VU Eliot
DALBIN Léo
TOISOUL Xavier
DAI Lianghe
NGUYEN Mathis

Rapport de projet Midterm

1. Introduction et Objectifs

Les marchés financiers sont un terrain complexe où se croisent une multitude de facteurs économiques, techniques et comportementaux. Le S&P 500, un indice regroupant les 500 plus grandes entreprises cotées en bourse aux États-Unis, est souvent considéré comme un baromètre de l'économie américaine. Les mouvements de cet indice sont étroitement surveillés pour évaluer la confiance des investisseurs et les perspectives économiques des États-Unis.Prédire les prix et les rendements des actifs de cet indice est un défi de taille, mais qui peut offrir des opportunités significatives pour les investisseurs et les gestionnaires de portefeuille. Dans ce contexte, le machine learning, avec ses capacités à traiter d'énormes volumes de données et à apprendre à partir de ces données, offre de nouvelles perspectives pour l'analyse prédictive. Contrairement aux méthodes traditionnelles, les méthodes de machine learning peuvent détecter des patterns complexes et non linéaires qui pourraient échapper aux techniques d'analyse classique. Les techniques avancées d'apprentissage automatique, telles que les différents modèles de réseaux de neurones, peuvent se montrer particulièrement efficaces pour traiter des séries temporelles, qui sont courantes dans les données financières. Ces modèles sont capables de capturer les dépendances temporelles et de prévoir les tendances futures avec une plus grande précision. De plus, les algorithmes de machine learning peuvent être continuellement mis à jour avec de nouvelles données, améliorant ainsi constamment leurs performances prédictives.

Ce projet vise à utiliser des techniques avancées de machine learning pour prédire les prix et les rendements des actifs du S&P 500. Nous appliquerons des méthodes d'analyse de séries temporelles et de modélisation prédictive pour développer des modèles performants. De plus, nous évaluerons rigoureusement les performances de ces modèles à l'aide de métriques appropriées, et nous interpréterons les résultats pour proposer des recommandations d'investissement éclairées.

Objectifs du Projet

1. **Collecte et préparation des données financières** :
- Récupérer les données historiques du S&P 500, y compris les prix de clôture et les volumes, sur une période de 5 ans via des API comme Yahoo Finance ou Alpha Vantage.
2. **Développement d'un modèle prédictif** :
- Utiliser deux algorithmes de machine learning adaptés pour prédire l'évolution temporelle du S&P500

3. **Évaluation des performances du modèle** :

- Évaluer la précision et la robustesse des modèles prédictifs avec des métriques

4. **Interprétation des résultats et recommandations** :

- Analyser les performances des modèles et interpréter les résultats à l'aide de visualisations graphiques.
- Formuler des recommandations d'investissement basées sur les prédictions du modèle

2. Méthodologie (Préparation des données/ Choix des modèles)

2.1. Préparation des données

2.1.1. Importation des bibliothèques

Les bibliothèques utilisées Python sont:

- pandas et numpy pour la manipulation et l'analyse des données.
- matplotlib et seaborn pour les visualisations.
- sklearn pour les algorithmes de machine learning.
- statsmodels pour les analyses statistiques et ARIMA.
- tensorflow pour le développement du modèle LSTM.

2.1.2. Collecte des données

Les données, issues de Yahoo Finance via la bibliothèque yfinance, concernent les actifs définis dans les variables asset et macro_indicators. Il s'agit de données historiques sur les prix (ouverture, clôture, volume, etc.), collectées sur une période de 5 ans avec une fréquence quotidienne à l'aide de la fonction yf.download(), pour refléter au mieux les dynamiques de marché.

***Les Tickers collectés** :*

1. **Actifs principaux :**
   - ^GSPC (Indice S&P 500)
   - ^IXIC (Nasdaq Composite)
   - TSLA (Tesla)
   - LOTB.BR (Lotus Bakeries)
   - MC.PA (LVMH)
   - BLK (BlackRock)
   - AAPL (Apple)

2. **Indicateurs macroéconomiques :**
   - DX-Y.NYB (US Dollar Index)
   - ^VIX (Indice de volatilité)
   - ^TNX (Rendement des obligations à 10 ans)
   - CL=F (Prix du pétrole brut)
   - GC=F (Prix de l'or)

2.1.3. Exploration, Nettoyage, Transformation des données

Les valeurs manquantes ont été détectées et supprimées, tandis que les doublons ont été identifiés et éliminés pour garantir la cohérence et éviter les biais dans l'analyse.

