule des variables de sortie du modèle, $G(X)$ est un scalaire.

Pour pouvoir écrire la décomposition ANOVA, l'hypothèse 1 suivante doit être vérifiée.

- *Hypothèse 1 :*

Chaque X_i , $i = 1, \dots, d$, est à valeurs dans un espace mesurable polonais (espace métrique complet et séparable) abstrait $(E_i, B(E_i))$. Ici, $B(E_i)$ désigne la tribu borélienne associée à E_i . Pour tout sous-ensemble non vide d'indices $A \in P_d$ ($P_d = P([1 : d])$, l'ensemble de tous les sous-ensembles de $[1 : d] = \{1, \dots, d\}$), on définit $(E_A, \epsilon_A) = (\Pi_{i \in A} E_i, \otimes_{i \in A} B(E_i))$. On pose $(E, \epsilon) = (E_{[1\dots d]}, \epsilon_{[1\dots d]})$. Soit P_X la distribution de probabilité du vecteur aléatoire X . Les composantes X_i du vecteur aléatoire X sont supposées être indépendantes ; ainsi, nous avons $P_X = \prod_{i=1}^d P_{X_i}$ avec P_{X_i} la distribution de probabilité de X_i .

On pose, pour tout $A \in P_d$:

- $L^2(P_X) = \{\text{fonctions } f \text{ mesurables sur } (E, \epsilon) : E[f^2(X)] < +\infty\}$
- $L_A^2 = \{f \in L^2(P_X) : f \text{ est mesurable sur } (E_A, \epsilon_A)\}$

On va maintenant chercher à décomposer des fonctions de $L^2(P_X)$ sur des sous-espace de facteurs appropriés. Soit $G \in L^2(P_X)$. Sous l'hypothèse 1, il existe une décomposition unique de G dans $L^2(P_X)$ de la forme

$$G(x) = \sum_{A \in P_d} G_A(x_A)$$

telle que les deux propriétés suivantes soient satisfaites :

1. G_\emptyset est constant.
2. $\forall A \in P_d, A \neq \emptyset, \forall i \in A, \int_{E_i} G_A(x_A) P_{X_i}(dx_i) = 0$

La solution unique s'exprime, $\forall A \in P_d$,

$$G_A(x_A) = \sum_{B \subseteq A} (-1)^{|A|-|B|} \mathbb{E}[G(X) | X_B = x_B]$$

Pour tout $A \in P_d, A \neq \emptyset$, posons $V_A = \text{Var } G_A(X_A)$. Alors, sous l'hypothèse 1, nous obtenons

$$V = \text{Var } G(X) = \sum_{A \in P_d, A \neq \emptyset} V_A.$$

De plus, pour tout $A \in P_d, A \neq \emptyset$, on :

$$V_A = \sum_{B \subset A} (-1)^{|A|-|B|} \text{Var } \mathbb{E}[G(X) | X_B]$$

On peut démontrer que G_A est la meilleure approximation de G dans le sens de $L^2(P_X)$, appartenant à L_A^2 et étant orthogonal (par rapport au produit scalaire hilbertien dans $L^2(P_X)$) à toute fonction dans $L_{A_0}^2$, $A_0 \subsetneq A$.

Maintenant que nous avons défini le cadre théorique sur lequel repose la définition des indices de Sobol, voyons leur expression mathématique.

Soit $G \in L^2(P_X)$ et on suppose que l'hypothèse 1 est satisfaite. Soit $A \in P_d$.

- L'indice de Sobol associé à A est défini tel que :

$$S_A = \frac{V_A}{V} = \frac{\sum_{B \subseteq A} (-1)^{|A|-|B|} \text{Var } [E[G(X) | X_B]]}{\text{Var}[G(X)]}.$$

- S_j est l'indice de Sobol associé au singleton $\{j\}$. On l'appelle l'indice d'ordre un pour la variable d'entrée X_j . Plus généralement, si $p = |A|$, alors S_A est appelé l'indice de Sobol d'ordre p associé à X_A .

- L'indice de Sobol fermé associé à l'ensemble A est défini tel que

$$S_A^{\text{clos}} = \sum_{A' \subset A} S_{A'} = \frac{\text{Var}E[G(X)|X_A]}{\text{Var}G(X)}$$

Cet indice est également appelé l'indice de Sobol du premier ordre associé au vecteur d'entrée X_A .

