DataScientest

Projet OPA

Allocation dynamique de portefeuille d'investissement

Février - Septembre 2025

Rapport de projet

Samuel LEE KWET SUN - Gilles LENY - Mathis GENTHON - Jerry PETILAIRE

Contexte du projet

La quête d'une **surperformance durable** par rapport au marché est le **défi central de la finance quantitative**. Ce projet de fin d'études s'attaque à cette problématique en explorant l'application du **Machine Learning** à **l'analyse fondamentale**, une discipline qui vise à évaluer la santé financière intrinsèque des entreprises pour prédire leur trajectoire future.

Notre démarche a débuté avec une question : est-il possible de construire un modèle prédictif qui, en se basant uniquement sur des **données financières publiques** (bilans, comptes de résultat), pourrait identifier des actions destinées à surperformer le marché ? Cette hypothèse nous a conduits à travers un parcours méthodologique rigoureux, allant de **l'acquisition de données** via l'API de Financial Modeling Prep (FMP) à l'ingénierie de caractéristiques, en passant par le défi majeur de la définition d'une **variable cible temporellement cohérente**.

Cependant, nos recherches et itérations ont révélé une **vérité nuancée**. Si la prédiction de la "surperformance" (la classe 1) avec une certitude absolue reste un défi, notre modèle **RandomForestClassifier** a ceci-dit démontré une capacité fiable à identifier le scénario inverse : la **sous-performance** (la classe 0). Avec une excellente précision sur les **prédictions à haute confiance**, nous avons réalisé que la véritable valeur ajoutée de notre approche ne résidait pas dans la **sélection agressive de "gagnants"**, mais dans un mécanisme de gestion des risques : le **filtrage des "perdants"**.

Cette découverte a provoqué une réorientation stratégique du projet. L'objectif n'était plus de générer de l'alpha et de battre le marché, mais de fournir un outil de **gestion des risques**, capable de préserver le capital en émettant des **alertes fiables** sur les actions les plus fragiles.

Nous avons donc développé **Stella**, un agent conversationnel interactif. Construit sur un écosystème technologique moderne (Python, Streamlit, LangGraph, LLMs), qui encapsule toute la complexité de notre pipeline – de la collecte de données à l'inférence du modèle – derrière une **interface de chat simple et intuitive**.

En somme, ce projet va **au-delà de la simple modélisation prédictive**. Il aboutit à la création d'un produit "augmenté" : non pas une promesse de rendements, mais un assistant d'aide à la décision, dont la contribution espérée est **d'accompagner la gestion de portefeuille** par le prisme de la maîtrise du risque.

Objectifs du projet

Pour répondre à la problématique de l'allocation d'actifs et capitaliser sur notre découverte stratégique, le projet a été structuré autour de **quatre objectifs principaux**, qui ont été atteints de manière itérative :

1. Ingénierie d'une Variable Cible Robuste et Cohérente
L'objectif initial était de créer une fondation solide pour notre modèle de classification. La **simple comparaison** des rendements annuels **des actions à celui de l'indice s'est avérée inadéquate** en raison des décalages entre les années fiscales des entreprises et l'année calendaire.

2. Construction et Validation d'un Modèle de Classification Axé sur le Risque
L'objectif était de développer un modèle de Machine Learning capable de prédire notre variable cible à partir des **indicateurs fondamentaux.** Après avoir écarté la régression, nous nous sommes concentrés sur la classification.

3. Conception et Déploiement d'un Agent Conversationnel ("Stella")
Un modèle, aussi performant soit-il, a une utilité limitée s'il n'est pas accessible. L'objectif était de créer une **interface agréable** qui permettrait à un **utilisateur non-technique** d'exploiter la puissance de notre modèle.

4. Enrichissement de l'Agent avec des Fonctionnalités d'Analyse Financière Étendues
Pour que Stella soit un véritable assistant, il ne devait pas se limiter à notre seule prédiction de risque. L'objectif était de l'équiper d'une **panoplie d'outils** pour permettre une analyse financière augmentée.

Ces quatre objectifs, une fois atteints, ont transformé un projet de modélisation en **une solution applicative complète**, robuste et centrée sur l'utilisateur.

Niveau d'expertise

Mathis GENTHON : Pas d'expertise professionnelle en finance. Cependant, j'investis personnellement dans différents types d'actifs et opère une veille quotidienne sur les marchés et l'analyse financière.

Jerry PETILAIRE : Pas d'expertise professionnelle en finance et inexpérimenté en investissement. Je m'intéresse aux grandes notions d'économie et finance à travers l'écoute de podcasts et ce projet m'a séduit par sa valeur concrète.

Samuel LEE KWET SUN : Pas d'expertise en finance.

Gilles LENY: Expert en ingénierie financière, j'élabore et optimise des montages complexes alliant financements publics et privés, avec une maîtrise de l'analyse, de la modélisation et de la gestion des risques.

Cadre du projet

Deux approches **ont été utilisées dans ce projet :**

- **Analyse technique** : Cette approche postule que **toute l'information pertinente est déjà reflétée dans le prix de l'actif** et s'appuie sur l'étude des graphiques, du momentum, de la volatilité, etc.
- **Analyse fondamentale** : Elle vise à évaluer la **valeur intrinsèque d'une entreprise** en analysant ses **données financières** (bilans, revenus, ratios) et son **environnement économique** pour estimer sa performance future.

L'analyse fondamentale a été retenue de **part de ses résultats**, et constitue le sujet principal de ce rapport

Nous avons utilisé des **données financières fondamentales** d'entreprises et des **données de cours** (OHLC - Open, High, Low, Close) pour un ETF NASDAQ 100 (QQQ), un ETF représentant la valeur de l'or (GLD) et un ETF représentant la valeur du pétrole (USO) ainsi que **chaque entreprise du NASDAQ 100** individuellement. Ces données ont été obtenues via **l'API de Financial Modeling Prep** (FMP). **L'approche fondamentale du projet** consiste donc à **classifier les entreprises** selon leur rendement par rapport au marché à long terme, en se basant sur les **indicateurs fondamentaux** présents dans les données, et complétées avec des **valeurs OHLC** pour création de la variable cible.

FMP propose un accès gratuit limité, suffisant pour les usages du projet. Cependant, l'accès à des historiques plus longs et à une gamme complète nécessite un abonnement. Ceci constitue une des limites du projet, puisque nous avons accès à des données sur une plage de seulement 5 ans. Le propriétaire de ces données est Financial Modeling Prep (FMP).

