Projet de fin d'études

Réalisé par :
Said Samer

Encadré par :

Année universitaire 2018/2019

Mention de confidentialité

Ce document est confidentiel. Il ne peut pas donc être communiqué à l'extérieur.

Remerciement

Je voudrais exprimer ma gratitude envers Didier M'TAMON, le responsable d'équipe, pour m'avoir laissé travailler de façon autonome, ce qui m'a permis d'améliorer ma maîtrise des moyens de recherche documentaire et de sources de données.

Une reconnaissance particulière à mes deux tuteurs, François Morois et Eléonore Moubachir, pour m'avoir inclus dans leur équipe pour ce projet de fin d'études et pour leur disponibilité à m'aider chaque fois que j' en'avais besoin ou me conseiller dans mon projet. Enfin, je voudrais remercier toute l'équipe du stage, en particulier Achraf Seddik, présent pour répondre à mes questions et me donner des suggestions concernant mon travail. C'est une excellente équipe.

Je suis aussi très chanceux d'avoir M.Francesco Russo et M.Christian Francq comme mes enseignants référents, dont la porte était toujours ouverte et qui étaient toujours patients avec moi.

Resumé

Lors de mon stage, j'ai intégré l'équipe « Modèles Quantitatifs de Portefeuille» au sein de la direction Risk and Permanent Control (RPC) chez Crédit agricole CIB. Dans ce cadre, j'ai contribué à l'automatisation, la fiabilisation et la documentation de la procédure d'estimation actuelle des LGD sur les financements d'actifs maritimes, la construction de nouveaux modèles de projection des valeurs d'actifs maritimes dont les modèles existants sont de performance faible et finalement l'amélioration du temps d'exécution dans l'estimation des valeurs d'actifs via l'implémentation de techniques avancées de machine learning. Il s'agit de réaliser des études sur des sujets quantitatifs, documenter les travaux réalisés, maintenir et améliorer les outils de calcul.

L'objet de ce mémoire consiste à améliorer les modèles existants pour le pricing d'actifs maritimes basés sur des statistiques avancées et des techniques de machine learning, Développer la librairie des modèles existante en R (Automatisation des calculs réalisés...) et implémenter de nouvelles techniques de machine learning (comme la régression lasso, la forét aléatoire, etc.) afin d'élever la performance des modèles existants.

Abstract

During my internship, I joined the team "Quantitative Portfolio Models" within the Risk and Permanent Control (RPC) department at Crédit Agricole CIB. In this context, I have contributed to the automation, reliability and documentation of the current estimation procedure of LGDs on the financing of maritime assets, the construction of new projection models of maritime asset values whose existing models are of poor performance and ultimately the improvement of the execution time in estimating asset values through the implementation of advanced machine learning techniques. It involves carrying out studies on quantitative subjects, documenting the work done, maintaining and improving the calculation tools.

The purpose of this thesis is to improve the existing models for the pricing of maritime assets based on advanced statistics and machine learning techniques, to develop the existing model library in R (automation of calculations performed ...) and to implement new machine learning techniques (like lasso regression, random forest, etc.) to raise the performance of existing models.

Table des matières

Introduction

En mai 2004, **Crédit Agricole Corporate and Investment Bank** naît du rapprochement de la Banque de Financement et d'Investissement du Crédit Lyonnais et de Crédit Agricole Indosuez. Actrice internationale des marchés financiers mondiaux, elle développe sa gamme de produits et ses services aux grandes entreprises. Ses activités s'articulent autour de quatre pôles majeurs : Coverage and Investment Banking pour le suivi et le développement des clients entreprises et des institutions financières, en France ou à l'étranger, Equity Brokerage and Derivatives pour le courtage actions en Europe, en Asie et aux Etats-Unis ainsi que pour les activités de trading et dérivés sur actions et fonds, Fixed Income Markets pour l'ensemble des activités de trading et de vente de produits de marché standards ou structurés, Structured Finance pour le métier de financement structuré des opérations de grande exportation et d'investissement, reposant sur des garanties sécurisées (avions, bateaux ...) ou encore des crédits complexes et structurés. A cette structure vient s'ajouter un ensemble de fonctions support.

La Direction RPC, Risk and Permanent Control, est précisément une fonction support de Crédit Agricole CIB. Elle assure le contrôle de l'ensemble des risques du groupe et de la Banque Privée, afin de contrôler le risque des différents métiers, au titre des risques de contrepartie, des risques de marché, des risques pays ou de portefeuille, et des risques opérationnels. Rattachée à la Direction Générale du groupe, elle exerce ses missions dans le cadre de la ligne métier "Risques" du Groupe CA s.a.. Ce département RPC ont les missions principales comme suites :

- Modélisation du risque de crédit du portefeuille bancaire sous l'approche économique (capital économique, risque de concentration)
- Stress-tests « risque de crédit » du portefeuille bancaire de CACIB et stress-tests ad hoc
- Modélisation et calcul des provisions collectives (IFRS 9) au titre du risque de crédit du portefeuille bancaire
- Définition et production des indicateurs de risque issus des modèles développés au sein de l'unité

Mon travail au sein de Crédit Agricole CIB s'est articulé autour de la norme IFRS9, la nouvelle norme internationale d'information financière, qui apporte des changements fondamentaux aux modèles de risque, notamment le modèle de Expected Credit Loss (ECL), construit de la probabilité de défaut (PD) et du taux de perte en cas de défaut (LGD), dans lesquels j'ai plongé pendant ce stage. Mes **missions de stage** se sont articulées autour de la modélisation des LGD sur les financements maritimes :

- Amélioration, Automatisation, fiabilisation et documentation de la procédure d'estimation actuelle des valeurs d'actifs maritimes
- Construction de nouveaux modèles de projection des valeurs d'actifs maritimes dont les modèles existants sont de performance faible
- Amélioration du temps d'exécution dans l'estimation des valeurs d'actifs via l'implémentation de techniques avancées de machine learning

Chapitre 1

La norme IFRS9

IFRS 9 est une norme internationale d'information financière (IFRS) publiée par l'International Accounting Standards Board (IASB). Elle traite de la comptabilisation des instruments financiers. Elle contient trois thèmes principaux : le classement et l'évaluation des instruments financiers, la dépréciation des actifs financiers et la comptabilité de couverture. La norme est entrée en vigueur le 1er janvier 2018, remplaçant l'ancienne IFRS pour les instruments financiers, IAS 39.

I Comptabilisation des dépréciations selon IFRS 9

L'une des principales lacunes d'IAS 39 était la génération d'écarts entre les provisions effectives et les pertes attendues. Celle-ci provenait du fait que seuls les instruments, dont la dégradation du risque associé était avérée, étaient soumis à un provisionnement. Sous IFRS 9, il sera nécessaire de provisionner pour les produits éligibles dès l'origination ou l'instant initial de détention. Le montant des provisions sera égal aux pertes moyennes attendues, nommées ECL (pour Expected Credit Loss). Par perte de crédit, nous entendons les pertes futures c'est-à-dire que la norme adopte un point de vue prospectif (forward). En outre, tous les produits ne seront pas soumis à la même échéance de calcul. Cette maturité sera déterminée par la significativité de la détérioration du risque de crédit constatée sur la contrepartie à partir de sa date d'achat ou d'origination. En pratique, la norme exige qu'une segmentation en deux paniers (buckets) soit réalisée suite à une quantification de la dégradation du risque, dont la mise en place du processus est à la charge de l'institution financière et peut reposer sur le système de notation interne (Internal Rating Based, IRB).

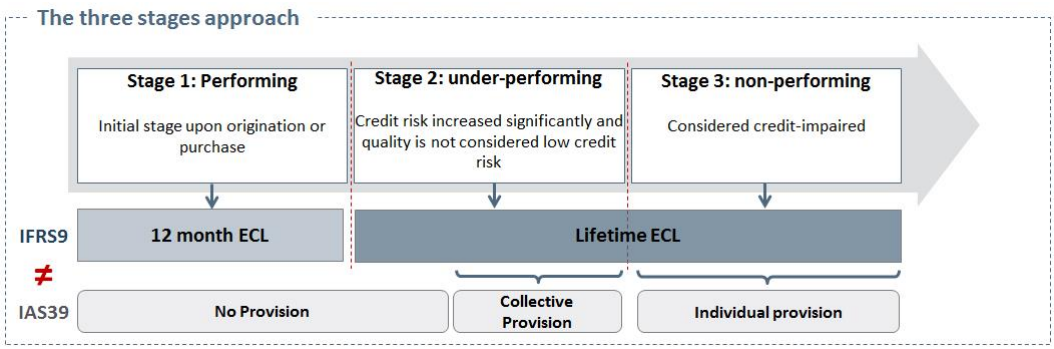


FIGURE 1.1 – Différence entre IFRS9 et IAS39 en termes de comptabilisation des ECLs [1].

II Trois étapes de la dégradation

La dépréciation des prêts est comptabilisée sur le plan individuel ou collectif en trois buckets selon IFRS 9 :

- **Bucket 1 :** : Lorsqu'un prêt est contracté, les ECLs provenant d'événements de défaut possibles au cours des 12 prochains mois sont comptabilisés (ECL 12 mois) et une provision pour pertes est établie. Lors des dates de reporting ultérieures, la comptabilisation des dépréciations pour 12 mois est applicable également aux prêts existants sans la présence significative du risque de crédit depuis leur comptabilisation initiale.
  Dans le but de déterminer si une augmentation significative du risque de crédit est survenue depuis la comptabilisation initiale, une banque doit évaluer la variation éventuelle du risque de faire défaut sur la durée de vie attendue du prêt (c'est-à-dire la modification de la probabilité de défaillance, par opposition à la quantité d'ECLs).
- **Bucket 2 :** Si le risque de crédit a significativement augmenté depuis la comptabilisation initiale mais on ne peut pas le considérer faible, on se dirige vers les ECLs à vie.
- **Bucket 3 :** Ce bucket concerne les actifs en défaut, dont la méthodologie de calcul des pertes est inchangée entre l'ancienne et la nouvelle norme comptable.

III Pertes de crédit attendues sur 12 mois et sur la durée de vie attendue

$$ECL = \int_{t_0}^{M} [\sum_{k=k_t}^{k_T} CF_k].LGD(t).D(t).dPD(t)$$

- M=1ans si on est dans bucket 1 et M=maturité si on est dans Bucket 2
- CF : Cash Flows at risk on/off BS (principal and interest)
- PD (t) : La probabilité de défaut [2] (PIT and Fwd Looking)
- LGD (t) : La perte encourue en cas de défaut de la part d'une contrepartie (PIT and Fwd Looking)
- D : Le facteur d'actualisation (Effective Interest Rate, or proxy)

Cette norme apporte des changements fondamentaux aux modèles de risque, notamment le modèle de l'Expected Credit Loss (ECL), construit de la Probabilité de défaut (PD) et de la Loss Given Default (LGD).

1. Dans le rapport, toutes les figures qui ne sont pas accompagnées de références bibliographiques sont faites par l'équipe du stage.
2. la probabilité qu'un débiteur ne puisse faire face à ses obligations de remboursement.

Chapitre 2

Modélisation des LGD des financements d'actifs Maritime

I Présentation du portefeuille Maritime

Le portefeuille des financements d'actifs Maritime est segmenté selon plusieurs axes, notamment :

- le type de navire
- le sous-type de navire

Le type de navire dépend de la nature des marchandises transportées, il en existe trois principales :

- les marchandises sèches en vrac : transportées par des vraquiers (appelés aussi Bulkers)
- les marchandises humides en vrac : transportées par des pétroliers (appelés aussi Tankers)
- les conteneurs : transportés par des porte-conteneurs (appelés aussi Containerships)

Pour chaque type de navire, il existe plusieurs sous-types de navires, déterminés par la capacité de charge. Pour la plupart des navires, la capacité de charge est exprimée en « Dead Weight Tons » (DWT), littéralement tonnes de poids mort, c'est-à-dire la masse d'eau de mer déplacée par le volume du navire lorsqu'il est complètement chargé. Les types de navires et leurs sous-types sont présentés dans le tableau ci-dessous :

Type d'actif	Sous-type d'actif
BULKERS	HANDYSIZE
BULKERS	HANDYMAX / SUPRAMAX
BULKERS	PANAMAX / POST PANAMAX
BULKERS	CAPESIZE & VLOC
CONTAINER SHIPS	HANDY
CONTAINER SHIPS	SUB PANAMAX
CONTAINER SHIPS	PANAMAX
CONTAINER SHIPS	POST PANAMAX <10 000 TEU
CONTAINER SHIPS	POST PANAMAX >= 10 000 TEU
TANKERS	PANAMAX
TANKERS	AFRAMAX
TANKERS	SUEZMAX
TANKERS	VLCC & ULCC

II Contexte des modèles

Comme mentionné dans le premier chapitre, la nouvelle norme de provisionnement IFRS 9 impose, lorsque le prêt est au stade 2, un calcul d'ECL à maturité. Cette étape exige la modélisation du taux de perte en cas de défaut (LGD), autour duquel mon travail s'est articulé. On a :

$$E[LGD_t|X] = \frac{E[max(Exposition_t - Prix_t, 0)|X]}{Exposition_t}$$

où X est l'ensemble de l'état de macro-économique. Elle est approximée par

$$E[LGD|X] \approx \frac{max(Exposition - E[Prix|X], 0)}{Exposition}$$

Du coup, la modélisation du paramètre de la LGD Shipping sur l'horizon de la maturité résiduelle de l'opération revient, à tout instant entre la date de calcul et la date de maturité, à comparer la valeur de l'actif au capital restant dû auquel sont ajoutés des intérêts. La modélisation de la LGD revient ainsi à modéliser la valeur d'actif du navire au cours du temps.

II.A Approche retenue

L'approche retenue consiste à modéliser le prix de l'actif à partir des historiques de prix disponibles. Il s'agit de trouver la relation statistique qui lie le prix d'un actif à des variables macroéconomiques.

La représentation par sous-type de navire du niveau des prix pour les âges 5 ans, 10 ans, 15 ans, 20 ans et/ou Scrap, permet de constater que pour un âge inférieur au scrap, les prix évoluent de façon très corrélées au cours du temps. Cela signifie que tous ces prix dépendent d'une combinaison de facteurs propres à chaque segment qui permet de déterminer l'évolution du cycle économique de chaque segment. On parle alors d'effet économique qui sera mesuré indépendamment de l'âge de chaque navire.

A chaque date, les prix des navires sont décroissants en fonction de l'âge de ces derniers. Il s'agit de la dépréciation de l'actif qui est plus ou moins importante en fonction de la période

considérée. Enfin, il convient de noter que les valeurs au scrap (SV) des navires constituent un plancher pour les prix quel que soit l'âge.

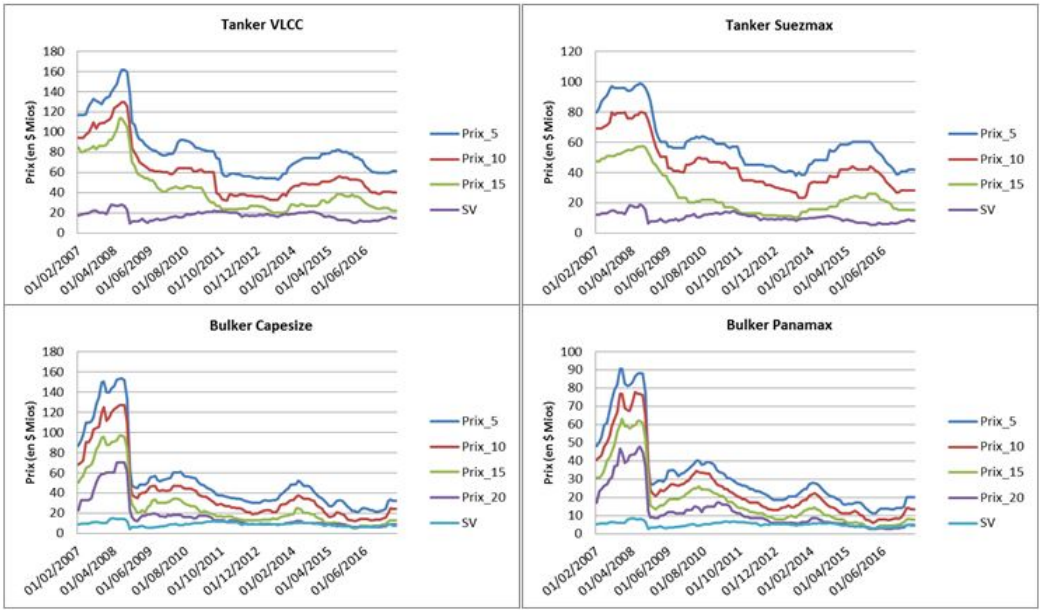


FIGURE 2.1 – L'évolution de prix de quelques navires de différents âges en fonction du temps.

Voir l'annexe A pour plus de figures sur l'évolution de prix d'autres navires.

Ainsi, à partir d'un prix à une date donnée (date d'arrêté de calcul des dépréciations), la projection future du prix doit tenir compte de :

- La dépréciation de l'actif liée au vieillissement de celui-ci (effet âge)
- La baisse de prix du navire engendrée par la conjoncture économique (effet économique)
- La limitation de la baisse de valeur de l'actif à la valeur ferraille (Scrap Value) [1]

L'approche de modélisation se décompose donc en 3 étapes :

- La modélisation de l'évolution du prix 5 ans représentant l'effet économique
- La modélisation de l'évolution ou du niveau des déviations des prix 10 ans, 15 ans, 20 ans par rapport au prix 5 ans représentant l'effet âge
- La modélisation de l'évolution de la valeur ferraille (Scrap Value), représentant la valeur plancher des valeurs d'actifs modélisées au-delà de 20 ans

1. La projection future du prix de l'actif va également tenir compte des coûts de repossession et décote de vente forcée, c'est-à-dire des coûts et décotes liés à la vente forcée de l'actif en cas de défaut de la contrepartie.

III La démarche de construction du modèle

III.A Construction de la base de calibrage

III.A.1 Historique, fréquence et source de données

III.A.1.1 Données de prix des navires

Bloomberg a permis de constituer un historique de prix des navires sur les tankers et les bulkers sur la période 2005 à 2018 inclus, issus du courtier Simpson Spence Young (SSY). Les prix sont fournis par :

- Type (Tankers, Bulkers)
- Sous-type (Aframax, Suezmax, VLCC, Capesize, Panamax, Handymax, Handysize)
- Age (5 ans, 10 ans, 15 ans)

Les prix 20 ans et/ou scrap des tankers et bulkers ont été récupérés via le courtier Clarksons. Les prix 5 ans, 10 ans, 15 ans et scrap des Tankers Panamax et Containerships ont également été récupérés via le courtier Clarksons.

Les sources des données de prix des navires sont ainsi récapitulées :

Source	Variables
SSY	Prix 5 ans : Tanker Aframax, Tanker Suezmax, Tanker VLCC, Bulker Panamax, Bulker Capesize, Bulker Handymax, Bulker Handysize Prix 10 ans : Tanker Aframax, Tanker Suezmax, Tanker VLCC, Bulker Panamax, Bulker Capesize, Bulker Handymax, Bulker Handysize Prix 15 ans : Tanker Aframax, Tanker Suezmax, Tanker VLCC, Bulker Panamax, Bulker Capesize, Bulker Handymax, Bulker Handysize
Clarksons	Scrap : Tanker Aframax, Tanker Suezmax, Tanker VLCC, Tanker Panamax, Bulker Panamax, Bulker Capesize, Bulker Handymax, Bulker Handysize, ContainerShips Prix 5 ans : Tanker Panamax, ContainerShips Prix 10 ans : Tanker Panamax, ContainerShips Prix 15 ans : Tanker Panamax, ContainerShips Prix 20 ans : Bulker Panamax, Bulker Capesize, Bulker Handymax, Bulker Handysize

FIGURE 2.2 – Récapitulation des sources des données de prix des navires.