Calcul des rendements logarithmiques (Log Returns):

Ensuite, pour mieux modéliser les séries temporelles et de stabiliser la variance, les rendements logarithmiques ont été calculés pour chaque actif à l'aide de la formule suivante:

$$Log\ Returns\ =\ 100\ \times\ log(\frac{Prix\ Actuel}{Prix\ Précédent})$$

Les valeurs NaN résultant des calculs ont été supprimées.

2.1.4. Stationnarité et autocorrélations

Stationnarité

Pour pouvoir utiliser des modèles comme ARIMA qui supposent que les données sont stationnaires, on doit vérifier la stationnarité des données.

- Le test ADF a été utilisé pour évaluer si les rendements log étaient stationnaires.

On a réalisé deux tests ADF, un premier ne prenant pas en compte les dépendances temporelles (lags = 0), et un second incluant 8 lags. Les deux tests concluent à la stationnarité de la série. Mais le second qui tient compte des dépendances temporelles est plus robuste.
La série des rendements logarithmiques est stationnaire et prête pour une analyse ou une modélisation basée sur des séries temporelles, sans nécessiter de transformations supplémentaires.

Autocorrélations

Déterminer le bon lag est essentiel pour identifier les dépendances temporelles significatives et construire un modèle prédictif efficace. On a alors tracé pour les rendements logarithmiques, les fonctions d'autocorrélation (ACF) et d'autocorrélation partielle (PACF) ont été tracées pour les rendements logarithmiques afin d'identifier la structure des dépendances temporelles et les corrélations significatives à différents décalages.

En analysant le PACF, nous constatons que les dépendances directes sont significatives jusqu'au lag 9, ce qui indique que les 9 premiers lags doivent être pris en compte dans la composante autorégressive (p=9). En revanche, l'ACF montre que les corrélations sont nulles dès le lag 1 (inclus), ce qui suggère que la composante moyenne mobile (q) est inutile (q=0).

2.1.5. Création d'indicateurs

Les indicateurs techniques, comme le MACD, RSI, SMA ou les Bandes de Bollinger, sont construits pour détecter les tendances, la volatilité et les conditions de surachat/survente. Ces outils permettent d'identifier des signaux clés pour la prise de décision en trading ou en gestion de portefeuille. Ils facilitent également l'interprétation des corrélations entre les actifs et leurs comportements temporels.

Moving Average Convergence Divergence (MACD)

Le MACD est un indicateur de momentum largement utilisé en analyse technique pour détecter les tendances et les changements de direction dans les séries temporelles.

Relative Strength Index (RSI)

Le RSI mesure la vitesse et l'amplitude des mouvements de prix, utile pour repérer les conditions de surachat ou de survente.

Simple Moving Average (SMA)

La moyenne mobile simple (SMA) est un indicateur de tendance qui lisse les fluctuations en calculant la moyenne des prix sur une période donnée.

Bollinger Bands

Les Bandes de Bollinger mesurent la volatilité en traçant deux bandes autour d'une moyenne mobile centrale.

Nettoyage des données NaN

Tous ces indicateurs (MACD, RSI, SMA, Bollinger Bands) introduisent des valeurs NaN en début de série à cause des fenêtres utilisées dans les calculs (exemple : une SMA 20 nécessite au moins 20 jours de données). On a supprimé ces lignes NaN pour s'assurer que les modèles ultérieurs n'utilisent que des données complètes et exploitables.

2.1.6. Corrélation

Une matrice de corrélation est calculée et visualisée sous forme de heatmap pour évaluer les relations linéaires globales entre les variables du dataset, telles que les prix ajustés, les indicateurs techniques (MACD, RSI, SMA), et d'autres paramètres financiers. Cela permet d'identifier des relations significatives ou des redondances potentielles, essentielles pour affiner la sélection des features et minimiser les biais liés à la multicolinéarité.

