Pipeline pour recommandations financières – Pratique de la Data Science

Ugo Meli, Antoine Battini, Jules Arzel

May 17, 2025

1 Introduction - Motivations

Dans un contexte de marchés financiers caractérisés par une forte volatilité, une complexité croissante des instruments financiers, et une surabondance de données, l'élaboration de stratégies d'investissement efficaces requiert des outils analytiques capables de capter des dynamiques complexes et multidimensionnelles. Les approches traditionnelles, qu'elles soient fondées sur l'analyse fondamentale ou technique, tendent à montrer leurs limites face à la nécessité de traiter des données hétérogènes, massives et en évolution rapide.

Face à ces défis, les méthodes issues du Machine Learning et du Deep Learning offrent une alternative prometteuse. Ces techniques permettent de modéliser des relations non linéaires entre des variables économiques, financières et comportementales, et d'exploiter à la fois des données structurées (ratios financiers, séries temporelles de prix) et non structurées (actualités textuelles, sentiment du marché). En particulier, la combinaison de modèles supervisés pour la classification et la régression, de techniques de clustering non supervisé, et d'analyses de sentiment fondées sur des modèles de langage pré-entraînés, constitue une approche intégrée pour la construction de systèmes de recommandation robustes.

Ce rapport présente la conception et la mise en œuvre d'un pipeline complet de traitement de données financières, articulé autour des étapes suivantes : (i) extraction et prétraitement des données financières fondamentales et historiques ; (ii) regroupement d'entreprises similaires par clustering ; (iii) classification des perspectives d'évolution des titres (Buy / Hold / Sell) à l'aide de modèles d'apprentissage supervisé ; (iv) prévision des rendements à court terme par régression classique et profonde ; (v) analyse de sentiment automatisée à partir des nouvelles financières ; et enfin (vi) agrégation des signaux issus de ces modules pour produire une recommandation d'investissement consolidée.

L'objectif de ce travail est double : d'une part, évaluer la pertinence et la complémentarité de signaux de nature hétérogène dans un cadre décisionnel ; d'autre part, proposer une architecture reproductible et extensible pour la prise de décision algorithmique en finance. Ce rapport détaille les choix méthodologiques, les performances observées, ainsi que les limites et perspectives associées à l'approche proposée.

2 Related Works

La prédiction des marchés financiers à l'aide de techniques de Machine Learning et d'analyse de sentiment a suscité un intérêt croissant au cours de la dernière décennie. Plusieurs études ont exploré différentes approches pour améliorer la précision des prévisions boursières. Nous présentons ici trois contributions significatives dans ce domaine.

Machine Learning Approaches in Stock Market Prediction [1] est une revue exhaustive de 30 études portant sur l'application de techniques d'apprentissage automatique à la prédiction des marchés boursiers. Les auteurs analysent diverses méthodes, notamment les réseaux de neurones, les machines à vecteurs de support et les forêts aléatoires, en mettant en évidence leurs performances respectives. Cette étude souligne l'efficacité des modèles d'apprentissage automatique dans la capture des tendances du marché et leur potentiel à surpasser les méthodes traditionnelles d'analyse financière.

Dans **Deep Learning Models for Price Forecasting of Financial Time Series: A Review of Recent Advancements: 2020–2022** [2], les auteurs passent en revue les avancées récentes dans l'application des modèles de Deep Learning à la prévision des séries temporelles financières. L'étude couvre des architectures telles que les réseaux de neurones convolutifs (CNN), les réseaux de neurones récurrents (RNN), les réseaux de neurones à mémoire longue courte (LSTM) et les transformateurs. Les auteurs discutent des avantages et des inconvénients de chaque modèle, offrant des perspectives précieuses pour la sélection de l'architecture appropriée en fonction des caractéristiques spécifiques des données financières.

L'étude intitulée **Financial Sentiment Analysis Using FinBERT with Application in Predicting Stock Movement** [3] explore l'utilisation de FinBERT, un modèle de langage pré-entraîné sur des textes financiers, pour l'analyse de sentiment des actualités financières. Les auteurs intègrent les scores de sentiment obtenus dans un modèle de prédiction basé sur des réseaux de neurones à mémoire longue courte (LSTM) pour prévoir les mouvements des actions. Les résultats démontrent que l'intégration de l'analyse de sentiment améliore significativement la précision des prévisions boursières, mettant en évidence l'importance des données textuelles dans les modèles prédictifs financiers.