La fréquence des prix retenue pour le calibrage est trimestrielle.

A partir des séries de prix récupérées, des données génériques par type ont été créées en effectuant la moyenne des rendements de prix des sous-types du type considéré afin de couvrir les autres sous types existant dans le portefeuille maritime dont on n'a pas des données :

- Bulker générique : la moyenne des rendements de prix des Bulkers Capesize, Bulkers Panamax, Bulkers Handymax et Bulkers Handysize permet de couvrir le reste des actifs Bulkers du portefeuille de CACIB
- Tanker générique : la moyenne des rendements de prix des Tankers Aframax, Tankers Suezmax, Tankers VLCC et Tankers Panamax permet de couvrir le reste des actifs Tankers du portefeuille de CACIB

III.A.1.2 Critères explicatifs

Les critères retenus pour le calibrage du modèle caractérisent l'offre et la demande du secteur Maritime. Voici la liste non exhaustive des indicateurs retenus :

Indicateurs macro-économiques	**Indicateurs spécifiques au Maritime**	
	Offre	**Demande**
Croissance du PIB Chine	Carnet de commande (en % flotte en service)	Exportation minerai de Fer Australie vers Chine
Croissance du PIB US	Flotte en service (en effectif et en capacité)	Exportation minerai de Fer Brésil
Croissance du PIB Europe	Part des Navires lancés dans le carnet de commande (en capacité)	Importation Charbon Chine
Production industrielle Chine	Part des Navires construits et lancés dans le carnet de commande (en capacité)	Importation Minerai de fer Chine
Production industrielle US	Navires en démolition (en capacité)	Importations de pétrole Chine
Prix du pétrole brut Brent	Baltic Index (Tanker, Handysize, Panamax, Supramax)	Importations de charbon Japon
Demande mondiale de pétrole brut	Taux de frêt des exportations chinoises conteneurisées	Production mondiale d'Acier

La fréquence des critères retenus pour le calibrage est trimestrielle.

III.A.2 La transformation des variables

Une fois les variables brutes extraites de Bloomberg, plusieurs transformations sont appliquées sur celles-ci.

III.A.2.1 Variables à expliquer

Conformément à l'approche méthodologique développée dans le paragraphe II.A, pour chaque Type x Sous-type :

- L'effet économique est modélisé par l'évolution dans le temps du rendement du prix 5 ans
- L'effet âge est modélisé par la déviation des prix 10 ans et 15 ans (voire 20 ans pour les Bulker) par rapport au prix 5 ans
- La modélisation de l'évolution dans le temps du rendement du prix Scrap de chaque sous-type permet d'obtenir une valeur plancher des projections des valeurs d'actifs

Pour chaque Type x Sous-type, les grandeurs « Rendement Prix 5 ans », « Pente 10 ans », « Rdt Pente 10 ans », « Pente 15 ans », « Rdt Pente 15 ans », « Pente 20 ans », « Rdt Pente 20 ans » « Rdt Scrap » sont calculées à partir des prix bruts extraits. Les formules de calcul sont les suivantes :

- $Rdt\ Prix\ 5ans_t = \ln\left(\frac{Prix\ 5ans_t}{Prix\ 5ans_{t-4}}\right)$
- $Pente\ 10ans_t = \frac{(Prix\ 10ans_t - Prix\ 5ans_t)}{Prix\ 5ans_t}$
- $Rdt\ Pente\ 10ans_t = \ln\left(\frac{Pente\ 10ans_t}{Pente\ 10ans_{t-4}}\right)$

- $Pente\ 15ans_t = \frac{(Prix\ 15ans_t - Prix\ 5ans_t)}{Prix\ 5ans_t}$
- $Rdt\ Pente\ 15ans_t = \ln\left(\frac{Pente\ 15ans_t}{Pente\ 15ans_{t-4}}\right)$
- $Pente\ 20ans_t = \frac{(Prix\ 20ans_t - Prix\ 5ans_t)}{Prix\ 5ans_t}$
- $Rdt\ Pente\ 20ans_t = \ln\left(\frac{Pente\ 20ans_t}{Pente\ 20ans_{t-4}}\right)$
- $Rdt\ Scrap_t = \ln\left(\frac{Scrap_t}{Scrap_{t-4}}\right)$

Il y a donc 75 variables à expliquer, dont :

- 40 sur les bulkers
- 30 sur les tankers
- 5 sur les porte-conteneurs

III.A.2.2 Variables explicatives

Les transformations appliquées aux variables explicatives sont les suivantes :

- L0.lvl.ticker.Index : variable initiale
- L0.rdt.ticker.Index : log-rendement de la variable initiale
- L0.diff.ticker.Index : différence de la variable initiale ($Variable_t - Variable_{t-4}$)
- L1.lvl.ticker.Index : variable initiale retardée d'un an
- L1.rdt.ticker.Index : log-rendement de la variable initiale retardée d'un an
- L1.diff.ticker.Index : différence de la variable initiale retardée d'un an
- L2.lvl.ticker.Index : variable initiale retardée de deux ans
- L2.rdt.ticker.Index : log-rendement de la variable initiale retardée de deux ans
- L2.diff.ticker.Index : différence de la variable initiale retardée de deux ans

De plus, pour chaque type x sous-type, des variables d'offre représentant l'état de la flotte sont également extraites. Des transformations et des calculs de nouvelles variables à partir ces variables brutes de flotte sont également opérées.
Par exemple, pour le sous-type des Tanker VLCC, les variables de flotte ci-dessous sont extraites (avec leur ticker Bloomberg et définition associés) :

Ticker Bloomberg	Libellé	Description
VESLVLIS Index	VLCC Tanker In Service	Nombre de tankers VLCC en service
VESLVLOO Index	VLCC Tanker on order	Nombre de tankers VLCC commandés, dont la construction n'a pas débuté et qui seront livrés au-delà d'1 an ou annulés
VESLVLUC Index	VLCC Tanker under construction	Nombre de tankers VLCC commandés, dont la construction a débuté, et qui seront livrés d'ici 1 an
VESLVLLA Index	VLCC Tanker launched	Nombre de tankers VLCC dont la livraison est imminente (dans les 6 mois)
VESLVLPC Index	VLCC Tanker orderbook as a % of DWT	Carnet de commande exprimé en % de la capacité de la flotte actuelle
VESLVLBU Index	VLCC Tanker broken up	Nombre de tankers VLCC démolis
VESLVLDW Index	VLCC Tanker Total Capacity	Capacité totale de la flotte en service des tankers VLCC

A partir de ces variables, des variables supplémentaires sont calculées avec leur formule associée :

Ticker Bloomberg	Variables	Description
Calculated variable	VLCC Tanker broken up*(VLCC Tanker Total Capacity / VLCC Tanker In Service)	Tanker_VLCC_Demolition
Calculated variable	VLCC Tanker on order+ VLCC Tanker under construction + VLCC Tanker launched	Tanker_VLCC_CarnetOrdre
Calculated variable	VLCC Tanker launched/(VLCC Tanker on order + VLCC Tanker under construction + VLCC Tanker launched)	Tanker_VLCC_PartLances_CarnetOrdre
Calculated variable	(VLCC Tanker under construction + VLCC Tanker launched)/(VLCC Tanker on order + VLCC Tanker under construction + VLCC Tanker launched)	Tanker_VLCC_PartLancesConstruction_CarnetOrdre

Ce calcul de variables complémentaires de flotte est réalisé sur l'ensemble des segments modélisés (4 sous-types Tankers, 4 sous-types Bulkers, le ContainerShips et les deux sous-types génériques de Tanker et Bulker). Le sous type générique représente la moyenne entre les valeurs des autres sous type de même type. Par exemple :

$$L0.rdt.Tanker\_Generique\_5Y = \frac{L0.rdt.Tanker\_Aframax\_5Y + L0.rdt.Tanker\_Panamax\_5Y + L0.rdt.Tanker\_Suezmax\_5Y + L0.rdt.Tanker\_VLCC\_5Y}{4}$$

III.B Procédure de calibrage

Pour un sous-type de navire, les grandeurs suivantes sont modélisées : Rendement Prix 5 ans, Pente 10 ans, Rendement Pente 10 ans, Pente 15 ans, Rendement Pente 15 ans et Rendement Prix Scrap. A noter que pour les sous-types de Bulkers, les grandeurs supplémentaires de la Pente 20 ans et du Rendement Pente 20 ans sont également modélisés.

Deux jeux de données distincts ont été constitués pour la calibration :

- 1 jeu contenant les variables explicatives sous la forme des transformations décrites dans la section précédente A.3 en ne considérant pas les variables retardées de 2 ans (retard d'1 an au plus) : jeu L1
- 1 jeu contenant les variables explicatives sous la forme des transformations décrites dans la section précédente A.3 en considérant les variables retardées de 2 ans (retard de 2 ans au plus) : jeu L2

Pour chacun de ces jeux de données, pour chaque sous-type, pour chaque grandeur citée ci-dessus, toutes les combinaisons de régressions linéaires à 1, 2, 3 et 4 variables sont testées. Les combinaisons respectant certaines propriétés statistiques sont ensuite agrégées selon la méthode d'agrégation appelée « Stacking ». Cette méthode recherche l'approximation $\hat{F}(X)$ de variable à expliquer sous forme de somme pondérée de fonctions basiques h_i(X), représentant des régressions linéaires à 1, 2, 3 ou 4 variables, acceptées par les tests économiques et statistiques.

$$\hat{F}(x) = \sum_{i=1}^{M} \gamma_i h_i(x) \mathbb{1}_{\{Accepted=TRUE\}} \tag{2.1}$$

III.B.1 Critère de sélection des modèles satellites

Comme évoqué précédemment, pour chaque sous-type, pour chaque grandeur à modéliser, toutes les combinaisons de régressions linéaires à 1, 2, 3 et 4 variables avec constante sont testées. Sur le 1er jeu de données cela représente :

- 112 variables candidates pour les bulkers (6Mios de combinaisons)

- 78 variables candidates pour les tankers (2Mios de combinaisons)
- 68 variables candidates pour les porte-conteneurs (1Mios de combinaisons)

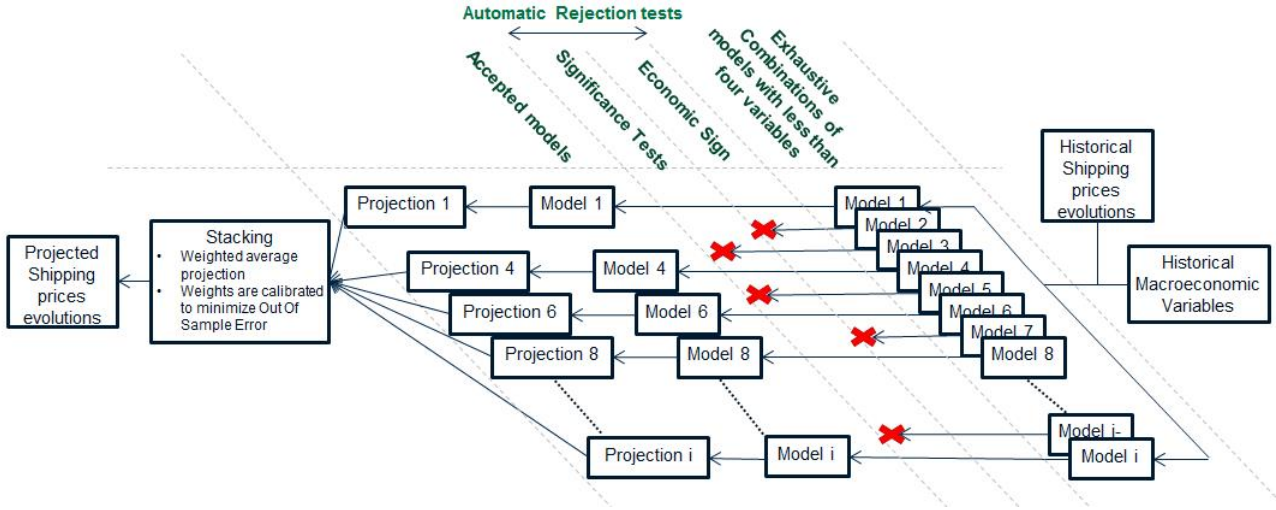


FIGURE 2.3 – Schéma récapitulatif de la procédure de sélection de modèles satellites.

La sélection de ces sous-modèles satellites se fait selon deux critères :

III.B.1.1 Critère économique

Avant d'aborder la procédure de calibrage, l'équipe CASA/ECO/DIS en charge du secteur Maritime intervient pour définir des corrélations a priori entre les variables à expliquer et les variables explicatives. Voir l'exemple sur Bulker Panamax dans l'Annexe A. D'après le tableau de corrélations attendues de l'annexe, on peut faire l'interprétation suivante :

- Un signe égal à 1 signifie qu'un coefficient positif de régression de cette variable est attendu
- Un signe égal à -1 signifie qu'un coefficient négatif de régression de cette variable est attendu

Les modèles de régression linéaire présentant ces variables avec un coefficient de signe opposé à celui du tableau sont exclus de l'étude et ne sont pas considérés dans le processus d'agrégation des modèles. Lorsque il y a une variable avec un signe pas précisé par CASA, on doit exclure cette variable des variables explicatives avant le lancement de la recherche exhaustives de tous les modèles linéaires [2].

III.B.1.2 Critères statistiques

- Stationnarité des résidus : **Test de Dickey Fuller augmenté**
- Indépendance des résidus : **Test de Box**
- Normalité des résidus du modèle satellite : **Test de Jarque-Bera**
- Homoscédasticité : Test de Goldfeld Quant, **Test de Studentized Breusch Pagan**
- Tendance et stabilité des coefficients : **Test de Cusum**
- Multicolinéarité : **Variance Inflation Factor (VIF)**

2. Il n'y a pas une variable explicative dont le signe peut être négatif ou positif.

Et les tests statistiques pour tester la pertinence et la robustesse du modèle :

- Significativité globale du modèle satellite : **Test de Fisher**
- Significativité des coefficients du modèle satellite : **Test de Student**

Par défaut, Le niveau de confiance de ces tests statistiques est de 95%. Voir l'annexe A pour plus de détails sur les tests statistiques.

III.B.2 Agrégation entre les modèles satellites acceptés

Une fois les modèles satellites sélectionnés, on passe à la partie de l'agrégation. Parmi quelques méthodes testées, la méthode stacking a été retenue. Le stacking (ou dit parfois blending) est une technique très répandue en apprentissage ensembliste [3] qui combine plusieurs modèles de classification ou de régression par un autre algorithme de Machine Learning.
D'une autre façon, on cherche à pondérer les modèles satellites d'une manière permettant d'avoir un modèle agrégé final ayant l'erreur de prédiction la plus minimale. On dispose de la matrice A de prédictions des modèles satellites sur la base de données. Il s'agit d'une matrice de dimensions n x p, où n est le nombre d'observations dans la base de données et p est le nombre de modèles satellites acceptés. Chaque ligne de la matrice A est représentée par les prédictions par la méthode de la validation croisée k-fold de chaque modèle accepté.

Soit B le vecteur des valeurs historiques de la variable à expliquer et il est de dimension n, x est le vecteur à trouver de poids des modèles satellites, il est de taille p (nombre de modèles satellites acceptés). Le problème de stacking est modélisé par le problème d'optimisation avec égalités et inégalités suivant :

$$\begin{cases} \underset{x\in\mathbb{R}^p}{argmin}||Ax-B||^2 \\ Ex=1 \\ Gx\geq 0 \end{cases} \tag{2.2}$$

Où E est le vecteur de taille p constitué des 1 pour s'assurer que la somme des coefficients de stacking est égale à 1 et G est une matrice diagonale de taille p x p ayant des 1 sur la diagonale (matrice unité) afin de considérer les x avec des coefficients positifs seulement.

Cette méthode est caractérisée par la bonne performance, la faible sensibilité à l'ajout ou au retrait d'une variable explicative.

III.B.3 Mesure de performance du modèle

Pour chaque sous-type, pour chaque grandeur modélisée, la performance du modèle agrégé est évaluée selon plusieurs critères, estimés de deux façons :

- In-sample : sur l'échantillon d'apprentissage
- Out of sample : par la méthode de validation croisée servant à découper l'historique en fenêtres de 3 années consécutives et pour chacune des fenêtres :
  - ◊ Les coefficients du modèle sont estimés sur tout l'historique sauf sur la fenêtre considérée
  - ◊ Une prédiction du modèle est réalisée sur la fenêtre considérée

3. En statistique et en Machine Learning, l'apprentissage ensembliste combine plusieurs algorithmes d'apprentissage par une méthode précise (bagging, moyenne arithmétique, etc.) afin d'obtenir de meilleures prédictions qu'avec un seul algorithme.

◊ L'opération est répétée autant de fois qu'il existe de fenêtres composant l'historique complet d'estimation.
◊ La concaténation de ces estimations correspond à l'estimation out of sample de la grandeur réalisée

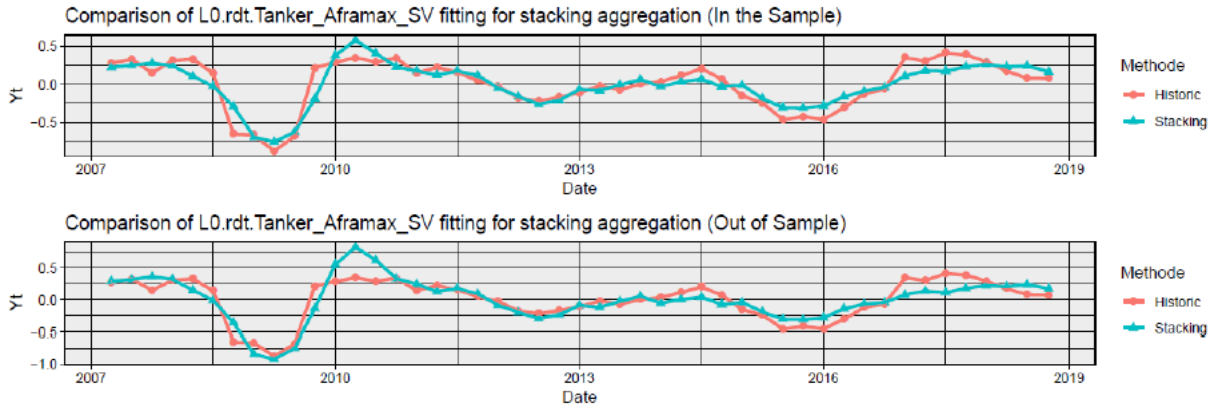


FIGURE 2.4 – Le graphe d'estimation In sample et Out of sample par le modèle stacking L0.rdt.Tanker_Aframax_SV comparé à celui de l'historique.