Volumétrie des données

Données tabulaires de 2020 à 2024	Observations	Variables
Indicateurs fondamentaux NASDAQ 100	504	61
OHLC NASDAQ 100	123k	14
OHLC ETF QQQ / GLD / USO	1258	14

Pertinence, pre-processing et feature engineering

Les données sont passées par plusieurs phases de traitement, à différentes étapes du projet :

- **Sélection des variables pertinentes pour l'analyse fondamentale** (61 colonnes dans le dataset originel)
- **Import de données venant d'autres datasets** (OHLC des entreprises, OHLC de l'ETF QQQ)
- **Création de features** (`marginProfit`, `return`, `benchmark`, `target`, colonnes `YoY_Growth`)
- **Sélection des colonnes pertinentes pour la modélisation** (retrait de certaines colonnes en se basant sur les valeurs de Shapley)

1. Sélection des variables

En **analyse fondamentale,** l'objectif est d'évaluer la **santé financière** et la valeur intrinsèque d'une entreprise en se basant sur ses données issues des **états financiers** (compte de résultat, bilan, etc) et d'autres facteurs.

Le jeu de données a été récupéré via la section de l'**API de Financial Modeling Prep** qui était orientée **métriques fondamentales** :

Échantillon d'une réponse de l'API sur l'endpoint key-metrics (Apple Inc.)


```
[
      {
            "symbol": "AAPL",
            "date": "2022-09-24",
            "calendarYear": "2022",
            "period": "FY",
            "revenuePerShare": 24.31727304755197,
            "netIncomePerShare": 6.154614437637777,
            "operatingCashFlowPerShare": 7.532762624088375,
            "freeCashFlowPerShare": 6.872425646259799
      }
]
```

Beaucoup de variables qui nous étaient nécessaires **étaient déjà calculées.** Ceci nous a permis de gagner un temps considérable en début de projet et de débuter rapidement les **premières phases exploratoires.**

Le tableau ci-dessous illustre ces variables.

Variables sélectionnées présentes dans le jeu de données originel

Indicateur	Maximiser / Minimiser ?	Pourquoi ?
ROE (Return on Equity)	**Maximiser**(>15% idéalement)	Plus le **ROE est élevé**, plus l'entreprise est rentable pour ses actionnaires.
ROIC (Return on Invested Capital)	**Maximiser**	Un **ROIC élevé** indique une entreprise efficace pour transformer le capital investi en profit.
Earnings Yield (E/P)	**Maximiser**	L'Earnings Yield représente **le rendement généré par l'entreprise pour chaque euro investi**. Il sert à identifier si une entreprise est potentiellement sous / surévaluée.
Debt to Equity (D/E)	**Minimiser**	Un ratio faible montre une faible dépendance à l'endettement. Trop élevé, il peut indiquer un risque financier accru.
Revenue per Share	**Maximiser**	Un chiffre d'affaires par action élevé peut indiquer une forte capacité de génération de revenus, ce qui est favorable pour la croissance future.
Market Capitalization	**–**	Une capitalisation élevée peut refléter une entreprise stable et bien établie, mais les petites capitalisations peuvent offrir un potentiel de croissance plus élevé.

La description complète des variables du jeu de données fondamentales est disponible en **annexe 1 "nov24_cds_opa - Rapport exploration des données (fondamentales).xlsx".**

2. Import de données venant d'autre datasets

Afin de créer une variable cible (**target**) pour un modèle de classification, nous avons dû d'abord estimer si l'entreprise X à l'année N avait surperformé ou non le marché. Il nous a donc été nécessaire d'utiliser des données **OHLC (Open High Low Close)**, ou données de cours afin d'identifier les entreprises qui avaient **surperformé le marché** en calculant leur rendement sur chaque année.

Pour ce faire nous avons importé, via l'API de Financial Modeling Prep, puis préparé deux datasets :

- **Données OHLC des entreprises du NASDAQ 100**

- **Données OHLC de l'ETF QQQ** (ETF suivant l'indice NASDAQ 100 faisant office de marché)

Échantillon d'une réponse de l'API sur l'endpoint key-metrics (Apple Inc.)


```
{
  "symbol" : "AAPL",
  "historical" : [ {
      "date" : "2021-10-08",
      "open" : 144.03,
      "high" : 144.17,
      "low" : 142.56,
      "close" : 142.9,
      "adjClose" : 142.9,
      "volume" : 5.545036E7,
      "unadjustedVolume" : 5.545036E7,
      "change" : -1.13,
      "changePercent" : -0.785,
      "vwap" : 143.21,
      "label" : "October 08, 21",
      "changeOverTime" : -0.00785
    } ]
}
```

3. Données OHLC des entreprises du NASDAQ 100

Pour pouvoir définir si une entreprise surperforme le marché, il est d'abord nécessaire de **connaître la valeur de l'action** à la **date de publication du rapport fiscal**. Nous avons donc fusionné les données fondamentales avec les **prix de clôture** (close) de chaque action.

Défi `merge_asof` : La **fusion initiale directe** par `symbol` et `date` a révélé un nombre significatif de valeurs manquantes (254 `NaN`) pour la colonne `close`. Ceci est dû au fait que les **rapports financiers** (données fondamentales) sont publiés à des dates **différentes** de l'année fiscale **pour chaque entreprise**, tandis que les prix de clôture sont **quotidiens et sur les jours ouvrés**. Les deux ne concordent donc pas forcément. Pour pallier ce problème, nous avons trié les deux DataFrames par `date` et `symbol`, puis utilisé une jointure `pd.merge_asof`. Cette méthode permet de fusionner en trouvant la date la plus proche ou égale dans le passé, garantissant ainsi l'association correcte des prix de clôture aux données fondamentales.

Après la fusion, un **nettoyage méticuleux des données** a été effectué pour assurer la **qualité et la complétude** du jeu de données final :

- **Gestion des `NaN` résiduels :** Malgré l'utilisation de `merge_asof`, quelques valeurs manquantes demeuraient pour des cas spécifiques (ex: `CCEP`, `PDD`, `APP`, `CEG`, `GEHC`, `ARM`). Ces cas ont été traités manuellement en imputant la valeur de clôture la plus proche disponible ou, si non pertinente, en décidant de la suppression ultérieure de ces observations si leur impact était minime. Les observations avec des `NaN` non imputables ont été supprimées (`dropna()`).
- **Exclusion de `GOOG` :** L'entreprise Alphabet (GOOG, catégorie C) a été retirée du dataset pour des raisons de cohérence, Google possédant deux tickers dans le NASDAQ. Nous avons ainsi retenu le ticker `GOOGL`, l'action de classe A, plus représentatif des investisseurs institutionnels

Défi Années Fiscales et Calendaires : La plage temporelle des observations nous a posé problème.. En effet, chaque entreprise publie ses rapports à des dates différentes de **l'année fiscale**, propre à l'entreprise elle-même. Les rendements du **NASDAQ-100** sont publiés sur **l'année calendaire**. Il n'est donc pas pertinent de

comparer le rendement de ces entreprises sur leurs années fiscales respectives aux rendements publiés par le NASDAQ.

Nous avons donc créé une matrice des rendements du **NASDAQ-100** prenant en colonnes et lignes **toutes les dates d'exercice de l'indice entre 2020 et 2024** (Notebook OPA#9 - Récupération des données OHLC - ETF QQQ). Cela nous a permis de comparer le rendement des entreprises sur leur années fiscales, directement avec le rendement de l'indice sur la même période et d'obtenir une classification (**0 - sous performe**, **1 - surperforme**) cohérente.

