Prédiction de la performance boursière, une étude comparative entre modélisations économétriques et modélisations basées sur l'intelligence artificielle

Manel Labidi[1], Ying Zhang[1], Matthieu Petit Guillaume[1], Aurélien Krauth[1]

[1] Leviatan, 725 Bd Robert Barrier, 73100 Aix-les-Bains

{m.labidi, y.zhang, matthieu, aurelien}@leviatan.fr

Résumé

Dans cet article nous présentons une étude comparative des performances des modèles économétriques (modèle de Mundlak et modèle GEE-Logit) et ceux issus de l'intelligence artificielle comme le modèle en empilement et le modèle ensembliste intégrant XGBoost et LightGBM, ainsi que les modèles d'apprentissage profond (LSTM, GRU, encodeur-décodeur basé sur les Transformers, TCN) dans une tâche de classification de titres cotés en titres sous-performants et titres surperformants, pour un horizon d'investissement à un an. Nous utilisons des données historiques annuelles de 2019 à 2021. Les résultats montrent qu'un modèle de classification en empilement surpasse les autres modèles et offre un meilleur équilibre entre le taux de vrais positifs (70%) et le taux de vrais négatifs (67%).

Mots-clés

Gestion de portefeuilles, décision d'investissement, eXtreme Gradient Boosting, Long Short-Term Memory, Light Gradient Boosting, Gated Recurrent Unit, Temporal Convolutional Network, modèle à pile, modèle GEE-Logit, modèle de Mundlak.

Abstract

In this article, we present a comparative study of the performance of econometric models (Mundlak model and GEE-Logit model) and artificial intelligence-based models, such as stacking model and ensemble model integrating XGBoost and LightGBM, as well as deep learning models (LSTM, GRU, Transformer-based encoder-decoder, TCN) in a classification task of listed securities into underperforming and outperforming stocks, with a one-year investment horizon. We use annual historical data from 2019 to 2021. The results show that a stacking classification model outperforms the other models and offers a better balance between the true positive rate (70%) and the true negative rate (67%).

Keywords

Portfolio management, investment decision, eXtreme Gradient Boosting, Long Short-Term Memory, Light Gradient Boosting, Gated Recurrent Unit, Temporal Convolutional Network, stacking model, GEE-Logit model, Mundlak model

1 Introduction

Les techniques d'intelligence artificielle (IA) sont aujourd'hui de plus en plus déployées en finance, notamment dans la gestion d'actifs financiers, les algorithmes de trading, les crypto-actifs, l'évaluation du risque de crédit, etc. L'intégration de l'IA en finance serait très avantageuse pour les sociétés financières en leur permettant le développement d'un avantage concurrentiel, en améliorant leur efficacité, notamment par la réduction des coûts et l'optimisation de la productivité, comme le souligne l'OCDE [32] dans son rapport sur l'IA et la finance. L'utilisation de l'IA permet de réduire les coûts liés à la gestion des actifs financiers, en utilisant des algorithmes de trading et des outils capables d'analyser rapidement des données volumineuses et parfois peu structurées, favorisant ainsi la prise de décision en temps opportun.

La sélection d'actions représente un processus à la fois complexe et fondamental en gestion de portefeuille. Les investisseurs mobilisent principalement deux approches complémentaires : l'analyse fondamentale, qui évalue la valeur intrinsèque des titres afin d'identifier ceux sous-évalués ou surévalués par le marché, et l'analyse technique, qui décode des indicateurs spécifiques tels que les moyennes mobiles, le momentum et les niveaux de support pour anticiper les évolutions du marché. Ces indicateurs techniques permettent de détecter les tendances futures haussières, baissières ou neutres et d'ajuster stratégiquement l'exposition des portefeuilles. Le stock-screening constitue une méthode alternative efficace, consistant à filtrer les titres selon un ensemble prédéfini de critères, combinant souvent indicateurs fondamentaux et techniques. Néanmoins, la profusion de critères disponibles (plusieurs centaines) rend cette démarche particulièrement exigeante et nécessite une expertise pointue pour des résultats optimaux.

L'avènement de l'intelligence artificielle en finance a transformé ces approches traditionnelles en offrant de nouvelles méthodes pour modéliser les relations complexes entre les variables de marché, dépassant ainsi les limites des modèles linéaires. Dans le domaine de la gestion d'actifs, l'IA per-

met désormais de tester les performances de portefeuilles selon des milliers de scénarios, renforçant considérablement la gestion du risque et l'efficacité des décisions d'investissement.

La sélection de titres constitue une étape cruciale dans la constitution d'un portefeuille diversifié, nécessitant la prise en compte des objectifs d'investissement (à court, moyen et long terme) et des contraintes propres à chaque investisseur (ressources financières, tolérance au risque, etc.). Les investisseurs présentent des profils intrinsèquement hétérogènes (en termes d'exposition au risque, de temporalité, de contraintes de liquidité, etc.), et des objectifs variés (croissance du capital, génération de revenus, préservation du capital, etc.) qui influencent significativement leurs décisions d'investissement [24, 29, 38]. Si certains privilégient des stratégies à court terme pour des profits rapides en spéculant sur les fluctuations de marché [2], d'autres adoptent une approche de diversification et d'optimisation du risque en investissant à moyen, ou long terme [29].

Dans cette étude, nous poursuivons un double objectif. Premièrement, nous comparons plusieurs familles de modèles (méthodes économétriques, algorithmes d'apprentissage automatique et architectures d'apprentissage profond) dans une tâche de classification des actions cotées en titres surperformants ou sous-performants dans un objectif d'aide à la prise de décision d'investissement à moyen terme (1 an). Deuxièmement, nous évaluons l'efficacité de notre système de classification basé sur un modèle en empilement. Notre approche parvient à distinguer les titres surperformants des titres sous-performants, et pourrait constituer un outil utile, tant pour les investisseurs individuels, que les gestionnaires dans la constitution du portefeuille optimal.

2 Application de l'IA en finance

Nous constatons depuis quelques années, l'intérêt grandissant pour l'utilisation de l'intelligence artificielle dans les modélisations financières. Cet engouement pourrait s'expliquer par les limites des modèles traditionnels et la capacité de l'IA à modéliser des relations complexes avec une meilleure performance. Malgré leur nature opaque, les méthodes d'apprentissage automatique parviennent à distinguer les actions sous-évaluées de celles surévaluées par le marché boursier, et à identifier la plupart des anomalies boursières [1]. L'OCDE [32] rapporte que l'IA touche un éventail de plus en plus large d'activités au sein des entreprises financières. Par exemple, auprès des fonds spéculatifs, l'IA est utilisée à hauteur de 67% dans la génération d'idées, dans la construction de portefeuilles (58%), dans la gestion de risque (33%), et dans l'exécution de transactions financières (27%).