Les critères de performance sont le pouvoir explicatif du modèle évalué avec les R^2 In sample et le R^2 Out of sample (sur un échantillon test). Pour rappel la formule du R^2 est la suivante :

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{2.3}$$

- y_i : les réalisations historiques de la variable à expliquer
- $\hat{y}_i$: les estimationss de la variable à expliquer
- $\bar{y}$: la moyenne des réalisations historiques de la variable à expliquer

III.B.4 Choix final du modèle

Il y a deux phases de sélection dont la première est considérée par tous les modèles agrégés afin de trouver le modèle le plus pertinent, calibré selon l'un des jeux de données L1 ou L2, en fonction :

- des critères de performance R^2 in et out of sample énoncés précédemment
- du graphe de l'estimation par le modèle comparé à celui de l'historique
- du nombre de modèles satellite sélectionnés
- des R^2 des 10 modèles satellites les plus contributeurs en termes de poids dans l'agrégation stacking
- de la pertinence des 5 variables les plus contributrices au modèle agrégé

La contribution de chaque variable au modèle agrégé a été calculée selon le principe de la **"permutation feature importance"**, qui consiste à opposer les performances du modèle en prédiction in sample avec et sans la variable à évaluer, le critère de performance étant le taux d'erreur. Pour neutraliser la variable, celle-ci est remplacée par une constante correspondant à chacune des valeurs prises par cette variable dans l'échantillon. L'opération est ainsi réitérée autant de fois qu'il y a d'observations. La contribution de la variable correspond à la variation relative du taux d'erreur après permutation de l'échantillon par rapport au taux d'erreur initial du modèle.

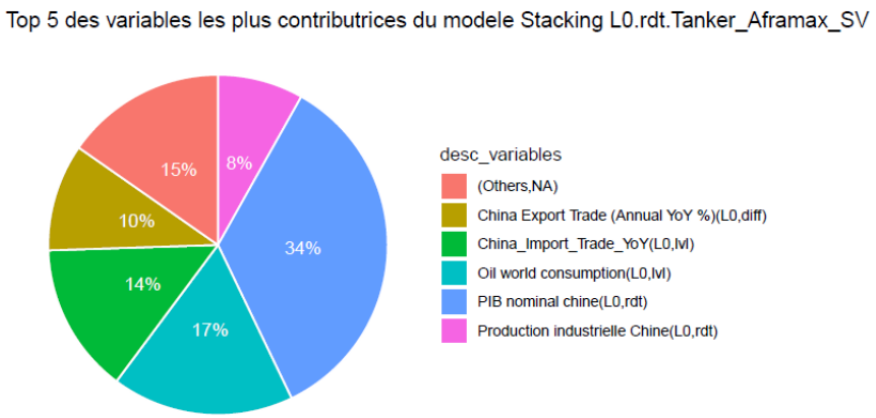


FIGURE 2.5 – Les variables les plus contributrices en termes d'importance dans le modèle L0.lvl.pente.Bulker_Panamax_20Y.

Avant de retenir le jeu de données L2, même si les critères précédents sont vérifiés, on doit s'assurer que la contribution des variables retardées de 2 ans était limitée (doit être normalement inférieure à 20%), pour assurer **le caractère prédictif** du modèle.

Une fois le jeu de donnée choisi, une deuxième sélection est faite pour les modélisations de pentes. En effet, pour un sous-type, pour un type de pente, deux modèles sont estimés : le rendement de pente et la pente. Le choix a été effectué en fonction des mêmes critères. Voir un exemple détaillé de modélisation dans l'annexe A.

III.C Calcul des projections

III.C.1 La procédure du calcul des projections

Une fois les modèles calibrés, des projections trimestrielles sur 3 ans sont réalisées. Les valeurs d'actifs sont projetées en tenant compte des scénarios sur 3 ans définis par les économistes du département Casa Eco.

Pour produire les projections des différents indicateurs modélisés, les modèles décrits dans la partie précédente sont appliqués sur une table de projection. Cette table de projection est un mix réalisé entre les prédictions produites par CASA ECO et les réalisations historiques des autres variables sur 3 années glissantes ; ces derniers s'appellent variables « historiques » et correspondent au complémentaire des variables CASA ECO sur l'ensemble global de variables explicatives.

Ainsi, selon la nature de la variable (spot, retardée, en niveau ou en variation) et sa présence ou non dans les scénarios de Casa ECO, la construction d'une projection des variables sur 3 ans est constituée de la manière suivante :

	Niveau			Variation		
	Spot	**Retardée d'un an**	**Retardée de 2 ans**	**Spot**	**Retardée d'un an**	**Retardée de 2 ans**
Prédite par ECO	Récupération de la prédiction fournie par ECO					
Non prédite par ECO	Recalcul du niveau spot à partir de la variation spot et du dernier niveau historique ayant servi au calibrage du modèle	Application de la fonction lag1 sur le niveau spot	Application de la fonction lag2 sur le niveau spot	Récupération de la variation historique sur 3 ans	Application de la fonction lag1 sur la variation spot	Application de la fonction lag2 sur la variation spot

III.C.2 Exemple d'usage du modèle

Voici, à titre illustratif, les prédictions sur les années 2019, 2020 et 2021 issues du modèle Bulker Panamax pente 20 ans sur la base des prédictions ECO des scénarios Baseline [4] et Adverse [5] pour les variables prédites par ECO et des scénarios historiques pour les variables non prédites par ECO. Pour chaque date on choisit le quantile à 25% de toutes les prédictions. Le graphe ci-dessous affiche les points de chaque année de prédiction, à la suite de l'historique de la pente 20 ans observée entre 2002 et 2018. A noter que les prédictions sont faites en utilisant le quantile à 25% au titre de la marge pour incertitude sur l'estimation des modèles.

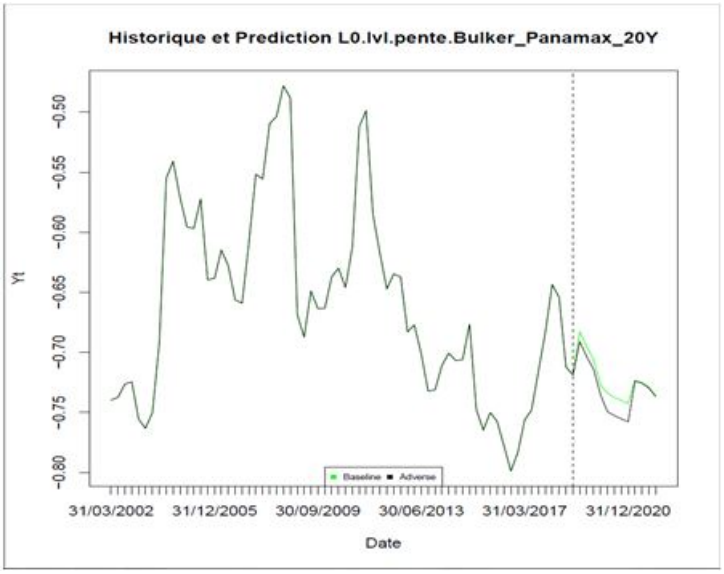


FIGURE 2.6 – Le graphe de réalisations historiques et de prédictions des variables L0.lvl.pente.Bulker_Panamax_20Y.

IV Bilan récapitulatif de tous les modèles

Comme on a vu dans la partie précédente, Il y a 75 variables à expliquer. Les importants détails en termes de complexité de calcul sont cités ci-dessous en fonction de type de navire :

4. La vision économique attendue
5. La vision économique pessimiste, en cas de dégradation éonomique par exemple

Type du navire	Jeu de données	Nombre de variables explicatives	Nombre de combinaisons	Temps de calcul	Nombre de variables à expliquer modélisées
Bulker	L1	112	∼6 Millions	1,16 hrs	40/40
	L2	168	∼32 Millions	∼12hrs	
Tanker	L1	78	∼2 Millions	16,3 mins	27/30
	L2	117	∼7 Millions	1.38 hrs	
Porte-conteneur	L1	68	∼1 Million	9,4 mins	5/5
	L2	102	∼4 Millions	46,8 mins	

D'après le tableau précèdent, on constate deux choses très importantes qui vont orienter plus notre projet vers la recherche de l'amélioration et la performance plus élevée :

- La grande complexité de calcul lors de la recherche exhaustive qui pose la question si on peut diminuer le nombre de combinaisons d'où la réduction de dimensions
- Il y a trois variables à expliquer dont notre procédure n'a pas trouvé encore de modèles, de type Tanker, qui sont L0.rdt.Tanker_Aframax_5Y, L0.rdt.Tanker _Panamax_5Y et L0.rdt.Tanker_Generique_5Y

Par conséquent, on doit trouver d'autres solutions alternatives servant à résoudre ces problèmes.

Chapitre 3

Optimisation de la procédure d'estimation

La modélisation décrite dans la partie précédente est réalisée sous le logiciel R. Plusieurs inputs ont été utilisés et plusieurs modules ont été implémentés. Le long de mon stage, j'ai développé et optimisé plusieurs modules, des fonctions permettant d'analyser la performance des modèles stacking obtenus, des fonctions de prédiction pour un scénario (Baseline ou Adverse) et des fonctions récapitulatives rassemblant les propriétés de chaque modèle satellite avec décision de sélection.

I Description générale du modèle

I.A Inputs

Les principaux inputs des programmes sont mentionnés ci-dessous :

- Deux fichiers comportant toutes les données provenant de Bloomberg et Clarksons, où figurent les prix
- Un fichier contenant les projections des variables macro-économiques que CASA Eco est en mesure de prédire
- Un fichier permettant de configurer et paramétrer la préparation de données avant de commencer le calibrage, il contient pour chaque donnée brute (prix, donnée macroéconomique, donnée de flotte) et recalculée (prix génériques, données de flotte) sa description, sa nature, son périmètre, son type et sous-type, transformations nécessaires et son type de prédiction (par CASA ou par l'approche historique).

I.B Modules

Des modules ont été déjà implémentés sous R pour les estimations et projections des modèles.

- Modules d'estimation des modèles : ce module, en programmation objet, prend en compte une classe « mère » (**cAverageModel**) et une classe « fille » (**cLinearModel**) qui hérite des propriétés de la classe « mère ».

  - ◊ Dans la classe **cLinearModel** sont implémentées les fonctions s'appliquant à un sous-modèle satellite. Cette classe est appelée lors de la recherche exhaustive lorsque toutes les combinaisons de modèles linéaires à 1, 2, 3 et 4 variables sont testées ; de plus, les fonctions d'estimation de modèles linéaires, les tests statistiques et les prédictions In et OutSample s'appliquant à un modèle linéaire sont implémentées dans cette classe.
  - ◊ Dans la classe **cAverageModel** sont implémentées les fonctions d'agrégation ; toute fonction s'appliquant à un niveau agrégé est programmée comme par exemple, le calcul des poids de la méthode d'agrégation Stacking ou les fonctions de prédiction s'appliquant à des modèles agrégés.

- Module regroupant des fonctions propres à la modélisation Shipping (**FunctionsForShippingModels.R**). Par exemple, l'import des données avec la possibilité de changement de leur fréquence, les fonctions de transformations de données brutes et le processus de prédictions détaillé dans la partie précédente (prédiction historique de variables explicatives, prédiction du modèle agrégé...).
- Module « **Main** » qui appelle :
  - ◊ Les fonctions d'estimation des modèles satellites
  - ◊ Les fonctions permettant d'analyser la performance des modèles stacking obtenus et retourner en sortie un fichier PDF par type de navire modélisé
  - ◊ Les fonctions de prédictions pour un scénario

I.C Outputs

On va lister les sorties principales de programme qu'on a construit, on peut distinguer :

- L'objet de la classe **cAverageModel**, le modèle final agrégé qu'on a obtenu après la procédure de calibrage, ce dernier comporte plusieurs attributs décrits dans l'annexe B
- Le fichier PDF récapitulatif du modèle agrégé final consistant à rassembler :
  - ◊ Toutes les propriétés statistiques (min p-value des tests statistiques ; minimum sur tous les modèles satellites acceptés, RMSE, R^2 In et R^2 Out)
  - ◊ Nombre de sous-modèles acceptés
  - ◊ Le Tableau **idAcceptedModels** décrit dans l'annexe B
  - ◊ Les variables les plus contributrices au modèle agrégé
  - ◊ Les graphes comparant l'historique et l'estimé (In the Sample et Out of Sample)
- Le fichier PDF de projection à 3 ans prochains contenant un graphe constitué de la partie historique de la variable à expliquer continuée par ses prédictions dans 3 ans (voir la figure 2.7 du chapitre précèdent)

I.D Tests de fiabilité et documentation de la procédure d'estimation

Puisque le modèle devrait être soumis à l'audit interne, on a dû :

- s'assurer que tout le code du modèle est bien documenté
- décrire les fonctions principales du modèle appelées dans **« Main »** (pour l'estimation et la prédiction)
  - — Lister tous les fonctions avec leurs retours
  - — Décrire les arguments de chaque fonction

— Indiquer s'il y a une connexion entre une fonction et une autre

A côté de la documentation, on a construit un ensemble de tests consistant à vérifier la robustesse des fonctions considérées devant les cas exceptionnels. En effet, on a implémenté les fonctions de base à tester avec Excel afin de comparer leurs résultats avec ceux de R. On a trouvé presque les même résultats mais ça nous a permis aussi de repérer quelques erreurs logiques qui nous a amené à les résoudre. Par exemple, on a testé l'algorithme de stacking servant à résoudre le problème d'optimisation (2.2) du chapitre 2, où on a montré qu'on peut trouver une autre approximation de la solution exacte donnant un modèle agrégé final plus performant en termes de (R^2.oos). Voir l'annexe B pour plus de détails sur le test stacking.

II Amélioration et innovation

Dans cette section, on va détailler les parties principales que j'ai abordées soit par l'amélioration, soit par le développement ou par la création.

II.A Estimation et analyse

Dans cette partie, j'ai fait l'automatisation de plusieurs tâches entrant dans la procédure d'estimation, en plus de la construction de différent tableaux récapitulatifs décrivant les modèles construits.

Fichier de développement	
fichier 1 :	**cAverageModel**
	• Rassembler dans un tableau quelques caractéristiques du modèle agrégé (RMSE, R^2 In Sample, R^2 out of Sample, nombres de variables de modèle, nombre de sous modèles satellites acceptés et les min et max de pvalues de tests statistiques de tous les modèles satellites)
fichier 2 :	**FunctionsForShippingModels**
	• Créer la fonction **dataExhaustive** retournant la table de données prête pour le calibrage (concaténation entre la partie historique et celle des variables explicatives prédites par CASA Eco après tous les traitements et les transformations possibles). Cette table représente la table finale de variables à expliquer et explicatives. Cette méthode n'était pas encapsulée avant. • Automatiser le choix entre le jeu de données L1 et L2 et le choix de modélisation entre le rendement de la pente ou la pente elle-même en se basant sur les critères mentionnés dans la partie III.B.4 du chapitre 2. Avant, la sélection était faite manuellement • Rassembler toutes les caractéristiques des modèles entrant dans la procédure de la sélection dans un tableau contenant : Voir table 3.1 ci-dessous ◊ tous les résultats de la fonction précédente de **cAverageModel** ◊ l'existence de variables de flotte ou non parmi les variables explicatives. Le but étant de sélectionner des modèles comportant à la fois des variables de demande et offre ◊ nombre de variables explicatives prédite par CASA Eco et nombre de variables prédites par l'approche historique. Le but étant de privilégier les modèles ayant plus de variables prédites par CASA ◊ test de scénarios (vérifier que les prédictions respectent la vision économique considérée, c'est-à-dire, les prédictions de la vision Baseline devraient être supérieures à celles de la vision adverse) ◊ pour le jeu de données L2, l'importance cumulée de variables en L2. Le but étant de minimiser la part des variables L2 dans le modèle • Récupérer les modèles dont le jeu de données sélectionné est L1, d'après le tableau de la fonction précédente, tous seuls dans un dossier spécifique et L2 dans un autre • Automatiser autres procédures secondaires mais nécessaire pour l'optimisation du code, par exemple, le calcul de rendements génériques, les transformations, procédure de sélection des modèles, etc

Target	**L0.rdt.Bulker _Capesize_5Y**	**L0.lvl.pente.Bulker _Capesize_10Y**	**L0.rdt.pente.Bulker _Capesize_10Y**
L1.R^2_IS[1]	0.93	0.92	0.82
L1.R^2_OOS[2]	0.86	-0.42	0.71
L1.Nbre_Models_Accepted	14 models have been accepted	10 models have been accepted	9 models have been accepted
L1.Existence_de_variables_ de_flotte	TRUE	TRUE	TRUE
L1.Niveau_confiance	0.95	0.95	0.95
L1.nombre_de_variables_ECO	10	5	4
L1.nombre_de_variables_Histo	20	19	16
L1.nombre_de_variables_total	30	24	20
L1.test_scenario	TRUE	TRUE	TRUE
L2.R^2_IS	0.91	0.92	0.78
L2.R^2_OOS	0.87	0.81	0.61
L2.Nbre_Models_Accepted	14 models have been accepted	13 models have been accepted	11 models have been accepted
L2.Existence_de_variables_ de_flotte	TRUE	TRUE	TRUE
Poids_cumulee_des_variables_ retardees_de_2_ans	2.28	25.27	26.7
L2.Niveau_confiance	0.95	0.95	0.95
L2.nombre_de_variables_ECO	11	6	3
L2.nombre_de_variables_Histo	22	23	23
L2.nombre_de_variables_total	33	29	26
L2.test_scenario	TRUE	TRUE	TRUE
Choix_L1_L2	L2	L2	L1
Choix_pente_ou_rdt_pente		ok	ko

TABLE 3.1 – Le tableau récapitulatif rassemblant tous les caractéristiques de trois modèles.

II.B Prédiction

Dans cette partie, j'ai fait l'automatisation de la procédure de prédiction et j'ai fait entrer des grandes modifications sur la procédure.

1. R^2_IS est le coefficient de détermination R^2 In sample
2. R^2_OOS est le coefficient de détermination R^2 Out of sample

Fichier	Développement
Prédiction	• Automatiser la procédure de prédiction en prenant comme paramètre principal le scénario économique adopté dans une fonction appelé **Historique_pred**. Avant, il fallait recréer la table finale de données de variables à expliquer et explicatives, à chaque fois quand on changeait le scénario économique, utilisée dans la procédure de prédiction. Cette nouvelle fonction permet de "batcher" les productions des prédiction

FunctionsForShippingModels	• Créer la fonction **Historique_pred** qui retourne une série temporelle représentant les réalisations historiques de la variable à expliquer accompagnée de ses projections à 3 ans • Corriger la procédure de calcul de scénarios historiques ◊ Il y a des ratios parmi les variables explicatives qui doivent être inférieurs à 1, mais avec l'ancienne procédure (retour à L0.lvl), on pouvait avoir des ratios recalculés supérieurs à 1 ◊ Appliquer la procédure décrite dans le chapitre précédent aux variables entrant dans le calcul du ratio et refaire le calcul. Voir ci-dessous pour plus de détails

On rappelle que :

$$L0.lvl.Tanker\_Panamax\_PLAUC\_CO = \frac{L0.lvl.Tanker\_Panamax\_PLAUC\_LA + L0.lvl.Tanker\_Panamax\_UC}{L0.lvl.Tanker\_Panamax\_PLAUC\_LA + L0.lvl.Tanker\_Panamax\_UC + L0.lvl.Tanker\_Panamax\_OO}$$

Le ratio précèdent est constitué de variables de flottes qui sont toujours positives donc il devrait être impérativement inférieure à 1, mais avec l'ancienne procédure, on note l'existence de quelques valeurs supérieurs à 1 qui n'est pas acceptable et s'oppose à la significativité derrière cette variable.