Apercu de la matrice des rendements du NASDAQ 100 (toutes dates de 2020 à 2024)

date	2020-01-02	2020-01-03	2020-01-06	2020-01-07	2020-01-08	2020-01-09	2020-01-10	2020-01-13	2020-01-14	2020-01-15
date										
2020-01-02	0.000000	-0.915988	-0.277572	-0.291451	0.457994	1.309215	1.050148	2.215951	1.813472	1.855107
2020-01-03	0.924456	0.000000	0.644318	0.630311	1.386684	2.245775	1.984312	3.160893	2.754692	2.796713
2020-01-06	0.278345	-0.640193	0.000000	-0.013917	0.737614	1.591204	1.331416	2.500464	2.096864	2.138616
2020-01-07	0.292303	-0.626363	0.013919	0.000000	0.751636	1.605345	1.345520	2.514731	2.111075	2.152833

4. Création de features - marginProfit

La `marginProfit` (ou marge bénéficiaire) mesure la **part du chiffre d'affaires** qu'une entreprise conserve en **bénéfice net après avoir payé toutes ses charges**. Exprimée en pourcentage, elle indique la **rentabilité globale** : plus elle est élevée, plus l'entreprise transforme efficacement ses revenus en profits.

Classique de **l'analyse fondamentale**, elle est **absente de nos données d'origine**. Il a donc fallu la calculer pour l'ajouter aux variables avec la formule suivante :

$$\text{Marge bénéficiaire} = \frac{\text{Bénéfice net}}{\text{Chiffre d'affaires}} \times 100$$

5. Création de features - YoY_Growth

Toutes les variables précédemment sélectionnées et générées ont été déclinées en **croissance annuelle.**

Par exemple, le `revenuePerShare_YoY_Growth` (ou croissance du chiffre d'affaires) mesure **l'évolution du chiffre d'affaires généré** (par action) entre deux exercices

consécutifs. Exprimée en pourcentage, elle reflète la **dynamique de croissance d'une entreprise** : une valeur positive indique une progression du chiffre d'affaires. Centrale en analyse fondamentale, cette variable est également absente des données fondamentales brutes. Il a donc fallu la calculer manuellement à l'aide de la formule suivante :

$$\text{Croissance annuelle} = \left(\frac{\text{Chiffre d'affaires}_{\text{année actuelle}} - \text{Chiffre d'affaires}_{\text{année précédente}}}{\text{Chiffre d'affaires}_{\text{année précédente}}} \right) \times 100$$

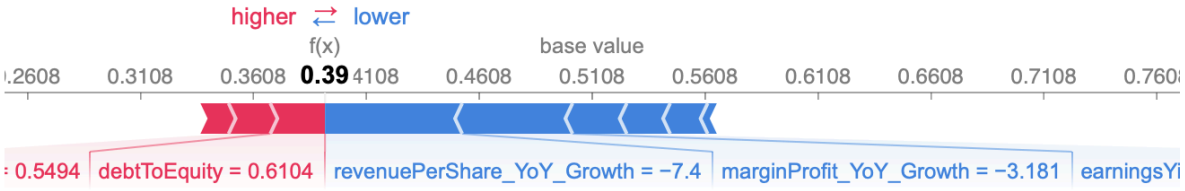
L'ajout de ces nouvelles variables **n'est pas sans impact**. La création de cette colonne nécessitant d'avoir des **données sur l'année précédente**, il n'est pas possible de l'effectuer pour les **observations datant de 2020** (la première année de notre dataset). On perd donc un cinquième de notre volumétrie de données.

Sélection des colonnes pertinentes pour la modélisation

Nous avons ensuite modélisé via un `RandomForestClassifier` (décrit plus loin) puis comparé l'impact des valeurs de chaque variable sur les **faux négatifs (FN) et faux positifs (FP)**. La démarche est de supprimer les variables qui poussent le modèle à **classifier de manière erronée**, sans pour autant aider sur d'autres observations à classifier correctement.

Exemple de classification influencée négativement par les variables YoY_Growth

```
--------------------------------------------------
Faux négatifs - Entreprise et année concernée : LRCX_2023
```



De cette démarche n'est restée que la variable `revenuePerShare_YoY_Growth`, ce qui est intéressant car c'est une valeur classique et très importante en analyse fondamentale. En cherchant de nouvelles variables explicatives et en explorant d'autres manières de faire, la démarche nous a **rapproché des pratiques métier**.

6. Traitements des données non-implémentés

Les données étant tabulaires et d'un volume relativement faible, nous n'avons pas eu recours à de la réduction de dimension. La normalisation a également été écartée, le `RandomForestClassifier` étant intrinsèquement **insensible à l'échelle des caractéristiques**.

Visualisations

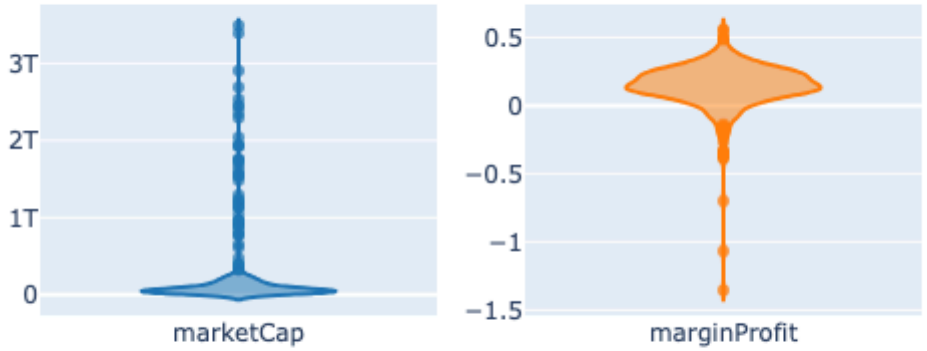
1. Introduction

Après la collecte et le nettoyage des données, cette section est consacrée à leur **exploration visuelle et statistique**. L'objectif est de développer une compréhension intuitive des données avant de procéder à la modélisation. On cherche à identifier les **caractéristiques principales de nos variables**, telles que leur distribution, la présence de valeurs aberrantes (outliers), et les relations qu'elles entretiennent les unes avec les autres.