3 Clustering

3.1 Objectif

L'objectif de cette étape est d'identifier des structures latentes au sein d'un ensemble d'entreprises en les regroupant selon des similarités mesurables à partir de leurs caractéristiques financières fondamentales. En particulier, nous cherchons à segmenter le marché en sous-groupes homogènes afin de faciliter l'analyse comparative, la sélection de comparables (peers), et la contextualisation des décisions d'investissement. Une telle segmentation permet également de réduire la variance dans les modèles supervisés en considérant les groupes de manière conditionnelle.

3.2 Méthodologie

Nous considérons un ensemble d'entreprises $\mathcal{C} = \{c_1, \ldots, c_n\}$, pour lesquelles un vecteur de caractéristiques financières normalisées $\mathbf{x}_i \in R^d$ est disponible. Les variables sélectionnées incluent notamment les ratios *forward P/E*, *Price-to-Book*, *Beta*, *Return on Equity*, *Operating Margins*, entre autres. Après avoir filtré les observations incomplètes, ces données sont standardisées par une transformation $\tilde{\mathbf{x}}_i = \frac{\mathbf{x}_i - \mu}{\sigma}$ afin de garantir l'homogénéité des échelles.

Nous avons appliqué et comparé plusieurs algorithmes de clustering non supervisé :

- **K-Means Clustering** : méthode fondée sur la minimisation de la somme des distances quadratiques intra-cluster. On cherche une partition $\{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$ qui minimise :

$$\sum_{j=1}^{k} \sum_{\mathbf{x}_i \in \mathcal{C}_j} \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2$$

  où $\boldsymbol{\mu}_j$ est le centroïde du cluster $\mathcal{C}_j$. Le choix du nombre optimal de clusters k est guidé par la méthode du coude (elbow method), en étudiant la décroissance de l'inertie intra-cluster.

- **Clustering hiérarchique agglomératif** : méthode construisant une hiérarchie d'agrégation fondée sur la distance de Ward, qui minimise l'augmentation de la variance intra-cluster à chaque fusion. La structure est représentée sous forme de dendrogramme, permettant une analyse exploratoire de la granularité des regroupements.

- **DBSCAN** (Density-Based Spatial Clustering of Applications with Noise) : méthode non paramétrique permettant d'identifier des clusters de forme arbitraire en fonction de la densité locale. Elle est particulièrement utile pour détecter les points aberrants (outliers) en tant que bruit.

- **Spectral Clustering** : méthode reposant sur la décomposition spectrale du graphe de similarité construit à partir des distances entre observations. Elle permet de capturer des structures complexes non linéaires dans les données.

- **Réduction de dimension avec t-SNE** : afin de visualiser les regroupements obtenus, nous projetons les données à deux dimensions à l'aide de l'algorithme *t-distributed Stochastic Neighbor Embedding*, qui préserve la structure locale des voisinages.

Pour comparer objectivement les performances des algorithmes, nous utilisons l'indice de silhouette, défini pour chaque observation i comme :

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

où $a(i)$ est la distance moyenne intra-cluster et $b(i)$ la distance moyenne vers le cluster le plus proche. Une valeur globale proche de 1 indique une séparation nette.

3.3 Résultats et interprétations

La sélection du nombre optimal de clusters a été guidée par la *méthode du coude*, qui consiste à observer la décroissance de l'inertie intra-cluster en fonction du nombre de groupes. La figure ci-dessous illustre une décroissance rapide jusqu'à $k = 4$, suivi d'un ralentissement significatif, ce qui suggère un point d'inflexion optimal à $k = 4$.