L'évaluation des actifs financiers (asset pricing) est un domaine de recherche très important en finance de marché, et en constante évolution. Son objectif principal est de développer des modèles permettant, d'une part, de comprendre la formation des prix sur les marchés boursiers et la manière dont les investisseurs évaluent les actions, obligations, options, et autres instruments financiers. Et d'autre part, d'identifier les facteurs expliquant les rentabilités boursières. Plusieurs modélisations théoriques existent déjà (Capital Asset Pricing Model (CAPM) [35], Arbitrage Pricing Theory (APT) [34], les modèles de Fama-French à trois facteurs [15] et à cinq facteurs [16], le modèle de Black-Scholes [4]).

De nombreux travaux de recherche en finance ont utilisé les modélisations proposées par l'intelligence artificielle pour répondre à des problématiques de marchés financiers. Les arbres décisionnels ont été utilisés dans l'identification des facteurs déterminants des rentabilités boursières. Nous pouvons citer les travaux de Coqueret et Guida [12] dans lesquels les auteurs utilisent les forêts d'arbres décisionnels de régression afin d'identifier les caractéristiques d'entreprises qui expliquent les rendements boursiers futurs. Parmi trente variables étudiées, les auteurs concluent que le momentum représente le facteur le plus important dans l'explication des rentabilités boursières, suivi par la volatilité, puis les indicateurs de liquidité.

En se basant sur les résultats des travaux de recherche de Green et al. [20] concluant au déclin significatif du pouvoir prédictif de 94 caractéristiques d'entreprises cotées, Han et al. [23] examinent la validité de ce constat scientifique. Au lieu d'utiliser une modélisation basée sur une régression linéaire par les MCO (Moindres Carrés Ordinaires), les auteurs utilisent un modèle combiné basé sur deux algorithmes d'apprentissage automatique, le LASSO (Least Absolute Shrinkage and Selection Operator) et ElasticNet. Leurs résultats mettent en évidence que le pouvoir prédictif des variables d'intérêt, au contraire, n'a pas diminué. D'après les auteurs, 30 variables sont pertinentes pour la prédiction des rendements boursiers sur la période de l'étude. Chen et al. [8] développent un modèle intégrant une combinaison de trois modèles d'IA : un réseau de neurones à propagation avant pour capter la non-linéarité du marché, un réseau de neurones LSTM, et un réseau antagoniste génératif. Les auteurs montrent que leur modèle offre une performance supérieure à celle des modèles classiques. Gu et al. [21] effectuent une analyse comparative des modèles d'évaluation de la prime de risque entre un ensemble de méthodes traditionnelles et un ensemble de modèles d'apprentissage automatique et concluent à la performance supérieure de l'IA dans la compréhension de l'évolution des prix des actifs financiers et de la prime de risque sur les marchés boursiers. Cette meilleure performance résulte de la capacité de l'IA à modéliser des interactions non linéaires. Chinco et al. [10] utilisent l'intelligence artificielle pour prédire les rendements à un intervalle de temps d'une minute en utilisant une régression LASSO avec les rendements passés comme variables prédictives.

Ces études démontrent que les modèles d'intelligence artificielle constituent non seulement une alternative prometteuse aux modèles traditionnels d'évaluation des actifs financiers, mais offrent également un avantage significatif grâce à leur capacité à capturer les relations non linéaires complexes, à intégrer de multiples facteurs explicatifs, et à s'adapter aux dynamiques changeantes des marchés financiers, ouvrant de nouvelles perspectives pour la prédiction

des rentabilités et la compréhension des mécanismes de formation des prix.

3 Modèles d'intelligence artificielle

Contrairement aux modèles econométriques, les approches d'intelligence artificielle permettent de modéliser des relations non linéaires complexes, offrant ainsi un cadre plus adapté à l'analyse des données financières. Parmi ces approches, les d'arbres décisionnels introduits par Breiman et al. [6]. L'évolution de ces modèles a conduit au développement de techniques d'ensemble comme l'agrégation bootstrap (bagging) [5] où plusieurs arbres sont entraînés sur différents échantillons puis combinés, et les forêts aléatoires [5] ajoutant une sélection aléatoire des variables. D'autres algorithmes d'apprentissage automatique ont émergé, notamment les machines à vecteurs de support (SVM) [13], le AdaBoost [18], le XGBoost [9], et LightGBM [25], chacun apportant des améliorations en termes de performances ou d'efficacité computationnelle.
Au-delà de ces modèles de machine learning traditionnels, l'évolution des capacités computationnelles a permis le développement de l'apprentissage profond (Deep Learning), qui comprend diverses architectures adaptées aux données financières. Les réseaux de neurones récurrents (RNN) sont utilisés pour la modélisation de séries temporelles grâce à leur capacité à capturer les dépendances temporelles, malgré leur limitation concernant l'atténuation du gradient. Les architectures basées sur les Transformers [40] offrent une alternative aux architectures récurrentes avec leur mécanisme d'attention permettant de capturer efficacement les dépendances à long terme. Les modèles LSTM [36], en introduisant des états de mémoire à long terme et un mécanisme de portes, permettent de contrôler le flux d'information à travers le réseau. Les Gated Recurrent Units (GRU) [14] constituent une alternative aux LSTM, offrant une architecture plus simple avec moins de paramètres, tout en maintenant une capacité de mémorisation similaire. Les Temporal Convolutional Networks (TCN) [11] sont des réseaux neuronaux basés sur des convolutions causales, permettant de modéliser les séries temporelles sans les contraintes des architectures récurrentes.
Cette revue non exhaustive présente des approches d'IA appliquées à la finance. Il convient également de noter l'existence d'autres architectures importantes comme les réseaux de neurones à convolution (CNN) [33], les réseaux antagonistes génératifs (GAN) [19], les modèles d'apprentissage par renforcement pour l'optimisation de portefeuille, les approches hybrides, etc. Le choix du modèle optimal dépend des spécificités du problème financier abordé, des caractéristiques des données disponibles, et des contraintes computationnelles.

4 Méthodologie

Dans cette section, nous présentons la méthodologie adoptée pour répondre à la problématique de classification des titres cotés en titres surperformants et titres sous-performants dans les décisions d'investissements à moyen terme. En premier lieu, nous présentons un descriptif du jeu de données, notamment les données d'entrée et les données de sortie. En second lieu, nous détaillons la stratégie adoptée pour la sélection des caractéristiques ainsi que les résultats de celle-ci. Ensuite, nous abordons la méthode adoptée pour l'entraînement des modèles et l'optimisation des hyperparamètres. Puis, nous présentons les différentes modélisations choisies. Enfin, nous présentons et discutons les résultats.