Le principe de la correction a consisté à récupérer pas l'évolution du ratio, mais plutôt l'évolution des variables composant le ratio.

Dans la figure suivante, on souligne l'importance de cette nouvelle procédure sur la transition entre les valeurs historiques et les prédictions et leur évolution.

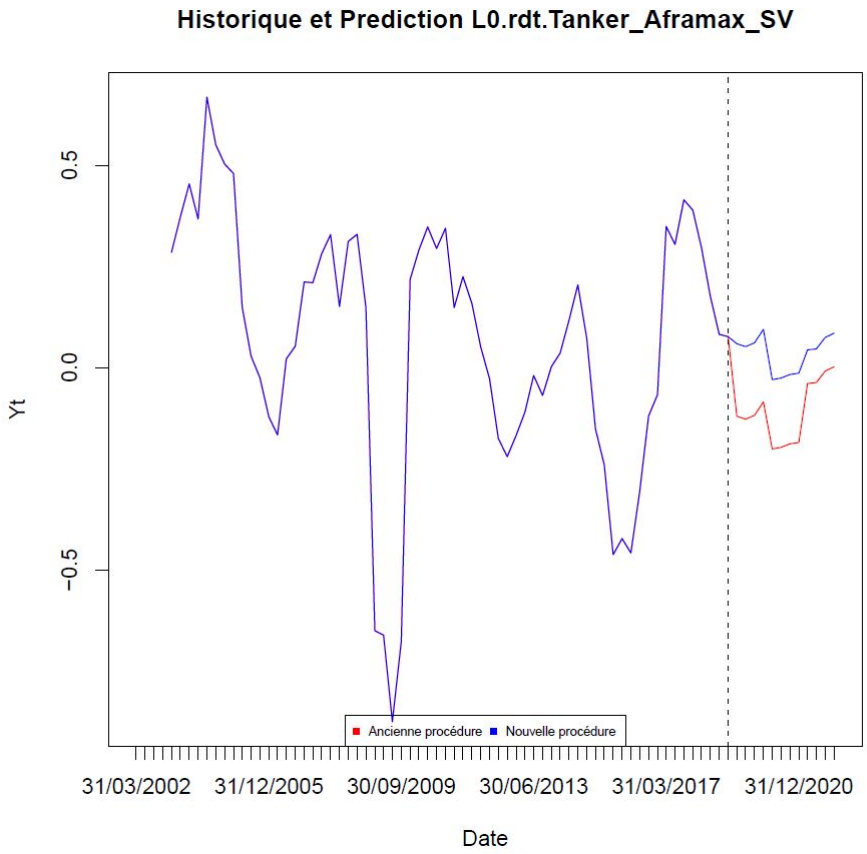


FIGURE 3.1 – Le graphe de réalisations historiques et de prédictions de variable L0.rdt.Tanker_Aframax_SV par les deux procédures.

D'après ce graphe, on remarque une amélioration pertinente dans la procédure de prédiction. Avec l'ancienne procédure, on note une décroissance rapide. Après les corrections, l'évolution des prédictions juste après l'historique est plus acceptable et plus en harmonie avec les dernières observations.

III Construction de nouveaux modèles

Dans l'annexe A, on a bien expliqué les tests que les sous modèles satellites devraient passer. Pour une variable à expliquer, si tous les modèles satellites sont rejetés, on doit chercher une solution alternative.

Aujourd'hui, il y a des recherches qui ont proposé des solutions de modélisation alternatives à tester pour chaque test statistique non passé, mais en général, il y a une méthode à suivre qui peut résoudre le problème de rejeter tous les modèles satellites, c'est **la transformation de variables** [8]. En effet, on parle de l'extension de la base de données par des nouvelles variables explicatives calculées à partir des anciennes, à côté de log-rendement et de la différence.

La transformation des données nécessite l'approche "trial and error". Lors de la construction du modèle, on teste une transformation, puis on vérifie si la transformation résout les problèmes du modèle. Si cela n'aide pas, on teste une autre transformation et ainsi de suite. On continue ce

processus cyclique jusqu'à ce que on ait construit un modèle approprié et utilisable. En d'autres termes, ce processus inclut la formulation, l'estimation et l'évaluation de modèle. L'évaluation est résumée principalement selon les trois critères décrits dans l'exemple suivant (R^2 In sample, R^2 Out of sample et nombre de sous modèles acceptés) :

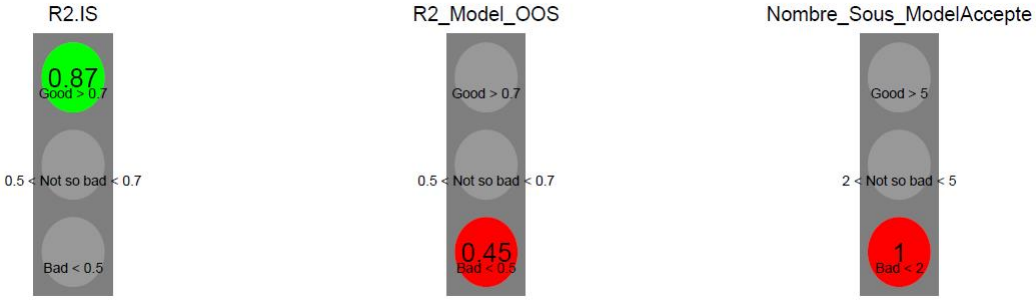


FIGURE 3.2 – La grille de performance.

Par considération au critère économique de sélection, on a choisit de tester seulement les transformations croissantes sur les variables considérées afin d'attribuer les signes de leurs coefficients à ceux de nouvelles variables créées. A côté des transformations, on a relâché aussi le niveau de confiance de 0.95 à 0.9 et on a utilisé le jeu de données L2 (plus de variables explicatives) pour augmenter la chance d'avoir un modèle final assez pertinent.

III.A Régression polynomial multiple

Il suggère d'enrichir notre base de données pour y inclure des termes linéaires et quadratiques [12]. En effet, on devrait construire un modèle qui inclut les deux prédicteurs, x et x^2 [6]. Après l'extension de la base de données, on a calibré des modèles avec la nouvelle base pour les variables à expliquer qui n'ont pas encore de modèles qui les expliquent (L0.rdt.Tanker_Aframax_5Y, L0.rdt_Tanker_Panamax_5Y et L0.rdt_Tanker_Generique_5Y).

On a appliqué cette transformation seulement sur les variables qui sont toujours positives, car la fonction carré est croissante sur la partie positive. On va discuter ici les résultats mesurant la performance de modèles construits, le reste (les variables les plus contributrices dans le modèle, graphes d'estimations et propriétés statistiques du modèles) se trouve dans l'annexe B.

- **Modèle Stacking L0.Tanker_Aframax_5Y**

Modélisation avec la transformation polynomiale						
RMSE	R^2.is	R^2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Jeu de données	Commentaires
0.08	0.94	0.8	7	150	L2	(+) Performance très élevée

- **Modèle Stacking L0.Tanker_Panamax_5Y**

Modélisation avec la transformation polynomiale						
RMSE	R^2.is	R^2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Jeu de données	Commentaires
0.07	0.92	0.85	2	150	L2	(+) Modèle assez performant (-) Un petit nombre de sous modèles acceptés

- **Modèle Stacking L0.Tanker_Generique_5Y**

Modélisation avec la transformation polynomiale						
RMSE	R^2.is	R^2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Jeu de données	Commentaires
0.14	0.91	-0.48	4	150	L2	(-) Modèle pas performant (-) RMSE grande (-) R^2 out sample très faible

D'après les résultats trouvés ci-dessus, on peut accepter les deux premiers modèles en termes de performances (comme indiqué dans la figure 3.2) mais pour le dernier, le modèle Stacking L0.Tanker_Generique_5Y, on doit chercher une autre solution alternative. D'après les autres résultats de ces modèles dans l'annexe B, on constate qu'on n'a pas besoin de tester la modélisation aussi avec le jeu de donnée L1, car il y a un peu de de modèles satellites acceptés et la plupart contiennent des variables explicatives retardées de 2 ans, d'où leur élimination implique la dégradation significative de la performance des modèles. On s'est dirigé vers la modélisation avec d'autres transformations afin d'améliorer plus la performance des modèles, surtout celle de stacking L0.Tanker_Generique_5Y.

III.B Transformation logarithmique [1]

Dans cette partie, on a étendu la base de données initiale avant la partie de régression polynomiale, par le logarithme de variables explicatives qui sont toujours positives [6], puis on refait le calibrage sur les variables à expliquer mentionnées dans la partie précédente.

- **Modèle Stacking L0.Tanker_Aframax_5Y**

Modélisation avec la transformation logarithmique						
RMSE	R^2.is	R^2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Jeu de données	Commentaires
0.18	0.87	0.43	5	150	L2	(-) Performance faible (-) RMSE un peu grand (-) R^2 out sample <0.5

- **Modèle Stacking L0.Tanker_Panamax_5Y**

Modélisation avec la transformation logarithmique						
RMSE	R^2.is	R^2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Jeu de données	Commentaires
0.07	0.92	0.86	1	150	L2	(+) Modèle assez performant (-) Un petit nombre de sous modèles acceptés

- **Modèle Stacking L0.Tanker_Generique_5Y**

Modélisation avec la transformation logarithmique						
RMSE	R^2.is	R^2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Jeu de données	Commentaires
0.09	0.95	0.58	3	150	L2	(+) Performance moyenne (-) R^2 out sample <0.7 (-) Un petit nombre de sous modèles acceptés

III.C Transformation avec la racine carré [6]

On a testé la modélisation avec cette transformation mais sans résultats pertinents par rapport aux précédents.

III.D Interprétation des résultats

D'après les résultats mentionnés ci-dessus, on peut noter qu'on a trouvé un modèle stacking de performance acceptable pour L0.Tanker_Generique_5Y, celui de la transformation logarithmique.

Pour le modèle stacking L0.Tanker_Aframax_5Y, on note que celui de la régression polynomiale est plus performant (RMSE plus faible, R^2 out sample plus grand et in sample aussi) et d'après les résultats de l'annexe B, la contribution de variables retardées de 2ans est moins importante ($9\% < 40\%$) donc on récupère le premier.

Pour le modèle stacking L0.Tanker_Panamax_5Y, les performances des modèles trouvés sont équivalentes et leurs contributions de variables retardées de 2ans sont aussi équivalentes malgré leur grande importance. On choisit finalement celui de la régression polynomiale car il possède plus de sous modèles satellites acceptés. Voir les graphes de prédictions de ces trois modèles finalement approuvés dans le paragraphe C.3 de l'annexe B.

On a décidé d'arrêter ici la recherche d'autres transformations, car ça devient de plus en plus compliqué avec les signes de coefficients fixés par CASA (le critère économique de sélection) et les variables explicatives nouvellement créées. On était obligé de tester seulement les transforamtions croissantes.

En plus de transformation de variables comme on a mentionné au début de cette partie, si un modèle satellite est rejeté par un test statistique, on a d'autres procédures de calibrage à tester. En effet, on a diminué significativement le niveau de confiance pour détecter les tests statistiques qui ne sont pas passés, on a trouvé notamment que le test de Box qui vérifie l'indépendance des résidus est le test qui rejette la plupart de sous-modèles satellites. Comme autre solution alternative, théoriquement on peut tester les modèles de séries temporelles, par exemple, la régression avec des erreurs autorégressives. C'est une ouverture pour le travail de l'équipe.

Chapitre 4

Implémentation d'algorithmes de réduction de dimensions

Lors du calibrage des modèles, on a constaté que la recherche exhaustive de sous modèles satellites de certaines variables à expliquer a pris un temps de calcul important (entre 1h et 2h surtout pour un jeu de données L2) en raison du grand nombre de combinaisons testées (modèles satellites à 4 variables parmi 168 variables pour les Bulkers avec jeu de données L2 par exemple). Par conséquent, nous avons cherché une méthode pour réduire le temps de calcul.

On a cherché à réduire le nombre de variables explicatives afin de réduire le nombre de combinaisons, ainsi le temps de calcul. On parle ici de techniques de sélection de variables, **Feature selection en anglais**.

I Principes des techniques de sélection de variables

En machine Learning et statistiques, la sélection de variables est le processus de sélection d'un sous ensemble contenant les variables explicatives pertinentes pour la modélisation parmi tous les variables disponibles. Les techniques de sélection sont utilisées pour plusieurs raisons [9] :

- Simplification de modèles pour les rendre interprétables
- Réduction du temps d'apprentissage
- Réduction de la complexité des modèles (overfitting)

Un algorithme de sélection de variables peut être vu comme la combinaison d'une technique de recherche permettant de proposer de nouveaux sous-ensembles de variables, et d'une mesure d'évaluation qui attribue des scores à ces différents sous-ensembles. L'algorithme le plus simple consiste à tester chaque sous-ensemble possible de variables en recherchant celui qui minimise le taux d'erreur. Il s'agit d'une recherche exhaustive qui est intraitable sur le plan informatique, à l'exception de problèmes de petite dimension. Le choix de la métrique d'évaluation a une grande influence sur l'algorithme, et ce sont ces métriques d'évaluation qui distinguent les trois catégories principales d'algorithmes de sélection : les Wrappers, les filtres et les méthodes intégrées « embedded » [9].

Algorithmes	Description	Avantages/Inconvénients
Wrapper	• Les méthodes Wrapper utilisent un modèle prédictif pour évaluer les sous-ensembles de variables, le taux d'erreur trouvé sur un ensemble test ou par validation croisée est le score • Elles fournissent généralement le sous ensemble le plus performant en termes de taux d'erreur minimal avec le modèle prédictif utilisé	(+) Les méthodes Wrapper évaluent des sous-ensembles de variables, ce qui permet de détecter les interactions possibles entre les variables (-) Le risque d'overfitting croissant lorsque le nombre d'observations est insuffisant (-) Le temps de calcul significatif lorsque le nombre de variables est grand
Filtres	• Les méthodes de filtrage suppriment les variables les moins intéressantes selon un critère bien choisi tel que les métriques de corrélation avec la variable à expliquer (Pearson, Spearman, Distance), test de Chi-2 etc.	(+) Ces méthodes sont particulièrement efficaces en temps de calcul et robustes en overfitting (-) Les filtres ont tendance à sélectionner des variables redondantes en ne prenant pas en compte les relations entre les variables
Embedded	• Les méthodes Embedded combinent les avantages des deux méthodes précédentes. Un exemple de cette approche est la méthode LASSO, un modèle linéaire, qui pénalise les coefficients de régression avec une pénalité L1, ramenant beaucoup d'entre eux à zéro	(+) Un algorithme d'apprentissage tire parti de son propre processus de sélection de variables et effectue simultanément la sélection et la classification des variables

II Description des algorithmes testés

Dans cette partie, on a choisi de tester seulement les algorithmes de famille Embedded et Wrapper, car on a trouvé que les filtres sont performants pour les variables discrètes, contrairement aux propriétés des variables de macro-économiques utilisées.

II.A LASSO

LASSO, Least Absolute Shrinkage and Selection Operator, a été formulé pour la première fois par Robert Tibshirani en 1996. C'est une méthode puissante qui effectue deux tâches principales : la régularisation et la sélection de variables. La méthode LASSO ajoute un terme de régularisation à l'estimateur de moindres carrés ordinaires [7].

$$\hat{\beta}^n(\lambda) = \underset{\beta \in \mathbb{R}^p}{argmin}(\frac{1}{2}||Y^n - X^n\beta||_2^2 + \lambda||\beta||_1) \tag{4.1}$$

où $Y_n \in \mathbb{R}^n$ correspond aux n observations de la variable à expliquer Y, X_n est la matrice n $\times$ p des observations de variables explicatives telle que n est le nombre d'observations et p le nombre de variables explicatives, $\beta \in \mathbb{R}^p$ est le paramètre du modèle à estimer, $||x||_2^2 = \sum_{i=1}^n x_i^2$ et $||x||_1 = \sum_{i=1}^n |x_i|$.

Le paramètre $\lambda \geq 0$ contrôle la puissance de la régularisation. Si on prend $\lambda = 0$, le Lasso correspond à une régression linéaire classique. La méthode applique un processus de contraction (régularisation) dans lequel elle pénalise les coefficients des variables de régression, réduisant certaines d'entre elles à zéro. Pendant le processus de sélection, les variables qui ont des coefficients non nuls sont sélectionnés pour faire partie du modèle. Le problème (1) peut admettre plusieurs solutions à cause de condition d'inversibilité de matrices, il est démontré que si X^T X est inversible alors (1) admet une solution unique . On cherche à trouver λ^* qui minimise le RMSE du modèle parmi un ensemble prédéfini ϕ.

$$\lambda^* = \underset{\lambda \in \phi}{argmin}(||Y - X\hat{\beta}^n(\lambda)||_2^2) \tag{4.2}$$

Finalement, avec λ^*, on reproduit le modèle lasso et on récupère les variables sélectionnés comme variables pertinentes pour la modélisation [7].

On a choisi de tester l'algorithme de lasso car il comporte un processus de sélection de variables interne comme décrit ci-dessus. Puisque on a p » n, alors, d'après Valeria Fonti [4], le processus de sélection par lasso est instable, donc on a reformulé un algorithme qui améliore la stabilité.

Du coup, on a choisi de répéter l'algorithme lasso ordinaire décrite au dessus N fois et pour toute $\lambda \in \phi$, et compter, pour chaque variable, le nombre de fois que son coefficient soit non nul dans tous les modèles construits. Ensuite, on doit calculer un pourcentage d'apparition pour chaque variable, la proportion du nombre total des modèles lasso construits $(Taille(\phi) * N)$ représentant le nombre de fois total où le coefficient de la variable soit non nul, et puis éliminer les variable ayant un pourcentage inférieur à α (niveau de sélection fixé en avant). On répète la procédure précédente avec les variables restantes M fois pour avoir finalement les variables sélectionnés.

Algorithme 1 Lasso modifié

Inputs : Vecteur de variable réponse : $Y = Y_1, \ldots, Y_n$, matrice de variables explicatives : X $n \times p$, niveau de sélection : α, nombre d'échantillons : N, nombre de répititions : M, grille proposée pour : ϕ

liste_var est le vecteur de noms de toutes les variables explicatives
pour $t \leftarrow 1, M$ **faire**
    Initialiser un vecteur v de 0 de même taille que liste_var
    pour $i \leftarrow 1, N$ **faire**
        Pour chaque $\lambda \in \phi$, reproduire le modèle LASSO avec Y et X [, liste_var]
        Compter, pour chaque variable, le nombre de fois que son coefficient soit non nul, dans tous les modèles Lasso construits, dans le vecteur v
    fin pour
    Transformer v en vecteur de taux de pourcentage d'apparition
    liste_var sera le vecteur des noms de variables ayant un pourcentage supérieur à α
fin pour
retourner *liste_var*

II.B LASSO avec bagging

Francis R.Bach [10] a proposé de construire une méthode ensembliste permettant l'amélioration de la performance et de la robustesse du modèle lasso ordinaire, en utilisant la méthode de bootstraping (cas particulier de bagging). Cette méthode consiste à créer des échantillons à partir de l'ensemble de données initial, on récupère à chaque fois 70% des observations aléatoirement sans remplacement et le reste avec remplacement. On a choisi de tester cette méthode car elle est une méthode d'agrégation d'algorithmes de prédictions qui s'applique bien à des algorithmes instables, de variance forte comme dans le cas de lasso.