2. Analyse des visualisations

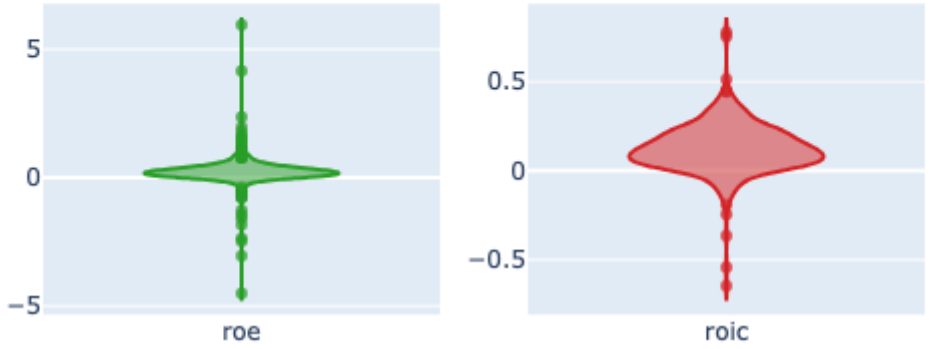
Graphiques 1 - Distribution des variables

Ces diagrammes en violon révèlent la **distribution, la densité et la dispersion** de nos principales variables quantitatives. On choisit une forme en **Violon** plutôt qu'en **Boxplot** car il y a une forte présence de valeurs extrêmes (bien que réelles) dans le dataset. La visualisation des quartiles, ne serait pas possible ici.



Pour la `marketCap`, la distribution est d'une **asymétrie positive extrême** : le corps du violon est écrasé près de zéro, indiquant qu'une très grande majorité des entreprises ont une **capitalisation boursière relativement "faible"**. La fine et longue pointe qui s'étend jusqu'à 3 trillions représente **les quelques "méga-capitalisations"** qui agissent comme des valeurs extrêmes massives.

En contraste, la `marginProfit` présente une distribution bien différente, ressemblant à une c**loche centrée légèrement au-dessus de zéro**. Cela signifie que la plupart des entreprises dégagent une **marge bénéficiaire positive et modérée**, bien que la queue de la distribution montre l'existence de sociétés subissant **beaucoup de pertes**.



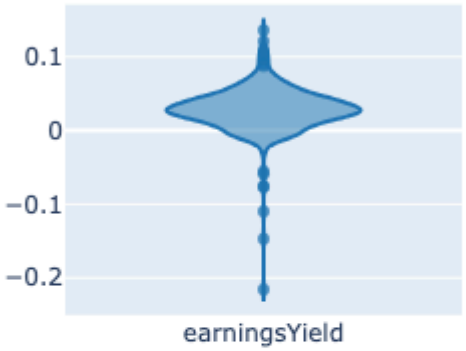
Ces deux distributions se concentrent sur **deux indicateurs de rentabilité clés** : le `roe` (Return on Equity) et le `roic` (Return on Invested Capital). La distribution du `roe` est particulièrement remarquable par sa **dispersion extrême**. Une forte concentration d'entreprises se situe autour d'un `roe` nul, mais les queues très allongées des deux côtés témoignent de la présence de valeurs extrêmes significatives : des entreprises avec une **rentabilité sur capitaux propres spectaculaire**, et d'autres avec **des pertes abyssales**. Le `roic`, bien que présentant également une large dispersion, semble **plus contenu et moins sujet aux valeurs extrêmes** que le `roe`. Cette distribution plus "stable" suggère que le `roic`, en incluant la **dette dans son calcul du capital** (la dette peut gonfler le `roe`), donne une image plus **normalisée** de la performance opérationnelle d'une entreprise.

On met ici en perspective **l'endettement** `debtToEquity` et la **croissance** `revenuePerShareYoYGrowth`.. La distribution du `debtToEquity` est, à l'image de la `marketCap`, fortement asymétrique. La quasi-totalité des entreprises sont

concentrées autour d'un ratio faible, proche de zéro, ce qui indique un **recours limité à l'endettement pour la majorité**. Cependant, la longue queue négative trahit la **présence d'entreprises très fortement endettées**, et surtout avec une équité négative, ce qui n'est **jamais un bon signal financier**.

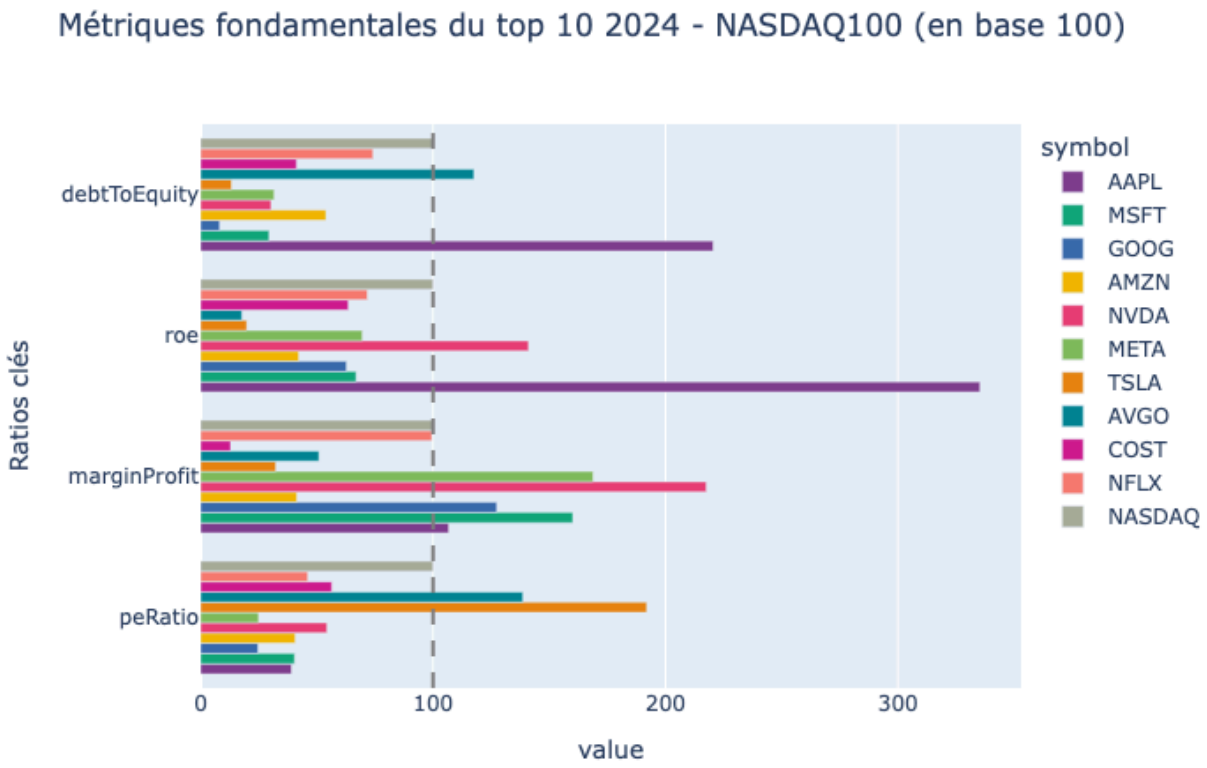
En ce qui concerne la queue positive, dans le **contexte du NASDAQ**, le plus probable est qu'il s'agisse d'entreprises faisant du **"Share Buyback",** qui rachètent leurs propres actions afin de réduire l'offre, et soutenir le cours. Cela a pour effet de **conforter les positions actuelles des actionnaires restants** et est **plus fiscalement avantageux** que de verser des dividendes.