4.1 DATA

4.1.1 Jeu de données

Le jeu de données utilisé dans cette recherche provient de la base de données Factset. Il comprend des données comptables et financières relatives à 1 935 entreprises cotées non financières, observées sur trois années consécutives de 2019 à 2021.

4.1.2 Données d'entrée

Le jeu de données présente des indicateurs comptables et financiers. Les indicateurs comptables incluent le ratio d'endettement net, la rentabilité des capitaux propres, la rentabilité de l'actif, et le ratio de liquidité. Les indicateurs financiers incluent le ratio valeur de marché/valeur comptable, le dividende, les rendements boursiers passés, la capitalisation boursière, et le volume d'échange. Les deux dernières variables présentent un taux élevée de données manquantes et ne sont pas retenues. Une seule variable binaire est présente : la place de cotation.

4.1.3 Données de sortie

La donnée de sortie est une variable binaire qui prend la valeur de 1 en cas de surperformance boursière, et 0 en cas de sous-performance au cours de l'année 2021. La performance boursière est définie par rapport à celle de l'indice boursier américain S&P500, calculée sur les dix dernières années à compter de 2011. Nous avons utilisé la librairie yfinance (Yahoo finance) pour récupérer les données historiques du prix de clôture de l'indice. Puis, nous avons calculé sa performance boursière moyenne entre 2011 et 2021. Elle est autour de 10%. Nous utilisons cette performance comme seuil de distinction des titres surperformants de ceux sous-performants.
Ainsi, les entreprises qui ont une rentabilité boursière en 2021 supérieure à 10% sont considérées comme surperformantes (classe 1) et celles ayant une rentabilité boursière inférieure ou égale à 10% sont considérées comme sous-performantes (classe 0). Le tableau 1 présente la répartition des classes de notre échantillon.

Données de sortie	**Proportion**
0 : sous-performance boursière	35 %
1 : surperformance boursière	65 %
Total : 1 935 entreprises	**100 %**

TABLE 1 – Données de sortie

4.2 Ingénierie des caractéristiques

L'ingénierie des caractéristiques ou feature engineering est une étape cruciale dans la construction d'un modèle de prédiction. C'est un processus qui vise à sélectionner les données d'entrée les plus appropriées. La sélection des données d'entrée est d'autant plus importante en présence d'un jeu de données de petite taille qui conduit inévitablement à des problèmes de surajustement. Plusieurs méthodes permettent la sélection des attributs les plus importants. Nous avons choisi quatre méthodes différentes. Puis nous avons combiné les résultats issus de chaque méthode pour identifier la liste des attributs à retenir. Les quatre méthodes sont l'élimination récursive (RFE - Recursive Feature Elimination) [22], la sélection basée sur le modèle LASSO [39], l'approche de Kofi et al. [31], et le gradient boosting decision tree [9]. Le tableau 2 ci-dessous présente les résultats des quatre processus d'ingénierie des caractéristiques. Pour être retenue, une variable doit être considérée comme étant importante par au moins 3 méthodes.

Caractéristiques	RFE	LASSO	RF	GBDT
Dividende	True	False	True	True
Liquidité	True	False	True	True
Rentabilité boursière passée	True	False	True	True
Ratio valeur de marché/valeur comptable	True	False	True	True
Ratio d'endettement net	True	False	True	True
Rentabilité des capitaux propres	True	False	True	True
Rentabilité de l'actif	True	False	True	False
Place de cotation	True	False	False	False

*N*ote : True signifie que l'attribut est important. False signifie qu'il ne l'est pas.

TABLE 2 – Ingénierie des caractéristiques

4.3 Optimisation et évaluation

Compte tenu de la taille limitée de notre échantillon et afin d'éviter le problème de surajustement, les données sont réparties en un ensemble d'entraînement (80%), de validation (10%), et de test (10%) lorsqu'on utilise des modèles de machine learning et de deep learning. En économétrie, une approche courante consiste à diviser le jeu de données en deux ensembles : in-sample (80%), utilisé pour l'estimation des paramètres du modèle prédéfini, et out-of-sample (20%), destiné à l'évaluation des performances prédictives du modèle. La mise à l'échelle des données est effectuée par standardisation (Z-Score). L'optimisation des hyperparamètres a été réalisée selon une approche méthodologique rigoureuse, combinant les modules GridSearch et Cross Validation de Scikit – Learn pour les modèles ensemblistes, et Optuna [37] pour les modèles de Deep Learning, avec une validation croisée à k-blocs fixée à 3.
La validation croisée à k-blocs a été spécifiquement choisie pour minimiser les risques de surajustement inhérents aux échantillons de petite taille, en permettant l'entraînement sur k-1 sous-ensembles et la validation sur le sous-ensemble restant, avec une rotation systématique des blocs. L'optimisation est réalisée en maximisant le score F1, une métrique qui combine précision et rappel afin de mieux évaluer les performances en présence de données déséquilibrées. Après avoir sélectionné les meilleurs modèles par validation croisée, une évaluation finale a été systématiquement conduite sur le jeu de données de test indépendant.

4.4 Expérimentations et résultats

4.4.1 Environnement de travail

Notre environnement de travail est une machine locale équipée d'un GPU, de 8 CPU et de 8 Go de RAM. Le language de programmation utilisé est Python. Les bibliothèques principalement utilisées pour les calculs financiers, la construction, l'entraînement, et le test des modèles sont : Numpy, Pandas, yfinance, Scikit-Learn, Tensorflow et Optuna [37].