Ensuite, une corrélation partielle est calculée pour examiner les dépendances directes entre deux variables spécifiques, comme Adj Close du S&P 500 et le VIX, tout en contrôlant l'effet des autres variables, comme la bande supérieure des Bollinger Bands.

2.2. Développement des modèles

2.2.1. Choix des modèles

Parmi les modèles de machine learning disponibles, nous avons choisi d'étudier les résultats de prédiction du gradient boosting et du LSTM.

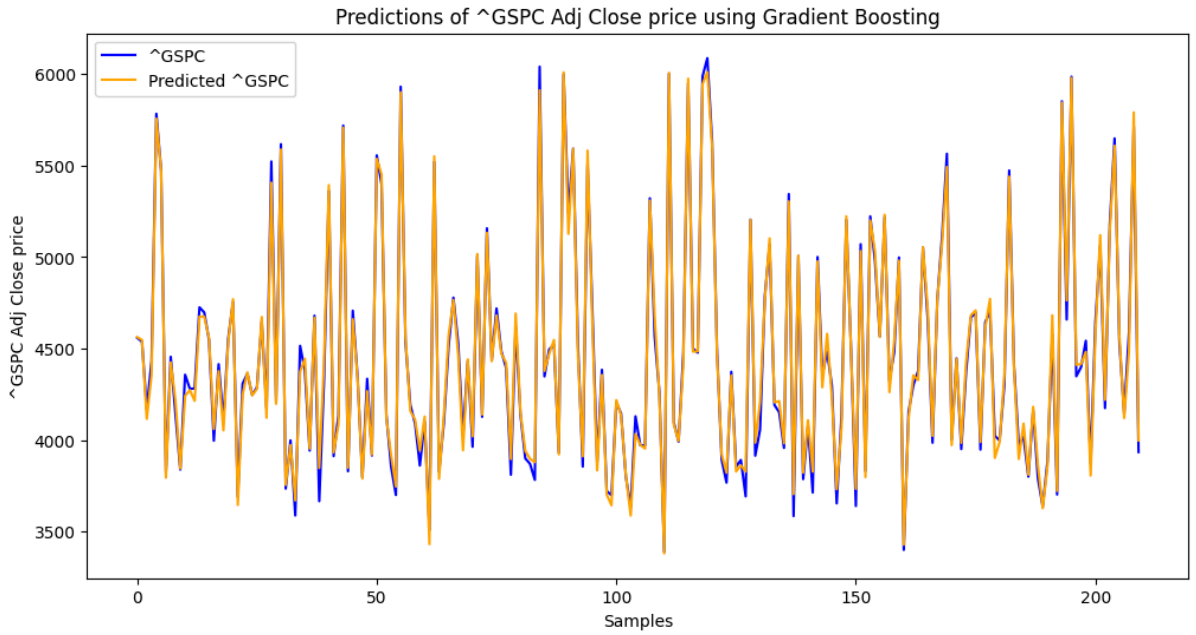
Le Gradient Boosting est une méthode d'ensemble qui combine les prédictions de plusieurs modèles faibles, généralement des arbres de décision, pour former un modèle robuste. L'idée clé est d'ajouter successivement des modèles pour corriger les erreurs des prédictions précédentes. Un avantage majeur du Gradient Boosting est sa capacité à fournir des performances élevées sur des tâches de classification et de régression, tout en étant particulièrement efficace pour gérer des données structurées. Cependant, il peut être sujet à un surajustement si le nombre d'arbres est trop grand ou si les hyperparamètres ne sont pas correctement réglés. De plus, le Gradient Boosting peut être trop coûteux à entraîner.

Les Long Short-Term Memory (LSTM) sont des réseaux de neurones récurrents conçus pour traiter les données séquentielles et capturer les dépendances temporelles à long terme. Les LSTM sont particulièrement utiles pour des tâches comme la prédiction de séries temporelles, le traitement du langage naturel, et la reconnaissance vocale. Leur principal avantage réside dans leur capacité à mémoriser les informations sur de longues périodes tout en atténuant le problème du gradient qui disparaît, ce qui est commun dans

les RNN traditionnels. Cependant, les LSTM peuvent être complexes à entraîner, nécessitent une grande quantité de données et de puissance de calcul, et sont plus difficiles à interpréter que les modèles de machine learning traditionnels comme les arbres de décision.