De son côté, la croissance du revenu par action (`revenuePerShareYoYGrowth`) est centrée sur zéro mais avec une forte variance (un violon assez large), illustrant la dynamique du secteur : **beaucoup d'entreprises affichent des croissances similaires**, mais un nombre non négligeable connaît soit une croissance explosive, soit un déclin marqué.



Enfin, le dernier graphique montre la distribution du `earningsYield` (rendement bénéficiaire). De toutes nos variables, c'est celle qui s'approche le plus d'une **distribution normale**. Le corps du violon est bien défini et centré sur une valeur positive, ce qui est cohérent, la plupart des entreprises ayant des bénéfices. Bien que des **outliers existent**, la dispersion est bien moins extrême que pour le `roe` ou la `marketCap`. Cette **relative normalité** pourrait en faire un indicateur plus simple à intégrer dans certains modèles prédictifs sans nécessiter de transformation complexe.

Graphique 2 - Analyser les entreprises grâce aux métriques fondamentales

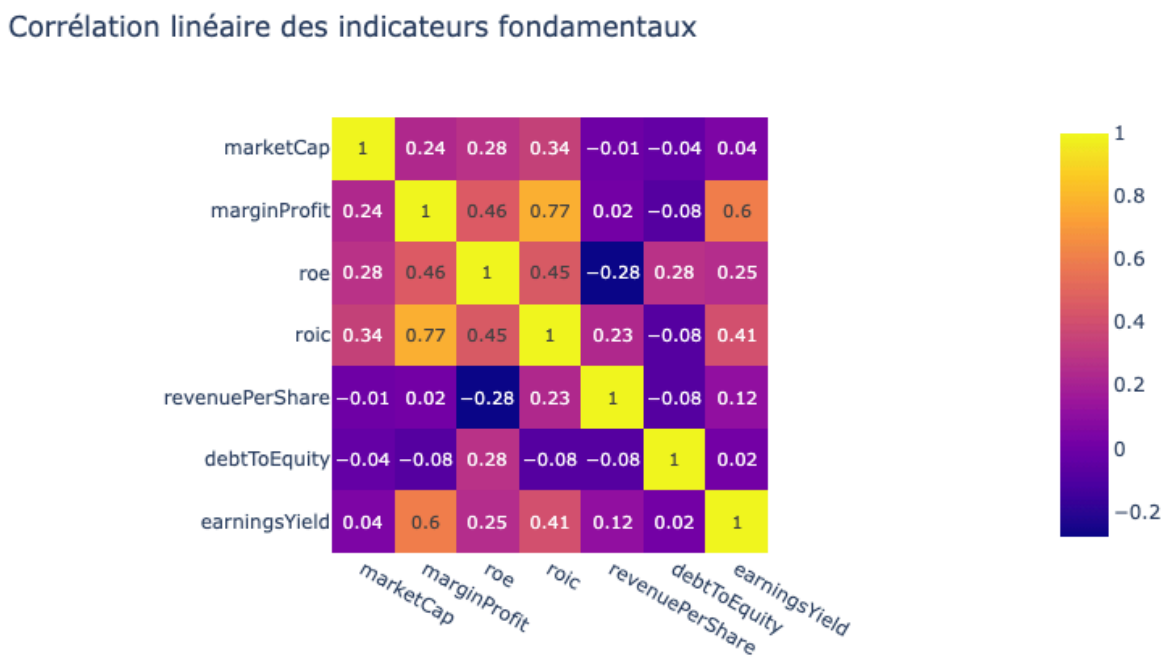


De l'importance de combiner les métriques fondamentales : Ce graphique illustre parfaitement l'importance de combiner les métriques fondamentales en analyse financière. On remarque en effet que le `roe` (Return on Equity) **d'Apple (AAPL) est extrêmement élevé,** dépassant de loin la moyenne du NASDAQ ainsi que tous les autres membres du groupe. On pourrait donc se dire au premier abord qu'Apple est très rentable par rapport à ses concurrents. Il convient néanmoins de regarder alors le ratio `debtToEquity`, qui est également le plus élevé du panel, et de très loin. On comprend qu'Apple utilise massivement **l'effet de levier** : l'entreprise contracte une dette importante pour amplifier artificiellement le retour pour ses actionnaires, une stratégie qui peut être très performante mais qui augmente aussi le **profil de risque de la société**.

Des stratégies diverses : L'analyse est d'autant plus révélatrice lorsqu'on observe la marge bénéficiaire (`marginProfit`). Contrairement à ce qu'on pourrait penser, celle d'Apple n'est pas la plus élevée ; des entreprises comme Microsoft (MSFT) ou NVIDIA (NVDA) affichent une **rentabilité opérationnelle supérieure**. Ce point est crucial : il confirme que le `roe` exceptionnel d'Apple est bien plus le fruit d'une **ingénierie financière pointue** que d'une simple supériorité sur ses marges.

Ainsi, le graphique expose des stratégies bien distinctes : une performance tirée par **l'excellence opérationnelle chez Microsoft et NVIDIA**, et une rentabilité spectaculaire mais **dépendante de l'endettement chez Apple**, dont le profil de risque est par conséquent différent.

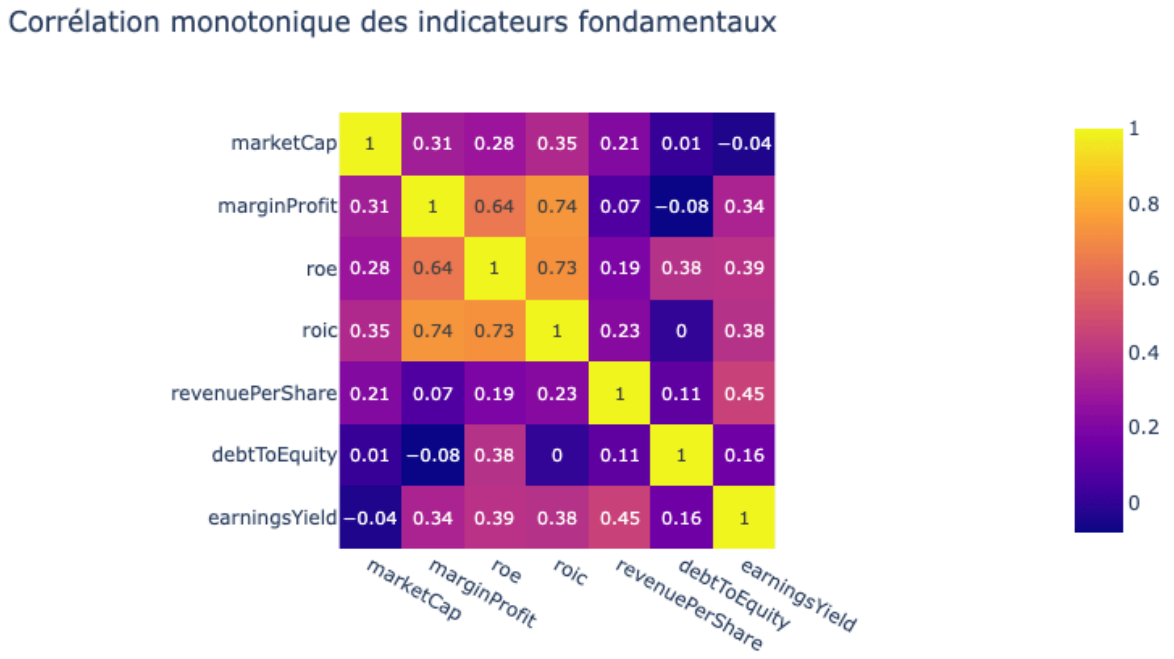
Graphique 3 - Relations entre les métriques fondamentales