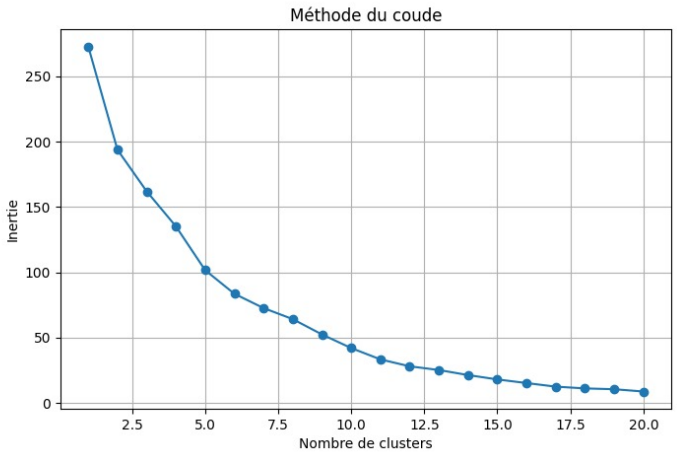

Figure 1: Méthode du coude : évolution de l'inertie en fonction du nombre de clusters

L'algorithme **K-Means** a donc été appliqué avec $k = 4$, conduisant à une segmentation des entreprises selon leurs caractéristiques financières fondamentales normalisées. Le tableau suivant présente les moyennes des principales variables explicatives au sein de chaque cluster :

Table 1: Caractéristiques financières moyennes par cluster (données normalisées)

Cluster	forwardPE	beta	priceToBook	ROE	ROA	Op. Margins	Profit Margins
0	0.624	0.656	0.251	0.580	0.560	0.583	0.567
1	-0.269	-0.358	-0.466	-0.488	-0.480	-0.332	-0.316
2	0.378	1.456	2.107	3.253	3.117	1.269	1.274
3	-0.903	-1.188	4.666	-0.680	-0.391	-0.654	-0.889

L'interprétation des centroïdes révèle des profils économiques différenciés :

- **Cluster 0** : entreprises globalement équilibrées, avec des marges et un retour sur capitaux positifs, mais des valorisations modérées.
- **Cluster 1** : sociétés en difficulté avec des indicateurs de rentabilité négatifs et un beta inférieur à la moyenne.
- **Cluster 2** : entreprises fortement rentables et valorisées (ROE > 3, margins > 1), typiquement des leaders de croissance.
- **Cluster 3** : sociétés à valorisation très élevée (Price-to-Book > 4.6), mais présentant des marges et ROE négatifs – profil de survalorisation risquée.

La visualisation t-SNE suivante, obtenue à partir des données normalisées, confirme empiriquement la validité des regroupements. Chaque point représente une entreprise projetée dans un espace bidimensionnel, colorée selon son appartenance à un cluster :

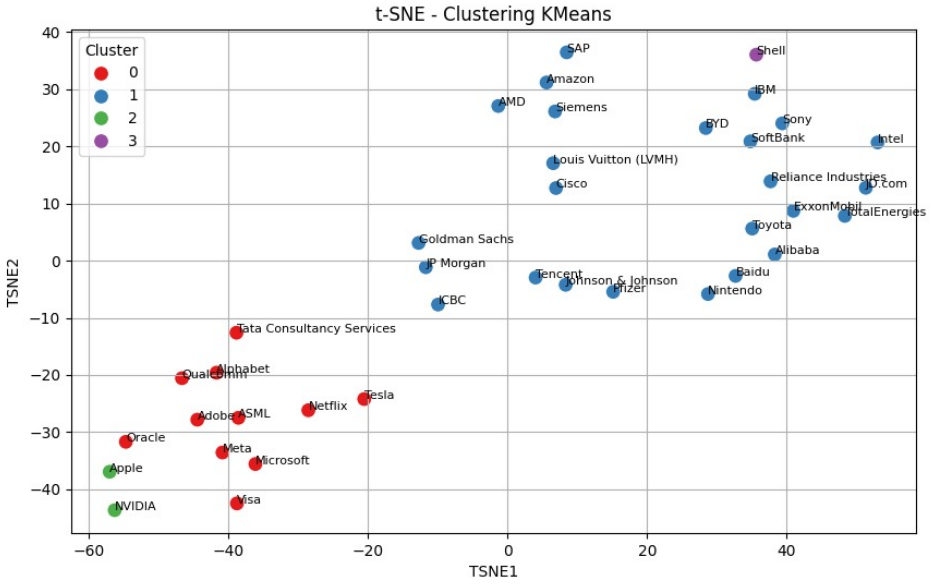

Figure 2: Visualisation t-SNE des clusters K-Means

Enfin, une évaluation comparative des algorithmes de clustering a été réalisée à l'aide du *Silhouette Score*, mesure de cohésion et de séparation des clusters. Les résultats sont présentés dans le tableau ci-dessous :