- L'indice de Sobol total associé à X_A est défini tel que :

$$S_A^T = 1 - S_{A^c}^{\text{clos}}.$$

6.2 Effets de Shapley

6.2.1 Contexte et prérequis

Les effets de Shapley permettent d'attribuer la valeur créée par une équipe à ses membres (Zaccour et al. (1988)). Appliqués à l'analyse de sensibilité, on peut considérer que les variables d'entrée (X_1, \dots, X_d) représentent les membres de l'équipe et que la valeur de la sortie $Y=G(X)$ représente la valeur créée par l'équipe. On suppose que $G(X) \in L^2(P_X)$ où P_X est la loi de probabilité de X. Les effets de Shapley résultent d'une allocation directe d'une part de la variance de la sortie à chaque entrée(Da Veiga et al. (2021)). Voyons cela plus en détails.

Considérons une fonction caractéristique val définie sur P_d ($P_d = P([1 : d])$), l'ensemble de tous les sous-ensembles de $[1 : d] = \{1, \dots, d\}$) et à valeurs dans \mathbb{R}_+ telle que $\text{val}(\emptyset) = 0$. Basée sur cette fonction caractéristique, une valeur Φ_j est attribuée à chaque variable d'entrée X_j (à chaque joueur dans le contexte de la théorie des jeux). La méthode d'attribution des valeurs Φ_j aux covariables doit vérifier les 4 propriétés suivantes :

- efficacité : $\sum_{j=1}^d \phi_j = \text{val}([1 : d])$. Cette propriété traduit le fait que les ressources disponibles pour la coalition des variables sont réparties entre elles.
- symétrie : Si $\text{val}(A \cup \{i\}) = \text{val}(A \cup \{j\})$ pour tout $A \subseteq -\{i, j\}$, alors $\Phi_i = \Phi_j$. Cette propriété traduit le fait que si 2 variables d'entrée ont la même contribution marginale à toute coalition alors on leur attribue la même valeur.
- variable muette : Si $\text{val}(A \cup \{i\}) = \text{val}(A)$ pour tout $A \in \mathcal{P}_d$, alors $\phi_i = 0$. Cette propriété traduit le fait que si quelque soit l'ensemble auquel une variable appartient elle ne change pas l'effet de cet ensemble sur la sortie alors on lui attribue une valeur nulle.
- additivité : Si val et val_1 ont respectivement des valeurs de Shapley Φ et Φ_1 , alors le jeu avec une valeur $\text{val} + \text{val}_1$ a une valeur de Shapley $\Phi_j + \Phi_{1j}$ pour $j \in [1 : d]$. La valeur est un opérateur additif dans l'espace de tous les jeux.

L'unique valeur ϕ qui satisfait les 4 propriétés attribue une valeur aux variables X_j selon la formule suivante (voir la preuve dans (Shapley (1953))) :

$$\phi_j = \frac{1}{d} \sum_{A \subset -j} \binom{d-1}{|A|}^{-1} [\text{val}(A \cup \{j\}) - \text{val}(A)]$$

6.2.2 Définition

Pour tout $j = 1, \dots, d$, on définit l'indice de Shapley pour X_j comme

$$\text{Sh}_j = \frac{1}{d} \sum_{A \subset -j} \binom{d-1}{|A|}^{d-1} (S_{A \cup \{j\}}^{\text{clos}} - S_A^{\text{clos}})$$

Cette définition correspond à la valeur ϕ_j obtenue en définissant la fonction caractéristique $val : \mathcal{P}_d \rightarrow \mathbb{R}^+$ comme

$$val(A) = S_A^{clos} = \frac{\text{Var}E[G(X)|X_A]}{V} \quad (1)$$

On peut choisir de définir la fonction caractéristique $val_0 : \mathcal{P}_d \rightarrow \mathbb{R}^+$ par $val_0(A) = \frac{E \text{Var}[G(X)|X_A]}{V}$ (2) pour obtenir les mêmes résultats. Cette fonction est utilisée dans certains algorithmes d'estimation des indices.

6.2.3 Propriétés

Les indices de Shapley sont compris entre 0 et 1 et on a $\sum_{j=1}^d Sh_j = 1$.

7 Les indices HSIC

Dans certains cas où la variance ne représente pas très fidèlement la variabilité de la distribution, la mesure d'importance du moment indépendant (voir Da Veiga (2021)) peut s'avérer très utile. On va alors chercher à définir la dépendance entre la sortie Y et chaque paramètre d'entrée X_k d'un point de vue probabiliste. Pour cela, l'idée est de trouver une fonction d qui mesure la similarité entre la distribution de Y et celle de $Y|X_k$. Ainsi, l'impact de X_k sur Y est donné par $S_{X_k} = E_{X_k}(d(Y, Y|X_k))$.