4.5 Modèles utilisés

4.5.1 Modèles économétriques

La modélisation en finance repose sur diverses méthodes économétriques pour analyser les relations entre variables économiques et comportements financiers. Parmi les plus utilisées, on trouve : la régression par la méthode des moindres carrés ordinaires (OLS-Ordinary Least Squares), adaptés aux relations linéaires et supposant une variance constante des erreurs (homoscédasticité) ; la régression par les moindres carrés généralisés (GLS- Generalized Least Squares) ; les modèles linéaires généralisés (GLM), qui capturent des relations plus complexes en adaptant la distribution des erreurs comme les modèles Logit et Probit, employés pour estimer les probabilités d'événements binaires, comme le risque de faillite d'une entreprise, etc. Les modèles autorégressifs et leurs extensions telles que ARMA[1] ou ARIMA[2], ARCH[3] ou GARCH[4] [17] sont utilisés pour les séries temporelles, où la valeur d'une variable à un instant donné dépend de ses valeurs passées.
Le choix du modèle approprié repose sur plusieurs critères, tels que le résultat des tests statistiques portant sur la distribution des variables explicatives (les entrées) et les termes d'erreur, la nature de la variable de sortie (continue ou discrète), etc. La performance future, conceptualisée comme une variable binaire (0 ou 1), nécessite une approche de modélisation adaptée aux données discrètes et à la distribution non normale. Deux modèles ont été sélectionnés pour notre analyse : le modèle GEE-Logit avec structure d'indépendance [28], et le modèle de Mundlak [30].

a. Modélisation des données panel

En présence de données panel (observations répétées pour un même ensemble d'unités statistiques sur plusieurs périodes), les hypothèses sous-jacentes aux moindres carrés

1. Autoregressive moving-average model
2. Autoregressive Integrated Moving Average model
3. Autoregressive Conditional Heteroskedasticity model
4. Generalized AutoRegressive Conditional Heteroskedasticity model

généralisés sont violées (homoscédasticité, absence de corrélation des termes d'erreur et homogénéité des observations). Pour gérer l'hétérogénéité non observée, deux approches principales existent [3] : le modèle à effets fixes, où les caractéristiques inobservables des individus (α_i) sont constantes dans le temps et potentiellement corrélées avec les variables explicatives qu'il faut modéliser, et le modèle à effets aléatoires, qui considère ces caractéristiques comme des variables aléatoires indépendantes, issues d'une distribution de probabilité, et non corrélées avec les variables explicatives. Dans un modèle à effets fixes (Fixed Effects, FE), α_i est un paramètre fixe propre à chaque individu. Tandis que dans un modèle à effets aléatoires (Random Effects, RE), le paramètre α_i suit une loi normale centrée, $\alpha_i \sim \mathcal{N}(0, \sigma_\alpha^2)$, indépendamment des X_{it}.

b. Modèle logit

Le modèle logit est fondamental pour l'analyse des variables dépendantes binaires. Un modèle logit pour des données de panel peut être soit à effets fixes, soit à effets aléatoires et peut s'écrire sous la forme suivante :

$$P(y_{it} = 1|X_{it}, \alpha_i) = \frac{\exp(X_{it}\beta + \alpha_i)}{1 + \exp(X_{it}\beta + \alpha_i)}$$

où :

- y_{it} : variable binaire de classification pour l'individu i à l'instant t
- X_{it} : vecteur des variables explicatives
- β : vecteur des coefficients à estimer
- α_i : effets individuels fixes à estimer pour chaque individu en cas de modèle logit à effets fixes ou aléatoires. Dans le cas d'un modèle logit à effets aléatoires, c'est une variable aléatoire suivant une loi normale de moyenne 0 et de variance σ_α^2 indépendante des variables explicatives x_{it}.

c. GEE-Logit avec structure d'indépendance

Le modèle GEE (Generalized Estimating Equations) [28] étend le logit aux données de panel en tenant compte de la corrélation entre les observations d'une même entité. Avec une structure d'indépendance, le modèle GEE-Logit s'écrit :

$$g[E(y_{it} \mid X_{it})] = \log\left(\frac{E(y_{it} \mid X_{it})}{1 - E(y_{it} \mid X_{it})}\right) = X_{it}\beta$$

où :

- $g(\cdot)$: fonction de lien logit.
- $E(y_{it} \mid X_{it})$: espérance conditionnelle de y_{it}
- X_{it} : vecteur des variables explicatives
- β : vecteur des coefficients à estimer

Contrairement au logit standard, le GEE estime les paramètres via des équations d'estimation généralisées plutôt que par maximum de vraisemblance, offrant une robustesse accrue aux spécifications incorrectes de la structure de corrélation.

d. Modèle de Mundlak

Le modèle de Mundlak [30] représente une approche alternative pour traiter l'hétérogénéité inobservable en introduisant les moyennes des variables explicatives. Il s'écrit de la manière suivante :

$$P(y_{it} = 1|X_{it}, \alpha_i) = \frac{e^{X_{it}\beta + \bar{X}_i\gamma + \alpha_i}}{1 + e^{X_{it}\beta + \bar{X}_i\gamma + \alpha_i}}$$

- $\bar{X}_i$: moyennes par entité des variables explicatives
- α_i : effets aléatoires.

4.5.2 Modèle en empilement

L'apprentissage en pile (stacking) [41] est une technique d'ensemble qui combine plusieurs modèles prédictifs en utilisant un méta-modèle (dans notre cas, nous utilisons une régression logistique). Dans cette approche, deux modèles de base, XGBoost et LightGBM, sont entraînés sur les données d'origine. Soit un ensemble de données financières, composé d'un vecteur de caractéristiques $X \in R^{n\times d}$, et d'une variable cible binaire $y \in \{0, 1\}^n$. L'objectif est de construire un classificateur robuste, minimisant une fonction de perte tout en tenant compte de la nature des données. Le modèle en empilement repose sur une modélisation en deux niveaux :

- Un ensemble de n modèles de base f1,f2,...,fn, chacun entraîné sur (X, y).
- Un méta-modèle g, entraîné sur les prédictions hors échantillon des modèles de base pour apprendre une combinaison optimale des prédictions.

L'estimation des méta-caractéristiques repose sur une validation croisée en k folds (k=3), notée CV_k. Le jeu de données est divisé en k sous-échantillons $S_1, S_2, \ldots, S_k$. Pour chaque modèle f_j, il est entraîné sur k – 1 folds, et validé sur le fold restant. Les prédictions hors échantillon sont stockées et forment la matrice des méta-caractéristiques $Z \in R^{n\times m}$. Formellement, les méta-caractéristiques sont définies par :

$$Z_{i,j} = f_j(X_i), \quad \forall i \in S_{valid}, \quad \forall j \in \{1, \ldots, m\}$$

Une fois la matrice Z construite, le méta-modèle g est ajusté sur (Z, y). L'évaluation finale du modèle est réalisée sur les données test.

4.5.3 Modèle ensembliste

Nous utilisons un modèle ensembliste basé sur la moyenne des probabilités comme méthode d'agrégation des prédictions de XGBoost et de LightGBM. Pour un échantillon X, chaque modèle prédit la probabilité $p_{i,j}$ que l'échantillon appartienne à la classe j. La probabilité agrégée $\hat{p}_j$ pour la classe j est obtenue par la moyenne des probabilités prédites par chaque modèle :

$$\hat{p}_j = \frac{1}{n}\sum_{i=1}^{n} p_{i,j}$$

où n est le nombre de modèles (ici, n = 2, pour XGBoost et LightGBM). Cette méthode permet de réduire la variance des prédictions individuelles en combinant les forces complémentaires des modèles.