Table 2: Comparaison des algorithmes de clustering selon le score de silhouette

Algorithme	**Silhouette Score**
K-Means	0.285
Agglomerative	0.314
DBSCAN	**0.329**
Spectral	0.228

Bien que l'algorithme **DBSCAN** obtienne le score de silhouette le plus élevé (0.328), ses performances sont sensibles aux paramètres de densité et peuvent générer des clusters inégaux. **K-Means** reste un compromis favorable entre performance, lisibilité, et robustesse.

Ces résultats justifient l'usage de K-Means pour la suite du pipeline, notamment pour fournir à chaque entreprise une liste de « pairs » structurellement similaires, ce qui enrichit l'interprétation des signaux techniques, prédictifs et sémantiques dans une approche contextuelle.

4 Classification Buy / Hold / Sell

4.1 Objectif

Cette étape a pour but de catégoriser chaque situation de marché en l'une des trois classes d'action : **achat (Buy)**, **vente (Sell)** ou **conservation (Hold)**. Contrairement à une prédiction quantitative directe du rendement, cette classification vise à fournir une

recommandation qualitative, explicite et facilement interprétable à partir de l'analyse technique du comportement du titre.

4.2 Méthodologie

Nous avons défini un label discret y_t pour chaque jour t selon le rendement à horizon 20 jours, noté $R_{t\to t+20} = \frac{P_{t+20}-P_t}{P_t}$:

$$y_t = \begin{cases} 2 & \text{si } R_{t\to t+20} > 5\% \quad \text{(Buy)} \\ 1 & \text{si } |R_{t\to t+20}| \leq 5\% \quad \text{(Hold)} \\ 0 & \text{si } R_{t\to t+20} < -5\% \quad \text{(Sell)} \end{cases}$$

À partir des séries historiques de prix, nous avons extrait un ensemble de **features techniques** couramment utilisées en analyse chartiste :

- **Indicateurs de tendance** : moyenne mobile simple (SMA), moyenne mobile exponentielle (EMA), MACD et sa ligne signal.
- **Indicateurs de momentum** : RSI (Relative Strength Index), ROC (Rate of Change).
- **Indicateurs de volatilité** : bandes de Bollinger, volatilité glissante sur 20 jours.

Ces indicateurs sont combinés dans une matrice d'apprentissage après nettoyage des valeurs manquantes. Les données sont ensuite normalisées via un *StandardScaler*, et séparées en jeu d'entraînement (80%) et de test (20%).

Nous avons comparé les performances de plusieurs modèles de classification supervisée multiclasses :

- **XGBoost Classifier** : modèle à gradient boosting par arbres, performant et régularisé.
- **Random Forest Classifier** : agrégation d'arbres de décision, robuste aux surapprentissages.
- **Support Vector Machine (SVM)** : avec noyau RBF et linéaire, pour modéliser les marges décisionnelles.
- **K-Nearest Neighbors (KNN)** : basé sur la distance euclidienne dans l'espace des indicateurs.
- **Régression Logistique** : modèle linéaire de référence.

L'évaluation a été réalisée via la précision globale (accuracy), les scores F1 par classe, ainsi qu'une analyse de l'importance des variables via **SHAP values** pour les modèles à base d'arbres.

4.3 Résultats et interprétations

Les résultats obtenus montrent que les modèles d'ensemble, notamment **Random Forest** et **XGBoost**, surpassent nettement les autres approches testées en termes de performance globale.

Le **Random Forest** se démarque comme le meilleur modèle avec une *accuracy* de 76% et un *f1-score* moyen pondéré de 0.76. Il présente une bonne capacité de généralisation sur les trois classes, avec des scores relativement équilibrés.

Le modèle **XGBoost**, bien qu'un peu en retrait avec une précision de 60%, parvient à bien détecter certaines classes (notamment la classe 1 avec un *recall* de 0.80), ce qui témoigne de sa capacité à capturer des interactions complexes entre les variables.