Algorithme 2 Lasso avec bagging

Inputs : Vecteur de variable réponse : $Y = Y_1, \ldots, Y_n$, matrice de variables explicatives : X $n \times p$, niveau de sélection : α, nombre d'échantillons : N, grille proposée pour λ : ϕ

pour $t \leftarrow 1, N$ **faire**
    Échantillonnage aléatoire sans remplacement de 0.7 d'observations distinctes (X_i, Y_i) avec $i \in [1, \ldots, n]$ et le reste avec remplacement
    pour $\lambda \in \phi$ **faire**
        Appliquer la validation croisée $10 - fold$ sur le problème (1) afin de calculer RMSE du modèle avec λ
        Calculer RMSE du modèle, le stocker dans le vecteur $VRMSE$
    fin pour
    λ^* est le paramètre ayant le RMSE minimal de $VRMSE$
    Stocker les noms de variables dont les coefficients dans le modèle lasso construit avec λ^* sont non nuls dans L
fin pour
liste_var est la liste de variables ayant une occurrence% supérieure à α dans L
retourner *liste_var*

II.C Random forest

C'est une méthode d'agrégation de modèles dont le but principal est d'augmenter la performance. Cet algorithme peut aussi être utilisé dans un objectif de sélection de variables au moyen de mesures d'importance. Il appartient à la famille Wrapper. Random forest est une procédure itérative qui consiste à :

- Faire un bootstrap de n observations dans l'échantillon
- Construire un arbre de décision où les noeuds sont déterminés à partir de M variables choisies de façon aléatoire
- Ne pas élaguer l'arbre

La quantification de l'importance d'une variable d'entrée sur la prédiction de la variable de sortie est faite par permutation, c'est l'augmentation de la moyenne de l'erreur de prédiction des arbres (MSE pour la régression) dans la forêt, lorsque les observations de la variable considérée sont aléatoirement permutées dans les échantillons OOB (out of bag [1]). Autrement dit, la variable X_j est considérée comme importante pour la prédiction de Y si en brisant le lien entre X_j et Y, l'erreur de prédiction augmente.

Le processus de calcul d'importance est sensible à la dimension de table de données. Plus la dimension augmente plus ce processus devient instable et l'ordre de grandeur de l'importance diminue. On a p » n, alors on doit, selon Robin Genuer et al. [11], changer le paramètre du modèle **mtry**, le nombre de variables choisis aléatoirement utilisées dans un arbre et **ntree**, le nombre de bootsraps. Puisque on est dans la même situation décrite dans l'article de Robin Genuer et al.[11], alors on a choisi les mêmes valeurs pour ntree et mtry que dans l'article. Ils ont démontré aussi que l'existence de variables fortement corrélées peut rendre le processus moins stable. Une variable fortement prédictive appartenant à un groupe de variables corrélées sera estimée moins importante qu'une variable indépendante et moins informative. On a choisi de tester cet algorithme car il permet de mesurer l'importance de chaque variable et on peut choisir un nombre minimal de variables à sélectionner au contraire des algorithmes précédents où la sélection est totalement occupée par l'algorithme. L'algorithme suivant explique les étapes de mesure d'importance et la sélection :

1. Out of bag correspond à l'ensemble des arbres dans random forest qui ne contiennent pas le couple (X_i, Y_i) à estimer lors de bootsrapping de données.

Algorithme 3 Random forest (RF)

Inputs : Vecteur de variable réponse : $Y = Y_1, \ldots, Y_n$, matrice de variables explicatives : X $n \times p$, nombre minimal de variables sélectionnées : k, nombre de répétitions de RF[9] : $runs$, nombre de bootsraps : $ntree = 2000$, nombre de variables choisis aléatoirement utilisées dans un arbre : $mtry = floor(p/2)$

pour $t \leftarrow 1, runs$ **faire**
    Entraîner un modèle random forest avec Y et X de paramètres principales $ntree$ et $mtry$
    Stocker les mesures d'importance trouvées dans la colonne t de la matrice VI $p \times runs$
fin pour
Calculer la moyenne des importances de chaque variable sur toutes les itérations dans un vecteur m
Ordonner m dans l'ordre décroissant
Calculer l'écart type des importances de chaque variable dans un vecteur sd
Ordonner sd dans le même ordre que m
Entraîner un modèle CART (un algorithme d'arbre de décision) de sd sur la série $1, 2, \ldots, taille(sd)$
Chercher threshold : la valeur estimée minimale du modèle CART précédent
Récupèrer les noms de variables avec une moyenne d'importance supérieure à threshold dans le vecteur liste_var en respectant l'ordre de m (décroissant)
pour $i \leftarrow 1, taille(liste\_var)$ **faire**
    Entraîner un modèle random forest avec Y et X $[, liste\_var[1, i]]$
    Calculer $RMSE$ du modèle et le récupérer dans le vecteur $nested\_models$
fin pour
Trouver i l'indice correspondant à la valeur minimale dans $nested\_models$ $[(k+1) : taille(liste\_var)]$
retourner $liste\_var$ $[1 : (i+k)]$

II.D Boruta

Boruta est une méthode de sélection de variables appartenant au groupe Wrapper. Elle consiste à trouver toutes les variables pertinentes en se basant sur les interactions entre les variables [3]. Elle consiste à résoudre le problème all-relevant dans lequel elle cherche à sélectionner toutes les variables explicatives pertinentes. Tandis que la plupart des autres algorithmes de sélection de variables appartenant au même groupe Wrapper suivent une méthode optimale minimale en s'appuyant sur un petit sous-ensemble de variables, ce qui produit une erreur minimale sur un classificateur choisi.

L'algorithme est basé sur le modèle random forest comme l'algorithme précédent mais avec l'amélioration de la mesure de l'importance de variable de la forêt aléatoire. La mesure d'importance est le score Z, le rapport entre la moyenne des pertes de performance et leur écart type, les pertes sont calculées séparément pour chaque arbre contenant la variable concerné après la permutation de ses observations. Puisque le score z n'est pas lié directement à la signification statistique de la mesure d'importance implémentée dans l'algorithme random forest, alors on doit le calculer après l'extension de la base de données par la création des variables « shadow », permutation aléatoire des observations de chaque copie d'une variable initiale. L'ensemble

2. Algorithme de random forest

des mesures d'importance de variables « shadow » est utilisé comme une référence pour identifier les variables qui sont vraiment importantes. L'algorithme suivant explique la procédure de sélection :

Algorithme 4 Boruta

Inputs : Vecteur de variable réponse : $Y = Y_1, \ldots, Y_n$, matrice de variables explicatives : X $n \times p$, nombre maximal d'itérations : $maxRuns$, nombre maximal de répétitions de random forest dans chaque itération : N, nombre de bootsraps : $ntree = 2000$, nombre de variables choisis aléatoirement utilisées dans un arbre : $mtry = floor(p/2)$

1: X_r=X
2: **tant que** (t$\leq$ maxRuns et Test==FALSE) **faire**
3: **pour** $i \leftarrow 1, N$ **faire**
4: Ajouter des copies de toutes les variables explicatives existantes dans X_r
5: Permuter les copies afin d'enlever leurs corrélations avec Y (les variables shadow)
6: Concaténer X_r et les variables shadow dans X_e
7: Performer un modèle random forest avec Y, X_e et les paramètres $ntree$ et $mtry$ puis calculer les scores Z
8: Trouver le score Z maximal ($MZSA$) parmi ceux de variables shadow
9: Marquer chaque variable non shadow dont le score est plus grand que $MZSA$
10: **fin pour**
11: Effectuer un test statistique bilatérale binomial (a two-sided equality test) pour toutes les variables afin d'identifier les variables importantes, pas importantes et les variables dont l'importance ne peut pas être jugée par ce dernier test. Marquer ces derniers par 'Tentative' (voir ci-dessous pour plus de détails sur le test)
12: Enlever toutes les variables shadow et les variables considérées pas importantes pour avoir finalement la nouvelle X_r
13: Vérifier si toutes les variables sont soit rejetées, soit considérées comme importantes ($Test == TRUE$)
14: **fin tant que**
15: **si** $Test == FALSE$ **alors**
16: Appeler la fonction **TentativeRoughFix**[10] pour évaluer finalement l'importance des variables marquées par 'Tentative'
17: Récupérer les variables considérées finalement comme importantes dans le vecteur $liste\_var$
18: **retourner** $liste\_var$

Dans le test statistique mentionné ci-dessus, l'hypothèse nulle est que l'importance de la variable est égale à l'importance maximale des attributs aléatoires ($MZSA$). Le test est un test d'égalité bilatéral - l'hypothèse peut être rejetée lorsque l'importance de l'attribut est significativement supérieure ou significativement inférieure à $MZSA$. Pour chaque variable, on compte combien de fois son importance était supérieure à $MZSA$ (un hit est enregistré pour la variable). Le nombre attendu de fois pour N exécutions est E (N) = 0,5 N avec un écart type S =$\sqrt{0.25N}$ (distribution binomiale avec p=q=0.5) [2].

La variable est considérée comme importante (acceptée) lorsque le nombre de fois est significativement plus élevé que la valeur attendue et considérée pas importante (rejeté) lorsque le nombre de fois est significativement inférieur à la valeur attendue. Il est facile de calculer les

3. Une fonction qui peut être utilisée pour obtenir toutes les décisions manquantes par une simple comparaison de la médiane du score Z de la variable considérée avec la médiane du score Z de la variable shadow la plus importante.

limites pour accepter et rejeter la variable pour un nombre quelconque d'itérations et pour un niveau de confiance souhaité. Les variables dont leurs importances ne peuvent pas être jugées par ce test sont considérées comme des variables avec une importance indéterminée (elles seront appelées 'Tentative') [2].

II.E Gradient Boosting Machine (GBM)

Gradient boosting est une technique d'apprentissage des problèmes de régression et de classification, qui produit un modèle de prédiction sous la forme d'une agrégation entre des modèles peu performants, généralement des arbres de décision. Cette méthode s'applique à des algorithmes fortement biaisés, mais de variance faible. Il construit le modèle étape par étape, en permettant l'optimisation d'une fonction de perte arbitraire différentiable [13].

L'objectif principal du problème est de minimiser le risque empirique, l'espérance d'une fonction de perte **L(y, F(x))** :

$$F = \underset{F}{argmin}\mathbb{E}_{x,y}[L(y, F(x))] \tag{4.3}$$

Dans ce cas, on a pris $L(y, F(x)) = \frac{1}{2}(y - F(x))^2$ de sorte que $\mathbb{E}_{x,y}[L(y, F(x))] = \frac{1}{n}\sum_i (y_i - \hat{y}_i)^2$

La méthode de Gradient Boosting recherche l'approximation $\hat{F}(X)$ de variable à expliquer sous forme de somme pondérée de fonctions basiques h_i(X) de classe H.

$$\hat{F}(x) = \sum_{i=1}^{M} \gamma_i h_i(x) + const \tag{4.4}$$

On applique ici le principe glouton, on commence par la fonction constante, puis on cherche itérativement les autres fonctions basiques :
$F_0(X) = \underset{\gamma}{argmin} \sum_{i=1}^{n} L(y_i, \gamma)$,
$F_m(x) = F_{m-1}(x) + \underset{h_m \in H}{argmin}[\sum_{i=1}^{n} L(y_i, F_{m-1}(x_i) + h_m(x_i))]$,
où h_m est une fonction basique. Puisque la complexité du problème précédent est très grande, On utilise la descente de gradient (la pente la plus raide) à chaque étape d'optimisation :
$F_m(x) = F_{m-1}(x) - \gamma_m \sum_i^n \nabla_{F_{m-1}} L(y_i, F_{m-1}(x_i))$,
$\gamma_m = \underset{\gamma}{argmin} \sum_i^n L(y_i, F_{m-1}(x_i) - \gamma \nabla_{F_{m-1}} L(y_i, F_{m-1}(x_i)))$

Dans l'algorithme, on doit calculer une approximation du gradient de L. Une fois le modèle final F_M construit, l'importance de variables est mesurée par l'influence relative décrite dans (Friedman, 2001) [5].

Puisque avec gradient boosting, comme dans le cas de random forest, on peut aussi avoir une mesure d'importance de chaque variable, alors on a décidé de le tester. L'algorithme générique de gradient boosting est :

Algorithme 5 Gradient Boosting Machine (GBM)

Inputs : Vecteur de variable réponse : $Y = Y_1, \ldots, Y_n$, matrice de variables explicatives : X $n \times p$, niveau de sélection : α, fonction de perte différentiable : $L\left(y, F(x)\right)$, nombre d'itérations : M

1: Initialiser la fonction constante $F_0(x) = \underset{\gamma}{argmin} \sum_{i=1}^{n} L(y_i,)$
2: **pour** $m \leftarrow 1, M$ **faire**
3: Calculer les résidus : $r_{it} = -[\frac{\partial L(y_i, F(x_i)}{\partial F(x_i)}]_{F(x)=F_{m-1}(x)}$, *pour* $i = 1...n$
4: Performer un modèle d'arbre de régression $h_m(x)$ sur les résidus, utilisant l'ensemble de données $\{(x_i, r_{im})\}_{i=1...n}$
5: Calculer le multiplicateur γ_m par la résolution du problème d'optimisation suivant :

$$\gamma_m = \underset{\gamma}{argmin} \sum_{i}^{n} L(y_i, F_{m-1}(x_i) + \gamma h_m(x_i)) \tag{4.5}$$

6: $F_m(x) = F_{m-1}(x) + \gamma_m h_m(x)$
7: **fin pour**
8: Récupérer les noms de variables dont l'importance dans F_M est supérieure au quantile de niveau α dans le vecteur *liste_var*
9: **retourner** *liste_var* =0

III Application aux modèles Maritime

Tous les algorithmes décrits dans la partie précédente ont été testés sur les données Maritime. Dans la modélisation initiale avec toutes les variables, on a trouvé d'après la recherche exhaustive, un petit nombre de modèles satellites acceptés parmi des millions. Par exemple, dans la partie B.2 de l'annexe B, L0.rdt.pente.Containership_15Y a 10 modèles satellites acceptés dont le nombre total de variables explicatives utilisées est 22. Ainsi, il y a 22 variables très importantes pour construire tous les modèles satellites acceptés, parmi 102 variables explicatives données.

En effet, on doit essayer d'avoir, à la fin du processus de sélection de variables, un ensemble de variables sélectionnées de taille important assurant l'existence le nombre le plus grand possible de variables parmi ces 22.

Ainsi, on a relâché les paramètres relatifs aux niveaux de sélection d'une manière permettant en même temps de capter un nombre assez important des variables importantes et réduire considérablement la complexité du modèle en termes du temps de calcul. On va détailler ici nos résultats sur deux exemples, **L0.rdt.Bulker_Panamax_SV** et **L0.rdt.Tanker_VLCC_SV**, on a testé la procédure de calibrage avec les variables sélectionnées seulement. D'autres exemples sont détaillés dans l'annexe C.

III.A Résultats

Les valeurs des paramètres des algorithmes de sélection ont été attribuées de la façon suivante afin de satisfaire les conditions ci-dessus :

Lasso modifié	M = 3 N = 20 $\lambda \in 10^{seq(1,\ -4,\ length=1000)}$ $\alpha = 0$
Gradient boosting	$\alpha = 0.5$
Random forest	K=25 Runs=50
Boruta	maxRuns=50
Baglasso	N = 100 $\alpha = 0.25$ $\lambda \in 10^{seq(1,\ -4,\ length=1000)}$

Ces paramètres ont été choisis pour respecter les conditions du paragraphe précédent.

- **L0.rdt.Bulker_Panamax_SV**

Modélisation initiale avec toutes les variables explicatives disponibles					
RMSE	R^2.is	R^2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
0.08	0.92	0.86	15	112	1,16 hrs

Modélisation après sélection à priori des variables						
Algorithme	RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
LASSO modifié	0.09	0.93	0.9	12	55	5.14 mins
Gradient boosting	0.12	0.89	0.77	9	40	1.96 mins
Random forest	0.11	0.9	0.83	11	48	3.28 mins
Boruta	0.1	0.9	0.85	10	56	5.47 mins
Baglasso	0.09	0.92	0.89	7	35	1.54 mins

- **L0.rdt.Tanker_VLCC_SV**

Modélisation avec toutes les variables explicatives disponibles					
RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
0.11	0.88	0.83	8	117	1.38 hrs

Modélisation avec les variables sélectionnées						
Algorithme	RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
LASSO modifié	0.12	0.9	0.84	5	55	5.01 mins
Gradient boosting	NA[11]	NA	NA	NA	NA	NA
Random forest	NA	NA	NA	NA	NA	NA
Boruta	0.21	0.8	0.55	2	39	1.85 mins
Baglasso	0.12	0.9	0.84	5	39	1.94 mins

4. NA signifie qu'il n'y a aucun modèle qui passe les tests décrits dans la section III.B.1 du chapitre 2.

III.B Interprétation des résultats

Pour **L0.rdt.Bulker_Panamax_SV**, on note en premier lieu que le temps de calcul est diminué d'une façon importante de 1,16 heures à 5.47 mins au minimum. Concernant la performance, on remarque qu'il n'y a pas une dégradation grave. En effet, on trouve que l'algorithme Lasso modifié nous a fourni un modèle après la sélection à priori de variables de performance élevée, notamment en termes de $R^2.is$ et $R^2.oos$ mais une petite dégradation au niveau du RMSE. Les autres algorithmes ont fourni aussi des modèles de performance acceptable mais moins importante que celle du modèle fourni par l'algorithme lasso modifié.

Pour **L0.rdt.Tanker_VLCC_SV**, on note aussi la diminution du temps de calcul d'une façon importante, de 1,38 heures à 5 mins au minimum. On remarque qu'il n'y a pas de modèles agrégés finals en utilisant l'algorithme gradient boosting et l'algorithme random forest. En effet, l'ensemble de variables sélectionnées par ces deux algorithmes ne permettent pas d'avoir des modèles satellites acceptés par les tests décrits dans la section III.B.1 du chapitre 2. On note une dégradation importante de la performance du modèle fourni par l'algorithme Boruta en termes de R^2 et de $RMSE$. Enfin, les deux algorithmes de lasso restants ont fourni deux modèles après la sélection à priori de variables de performance acceptable.

Dans certains cas, l'ensemble des algorithmes considérés ne permettent pas d'obtenir un modèle final après la sélection à priori de variables comme dans le cas de L0.lvl.pente.Tanker_Suezmax_15Y décrit dans l'annexe C. On peut expliquer ce problème par le nombre très faible de variables pertinentes (6 variables parmi 117 variables explicatives, d'où deux sous modèles acceptés au maximum dans le modèle initial). Le fait de ne pas détecter au moins une variable parmi les variables explicatives de chaque modèle satellite accepté dans le modèle initial augmente le risque de ne pas avoir un modèle final agrégé.

Pour conclure, si on considère tous les critères de performance décrits dans la partie III du chapitre 3 (R^2 In sample, R^2 Out of sample et nombre de sous modèles acceptés), on trouve que l'algorithme Lasso modifié est l'algorithme le plus stable en termes de performance parmi les algorithmes testés et sa performance est proche de celle du modèle initial.