Pour mesurer la dépendance entre X et Y on peut utiliser une mesure qui compare la distribution jointe $P_{X,Y}$ et le produit des distributions marginales indépendantes $P_X P_Y$. On peut, par exemple, utiliser la mesure de la divergence maximale de la moyenne (MMD) défini comme suit :

$$\begin{aligned} MMD^2(P_{Y,X}, P_X P_Y) &= \sup_{f \in \mathcal{F}} [\mathbb{E}_{P_{X,Y}} f(x, y) - \mathbb{E}_{P_X P_Y} f(x, y)]^2 \\ &= \|\mu_{P_{X,Y}} - \mu_{P_X P_Y}\|_{F \times G}^2 \\ &= HSIC(X, Y) \end{aligned}$$

On peut de plus montrer que cette mesure est égale au critère d'indépendance de Hilbert-Schmidt (HSIC), une mesure qui dépend d'un noyau défini dans l'espace joint (voir Da Veiga (2021) pour des explications détaillées).

L'indice de sensibilité basé sur le critère Hsic est défini de la manière suivante :

$$S_{X^k}^{HSIC_{F,G}} = R(X^k, Y)_{F,G}$$

où la corrélation de distance basée sur le noyau de kernel est donnée par

$$R^2(X, Y)_{F,G} = \frac{HSIC(X, Y)_{F,G}}{\sqrt{HSIC(X, X)_{F,F} HSIC(Y, Y)_{G,G}}}$$

Lorsque les variables d'entrée sont dépendantes, on peut utiliser l'effet de Shapley-HSIC défini comme suit :

$$Sh_j^{HSIC} = \frac{1}{HSIC(X, Y)} \frac{1}{p} \sum_{A \subset -j} \binom{p-1}{|A|}^{-1} (HSIC(X_{A \cup \{j\}}, Y) - HSIC(X_A, Y))$$

La propriété de la somme $\sum_{j=1}^d Sh_j^{HSIC} = 1$ reste valable.

Bibliographie

- Bénard, Clément, Sébastien Da Veiga, and Erwan Scornet. 2022. “Mean Decrease Accuracy for Random Forests: Inconsistency, and a Practical Solution via the Sobol-MDA.” *Biometrika* 109 (4): 881–900.
- Breiman, Leo. 2001. “Random Forests.” *Machine Learning* 45: 5–32.
- . 2002. *Manual on Setting up, Using, and Understanding Random Forests*. Statistics Department University of California Berkeley. USA, CA.
- . 2017. *Classification and Regression Trees*. Routledge.
- Cutler, Adele, D Richard Cutler, and John R Stevens. 2012. “Random Forests.” *Ensemble Machine Learning: Methods and Applications*, 157–75.
- Da Veiga, Sébastien. 2021. “Kernel-Based ANOVA Decomposition and Shapley Effects—Application to Global Sensitivity Analysis.” *arXiv Preprint arXiv:2101.05487*.
- Da Veiga, Sébastien, Fabrice Gamboa, Bertrand Iooss, and Clémentine Prieur. 2021. *Basics and Trends in Sensitivity Analysis: Theory and Practice in r*. SIAM.
- Genuer, Robin, Jean-Michel Poggi, and Christine Tuleau-Malot. 2010. “Variable Selection Using Random Forests.” *Pattern Recognition Letters* 31 (14): 2225–36.
- Mahévas, Stéphanie, and Dominique Pelletier. 2004. “ISIS-Fish, a Generic and Spatially Explicit Simulation Tool for Evaluating the Impact of Management Measures on Fisheries Dynamics.” *Ecological Modelling* 171 (1-2): 65–84.
- Scornet, Erwan. 2023. “Trees, Forests, and Impurity-Based Variable Importance in Regression.” In *Annales de l'institut Henri Poincaré (b) Probabilités Et Statistiques*, 59:21–52. 1. Institut Henri Poincaré.
- Shapley, Lloyd S. 1953. “A Value for n-Persons Game.” In *Contributions to the Theory of Games II*, edited by Harold W. Kuhn and Albert W. Tucker, volume 28:pages 307–317. Princeton, NJ: Princeton University Press.
- Williamson, Brian D, Peter B Gilbert, Noah R Simon, and Marco Carone. 2023. “A General Framework for Inference on Algorithm-Agnostic Variable Importance.” *Journal of the American Statistical Association* 118 (543): 1645–58.
- Wright, Marvin N, and Andreas Ziegler. 2015. “Ranger: A Fast Implementation of Random Forests for High Dimensional Data in c++ and r.” *arXiv Preprint arXiv:1508.04409*.
- Wright, Marvin N, Andreas Ziegler, and Inke R König. 2016. “Do Little Interactions Get Lost in Dark Random Forests?” *BMC Bioinformatics* 17: 1–10.
- Zaccour, Georges et al. 1988. “Valeur de Shapley Et Partage équitable Des Ressources.” *L'Actualité économique* 64 (1): 96–121.