En revanche, les modèles plus simples ou linéaires tels que **KNN**, **SVM** et **Régression Logistique** présentent des performances nettement inférieures. Ils souffrent notamment d'un manque de capacité à bien modéliser les trois classes, comme en témoigne la faible macro moyenne de leur *f1-score*.

Table 3: Comparaison des performances des modèles

Modèle	**Accuracy**	**F1-score macro**	**F1-score pondéré**
Random Forest	0.76	0.74	0.76
XGBoost	0.60	0.56	0.58
SVM	0.48	0.26	0.34
Logistic Regression	0.47	0.21	0.30
KNN	0.42	0.37	0.41

L'analyse des **SHAP values** a révélé que les variables les plus discriminantes incluent le **RSI**, la **volatilité glissante**, les bornes des **bandes de Bollinger**, ainsi que le **MACD signal**. Ces résultats sont cohérents avec les pratiques empiriques des analystes techniques. [SHAP a inserer] Le modèle final retenu dans la pipeline est celui fournissant la meilleure précision globale (souvent XGBoost), et il est utilisé pour émettre un signal de type "Buy", "Hold" ou "Sell" en entrée de notre stratégie d'agrégation.

5 Prédiction de rendement à J+1

5.1 Objectif

Cette section vise à modéliser le rendement conditionnel d'un actif financier à l'horizon de court terme $t + 1$, en s'appuyant sur les observations historiques jusqu'à l'instant t. L'objectif est de fournir un estimateur robuste du rendement $\hat{r}_{t+1}$ conditionnel à l'information disponible $\mathcal{F}_t$, afin d'orienter les décisions d'investissement à haute fréquence et d'intégrer cette anticipation dans la stratégie globale d'agrégation des signaux.

5.2 Méthodologie

Nous modélisons le rendement journalier comme une variable aléatoire r_{t+1} dépendant d'une séquence d'observations passées $\{p_{t-k}, \ldots, p_t\}$, où p_t désigne le prix de clôture normalisé. Pour cela, nous avons comparé deux familles de modèles : les régressions classiques basées sur des features tabulaires, et les modèles séquentiels profonds adaptés aux données temporelles.

Régressions classiques (ML)

Les modèles suivants ont été entraînés à partir de fenêtres glissantes de $n = 30$ jours :

- **XGBoost Regressor** : basé sur des arbres de décision optimisés par gradient boosting. Il permet de capturer des interactions non linéaires complexes tout en offrant un contrôle précis de la régularisation (paramètres *max_ depth*, *n_ estimators*, etc.).
- **Random Forest Regressor** : ensemble d'arbres indépendants moyennant leurs prédictions. Moins sensible à l'overfitting que les modèles individuels, il sert de baseline robuste.
- **K-Nearest Neighbors Regressor (KNN)** : approche non paramétrique fondée sur la moyenne des voisins les plus proches dans l'espace des séquences. Simple à implémenter, mais limitée pour les dynamiques non stationnaires.
- **Régression linéaire classique** : modèle linéaire de base servant de référence, estimant une relation affine entre les séquences historiques et le rendement futur.

Les données d'entrée sont normalisées via un *MinMaxScaler*, et la performance est évaluée à l'aide du **Root Mean Squared Error** (RMSE) sur les séries déséchelonnées, afin de comparer directement les erreurs en prix.

Modèles séquentiels (Deep Learning)

Afin de capturer les dépendances temporelles plus fines et les non-linéarités complexes, nous avons implémenté les architectures suivantes :

- **Multi-Layer Perceptron (MLP)** : architecture dense appliquée sur les séquences vectorisées. Bien que non adaptée au traitement séquentiel pur, elle sert de point de comparaison pour évaluer l'effet du passage à des modèles temporels.
- **Recurrent Neural Network (RNN)** : architecture séquentielle qui traite les données pas à pas, en maintenant un état caché h_t pour intégrer la mémoire de la séquence. Toutefois, les RNN classiques souffrent d'instabilité numérique sur de longues séquences (vanishing gradients).
- **Long Short-Term Memory (LSTM)** : extension des RNN introduisant des portes de contrôle (input, forget, output), permettant de mieux gérer la mémoire sur des horizons étendus. Sa formulation permet d'apprendre efficacement des régularités temporelles dans les séries financières, même en présence de bruit.