Dans le cas de la classification binaire, la classe prédite est celle pour laquelle la probabilité moyenne est maximale :

$$\hat{y} = \arg\max_{j}(\hat{p}_j)$$

La moyenne des probabilités permet ainsi d'améliorer la robustesse des prédictions en tirant parti de la diversité des modèles tout en maintenant une approche simple et efficace.

4.5.4 Modèles d'apprentissage profond

Nous analysons les données financières des entreprises cotées de 2019-2020 pour prédire leur performance en 2021, en exploitant quatre architectures de réseaux neuronaux : TCN, un encodeur-décodeur basé sur les Transformers, LSTM et GRU. Nous avons utilisé la librairie Optuna [37] afin d'optimiser les différents hyperparamètres. En plus de l'optimisation avec Optuna, l'entraînement est réalisé en validation croisée K-folds (k=3). Les modèles ont été évalués par la suite sur les données test. Nous utilisons un callback de planification de taux d'apprentissage qui implémente une stratégie de décroissance par paliers. Cette fonction réduit le taux d'apprentissage de moitié tous les 8 époques après la $16^{ème}$ époque. Nous présentons dans ce qui suit l'architecture de chaque modèle et les hyperparamètres après optimisation.

a. Architecture LSTM

L'architecture LSTM déployée se compose de quatre couches principales : deux couches LSTM séquentielles, la première maintenant la dimension temporelle de la séquence, tandis que la seconde génère une représentation vectorielle unique de la séquence. Chaque couche LSTM est suivie d'une couche de Dropout pour éviter le surapprentissage. Le modèle inclut ensuite une couche Dense avec activation ReLU pour capturer les relations non linéaires, et une couche de sortie avec activation sigmoïde, produisant une probabilité pour la classification binaire de la performance d'une entreprise (surperformante ou sous-performante). Les hyperparamètres utilisés dans l'entraînement du modèle LSTM sont :

- Couche LSTM 1 : 16 neurones
- Taux de dropout 1 : 0.15
- Couche LSTM 2 : 16 neurones
- Couche Dense : 8 neurones
- Taux de dropout 2 : 0.15
- Taux d'apprentissage initial : 0.0001
- Taille de batch : 32
- Nombre d'époques : 48

b. Architecture GRU

L'architecture GRU implémentée adopte une approche parallèle pour traiter séparément les données des deux années financières. Le modèle commence par séparer l'entrée bidimensionnelle en deux séquences distinctes via des couches Lambda, suivies d'un reshaping pour adapter le format aux couches GRU. Un Dropout est appliqué entre ces deux couches. Après traitement, les représentations vectorielles des deux années sont fusionnées par concaténation, puis traversent une couche Dense avec activation ReLU et une couche Dropout. La structure se termine par une couche de sortie sigmoïde produisant une probabilité pour la classification binaire de la performance boursière. Les hyperparamètres du modèle sont :

- Couche GRU 1 (unités) : 24
- Taux de dropout : 0.15
- Couche GRU 2 (unités) : 24
- Taux de dropout : 0.15
- Couche Dense (unités) : 32
- Taux d'apprentissage initial : 0.0001
- Taille de batch : 32
- Nombre d'époques : 56

c. Architecture encodeur-décodeur basée sur les Transformers

Cette architecture utilise le mécanisme d'attention multi-têtes pour traiter les données financières séquentielles. Le modèle repose sur un encodeur qui normalise d'abord les entrées, puis applique l'attention parallèle. Cette structure est complétée par deux connexions résiduelles et une couche feed-forward avec normalisation intermédiaire et un Dropout. Après encodage, les représentations sont agrégées par pooling global, puis traversent une couche Dense avec activation ReLU suivie d'un second Dropout. Une couche sigmoïde finale produit la classification binaire. Les hyper paramètres du modèle sont les suivants :

- Taille des têtes d'attention : 32
- Nombre de têtes : 4
- Dimension de la couche feed-forward : 32
- Unités de la couche Dense : 24
- Taux de dropout 1 : 0.1
- Taux de dropout 2 : 0.2
- Taux d'apprentissage initial : 0.0001
- Taille de batch : 64
- Nombre d'époques : 56

d. Architecture TCN

L'architecture TCN implémentée utilise des convolutions dilatées pour capturer les dépendances temporelles dans les données financières. Le modèle est composé de plusieurs blocs TCN, chacun comprenant deux couches Conv1D avec padding causal et activation ReLU, suivies de couches Dropout distinctes pour régularisation. La dilatation progressive élargit le champ réceptif sans augmenter le nombre de paramètres. Après le passage dans les blocs convolutionnels, un GlobalAveragePooling1D agrège les représentations, suivi d'une couche Dense (ReLU), d'une troisième couche Dropout et d'une couche sigmoïde finale pour la classification binaire. Les hyperparamètres sont les suivants :

- Nombre de filtres : 32
- Taille du noyau : 4
- Unités de la couche Dense : 24
- Taux de dropout 1, 2 et 3 : 0.15
- Taux d'apprentissage initial : 0.0001
- Facteurs de dilatation : [1, 2, 4]
- Taille de batch : 64
- Nombre d'époques : 32

4.6 Résultats

4.6.1 Résultats des modèles économétriques

Le test du rapport de vraisemblance pour les effets individuels a produit une statistique LR de 4804.27 (p-value < 0.001), excluant l'utilisation d'un modèle homogène (pooled model) et confirme la nécessité de prendre en compte le problème d'hétérogénéité. Pour discriminer entre effets fixes et effets aléatoires dans ce contexte de modèle logit, nous avons implémenté le test de Mundlak [30]. Le test indique une correlation significative entre les effets individuels et les variables explicatives.

Cette endogénéité détectée suggérerait normalement l'utilisation d'un modèle à effets fixes. Cependant, notre panel ne comprend que deux périodes d'observation (2019-2020), ce qui pose un défi méthodologique important. En effet, dans les modèles non linéaires comme le logit, les estimateurs à effets fixes conventionnels souffrent du problème des "paramètres incidents" avec des panels courts, conduisant à des estimations inconsistantes [27]. Pour résoudre ce dilemme, nous avons adopté l'approche de Mundlak-Chamberlain [7, 30] qui consiste à introduire dans le modèle de prédiction les moyennes individuelles des variables explicatives évoluant dans le temps. Cette méthode, bien établie dans la littérature économétrique [3, 42], permet de contrôler l'endogénéité tout en évitant les biais d'estimation associés aux effets fixes dans les panels courts.