La sélection de variables est très utile dans la réduction importante du temps de calcul (de ~2h à moins de 8 mins), ce qui nous a bien aidé dans la partie de construction de nouveaux modèles du chapitre 3, après l'extension de la base de données en ajoutant des transformations supplémentaires de données. Elle a facilité notre recherche en capacité de calcul, le coût de calcul de modèle après l'extension de la base de données surpasse la capacité du serveur avec 96 Go de RAM, d'où une réduction de dimension est nécessaire.

Conclusion

Pour conclure, j'ai effectué mon stage de fin d'études et en même temps de mon Master en statistiques et finance de l'ENSAE en tant que data scientist - analyste quantitatif au sein de département RPC chez Crédit agricole CIB avec l'équipe MQP. Lors de ce stage, j'ai pu mettre en pratique mes connaissances théoriques acquises durant ma formation académique, tout en étant confronté aux difficultés réelles du monde du travail.

Après ma rapide intégration dans l'équipe, j'ai eu l'occasion de réaliser plusieurs missions en risque de crédit et développer les modèles existants. J'ai contribué à l'amélioration, l'automatisation, la fiabilisation et la documentation de la procédure d'estimation actuelle des LGD sur les financements d'actifs maritimes, la construction de nouveaux modèles de projection des valeurs d'actifs maritimes dont les modèles existants sont de performance faible et finalement l'amélioration du temps d'exécution dans l'estimation des valeurs d'actifs via l'implémentation de techniques avancées de machine learning.

Cette expérience m'a permis de répondre aux questionnements que j'avais en ce qui concerne les moyens utilisés par les banques pour se prémunir du risque de crédit et sa modélisation par des modèles statistiques. J'ai eu l'opportunité de travailler avec des différentes techniques de machine learning comme l'apprentissage ensembliste, la technique de prédiction historique des variables explicatives dans le but de la projection de variable à expliquer et les différentes techniques de réduction de dimension. Parmi les modèles rencontrés : le modèle stacking, la régression lasso, le gradient boosting, la forêt aléatoire, bagging et Boruta. J'ai également eu l'occasion de me familiariser avec les tests statistiques (servant à vérifier les hypothèses de la régression linéaire et la nature de séries temporelles) et leur mise en oeuvre avec R.

Ce stage a été enrichissant pour moi, car j'ai eu l'opportunité d'améliorer mes compétences en programmation avec le langage R mais également en traitement de données avec microsoft Excel. Ainsi que dans l'utilisation de l'outil bloomberg pour l'exportation de données financières et macroéconomiques dans Excel, j'ai d'ailleurs eu la possibilité de faire une formation certifiée qui pourra m'aider dans mes expériences futures. Puisque la recherche est une partie primordiale dans le stage, j'ai eu l'occasion d'améliorer mes compétences en termes de la maîtrise des moyens de recherche documentaire. Ce stage m'a aussi permis de comprendre que les missions rassemblant le machine learning et la finance quantitative sont les plus adaptées à mon profil et que la recherche et l'innovation me passionent le plus. Je préfère ainsi de m'orienter vers un poste dont la grande partie est dédiée pour la recherche et le développement.

Dans ce rapport, on a abordé plusieurs applications de modélisation statistique en risque de crédit mais en raison de limite de temps de stage, il y a des aspects non explorés, par exemple :

- Tester d'autres transformations sur les variables explicatives, dans la partie de construction de nouveaux modèles, qui exigent la validation de l'équipe CASA ECO
- Tester les modèles de séries temporelles (ARIMA, GARCH, etc)
- Chercher une méthode plus performante que le stacking agrégeant les modèles satellites acceptés

Bibliographie

[1] Kenneth Benoit. Linear regression models with logarithmic transformations. March 2001.

[2] Miron B. Kursa et Witold R. Rudnicki. Boruta – a system for feature selection. *Fundamenta Informaticae*, 2010.

[3] Miron B. Kursa et Witold R. Rudnicki. Feature selection with the boruta package. *Journal of Statistical Software*, 2010.

[4] Valeria Fonti. Feature selection using lasso. 2017.

[5] Jerome H.Friedman. Greedy function approximation : A gradient boosting machine. 2001.

[6] Octavia Wong et C.M Wong Jolynn Peck. Data transformations for inference with linear regression : Clarifications and recommendations. *Practical Assessment, Research and Evaluation*, 2017.

[7] Serge Nakache. Aperçu des méthodes de sélection de variables (avec r).

[8] Eberly College of Science. Lesson 9 : Data transformations.

[9] Manish Pathak. Feature selection in r with the boruta r package.

[10] Francis R.Bach. Bolasso : Model consistent lasso estimation through the bootstrap. 2008.

[11] Christine Tuleau-Malot Robin Genuer, Jean-Michel Poggi. Variable selection using random forests. 2012.

[12] Jeffrey S.Simonoff. Transformation in regression. 2016.

[13] Wikipedia. Gradient boosting.

Annexe A

A Contexte des modèles

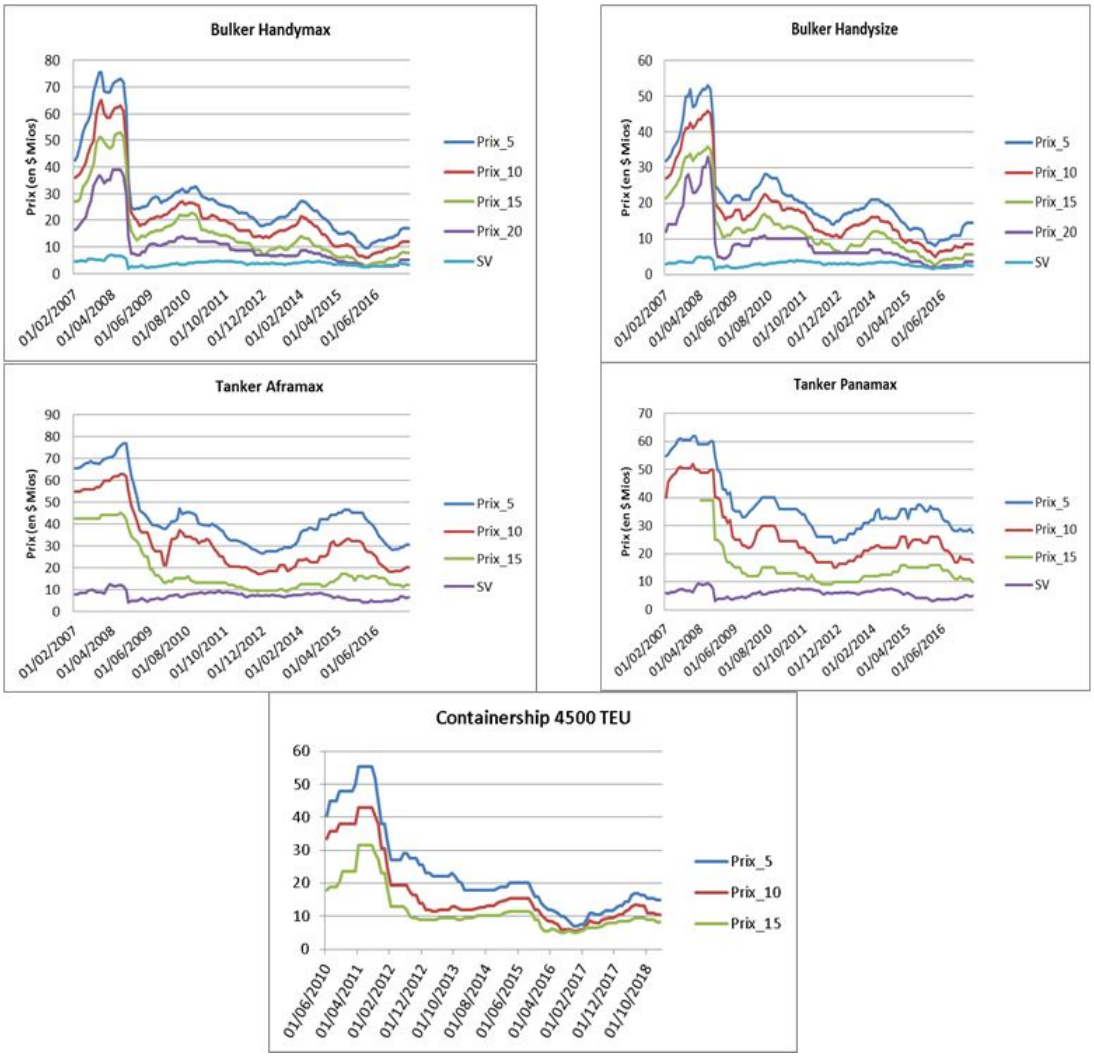


FIGURE 1 – L'évolution de prix de quelques navires de différents âges en fonction du temps.

Le tableau ci-dessous ne contient ici qu'un exemple de signes sur les variables de flotte pour les bulkers Panamax, le raisonnement étant identique pour les autres types/sous-types de navires, idem pour l'indice Baltique.

Variable	Description	Signe
CHVAIOY_Index_MA	Production industrielle Chine	1
OMRSD001_Index	demande_mondiale_petrole	1
IISTTOTL_Index	Production_Mondiale_Acier	1
IP_YOY_Index_MA	Production_Industrielle_US	1
CNMVIROM_Index	China Import Commodity Volume – Iron Ore & Concentrate	1
CNIVCERE_Index	China Import Commodity Volume – Cereals and Cereal Flour	1
CNIVCOAL_Index	China Import Commodity Volume – Coal & Brown Coal	1
CJPNGLBL_Index	Japanese Global Imports of Coa	1
GDP_CYOY_Index_MA	real GDP GA - US	1
EUGNEMUY_Index_MA	real GDP GA - Zone euro	1
BDIY_Index_MA	Baltic Dry Index	1
OCONTWLD_Index_YE	Oil world consumption YE	1
CNFREXPY_Index	China Export Trade (Annual YoY %)	1
EUCRBRDT_Index	Prix du Brent (USD/Baril)	1 sauf tanker (-1)
CPMINDX_Index	Production manufacturière Chine	1
CNIVCRUO_Index	Importation chinoise de pétrole	1
WIOPBRZL_Index	Production de fer Brésil	1
AUITIROV_Index	Exportation minerai de Fer Australie	1
AUITIRCV_Index	Exportation minerai de Fer Australie vers Chine	1
CNGDPYOY_Index_MA	PIB nominal chine	1
AUITSCKV_Index	Australia Steaming Coal Export	1
CNFREXPY_Index	China Export Trade (Annual YoY %)	1
CNFRIMPY_Index_MA	China_Import_Trade_YoY	1
CNGDPYOY_Index_MA	PIB nominal chine	1
EUGNEMUY_Index_MA	real GDP GA - Zone euro	1
GDP_CYOY_Index_MA	real GDP GA - US	1
OCONTWLD_Index_MA	Oil world consumption	1
Bulker_Panamax_IS*	Bulker Panamax Nombre navires en service	-1
Bulker_Panamax_PC*	Bulker Panamax Part du carnet de commande dans la flotte	-1
Bulker_Panamax_DW*	Bulker Panamax Capacité totale	-1
Bulker_Panamax_BU_DW*	Bulker Panamax Démolitions en capacité	1
Bulker_Panamax_CO*	Bulker Panamax Carnet d'ordre	-1
Bulker_Panamax_PLA_CO*	Bulker Panamax Part des navires lancés dans Carnet d'ordre	-1
Bulker_Panamax_PLAUC_CO*	Bulker Panamax Part des navires lancés et en construction dans Carnet d'ordre	-1
VESLBKDW_Index*	Vessel Fleet Status Bulk Carrier DWT In Service	-1

FIGURE 2 – Un exemple de signes de corrélations sur Bulker Panamax.

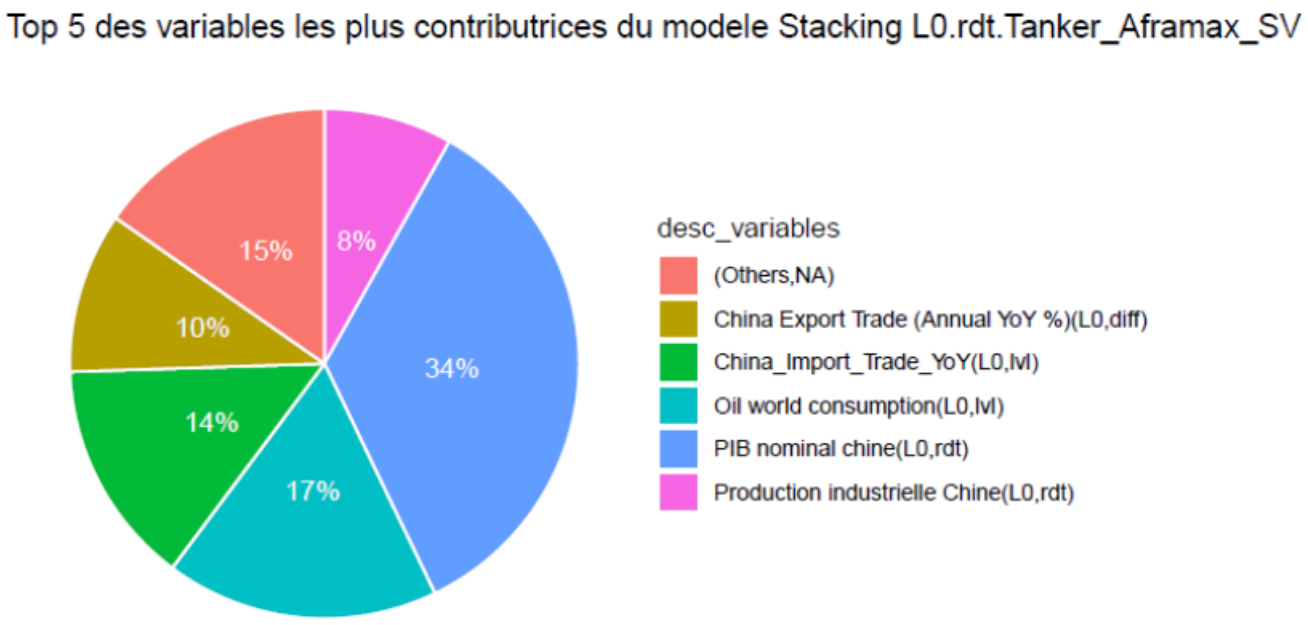
B Un exemple détaillé d'une modélisation

Le détail des résultats d'un modèle sont présentés ci-dessous, le modèle du log-rendement de la valeur scrap du Tanker Aframax. Ce modèle est issu de l'agrégation de 6 modèles satellites qui font intervenir 16 variables.

Indicateur	Valeur
RMSE	16,00%
R2.IS	81%
R2.OOS	73%
Number.of.sub.model	6 models have been accepted
Number.of.variables	16
max Fisher	0,00%
max Student	1,00%
max ADF	4,00%
max VIF	8,23
min JarqueBera	25,00%
min Box	7,00%
min GoldfeldQuandt	78,00%
min BreuschPagan	11,00%
min Cusum	16,00%

Le modèle (estimation L1) a été retenu en raison de ses critères de performance satisfaisants. En effet, les R^2 in et out of sample sont respectivement de 81% et 73%. Les 6 modèles satellites retenus sont les suivants, les R^2 ajustés de ces modèles unitaires vont de 69% à 80% :

Nom modele	Poids Stacking	R2	R2 ajusted
y~L0.lvl.CNFRIMPY_Index_MA+L0.rdt.CNGDPYOY_Index_MA+L0.lvl.OCONTWLD_Index_MA+L1.rdt.Tanker_Aframax_PLA_CO	0.43369833	0.7646530	0.7422390
y~L0.rdt.CHVAIOY_Index_MA+L1.lvl.CHVAIOY_Index_MA+L0.rdt.CNGDPYOY_Index_MA+L0.lvl.OCONTWLD_Index_MA	0.20860175	0.8213660	0.8043532
y~L0.diff.CNFREXPY_Index+L0.lvl.CNGDPYOY_Index_MA+L0.rdt.OCONTWLD_Index_MA+L0.rdt.Tanker_Aframax_PLAUC_CO	0.11306850	0.7617090	0.7390146
y~L1.lvl.BIDY_Index+L0.diff.CNFREXPY_Index+L1.diff.CNFREXPY_Index+L0.rdt.Tanker_Aframax_DW	0.10966143	0.7227601	0.6963563
y~L1.rdt.CHVAIOY_Index_MA+L0.diff.CNFRIMPY_Index_MA+L0.rdt.CNGDPYOY_Index_MA+L1.rdt.OMRSD001_Index	0.07994937	0.7179846	0.6911260
y~L1.lvl.BIDY_Index+L0.diff.CNFREXPY_Index+L0.rdt.OCONTWLD_Index_MA+L0.rdt.Tanker_Aframax_PLAUC_CO	0.05502062	0.7707897	0.7489602

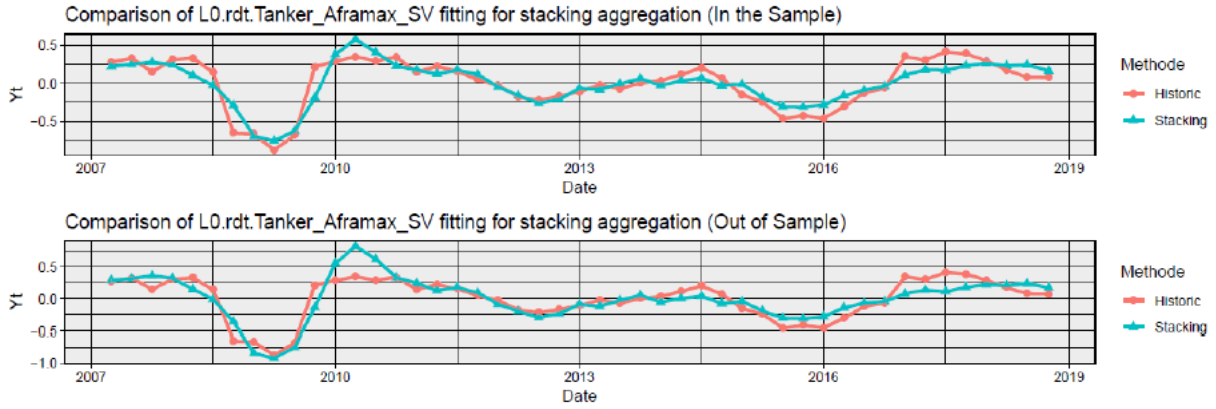


Les variables les plus contributrices au modèle agrégé sont :

- Le PIB Chine (34%)

- La consommation mondiale de pétrole (17%)
- Les importations chinoises (14%)
- Les exportations chinoises (10%)
- La production industrielle Chine (8%)

Sur l'historique de calibrage, les estimations in et out of sample sont comparées avec les valeurs observées :



L'estimation avec le jeu L1 a été préférée à celle du jeu L2 en raison des meilleures performances out of sample (R^2 out of sample 73% dans le jeu L1 versus 60 dans le jeu L2).

C Les tests statistiques :

Test de Dickey Fuller augmenté (ADF)

Le test ADF est un test statistique construit pour savoir si une série temporelle est stationnaire, c'est-à-dire si ses propriétés statistiques (espérance, variance, autocorrélation) varient ou pas dans le temps. Il est inspiré de test Dickey et Fuller (1974), un test de racine unitaire qui estime l'hypothèse nulle de racine unitaire (ou de non stationnarité) et ne concerne que les processus autorégressifs d'ordre un ou processus AR(1). Il estime trois modèles, le modèle sans constante ni dérive temporelle, un modèle avec constante et sans dérive temporelle et un modèle avec constante et dérive temporelle.