3. Résultats et analyse

1. Gradient Boosting



- Métriques :

```
Mean Squared Error (MSE), for Train: 1033.1062160681586
Mean Absolute Error (MAE), for Train: 25.16497462976574
Mean Squared Error (MSE), for Test: 2849.954753348085
Mean Absolute Error (MAE), for Test: 41.585232701424474
Mean Absolute Error Ratio (MSER): 2.7586270501735717
Mean Absolute Error Ratio (MAER): 1.6525044556268478
```

Les métriques montrent que le modèle de gradient boosting performe bien sur l'entraînement (MSE = 1033.11, MAE = 25.16) mais souffre de sur apprentissage, avec des erreurs significativement plus élevées sur le test (MSE = 2849.95, MAE = 41.59). Les ratios MSER (2.76) et MAER (1.65) confirment une dégradation des performances en généralisation. Bien que l'erreur moyenne absolue de test (41.59) reste raisonnable selon l'échelle du S&P500, le modèle peut être amélioré en validant par cross validation.

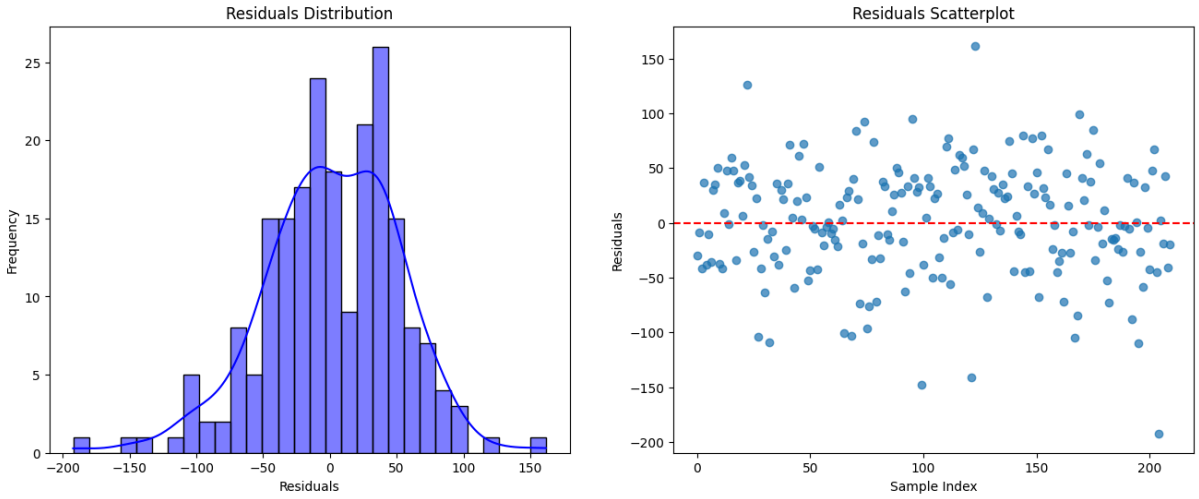
- Analyse des métriques par cross validation:

La validation croisée est une technique utilisée pour évaluer la performance d'un modèle en divisant les données en plusieurs sous-ensembles (**folds)**. Le modèle est entraîné sur k-1 parties et testé sur le sous-ensemble restant. Ce processus est répété k fois, chaque fold servant une fois comme test. Les métriques d'évaluation sont calculées pour chaque itération.

```
Cross-Validation MSE Scores: [11517.96514342  1067.42204698  2694.7111876   6803.41171054
  4699.51948419  2650.00587264  1580.02652812  2684.87790541
  3477.767286   93962.6389385 ]
Average CV MSE: 13113.834610341257
Standard Deviation of CV MSE: 27106.290151129324
```

Ici, une grande variabilité dans ces scores (de très faibles erreurs à des erreurs relativement élevées) indique que la performance du modèle dépend fortement des données spécifiques dans chaque sous-ensemble donc le modèle généralise bien en moyenne malgré des écarts en fonction des jeux de données. Néanmoins, la moyenne et l'écart-type de la CV MSE sont respectivement d'environ 13114 et 27106 ce qui est semble étonnamment démesuré. Ces résultats sont d'autant plus étranges que la prédiction paraît plus que satisfaisante sur le graphique ci-dessus. L'erreur provient sûrement du calcul de la MSE.