Les résultats de ce modèle corrigé des moyennes de groupes sont présentés dans le tableau 5. Cette spécification constitue le meilleur compromis entre rigueur théorique et faisabilité empirique compte tenu des contraintes de nos données.

Le tableau 3 montre les résultats de régression du modèle GEE-logit avec structure d'indépendance. Les résultats du modèle GEE-Logit mettent en évidence l'influence de plusieurs variables financières sur la probabilité d'une surperformance boursière. Le ratio valeur de marché sur valeur comptable (PBV - Price to Book Value) présente un effet négatif significatif (p-value < 0.05) indiquant qu'une augmentation du PBV réduit la probabilité d'une surperformance boursière. Cela peut être contre intuitif mais ce résultat peut s'expliquer par une surévaluation des actions par rapport aux fondamentaux de l'entreprise, ce qui peut entraîner des attentes excessives du marché et une correction ultérieure des prix. La rentabilité des actifs (RA) et le niveau d'endettement net sur capital total (NDTC) ont des effets positifs significatifs (p-value < 0.001), avec des odds ratios respectifs de 1.421 et 1.298, soulignant leur rôle déterminant dans l'amélioration de la performance boursière.

Les résultats du modèle Mundlak présentés dans le tableau 4 montrent que les effets temporels des variables financières sont déterminants dans la prédiction de la performance boursière. Le *PBV_mean* a un effet négatif significatif (p-value < 0.1, coef. = -0.2309), suggérant qu'un *PBV_mean* élevé réduit la probabilité d'une surperformance boursière, possiblement en raison de corrections du marché. À l'inverse, la rentabilité moyenne des actifs (*RA_mean*) a un impact positif significatif (p-value < 0.1, coef. = 0.298), indiquant que des actifs plus rentables sont associés à une surperformance boursière. Le test d'endogénéité (p-value = 0.006) confirme que les effets individuels doivent être pris en compte, justifiant l'approche de Mundlak et (*GEELogit*).

Variables	Coefficients	Odds Ratio
Constante	0.594***	1.811
DIV	0.065	1.067
PBV	-0.168*	0.846
QR	-0.028	0.972
NDTC	0.261***	1.298
RE	0.016	1.016
RA	0.352***	1.421
TR	-0.040	0.961

Note : * p<0.05, ** p<0.01, *** p<0.001.
Nombre d'observations : 3 096. DIV : Dividende. PBV : Valeur de marché sur valeur comptable. QR : Ratio de liquidité. NDTC : ratio d'endettement. RA : Rentabilité de l'actif. RE : Rentabilité des capitaux propres. TR : Rendement boursier.
L'Odds Ratio : le facteur par lequel les chances (odds) qu'un événement se produise changent lorsque la valeur d'une variable explicative augmente d'une unité, toutes choses égales par ailleurs. l'Odds Ratio d'une variable est obtenu en calculant l'exponentielle de son coefficient e^{β}.

TABLE 3 – Résultats de régression du modèle GEE-Logit avec structure d'indépendance

Le tableau 5 présente les résultats de performances out-of-sample du modèle Mundlak et du modèle GEE-Logit. Les deux modèles présentent des performances similaires dans la classification des performances boursières, avec des métriques clés très proches : une exactitude autour de 68-69%, un score F1 proche de 79-80%, et un taux de vrais positifs élevé à 94%. Toutefois, ces modèles révèlent une faiblesse commune : un taux bas de vrais négatifs (environ 20%), indiquant une difficulté significative à identifier correctement les titres sous-performants. Cette caractéristique suggère que les modèles sont biaisés vers la détection des performances positives, avec un risque élevé de faux positifs pour les performances boursières faibles.

4.6.2 Résultats des modèles de machine learning

Les résultats d'entraînement des modèles XGBoost, LightGBM, du modèle ensembliste et du modèle en empilement révèlent des forces distinctes pour chaque approche. Le modèle en empilement présente le meilleur équilibre global entre les critères d'évaluation, bien qu'il ne soit pas optimal sur chaque métrique individuelle. Son taux de vrais positifs élevé en comparaison des autres modèles (67.22 %) permet de mieux identifier les cas négatifs, réduisant ainsi les faux positifs, un aspect crucial en investissement. Avec une précision de 78.57 % et un score F1 de 74.42 %, il atteint un compromis entre précision et sensibilité (70.68 %). Bien que cette dernière soit plus faible que pour d'autres modèles, elle reste acceptable et est compensée par une meilleure gestion des faux positifs, limitant ainsi les risques financiers. Les hyperparamètres utilisés sont les suivants :

Variables	Coefficients
Constante	0.6076***
Div_mean	0.1261
PBV_mean	-0.2309*
QR_mean	0.000
NDTC_mean	0.250
RE_mean	0.10
RA_mean	0.298*
TR_mean	0.0541

*Note :*** p<0.01, ** p<0.05, * p<0.1. Nombre d'observations : 3096. Test LR d'endogénéité : Statistique LR : 25.7730, Degrés de liberté : 7, P-value : 0.0006. Div_mean : moyenne des dividendes. PBV_mean : moyenne de la valeur de marché/valeur comptable. QR_mean : moyenne du ratio de liquidité. NTDC_mean : endettement moyen. RA_mean : moyenne de la rentabilité de l'actif. RE_mean : moyenne de la rentabilité des capitaux propres. TR_mean : moyenne des rendements boursiers. Les moyennes des variables sont calculées sur les années 2019 et 2020.

TABLE 4 – Résultats de régression du modèle de Mundlak

Métriques	Modèle Mundlak	GEE-Logit
Acc.	68.10%	68.60%
Prec.	68.51%	69.00%
F1	79.30%	79.60%
Rec. A	94.13%	94.05%
Rec. B	19.91 %	21.11%

Note : Acc. : accuracy. Prec. : précision, F1 : score F1. Rec. A : taux de vrais positifs (titres surperformants). Rec. B : taux de vrais négatifs (titres sous-performants)

TABLE 5 – Performance Out-of-Sample des modèles Mundlak et GEE-Logit avec structure d'indépendance

pour XGBoost, 100 estimateurs, une profondeur maximale de 4 et un taux d'apprentissage de 0.10. Pour LightGBM, les hyperparamètres sont les suivants : 100 estimateurs, une profondeur maximale de 5, un taux d'apprentissage de 0.10 et 31 feuilles.