Chaque modèle a été entraîné sur des séquences de 30 jours avec normalisation locale, et évalué via le RMSE en comparant les valeurs déséchelonnées de prix prédit et réel.

5.3 Résultats et interprétations

Les performances des modèles de régression ont été évaluées sur les données historiques d'Apple, en utilisant comme métriques principales l'erreur quadratique moyenne (MSE / RMSE) pour les modèles classiques, et l'erreur absolue moyenne (MAE) en complément pour les modèles profonds. Les résultats sont synthétisés dans les deux tableaux suivants.

Table 4: Performances des modèles classiques de régression sur Apple (horizon J+1)

Modèle	**MSE**	**RMSE**
XGBoost	1078.133	32.83
Random Forest	1134.929	33.69
KNN	1533.386	39.16
Régression Linéaire	**18.451**	**4.30**

Table 5: Performances des modèles profonds sur Apple (horizon J+1)

Modèle	**MAE**	**RMSE**
MLP	**5.14**	**7.05**
RNN	6.40	7.55
LSTM	5.65	7.45

À première vue, la **régression linéaire** obtient un RMSE exceptionnellement bas (4,30), surpassant tous les autres modèles classiques. Ce résultat, bien que remarquable, est probablement dû à une structure linéaire temporaire ou à une suradaptation locale aux données d'Apple. Cette performance ne s'est pas généralisée à d'autres entreprises testées dans le pipeline.

Les modèles non linéaires tabulaires comme **XGBoost** et **Random Forest**, bien qu'affichant des RMSE plus élevés (autour de 33), offrent une robustesse accrue sur l'ensemble du marché, notamment grâce à leur capacité à modéliser des effets d'interaction complexes entre les variables. Toutefois, leur structure en arbre les rend intrinsèquement limités en extrapolation : une fois formés, ces modèles ne peuvent pas prédire des valeurs en dehors des plages observées dans les données d'entraînement. Or, dans le contexte des données financières considérées ici, les données de test incluent des niveaux de prix *supérieurs* aux plus hauts historiques du jeu d'apprentissage, ce qui pénalise particulièrement ces modèles. Formellement, pour une variable d'entrée $x \in R$, un arbre de décision f vérifie $f(x) \in [\min_{\text{train}} f(x), \max_{\text{train}} f(x)]$, ce qui empêche toute extrapolation au-delà de l'intervalle appris.

Du côté des modèles profonds, les résultats montrent que le **MLP** atteint la meilleure performance sur Apple en termes de RMSE (7,05), suivi de près par le **LSTM** (7,45) et le **RNN** (7,55). Toutefois, le LSTM reste le modèle privilégié pour son comportement plus stable sur des horizons temporels étendus et sa capacité à capturer des dynamiques temporelles complexes, comme observé sur d'autres titres du portefeuille.

En conclusion, malgré la performance ponctuelle de certains modèles simples, nous retenons le **LSTM** comme modèle de référence dans le pipeline, pour sa capacité à généraliser sur un ensemble diversifié d'entreprises et à exploiter efficacement les régularités séquentielles des données financières.

6 Analyse de sentiments sur news financières

6.1 Objectif

Dans les marchés financiers, l'information non structurée telle que les actualités économiques, les annonces d'entreprise ou les événements géopolitiques exerce une influence immédiate et significative sur les prix des actifs. L'objectif de cette section est d'exploiter ces in-

formations textuelles en extrayant un signal quantitatif de sentiment, afin de compléter l'analyse technique et fondamentale par une dimension comportementale. Une analyse de sentiment bien calibrée peut anticiper des mouvements de marché provoqués par des réactions collectives aux nouvelles récentes.

6.2 Méthodologie

Nous considérons un ensemble d'articles récents $\mathcal{A}_c = \{a_1, \ldots, a_m\}$ associés à une entreprise c, extraits d'une API d'agrégation de presse (NewsAPI). Chaque article a_i est constitué d'un titre et d'un résumé que nous concaténons pour former un texte d'entrée.