Cette première matrice, basée sur le **coefficient de Pearson**, mesure l'intensité d'une **relation linéaire** entre nos variables. Elle met en lumière plusieurs dynamiques clés. On observe en premier lieu une forte **multicolinéarité** entre les indicateurs de rentabilité : la `marginProfit` est fortement et positivement corrélée au `roic` (Return on Invested Capital) avec un **coefficient de 0.77,** et de manière plus modérée au `roe` (0.46). Cette forte liaison est **attendue**, une meilleure marge se traduisant logiquement par une meilleure rentabilité globale.

Un autre **enseignement majeur** est le caractère unique du ratio `debtToEquity`. Celui-ci ne présente quasiment **aucune corrélation linéaire** avec les autres métriques (par exemple, -0.08 avec la marge et -0.04 avec la capitalisation), ce qui confirme qu'il capture une **dimension différente de l'entreprise** : **sa structure financière** plutôt que sa performance opérationnelle. Enfin, on note une **corrélation positive et notable de 0.60** entre le `earningsYield` (rendement bénéficiaire) et la `marginProfit`, ce qui suggère de manière intuitive que les entreprises aux marges les plus saines offrent, à valorisation égale, un meilleur rendement à leurs actionnaires.

Graphique 4 : Valider les relations avec le coefficient de Spearman

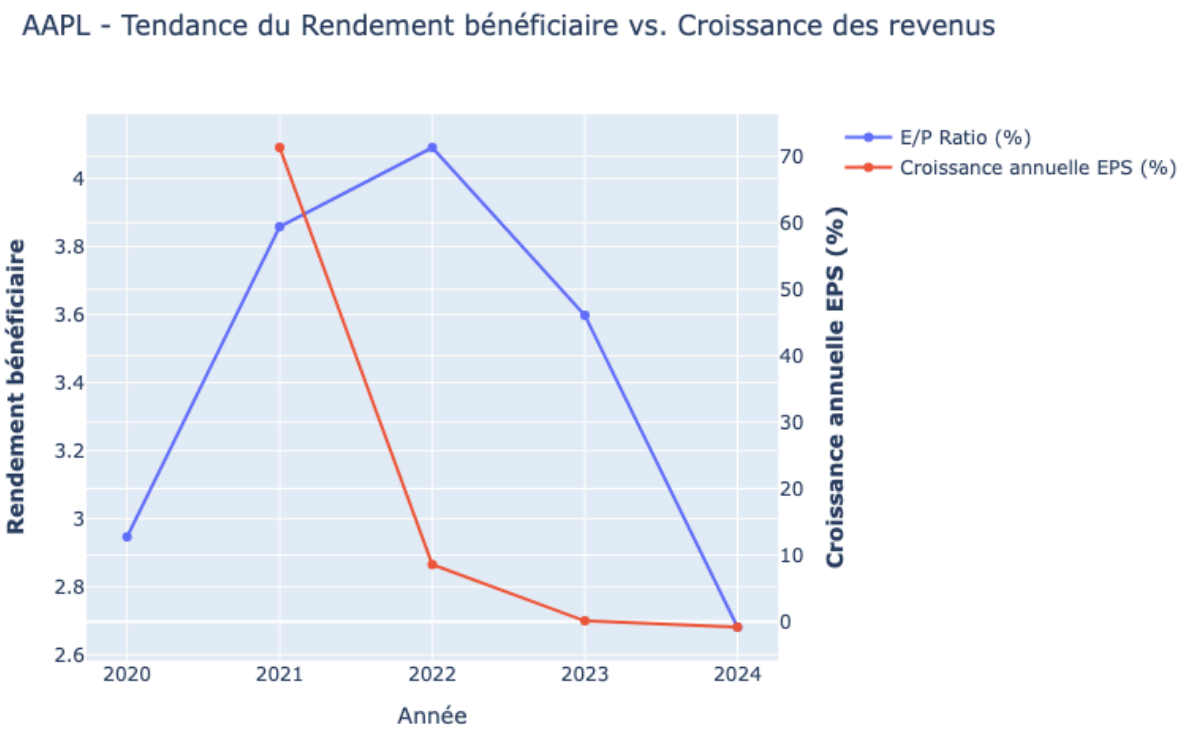


La seconde matrice, utilisant le **coefficient de Spearman**, est plus robuste aux valeurs aberrantes et détecte **toute relation où les variables évoluent dans la même direction**, même de manière non-linéaire. La comparaison avec la matrice de Pearson est riche d'enseignements. Les relations entre les métriques de rentabilité sont non seulement confirmées, mais même renforcées. La corrélation entre le `roe` et la `marginProfit` passe de **0.46 à 0.64**, et celle entre le `roe` et le `roic` bondit de **0.45 à 0.73**. Cette intensification suggère que les nombreuses valeurs extrêmes du `roe` (observées dans le graphique en violon) masquaient la véritable force de cette relation : la tendance est bien que **lorsque la rentabilité opérationnelle augmente, le `roe` augmente aussi**, et ce de manière très constante. De même, la corrélation entre `debtToEquity` et `roe` se renforce, passant de **0.28 à 0.38**. Bien que modérée, cette augmentation confirme la **robustesse de l'effet de levier** : un endettement plus élevé tend de manière plus claire à être associé à un `roe` plus élevé sur l'ensemble de notre échantillon. L'analyse monotonique valide donc les tendances linéaires et nous indique que ces relations sont **encore plus profondes et systématiques** que ce qu'une simple analyse linéaire laissait paraître.