Métriques	Empilement	Ensemble	XGB	LGBM
Acc.	68.73%	72.09%	73.39%	70.54%
Prec.	78.57%	73.74%	73.86%	73.52%
F1	74.42%	80.22%	81.44%	78.73%
Rec. A	70.68%	87.95%	90.76%	84.74%
Rec. B	67.22%	43.48%	42.03%	44.93%

Note : Acc. : accuracy. Prec. : précision, F1 : score F1. Rec. A : taux de vrais positifs (titres surperformants). Rec. B : taux de vrais négatifs (titres sous-performants). XGB : XGBoost. LGBM : LightGBM

TABLE 6 – Performance des modèles en empilement, ensembliste, XGBoost et LightGBM sur les données test

4.6.3 Résultats des modèles d'apprentissage profond

Les résultats, obtenus après optimisation des hyperparamètres avec Optuna [37] et ajustement des poids de classes, montrent que le TCN affiche une légère amélioration par rapport aux modèles LSTM, GRU et un encodeur-décodeur basé sur les Transformers, avec une exactitude de 67.12% contre environ 65% pour les autres, ainsi qu'un score F1 légèrement supérieur (66.78%). Les résultats montrent un taux de vrais négatifs faible pour tous les modèles (< 51%) indiquant une difficulté à détecter correctement les marchés baissiers. Cette limitation souligne la nécessité d'approfondir l'analyse, notamment via une exploration de modèles hybrides et un backtesting sur données réelles, afin d'évaluer si une amélioration observée se traduit par une performance exploitable en contexte de trading.

Modèle	LSTM	GRU	Enc.-dec.	TCN
Acc.	65.23%	65.57%	65.92%	67.12%
Prec.	65.53%	65.50%	65.70%	66.56%
F1	65.37%	65.53%	65.80%	66.78%
Rec. A	65.37%	65.53%	65.80%	66.78%
Rec. B	50.72%	48.79%	49.27%	49.75%

Note : Acc. : accuracy. Prec. : précision, F1 : score F1. Rec. A : taux de vrais positifs (titres surperformants). Rec. B : taux de vrais négatifs (titres sous-performants). Enc.-dec. : encodeur-décodeur.

TABLE 7 – Performance des modèles d'apprentissage profond sur les données test

4.7 Comparaison et discussion

Les modèles économétriques (Mundlak et GEE-Logit) présentent une forte asymétrie avec un taux de vrais positifs élevé (> 94%) mais un taux de vrais négatifs faible ($\sim$ 20%) montrant une capacité faible à discriminer efficacement les titres sous-performants. Ce résultat les rende inadaptés à la classification des performances financères futures. Les architectures d'apprentissage profond (LSTM, GRU, encodeur-décodeur basé sur les transformers, TCN) offrent un meilleur équilibre entre le taux de vrais positifs et le taux de vrais négatifs ($\sim$ 65% vs $\sim$ 50%) par rapport aux modèles économétriques mais reste toutefois insuffisant. Cette distribution plus équilibrée des performances s'explique par la capacité des réseaux neuronaux à modéliser des dépendances temporelles complexes et non linéaires dans les données financières. Cependant, le score F1 plus faible par rapport aux modèles d'ensemble suggère que ces architectures ne parviennent pas à extraire efficacement toute l'information pertinente des données analysées, particulièrement dans un contexte financier. Les méthodes d'ensemble et de boosting démontrent une supériorité sur la majorité des métriques. XGBoost se distingue particulièrement avec l'exactitude la plus élevée (73.39%) et le meilleur score F1 (81.44%), suggérant une meilleure capacité à identifier les signaux prédictifs pertinents, mais la capacité du modèle à distinguer les cas négatifs est faible.
L'approche par empilement offre quant à elle le meilleur équilibre entre le taux de vrais positifs (70.68%) et le taux de vrais négatifs (67.22%), constituant un compromis optimal lorsque les coûts associés aux deux types d'erreurs sont comparables. Cette caractéristique s'explique proba-

blement par la diversité des modèles de base intégrés dans la méta-architecture, permettant une réduction efficace de la variance prédictive tout en maintenant un biais acceptable. Ces résultats empiriques soulèvent des questions fondamentales concernant la modélisation prédictive en finance. La supériorité du modèle en empilement suggère que l'agrégation de modèles diversifiés permet de mieux capturer l'hétérogénéité des dynamiques financières, tandis que le déséquilibre prononcé entre le taux de vrais positifs et le taux de vrais négatifs observé dans les modèles économétriques met en évidence les limites des approches paramétriques traditionnelles face à la complexité des phénomènes financiers.

Conclusion

Dans ce travail de recherche nous menons une analyse comparative des modèles économétriques, d'apprentissage automatique et d'apprentissage profond pour la classification des actions cotées en fonction de leur performance boursière future. En exploitant les données de 2019 et 2020 pour prédire la performance boursière de l'année 2021, nous avons cherché à identifier les modèles les plus performants pour aider à la prise de décision d'investissement à moyen terme. Nos résultats montrent que, bien que les modèles économétriques (Mundlak et GEE-Logit) offrent une base solide pour l'analyse des titres boursiers, ils sont surpassés par les modèles d'apprentissage automatique et, en particulier, par le modèle en empilement basé sur XGBoost et LightGBM, qui se distingue comme l'approche la plus robuste.

L'analyse des performances met en évidence les limites des modèles d'apprentissage profond (LSTM, GRU, encodeur-décodeur basé sur le Transformers, TCN) dans notre contexte, particulièrement en ce qui concerne leur capacité à identifier correctement les titres sous-performants. Cette efficacité restreinte s'explique principalement par deux facteurs inhérents à notre jeu de données. D'une part, la taille limitée de l'échantillon constitue un obstacle majeur, les architectures d'apprentissage profond nécessitant généralement d'importants volumes de données pour optimiser leurs paramètres et généraliser efficacement. D'autre part, la brièveté de la séquence temporelle analysée (deux années seulement) affecte particulièrement les modèles récurrents comme le LSTM, dont la puissance prédictive se manifeste davantage sur des horizons temporels étendus permettant de capturer des dépendances à long terme. Les modèles XGBoost et LightGBM se montrent plus performants en termes de score F1 mais montrent également une limite dans la détection des titres sous-performants. Le modèle en empilement offre le meilleur équilibre dans la prédiction des deux classes.

L'interprétation des résultats ne peut se limiter aux scores globaux de performance (Score F1 et exactitude). Dans un contexte d'investissement, la nature des erreurs de classification est cruciale. Nous considérons que classifier à tort un titre comme faisant partie de la classe positive (i.e., un titre performant alors qu'il ne l'est pas) est plus préjudiciable qu'une erreur inverse. Cette observation s'appuie sur la théorie des perspectives de Kahneman et Tversky [24], qui est un pilier de la finance comportementale et de l'économie comportementale. Cette théorie démontre que les individus sont plus sensibles aux pertes potentielles qu'aux opportunités de gains manqués. En d'autres termes, un investisseur perçoit une perte financière comme plus douloureuse qu'un gain équivalent en valeur absolue.