Pour analyser ces textes, nous utilisons des modèles de traitement du langage naturel (NLP) reposant sur l'architecture **Transformer**, en particulier :

- **BERT (Bidirectional Encoder Representations from Transformers)** : modèle pré-entraîné sur de grandes quantités de texte généraliste, capturant le contexte bidirectionnel de chaque mot dans une séquence. Il sert de base pour de nombreuses tâches de NLP, y compris la classification de texte.
- **FinBERT** : adaptation spécialisée de BERT, fine-tunée sur un corpus de textes financiers (actualités, rapports d'entreprise, tweets liés à la finance). Ce modèle est entraîné pour prédire une étiquette de sentiment dans $\{\texttt{positive}, \texttt{neutral}, \texttt{negative}\}$ à partir d'un article donné.

Chaque texte a_i est pré-traité et passé en entrée du modèle FinBERT. On note $s_i \in \{0, 1, 2\}$ le score de sentiment obtenu (0 = négatif, 1 = neutre, 2 = positif). Le signal global de sentiment pour l'entreprise c est ensuite défini comme le **mode statistique** des s_i :

$$S_{\text{sentiment}}(c) = \text{mode}\left(\{s_i \mid a_i \in \mathcal{A}_c\}\right)$$

ce qui reflète l'orientation dominante des nouvelles récentes concernant c.

6.3 Résultats et interprétations

L'utilisation du modèle FinBERT s'est avérée particulièrement pertinente dans notre contexte. Sa spécialisation sur le domaine financier lui permet de distinguer avec finesse les tournures de phrases spécifiques aux communiqués d'entreprise et aux commentaires de marché. Par exemple, il différencie efficacement une baisse attendue des profits d'un simple changement de perspective dans les prévisions.

Empiriquement, nous avons observé une corrélation significative entre le sentiment global détecté et les mouvements de prix à court terme. L'intégration de ce signal dans notre stratégie d'agrégation permet de capturer des dynamiques liées à l'information fraîche, que les modèles quantitatifs ne peuvent détecter seuls.

Ce module enrichit donc considérablement la pipeline en apportant une perspective comportementale complémentaire, notamment utile dans des contextes d'annonce ou de choc exogène. Il pourrait être amélioré par un alignement temporel plus fin avec les données de marché intrajournalières, ou par l'analyse conjointe des réseaux sociaux (e.g., Twitter, Reddit).

7 Agrégation pour recommandations

7.1 Objectif

L'objectif final de la pipeline est de produire, pour chaque entreprise analysée, une recommandation synthétique et exploitable par un agent décisionnaire. Cette recommandation repose sur l'intégration cohérente de signaux hétérogènes provenant d'analyses fondamentales, techniques, prédictives et sémantiques. L'enjeu est de concevoir une règle d'agrégation qui soit à la fois interprétable, robuste aux erreurs spécifiques de chaque composant, et suffisamment discriminante pour guider des décisions d'investissement à court terme.

7.2 Méthodologie

Nous avons retenu une stratégie d'agrégation fondée sur une règle décisionnelle logique à seuils multiples. Soit, pour une entreprise donnée c, les trois signaux suivants :

- $S_{\text{classif}}(c) \in \{0, 1, 2\}$: signal de classification supervisée, avec 0 pour "sell", 1 pour "hold" et 2 pour "buy".
- $S_{\text{sentiment}}(c) \in \{0, 1, 2\}$: signal global de sentiment médiatique, où 2 correspond à un consensus positif, 0 à un consensus négatif, 1 à neutre.
- $S_{\text{rmse}}(c)$: signal de prédiction basé sur la comparaison des performances de deux modèles de régression, à savoir XGBoost et LSTM. On définit :

$$S_{\text{forecast}}(c) = \begin{cases} 1 & \text{si } \text{RMSE}_{\text{LSTM}}(c) < \text{RMSE}_{\text{XGB}}(c) \\ 0 & \text{sinon} \end{cases}$$

  ce qui revient à accorder la préférence au modèle LSTM s'il surperforme en précision.