Le test ADF cherche à détecter la présence d'une racine unitaire pour les processus de type AR(p). Par exemple, le modèle sans constante ni dérive temporelle est le modèle suivant :

$$\Delta y_t = \phi y_{t-1} + \sum_{j=2}^{p} \beta_j \Delta y_{t-1-j} + \epsilon_t \tag{6}$$

H0 : $\phi = 0$

Le test ADF consiste à comparer la valeur estimée du t de student associé au paramètre ϕ aux valeurs tabulées de cette statistique. L'hypothèse nulle H_0 et de la non stationnarité est rejetée au seuil de 5% lorsque la valeur observée du t de student est inférieure à la valeur critique tabulée ou $tobs < ADF_{.05}$.

Test de Box

Le test Ljung-Box (appelé aussi le test de Portmanteau) est utilisé pour tester si les observations

sont aléatoires et indépendantes. En particulier, pour k donné, il teste le suivant :

$$\left\{ \begin{array}{l} H_0 : r_1 = r_2 = ... = r_k = 0 \\ H_1 : \exists\ i\ \in\ \{1, 2, ..., k\}\ tq\ r_i \neq 0 \end{array} \right\} \tag{7}$$

r_i est l'autocorrélation des résidus. La statistique du test est donné par :

$$Q_k = n(n+2) \sum_{j=1}^{k} \frac{r_j^2}{n-j}$$

Avec :

- n = nombre d'observations
- k = ordre de lag maximal testé

La statistique suit asymptotiquement la loi khi 2 d'ordre k sous H_0. On rejette H_0 si la probabilité critique (p-value) est inférieure au risque de première espèce (0.05 ou 0.1), ou si la statistique de test est supérieure au seuil critique choisi.

Test de Jarque-Bera

On doit vérifier si les résidus ϵ_i suivent une loi normale. Ce test est basé sur le skewness S [2] et le kurtosis K [3]. Ce test est fondé sur la comparaison entre le skewness et le kurtosis estimé à partir d'un échantillon et les valeurs de référence de la loi normale. Une loi normale a un coefficient d'asymétrie = 0 et une kurtosis = 3. Les hypothèses du test sont :

$$\left\{ \begin{array}{l} H_0 : S(\epsilon) = 0\ et\ K(\epsilon) = 3 \\ H_1 : S(\epsilon) \neq 0\ et\ K(\epsilon) \neq 3 \end{array} \right\} \tag{8}$$

La statistique du test notée JB est :

$$JB = \frac{n-k}{6}(S^2 + \frac{(K-3)^2}{4})$$

Avec :

- n = nombre d'observations
- k = nombre de variables explicatives
- S = coefficient d'asymétrie : $\frac{\frac{1}{n}\sum_{i=1}^{n}(\epsilon_i - \bar{\epsilon})^3}{[\frac{1}{n}\sum_{i=1}^{n}(\epsilon_i - \bar{\epsilon})^2]^{3/2}}$
- K = kurtosis : $\frac{\frac{1}{n}\sum_{i=1}^{n}(\epsilon_i - \bar{\epsilon})^4}{[\frac{1}{n}\sum_{i=1}^{n}(\epsilon_i - \bar{\epsilon})^2]^2}$

Sous H_0, JB suit asymptotiquement la loi Khi 2. On rejettera H_0 si la statistique de test est supérieure au seuil critique ou si la probabilité critique est inférieure au risque de première espèce choisi (0.05 ou 0.1).

Test de Goldfeld Quant

C'est un test statistique, très connu en économétrie qui vise à savoir si les perturbations issues d'un modèle linéaire multiple estimé par la méthode des moindres carrés sont homoscédastiques ou hétéroscédastiques. La procédure du test est comme suit :

- Ordonner les données en fonction d'une variable X_k parmi les variables explicatives (soupçonné d'être source d'hétéroscédasité) par ordre croissant ou décroissant

2. Le coefficient d'asymétrie d'une distribution.
3. Le coefficient d'aplatissement.

- Enlever un part c d'observations au milieu l'échantillon classé (k peut être 1/4 ou 1/5 par exemple)

On obtient 2 échantillons de taille (n-c) / 2 contenant l'un les valeurs les plus faibles et l'autre les valeurs les plus grandes.

- Effectuer la régression par MCO [4] dans les deux échantillons et on trouve SCR_1 et SCR_2 respectivement la somme des carrés des résidus estimés.

$$\left\{ \begin{array}{l} H_0 : Var(\epsilon_1) = Var(\epsilon_2) \\ H_1 : Var(\epsilon_1) \neq Var(\epsilon_2) \end{array} \right\} \tag{9}$$

Sous H_0, la statistique du test, $f = fracSCR2SCR1$ suit asymptotiquement la loi de Fisher de paramètres [(n-c-k) / 2 ; (n-c-k) / 2], où n et k représentent respectivement le nombre d'observations et le nombre de variables explicatives. On rejettera H_0 si f est supérieure au seuil critique ou si la probabilité critique est inférieure au risque de première espèce choisi (0.05 ou 0.1).

Test de Studentized Breusch Pagan

C'est un test statistique ayant le même but que le test précédent, il consiste aussi à tester l'homoscédacité des résidus.

Variance Inflation Factor (VIF)

Variance inflation factor (VIF) détecte la multicolinéarité dans la régression. La multicolinéarité s'entend lorsqu'il existe une corrélation entre les prédicteurs (c'est-à-dire les variables explicatives) dans un modèle ; sa présence peut affecter négativement les résultats de régression. Le VIF estime à quel point la variance d'un coefficient de régression est gonflée en raison de la multicolinéarité dans le modèle. On l'obtient en faisant régresser chaque variable explicative, disant X_i sur les variables explicatives restantes, et en vérifiant quelle part (de X_i) est expliquée par ces variables.

$$VIF_i = \frac{1}{1 - R_i^2} \tag{10}$$

Où R_i est le coefficient de détermination de l'équation de la régression dans le paragraphe précédent. Enfin, on analyse la magnitude de la multicolinéarité en considérant le VIF_i. Une règle de base est que si $VIF_i > 10$ donc la multicolinéarité est pertinente.

Test de Fisher

Le test de Fisher consiste à tester la significativité globale de modèle satellite, tester l'existence d'une relation linéaire entre y et les variables explicatives. En particulier, il permet de réaliser un test de nullité jointe de l'ensemble des coefficients. Dans ce cas, on teste l'hypothèse

$$\left\{ \begin{array}{l} H_0 : \alpha_1 = ... = \alpha_k = 0 \\ H_1 : \exists j,\ \alpha_j \neq 0 \end{array} \right\} \tag{11}$$

La statistique du test s'écrit :

$$S = \frac{SSR}{SSE} = \frac{(SCT - SCR)/p}{SCR/(n - p - 1)}$$

- n : nombre d'observations
- p : nombre de variables explicatives

4. La méthode ordinaire des moindres carrés.

- SCT : $\sum_i (y_i - \bar{y})^2$
- SCR : $\sum_i (y_i - \hat{y}_i)^2$
- y_i : les réalisations historiques de la variable à expliquer
- $\hat{y}_i$: les estimations de la variable à expliquer
- $\bar{y}$: la moyenne des réalisations historiques de la variable à expliquer

La statistique de test S suit une loi de Fisher de paramètres (p, n-p-1). Dans ce cas, le rejet de l'hypothèse H_0 est souhaité et de même on rejettera H_0 si la probabilité critique est inférieure au risque de première espèce choisi (0.05 ou 0.1).

Test de Student Le test de student consiste à tester la significativité de coefficients de modèle satellite. L'hypothèse nulle ou H_0 considère qu'il y a au moins un coefficient pas significatif. Elle est rejetée si la probabilité critique (p-value) maximale parmi toutes les probabilités critiques de t-student associées aux coefficients est inférieure au risque de première espèce (0.05 ou 0.1).

Annexe B

A La conception d'un objet de calsse cAverageModel

- **dictModels** : une liste contenant tous les modèles satellites acceptés de classe **cLinearModel**
- **y** : jeu de données à expliquer
- **x** : tableau de données de variables explicatives
- **xnames** : noms de variables explicatives
- **variableEcosign** : tableau contenant les signes à priori des coefficients des variables explicatives
- **statTests** : variable booléenne indiquant si on doit appliquer les tests statistiques ou non
- **confidenceLevel** : le niveau de confiance pour les tests statistiques
- **RMSE_methodSubset** : la méthode pour le calcul de RMSE (par exemple, la validation croisée)
- **RMSE_nbout** : nombre de sous-groupes (KFolds) pour la validation croisée
- **whichScore** : méthode de calcul du score de modèle satellite
- **aggregationMethod** : la méthode ensembliste d'agrégation entre les modèles satellites
- **explorationTbl** : tableau rassemblant tous les modèles satellites acceptés accompagnés de leurs résultats de tests statistique et de test économique et leurs scores
- **idAcceptedModels** : tableau rassemblant tous les modèles satellites acceptés accompagnés de leurs poids d'agrégation selon la méthode utilisée (**aggregationMethod**)

B Le test sur la méthode stacking

B.1 Description

- La méthode stacking est un apprentissage, dont les données principaux sont les prédictions de sous modèles et les réalisations historiques de la variable à expliquer
- Le problème d'optimisation (2.2) est résolu par la fonction lsei "Least squares with equalities and inequalities", cherchant à approximer la solution exacte
- L'approximation est changée d'une exécution de l'algorithme à une autre, mais la différence n'est pas importante
  $\Longrightarrow$ On peut considérer que l'algorithme est assez stable mais il n'y a pas de solution unique
  Dans un algorithme d'apprentissage, des données extrêmes ou pas pertinentes peuvent diminuer la performance du modèle

$\Longrightarrow$ L'élimination des prédictions des modèles satellites non très performants peut aider à améliorer la résolution du problème d'optimisation

- On a testé le calcul de poids stacking avec toutes les combinaisons de sous modèles satellites et on a récupéré celle avec la performance (R^2.oos) maximale et même RMSE (la valeur à minimiser dans le problème d'optimisation) qu'avec tous les sous modèles

B.2 Résultats du test et discussion

- **L0.lvl.pente.Bulker_Capesize_20Y**

Modèle agrégé avec tous les modèles satellites				
RMSE	$R^2.is$	$R^2.oos$	Nombre de sous modèles acceptés	Nombre de variables explicatives de modèles acceptés
0.02	0.87	0.63	14	32

Modèle avec la combinaison de modèles satellites acceptés la plus optimale en performance				
RMSE	$R^2.is$	$R^2.oos$	Nombre de sous modèles acceptés	Nombre de variables explicatives de modèles acceptés
0.02	0.86	0.86	6	17

- **L0.rdt.pente.Containership_15Y**

Modèle agrégé avec tous les modèles satellites				
RMSE	$R^2.is$	$R^2.oos$	Nombre de sous modèles acceptés	Nombre de variables explicatives de modèles acceptés
0.1	0.81	0.3	10	22

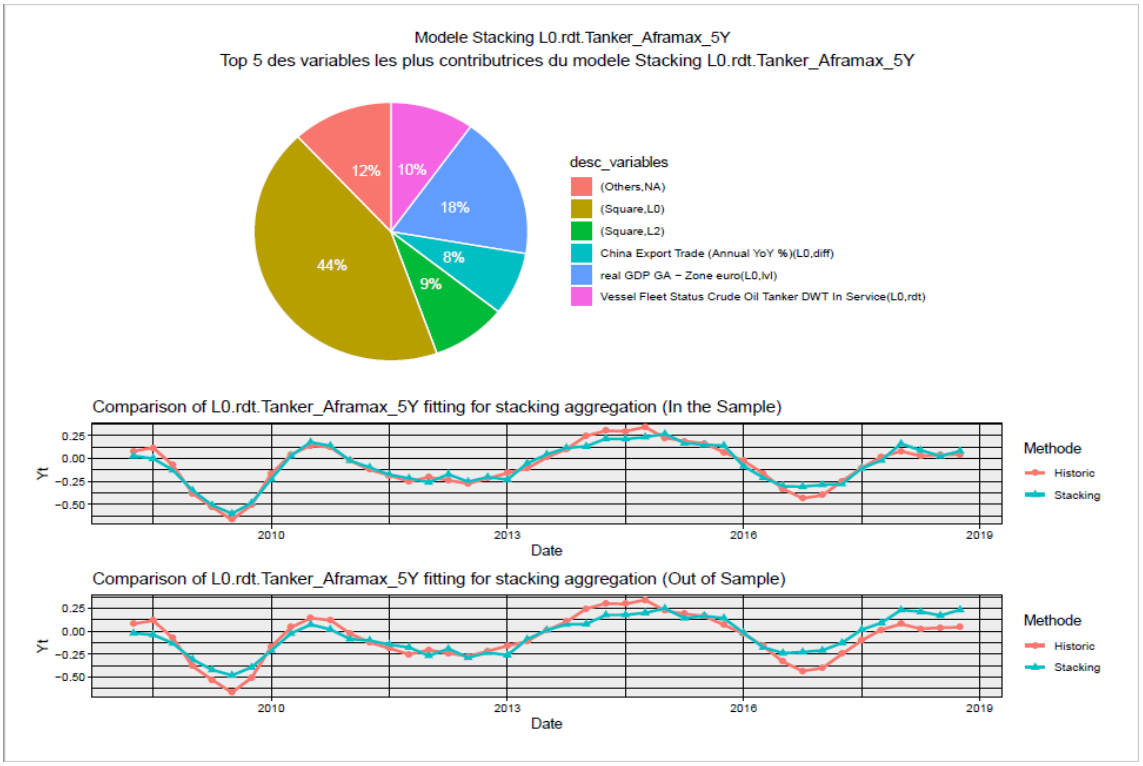
Modèle avec la combinaison de modèles satellites acceptés la plus optimale en performance				
RMSE	$R^2.is$	$R^2.oos$	Nombre de sous modèles acceptés	Nombre de variables explicatives de modèles acceptés
0.11	0.82	0.71	6	16

On note, d'après le premier exemple, qu'il n'y a pas une solution unique pour le problème et qu'on peut trouver d'autres solutions permettant d'avoir un modèle agrégé final plus performant en termes de $R^2.oos$. Dans le deuxième exemple, on a fixé une marge de 0.01 pour RMSE dans la recherche d'autres solutions pour le problème de l'optimisation 2.2 avec toutes les combinaisons possibles de modèles satellite afin de maximiser le plus possible la performance finale en termes de $R^2.oos$.

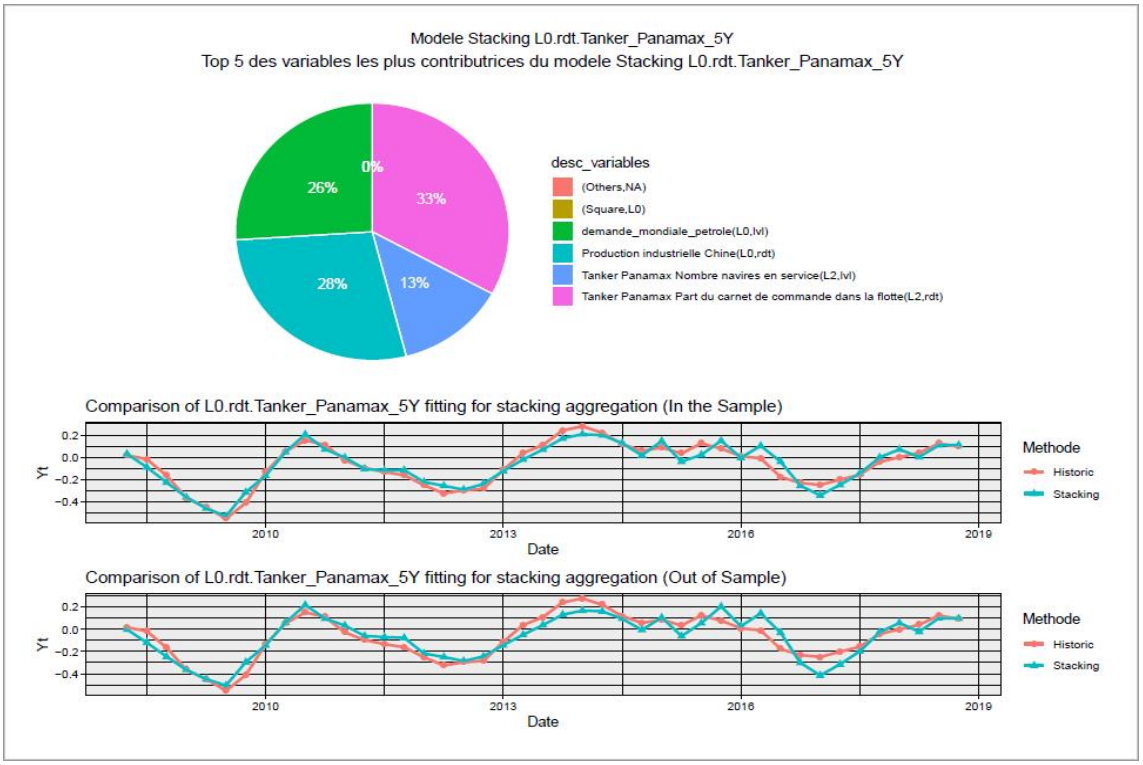
C Les résultats de modélisation de variables à expliquer sans modèles après les transformations