De plus, les travaux de Kent et al. [26] montrent que les investisseurs ont tendance à surréagir aux mauvaises nouvelles et sous-réagir aux bonnes nouvelles, ce qui entraîne à la fois une vente excessive de titres après une annonce négative et un manque d'achats de titres malgré de bonnes nouvelles. Dans ce cadre, le modèle en empilement présente un avantage décisif car il réduit la probabilité d'erreurs, notamment celles qui surestiment la performance d'un titre. Grâce à sa meilleure capacité de généralisation et de gestion des faux positifs, il offre un outil plus fiable pour les investisseurs, en minimisant les risques.

Notre étude ouvre plusieurs pistes de recherche : Étendre la période d'étude sur au moins cinq ans, afin d'évaluer la robustesse des modèles sur plusieurs cycles de marché. Adapter le modèle à des horizons d'investissement plus courts (hebdomadaire, mensuel), pour répondre aux dynamiques de marché en temps réel.

Références

[1] D. S. Avramov and L. Metzker. Machine learning versus economic restrictions : Evidence from stock return predictability. *SSRN Electronic Journal*, 2019.

[2] B. M. Barber and T. Odean. Trading is hazardous to your wealth : The common stock investment performance of individual investors. *The Journal of Finance*, 55(2) :773–806, 2000.

[3] A. Bell and K. Jones. Explaining fixed effects : Random effects modeling of time- series cross-sectional and panel data. *Political Science Research and Methods*, 3(1) :133153, 2015.

[4] F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3) :637–657, 1973.

[5] L. Breiman. Bagging predictors. *Machine Learning*, 24 :123–140, 1996.

[6] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. 1984.

[7] G. Chamberlain. Multivariate regression models for panel data. *Journal of Econometrics*, 18(1) :5–46, 1982.

[8] L. Chen, M. Pelger, and J. Zhu. Deep learning in asset pricing. *SSRN Electronic Journal*, 2019.

[9] T. Chen and C. Guestrin. Xgboost : A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, aug 2016.

[10] A. Chinco, A. D. Clark-Joseph, and M. Ye. Sparse signals in the cross-section of returns. *Journal of Finance*, 74(1) :449–492, 2019.

[11] L. Colin, R. Vidal, A. Reiter, and G. D. Hager. Temporal convolutional networks : A unified approach to action segmentation, 2016.

[12] G. Coqueret and T. Guida. Stock returns and the cross-section of characteristics : A tree-based approach. *SSRN Electronic Journal*, (2010) :1–17, 2018.

[13] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20 :273–297, 1995.

[14] R. Dey and F. M. Salem. Gate-variants of gated recurrent unit (gru) neural networks. In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 1597–1600, 2017.

[15] E. F. Fama and K. R. French. The cross-section of expected stock returns. *Journal of Finance*, 47(2) :427–465, 1992.

[16] E. F. Fama and K. R. French. A five-factor asset pricing model. *Journal of Financial Economics*, 116(1) :1–22, 2015.

[17] C. Francq and J. M. Zakoian. *GARCH Models : Structure, Statistical Inference and Financial Applications*. Wiley, 2 edition, 2019.

[18] Y. Freund and R. E. Schapire. Experiments with a new boosting algorithm. In *Machine Learning : Proceedings of the Thirteenth International Conference*, pages 1–9, 1996.

[19] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial networks, 2014.

[20] J. Green, J. R. M. Hand, and X. F. Zhang. The characteristics that provide independent information about average u.s. monthly stock returns. *Review Financial Studies*, 30(12) :4389–4436, 2017.

[21] S. Gu, B. T. Kelly, and D. Xiu. Empirical asset pricing via machine learning. *Review of Financial Studies*, 33(5) :2223–2273, 2018.

[22] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46 :389–422, 2002.

[23] Y. Han, A. He, D. Rapach, and G. Zhou. How many firm characteristics drive us stock returns ? *SSRN Electronic Journal*, 2018.

[24] D. Kahneman and A. Tversky. Prospect theory : An analysis of decision under risk. *Econometrica*, 47(2) :263–292, 1979.

[25] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, and Q. Ye. Lightgbm : A highly efficient gradient boosting decision tree. In *31st Conference on Neural Information Processing Systems*, pages 3147–3155, 2017.

[26] D. Kent, D. Hirshleifer, and A. Subrahmanyam. Investor psychology and security market under- and overreactions. *The Journal of Finance*, 53 :1839–1885, 1998.

[27] T. Lancaster. The incidental parameter problem since 1948. *Journal of Econometrics*, 95(2) :391–413, 2000.

[28] K. Y. Liang and S. L. Zeger. Longitudinal data analysis using generalized linear models. *Biometrika*, 73(1) :13–22, 1986.

[29] H. Markowitz. Portfolio selection. *The Journal of Finance*, 7(1) :77–91, 1952.

[30] Y. Mundlak. On the pooling of time series and cross section data. *Econometrica*, 46(1) :69–85, 1978.

[31] I. K. Nti, A. F. Adekoya, and B. A. Weyori. Random forest based feature selection of macroeconomic variables for stock market prediction. *American Journal of Applied Sciences*, 16(7) :200–212, 2019.

[32] OECD. *Artificial Intelligence, Machine Learning and Big Data in Finance : Opportunities, Challenges, and Implications for Policy Makers*. OECD Publishing, 2021.

[33] K. O'Shea and R. Nash. An introduction to convolutional neural networks. *CoRR*, abs/1511.08458, 2015.

[34] S. A. Ross. The arbitrage theory of capital asset pricing. *Journal of Economic Theory*, 13(3) :341–360, 1976.

[35] W. F. Sharpe. Capital asset prices : A theory of market equilibrium under conditions of risk. *Journal of Finance*, 19(3) :425–442, 1964.

[36] R. C. Staudemeyer and E. Rothstein Morris. Understanding lstm – a tutorial into long short-term memory recurrent neural networks, 2019.

[37] A. Takuya, S. Shotaro, Y. Toshihiko, O. Takeru, and K. Masanori. Optuna : A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631. ACM, 2019.

[38] R. H. Thaler. *Advances in Behavioral Finance*. Russell Sage Foundation, 1993.

[39] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society : Series B (Methodological)*, 58(1) :267–288, 1996.

[40] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[41] David H. Wolpert. Stacked generalization. *Neural Networks*, 5(2) :241–259, 1992.

[42] J. Wooldridge. *Introduction à l'économétrie : une approche moderne*. De Boeck, Louvain-la-Neuve, 2010.