- Analyse des résidus:



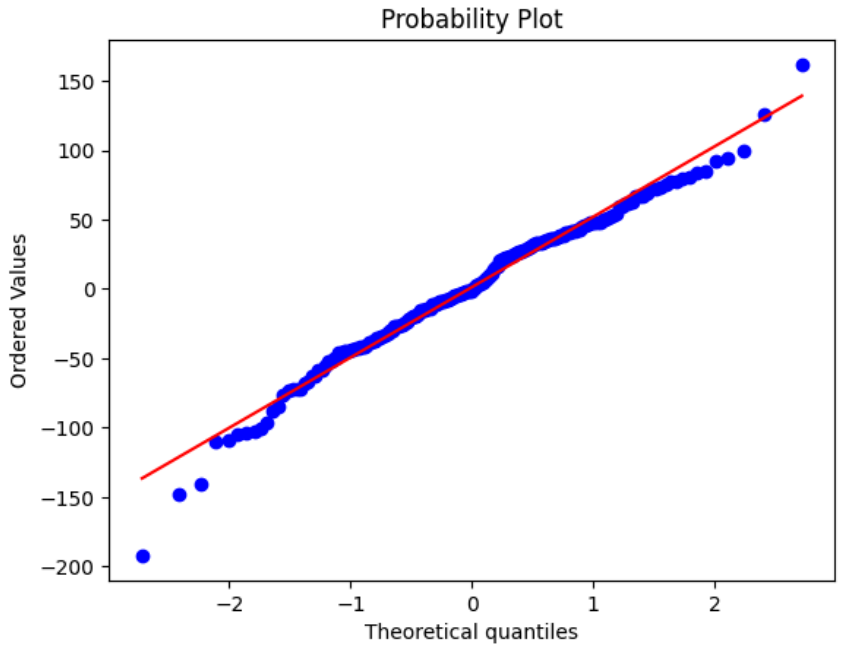
On a représenté la distribution des résidus avec un histogramme des résidus avec une courbe lissée représentant la distribution. Visuellement, les résidus sont majoritairement répartis autour de 0, malgré quelques valeurs extrêmes, ce qui est un bon signe : cela signifie que, en moyenne, les erreurs du modèle sont faibles.

```
Mean of residuals: 1.2659931321164484
Standard deviation of residuals: 50.67830787018526
```

La moyenne des résidus est très proche de 0, ce qui est un bon signe. Cela indique qu'en moyenne, les erreurs du modèle sont pratiquement nulles, ce qui signifie qu'il n'y a

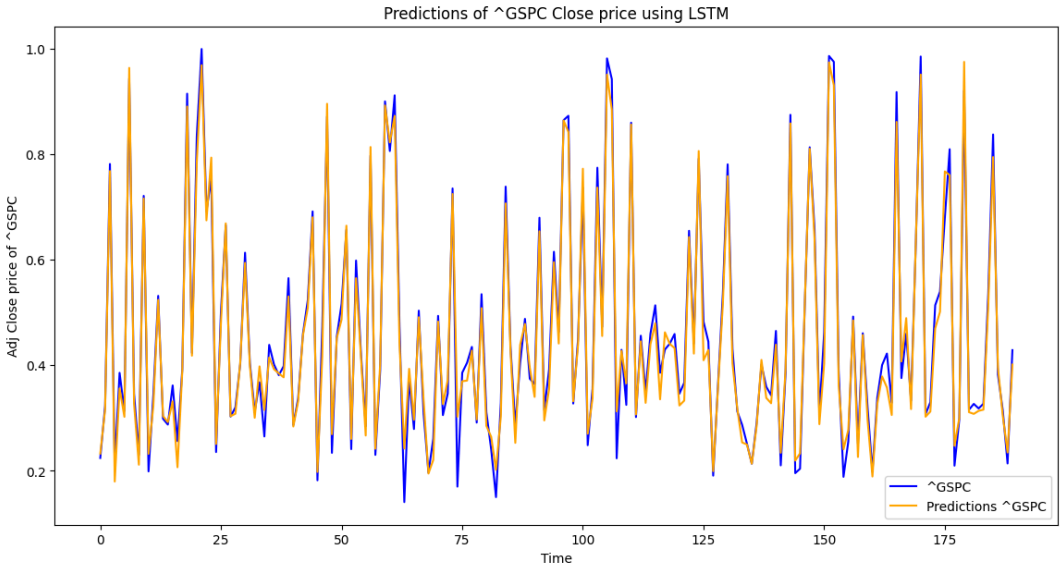
pas de biais systématique dans les prédictions (le modèle ne sous-estime ni ne surestime en moyenne).