C.1 Régression polynomial multiple

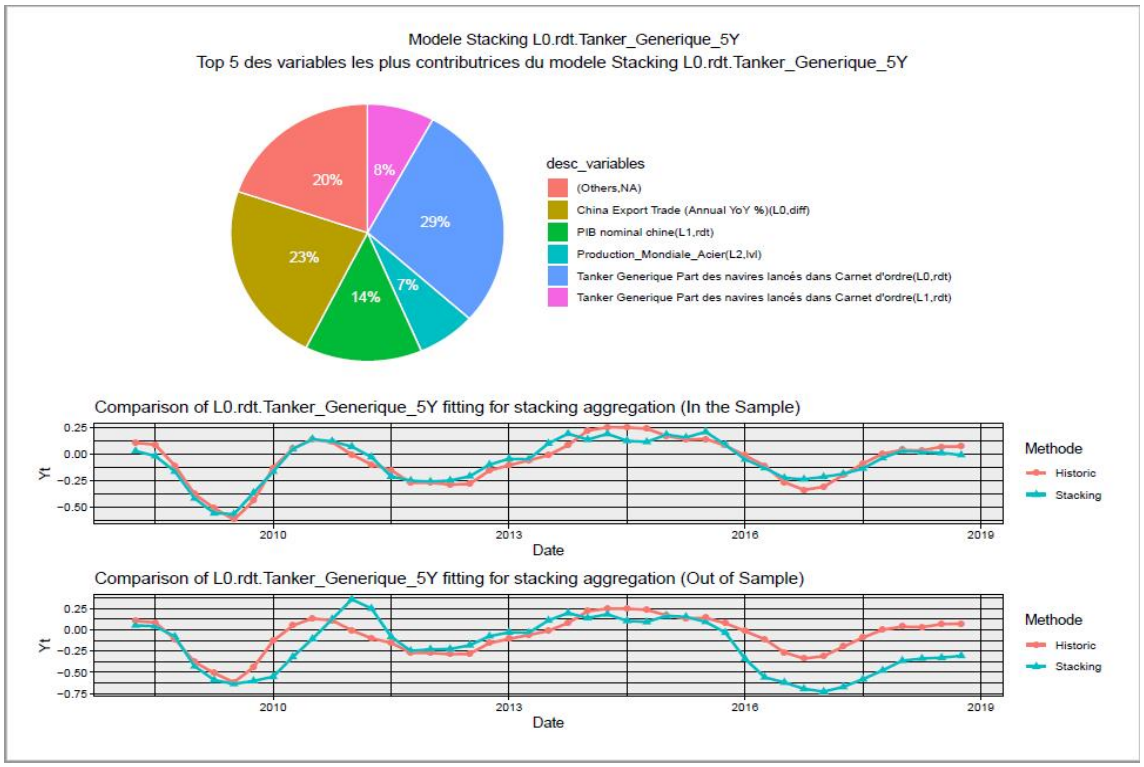
- **L0.rdt.Tanker_Aframax_5Y**



- **L0.rdt.Tanker_Panamax_5Y**

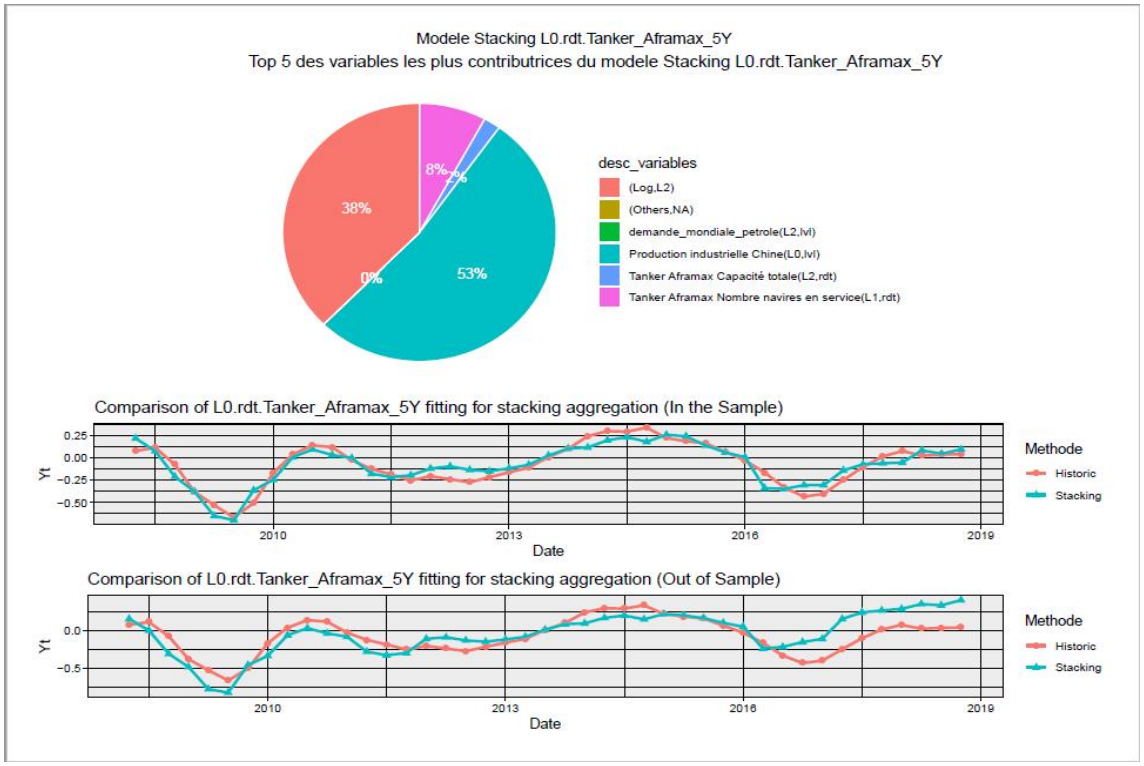


- **L0.rdt.Tanker_Generique_5Y**

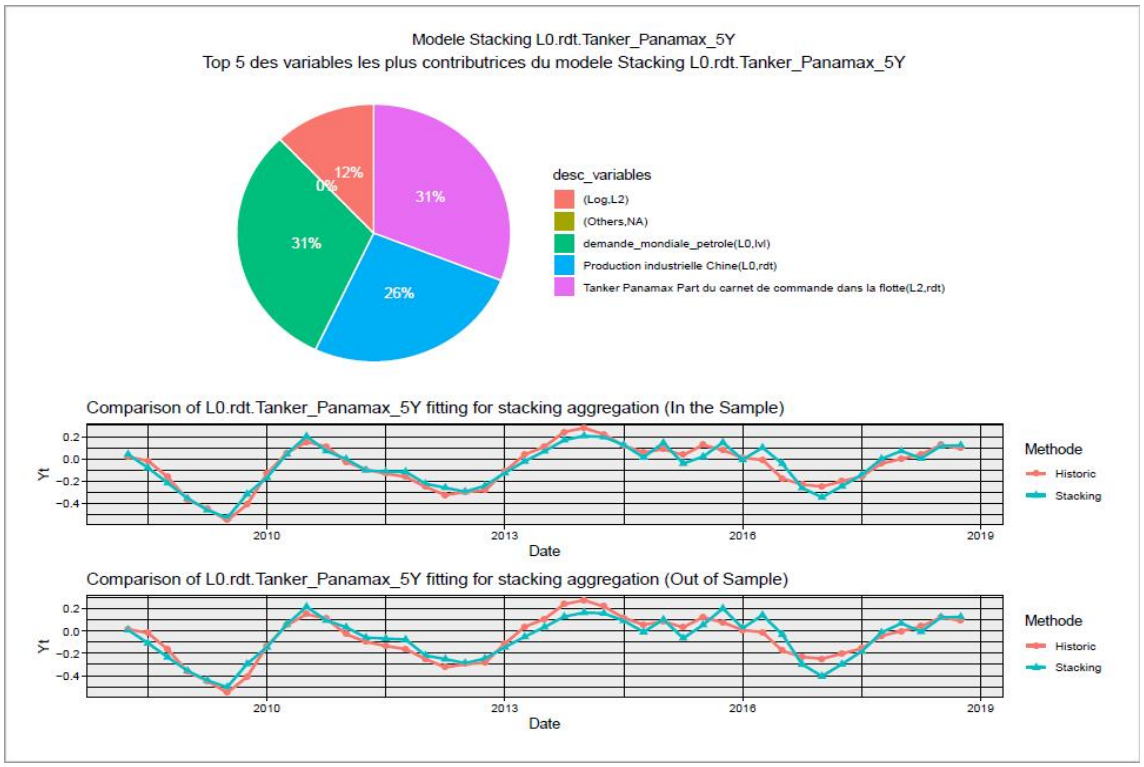


C.2 Transformation logarithmique

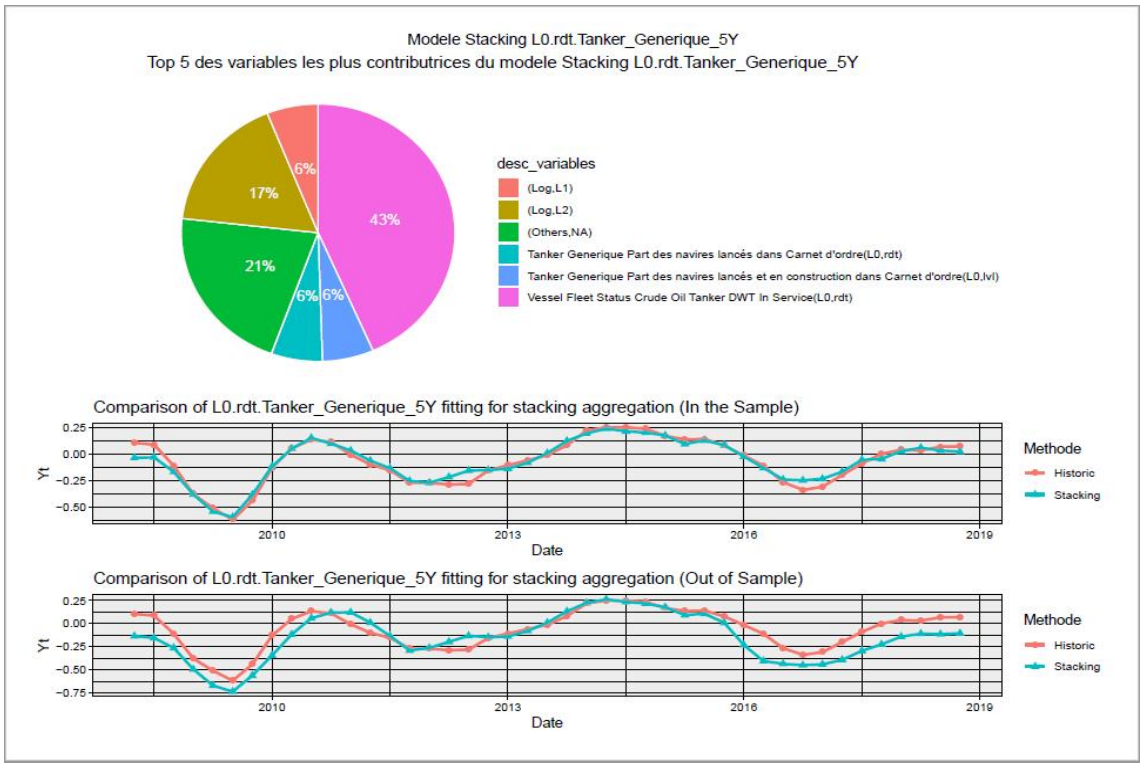
- **L0.rdt.Tanker__Aframax__5Y**



- **L0.rdt.Tanker__Panamax__5Y**



- **L0.rdt.Tanker__Generique__5Y**



C.3 Graphes de prédictions de modèles finalement approuvés

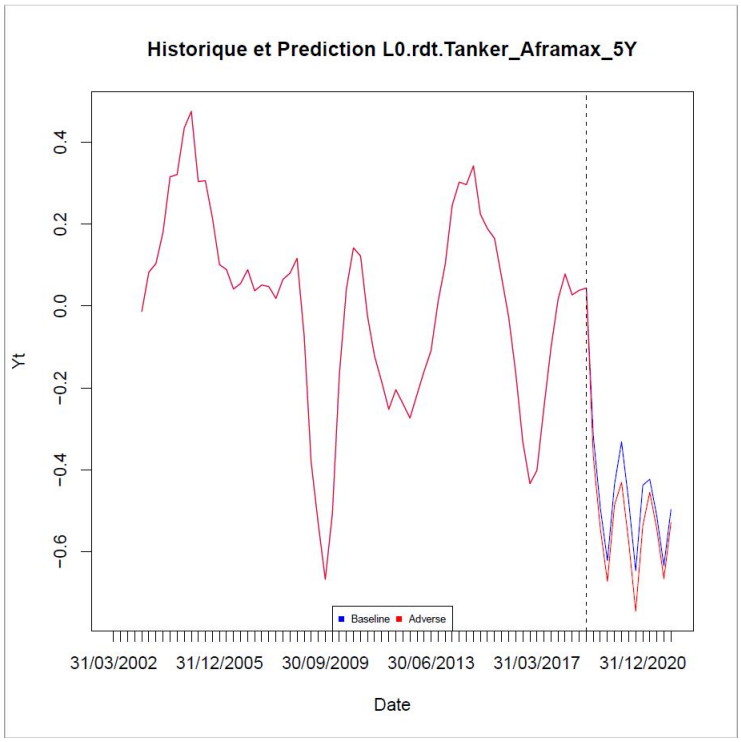


FIGURE 3 – Le graphe de réalisations historiques et de prédictions de variable L0.rdt.Tanker_Aframax_5Y.

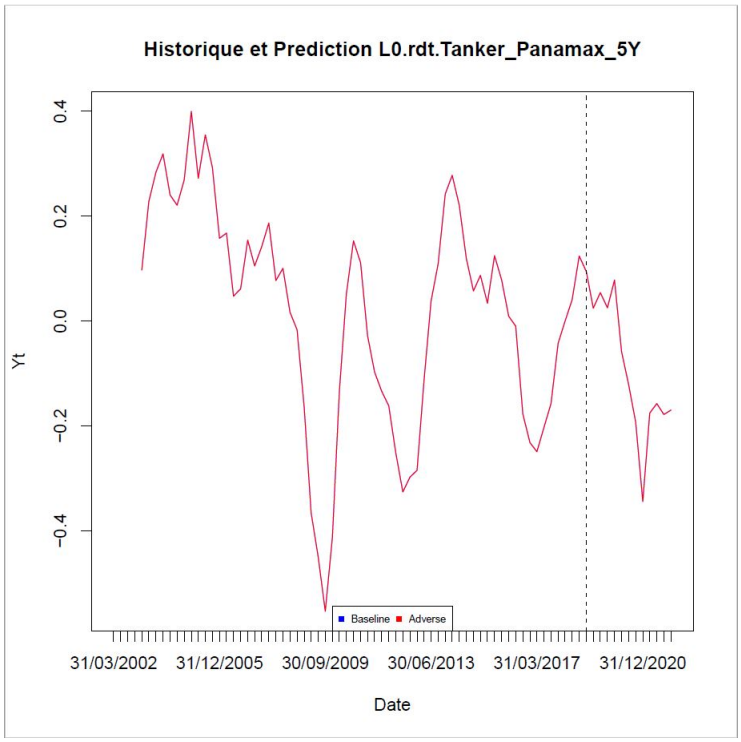


FIGURE 4 – Le graphe de réalisations historiques et de prédictions de variable L0.rdt.Tanker_Panamax_5Y.

On note la même prédiction donnée par les deux scénarios, ce qui est expliqué par le fait que les variables décrites par CASA ECO et existantes dans les deux scénarios n'entrent pas dans le modèle final comme variables explicatives, seulement des variables prédites par la méthode historique existent dans le modèle.

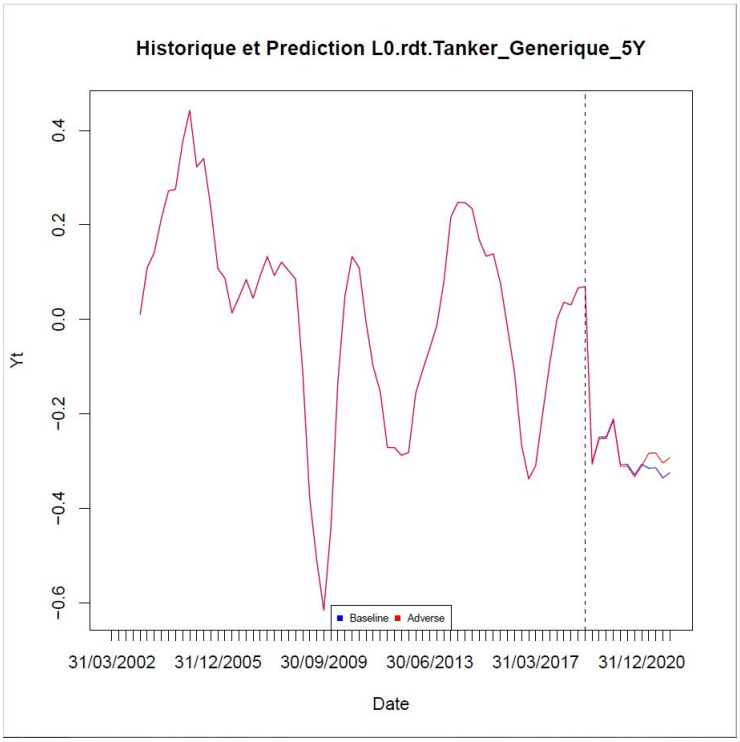


FIGURE 5 – Le graphe de réalisations historiques et de prédictions de variable L0.rdt.Tanker_Generique_5Y.

Annexe C

Les résultats de modélisation avec sélection de variables :

- **L0.lvl.pente.Bulker_Panamax_20Y**

Modélisation avec toutes les variables explicatives disponibles					
RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
0.02	0.94	0.88	17	112	1.16 hrs

Modélisation avec les variables sélectionnées						
Algorithme	RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
LASSO modifié	0.03	0.93	0.88	12	56	5.46 mins
Gradient boosting	0.03	0.93	0.81	12	47	3.07 mins
Random forest	0.03	0.93	0.81	12	29	1.13 mins
Boruta	0.03	0.93	0.81	12	56	5.48 mins
Baglasso	0.03	0.93	0.81	7	37	1.73 mins

- **L0.rdt.Bulker_Capesize_5Y**

Modélisation avec toutes les variables explicatives disponibles					
RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
0.11	0.93	0.86	14	112	1.16 hrs

Modélisation avec les variables sélectionnées						
Algorithme	RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
LASSO modifié	0.12	0.93	0.86	9	66	8.61 mins
Gradient boosting	0.16	0.91	0.85	5	37	1.63 mins
Random forest	0.17	0.9	0.79	5	44	2.54 mins
Boruta	0.14	0.91	0.86	5	56	5.42 mins
Baglasso	0.13	0.92	0.86	5	42	2.34 mins

- **L0.rdt.Bulker_Generique_5Y**

Modélisation avec toutes les variables explicatives disponibles					
RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
0.1	0.94	0.88	11	112	1.16 hrs

Modélisation avec les variables sélectionnées						
Algorithme	RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
LASSO modifié	0.11	0.93	0.9	9	63	8.043 mins
Gradient boosting	0.11	0.93	0.9	8	57	5.73 mins
Random forest	0.14	0.9	0.87	4	27	56.77 secs
Boruta	0.11	0.92	0.89	9	56	5.47 mins
Baglasso	0.12	0.92	0.85	6	42	2.3 mins

- **L0.rdt.Bulker_Panamax_5Y (jeu de données L2)**

Modélisation avec toutes les variables explicatives disponibles					
RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
0.1	0.92	0.88	12	168	12 hrs

Modélisation avec les variables sélectionnées						
Algorithme	RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
LASSO modifié	0.12	0.91	0.74	7	70	1.94 mins
Gradient boosting	0.13	0.91	0.83	9	63	8.35 mins
Random forest	0.15	0.89	0.75	2	25	53.74 secs
Boruta	0.14	0.89	0.76	6	31	1.26 mins
Baglasso	0.12	0.91	0.88	5	51	4.17 mins

- **L0.rdt.Tanker_VLCC_5Y (jeu de données L2)**

Modélisation avec toutes les variables explicatives disponibles					
RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
0.08	0.93	0.83	4	117	1.38 hrs

Modélisation avec les variables sélectionnées						
Algorithme	RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
LASSO modifié	0.09	0.92	0.84	2	56	4.17 mins
Gradient boosting	0.14	0.85	0.62	2	51	4.21 mins
Random forest	0.14	0.89	0.63	1	42	2.39 mins
Boruta	0.18	0.82	0.39	1	53	4.83 mins
Baglasso	NA	NA	NA	NA	NA	NA

- **L0.lvl.pente.Tanker_Suezmax_15Y (jeu de données L2)**

Modélisation avec toutes les variables explicatives disponibles					
RMSE	R2.is	R2.oos	Nombre de sous modèles acceptés	Nombre de variables explicatives	Temps de calcul
0.05	0.87	0.6	2	117	1.38 hrs

Pas de modèles trouvés avec la sélection de variables.