Tendances du rendement bénéficiaire vs. Croissance des revenus

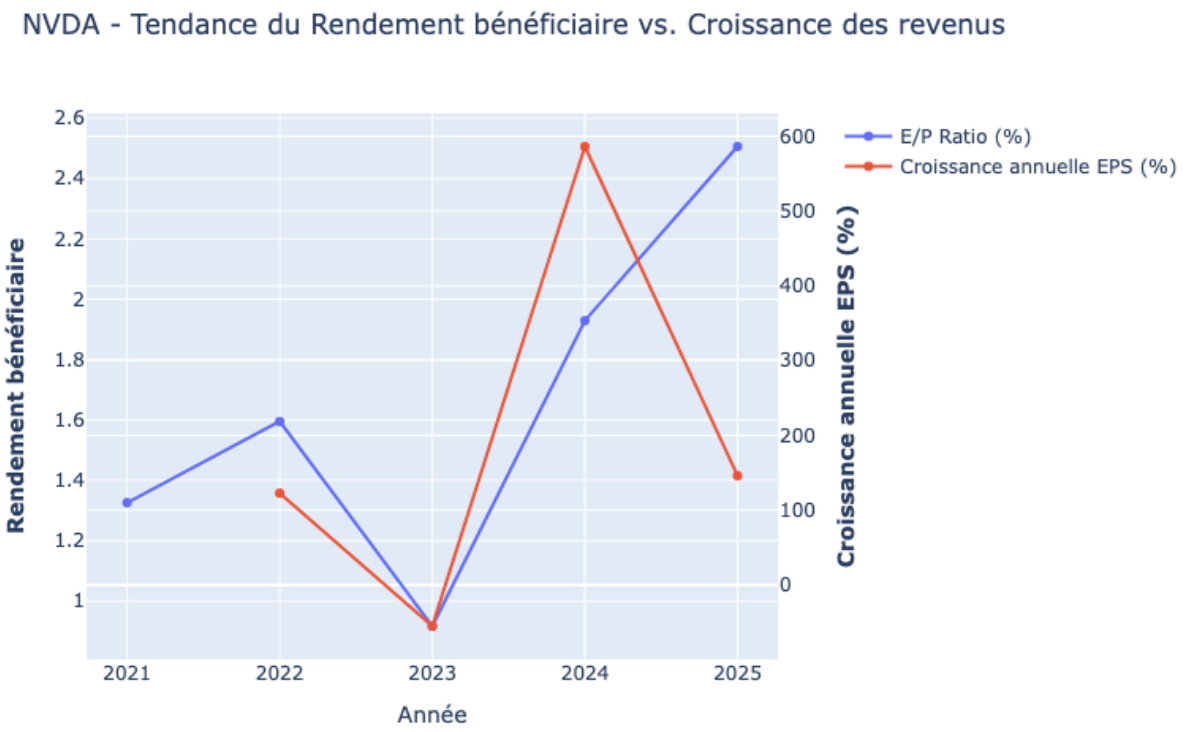
Ces derniers graphiques ajoutent une **dimension temporelle** essentielle à notre analyse en confrontant, pour trois entreprises emblématiques, l'évolution de leur rendement bénéficiaire (`earningsYield`), qui est une mesure de valorisation, à la croissance de leurs bénéfices (`revenuePerShareYoYGrowth`). Ils révèlent que la **relation entre performance et valorisation est loin d'être uniforme**.

Graphique 5 : Le cas d'Apple (AAPL)



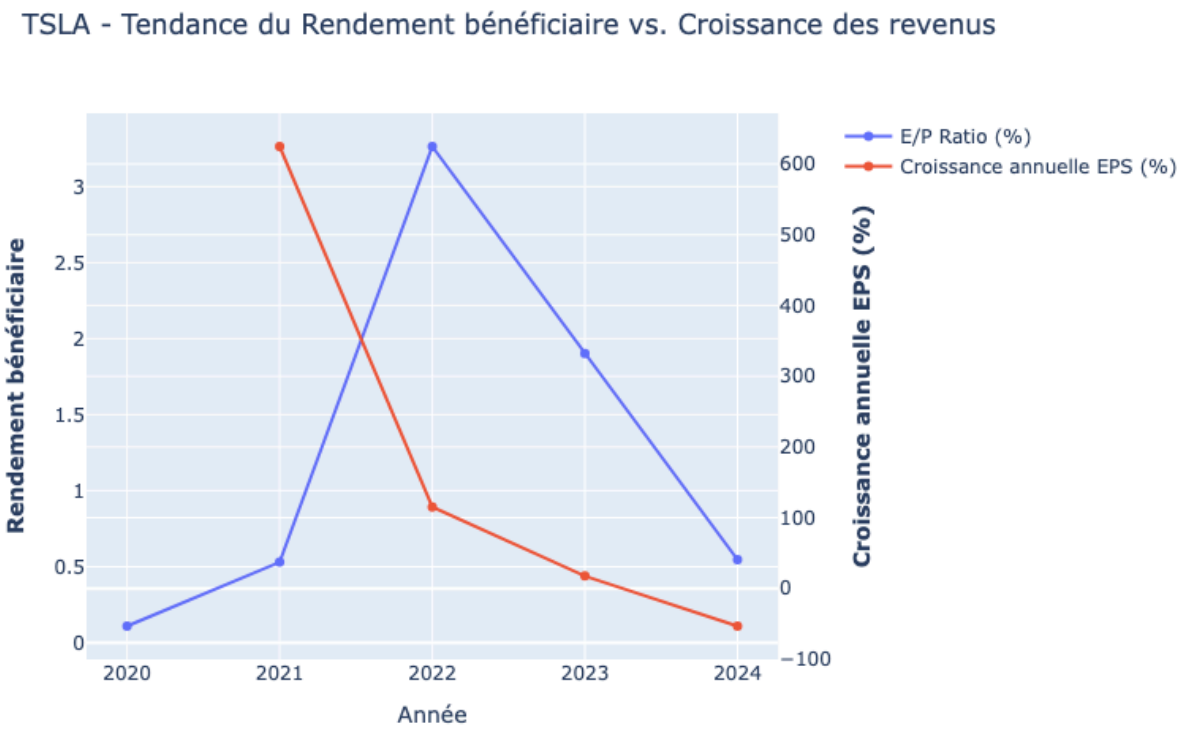
L'analyse du graphique d'Apple met en lumière un phénomène de découplage notable entre la croissance et la valorisation. On observe un **pic de croissance des bénéfices spectaculaire en 2021**, qui est suivi d'une décélération tout aussi drastique dès 2022. Pourtant, le rendement bénéficiaire (la courbe bleue) ne s'effondre pas ; il atteint au contraire son apogée en 2022, un an après le pic de croissance, avant de redescendre. Cela suggère une forte inertie du marché ou l'influence d'autres facteurs, comme les programmes massifs de rachat d'actions (Share buybacks) qui soutiennent le bénéfice par action et le cours de l'action. Le profil d'Apple est celui d'une entreprise mature dont la valorisation est moins dépendante des sursauts de croissance que de sa capacité à générer des flux de trésorerie stables et à rémunérer ses actionnaires.