L'écart-type indique la dispersion des erreurs autour de la moyenne. Avec une valeur de 50.67, cela montre que les erreurs peuvent être assez importantes dans certains cas, bien que la majorité des résidus soient concentrés autour de zéro (comme observé dans la distribution des résidus).



Les points suivent de près la droite rouge, qui représente une distribution normale théorique.Cela indique que les résidus (erreurs) suivent approximativement une **distribution normale**, ce qui soutient l'hypothèse que les erreurs sont aléatoires et indépendantes.
Dans les parties les plus extrêmes (en bas à gauche et en haut à droite), les points s'écartent légèrement de la ligne rouge.Cela signifie que les résidus présentent des écarts mineurs par rapport à une parfaite normalité dans les queues de la distribution (possible présence d'**outliers** ou de valeurs aberrantes).

2. LSTM



- Métriques :

```
Mean Squared Error (MSE), for Train: 0.0006770268954841508
Mean Absolute Error (MAE), for Train: 0.02102803577901547

Mean Squared Error (MSE), for Test: 0.000873885720257806
Mean Absolute Error (MAE), for Test: 0.02273133018105508

Mean Squared Error (MSE), naive methode: 0.08469190727640981
Mean Absolute Error (MAE), naive methode: 0.226558246033159

MSE Ratio : MSE Prediction over MSE Train (MSER): 1.2907695781168027
MAE Ratio : MAE Prediction over MAE Train (MAER): 1.0810011177429792
```

- Analyse des métriques par cross validation :

```
Cross-Validation MSE Scores: [0.005722862224215539,
0.0016105624350655084, 0.28444002661822754, 0.0021958496714746785,
0.28385324033754195]
Average CV MSE: 0.11556450825730505
Standard Deviation of CV MSE: 0.13765404445065246
```

On remarque une certaine variabilité du MSE (de très faibles erreurs à des erreurs relativement élevées) qui indique que la performance du modèle dépend également de la répartition des données. L'Average CV MSE est 0.11560 ce qui est relativement faible. En moyenne, le modèle fonctionne donc correctement. L'écart-type du MSE est 0.1377 et est supérieur à la moyenne du MSE. Cela reflète une certaine différence en fonction de la répartition des données.

A l'instar du Gradient Boosting, les résidus sont majoritairement répartis autour de 0, à l'exception de quelques valeurs extrêmes. Cela reste néanmoins un résultat satisfaisant.

```
Mean of residuals: 0.006308051183090521
Standard deviation of residuals: 0.028880689232241605
```

La moyenne des résidus est très proche de 0, il n'y a donc à nouveau pas de biais systématique dans les prédictions. En revanche, on remarque un écart-type bien plus faible que pour le Gradient Boosting ce qui signifie que les résidus seront plus concentrés autour de la moyenne.

4. Conclusion/Recommandations

Ce projet a étudié l'efficacité des techniques de machine learning pour prédire les prix et rendements des actifs du S&P 500. Les modèles LSTM peuvent se révéler supérieurs pour capturer les dépendances temporelles et fournir des prédictions précises des rendements logarithmiques, de par leurs capacités à modéliser les séries temporelles complexes grâce à leur capacité à retenir les informations sur de longues périodes et à capturer les relations non linéaires.

Enrichir les modèles avec des indicateurs comme le VIX et les taux d'intérêt, permet d'améliorer la précision des modèles en tenant compte des facteurs macroéconomiques influençant les marchés. Cela enrichit le contexte dans lequel notre modèle fait ses prédictions.

Étant donné la nature dynamique des marchés financiers, il est crucial de maintenir les modèles à jour avec les données les plus récentes. Des mises à jour régulières permettent de capturer les évolutions du marché et d'améliorer continuellement la précision des prédictions.