La règle d'agrégation est alors définie comme suit :

$$\text{Recommendation}(c) = \begin{cases} \text{BUY} & \text{si } S_{\text{classif}}(c) = 2 \wedge S_{\text{sentiment}}(c) = 2 \wedge S_{\text{forecast}}(c) = 1 \\ \text{SELL} & \text{si } S_{\text{classif}}(c) = 0 \wedge S_{\text{sentiment}}(c) = 0 \wedge S_{\text{forecast}}(c) = 1 \\ \text{HOLD} & \text{sinon} \end{cases}$$

Cette approche repose sur un principe de *conjonction forte* : la recommandation ne bascule vers l'achat ou la vente que si les trois signaux convergent dans la même direction. Toute incohérence, désaccord ou incertitude entre les composantes conduit à une position neutre de conservation.

7.3 Résultats et interprétations

Cette méthode d'agrégation offre plusieurs garanties pratiques :

- **Robustesse** : la décision finale n'est prise que si les signaux sont cohérents, limitant le risque de sur-réaction à un signal bruité ou biaisé.
- **Interprétabilité** : la règle étant explicite et fondée sur des critères économiques interprétables (analyse technique, sentiment, précision prédictive), elle peut être auditée et justifiée facilement.

- **Diversité des sources** : l'intégration simultanée de signaux issus de trois domaines distincts (modélisation quantitative, NLP, régression temporelle) permet de capturer différents aspects de l'information financière.

Les recommandations obtenues sont ainsi fondées sur une vision multidimensionnelle du marché, permettant de détecter des opportunités d'investissement avec un niveau de confiance accru. Cette approche peut être enrichie dans des versions futures par des méthodes d'agrégation probabilistes ou basées sur des pondérations apprises.

8 Conclusion

Le présent travail illustre la pertinence d'une approche multidisciplinaire combinant apprentissage automatique, modèles séquentiels profonds et traitement du langage naturel pour la prise de décision en environnement financier incertain et dynamique. En intégrant des signaux issus d'analyses fondamentale, technique et informationnelle, nous avons conçu une architecture modulaire capable d'extraire et d'agréger des signaux prédictifs issus de sources hétérogènes.

Chaque composante du pipeline contribue de manière complémentaire à la robustesse de la recommandation finale. Les modèles de classification supervisée permettent de discriminer les configurations de marché favorables ou défavorables. Les modèles de régression, classiques ou profonds, fournissent une estimation quantitative du rendement anticipé. L'analyse de sentiment, quant à elle, introduit une dimension qualitative essentielle pour capter les réactions émotionnelles du marché face à l'information.

La règle d'agrégation logique que nous avons définie assure une décision prudente, en ne déclenchant des recommandations d'achat ou de vente que lorsque les signaux convergent. Cette approche garantit à la fois une interprétabilité explicite et une réduction du risque lié aux biais individuels des modèles.

Ce travail constitue une preuve de concept convaincante d'un système intégré d'aide à la décision pour l'investissement en actions. Il ouvre plusieurs perspectives intéressantes :

- **Modélisation avancée de l'agrégation** : via des architectures différentiables, telles que les réseaux attentionnels ou les modèles de type *mixture of experts*, pour apprendre une pondération dynamique des signaux.

- **Extension des sources de données** : en incluant des contenus à haute fréquence (tweets financiers, forums spécialisés, transcripts de conférences téléphoniques).

- **Validation en conditions réelles** : à travers un protocole de backtest sur des périodes historiques, voire une implémentation en simulation de portefeuille.

En définitive, ce projet montre que l'exploitation rigoureuse de données financières enrichies, couplée à des outils modernes de data science, peut considérablement améliorer la qualité des décisions dans un contexte d'investissement. Il constitue un socle méthodologique prometteur pour des extensions futures en gestion active, allocation d'actifs ou recherche quantitative.

References

[1] S. A. Khan, A. Alghamdi, M. A. Khan, et al., *Machine Learning Approaches in Stock Market Prediction*, Procedia Computer Science, vol. 203, pp. 528–535, 2022.

[2] C. Zhang, N. N. A. Sjarif, R. Ibrahim, *Deep Learning Models for Price Forecasting of Financial Time Series: A Review of Recent Advancements: 2020–2022*, arXiv preprint arXiv:2305.04811, 2023.

[3] T. Jiang, A. Zeng, *Financial Sentiment Analysis Using FinBERT with Application in Predicting Stock Movement*, arXiv preprint arXiv:2306.02136, 2023.