Graphique 6 : Le cas de NVIDIA (NVDA)



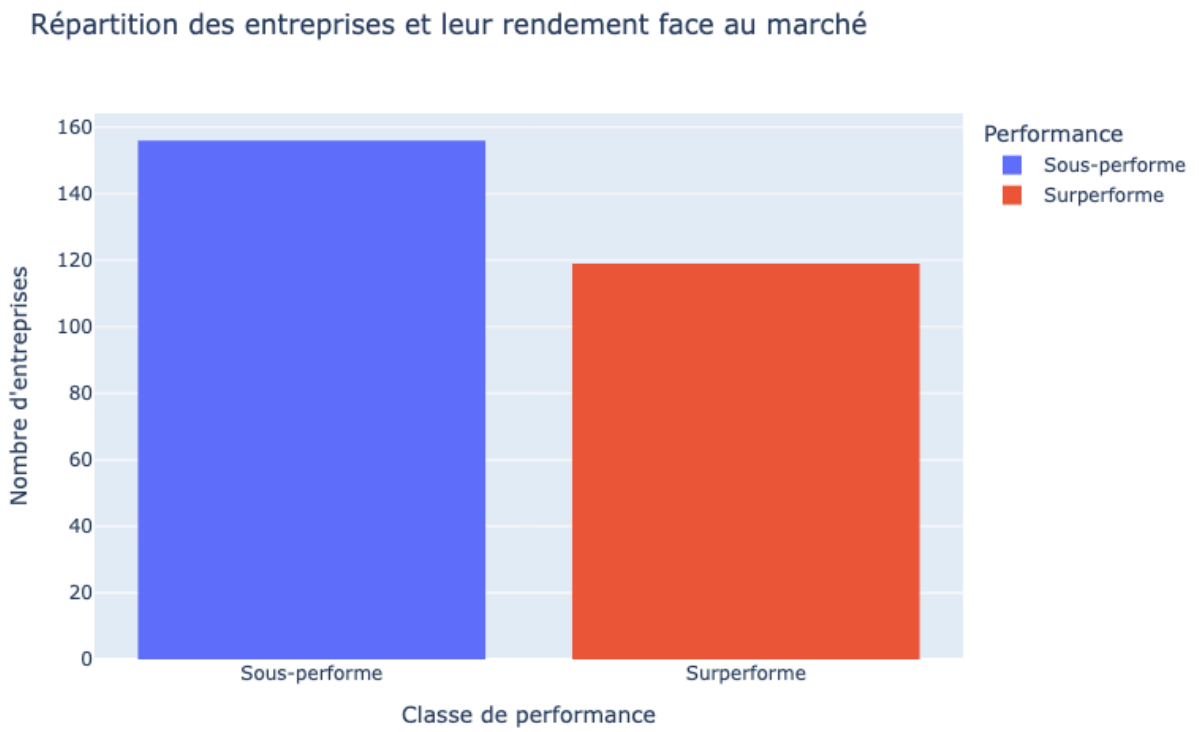
NVIDIA présente une dynamique **radicalement opposée avec AAPL**. La corrélation entre la croissance des bénéfices et le rendement bénéficiaire est ici **quasi parfaite et immédiate**. L'effondrement de la croissance en 2023 est accompagné d'une **chute du rendement** (le titre devient "cher"), tandis que l'explosion phénoménale de la croissance en 2024 se traduit instantanément par une e**nvolée du rendement**. Le marché semble ici hyper-réactif, **ajustant la valorisation en temps réel pour refléter les performances de croissance exceptionnelles**, principalement portées par le narratif de l'intelligence artificielle. Ce graphique illustre le cas d'une entreprise dont la valeur est intimement et directement liée à sa capacité à **maintenir une trajectoire de croissance exponentielle**.

Graphique 7 : Le cas de Tesla (TSLA)



Le graphique de Tesla est l'archétype d'une **valeur de croissance volatile**. À l'instar d'Apple, on y observe un pic de croissance extrême en 2021, suivi d'une forte décélération. Cependant, la réaction de la valorisation est différente. Pendant la phase d'hyper-croissance (2021), le rendement bénéficiaire est très faible, signifiant que le marché paie une **prime de valorisation extrême en anticipation des performances futures**. C'est seulement lorsque la croissance ralentit nettement que le **rendement bénéficiaire augmente**, atteignant son pic en 2022. Cela illustre le cycle typique d'une valeur de croissance : **une valorisation très élevée portée par l'anticipation, suivie d'une réévaluation** (le titre devient "moins cher") lorsque la croissance se normalise. La volatilité des deux courbes souligne le profil de risque élevé associé à ce type d'investissement.

Graphique 8 : Un aperçu de la classification en vue de la modélisation



Ce diagramme en barres expose la **distribution de notre variable cible**, qui classifie les entreprises en deux catégories : celles qui "**sous-performent**" (avec un rendement annuel de l'action inférieur au rendement annuel de l'indice) et celles qui "**surperforment**" le marché. L'observation immédiate est un déséquilibre clair mais non démesuré, entre les deux classes : le nombre d'entreprises en sous-performance (environ 155) est plus élevé que celui des entreprises en surperformance (environ 120). Cette information est cruciale pour la phase de modélisation. Un modèle entraîné sur un jeu de données déséquilibré pourrait développer un biais en faveur de la classe majoritaire. Il conviendrait **d'adapter les poids**, ou de synthétiser de la donnée, si ce déséquilibre n'était pas présent naturellement, et plutôt dû au manque de données. Dans la réalité, **il y a plus d'entreprises qui sous-performent plutôt que l'inverse**. La répartition des classes semble donc ici cohérente, et on choisit de **ne pas effectuer de traitement supplémentaire** pour le déséquilibre.

3. Conclusion et projection vers la modélisation

L'analyse exploratoire a permis de mettre en évidence plusieurs caractéristiques fondamentales de notre jeu de données. Nos variables explicatives sont marquées par des **distributions asymétriques** et la présence de nombreuses valeurs extrêmes mais réalistes, rendent l'utilisation d'un **modèle robuste** (tel que Random Forest) non pas optionnel, mais obligatoire.

Ces constat nous oriente naturellement vers des algorithmes de modélisation non-sensibles et capables de capturer des relations non-linéaires. Les **modèles ensemblistes,** tels que les forêts aléatoires (Random Forest) semblent particulièrement bien adaptés. Leur capacité à gérer nativement des interactions complexes entre variables et leur relative **insensibilité à la mise à l'échelle des données** en font des candidats de choix pour prédire la surperformance d'une entreprise sur la base des indicateurs fondamentaux que nous venons d'analyser.

Modélisation

Cette section détaille l'approche méthodologique adoptée pour la construction d'un modèle de machine learning visant à **prédire la direction relative des performances des entreprises** face à l'indice NASDAQ-100 via des métriques fondamentales.

1. Formulation du Problème et Ingénierie de la Variable Cible

Le projet a été formulé comme un problème de **classification binaire supervisée**. L'objectif principal est de prédire si une action **surperformera** (classe 1) ou **sous-performera** (classe 0) l'indice NASDAQ-100 sur une période **d'un an suivant la publication de ses résultats financiers.** Cette approche a été privilégiée en raison de la non-fiabilité notoire, voire de l'impossibilité, de prédire le rendement exact d'une action (problème de régression), comme le soutiennent les résultats exploratoires préliminaires (OPA#6) ayant révélés une inefficacité de la régression (score R^2 négatif). Il a été conclu que les indicateurs fondamentaux sont plus pertinents pour **prédire la direction relative des rendements** par rapport au marché.

La **définition de la variable cible** a été une étape critique et a évolué en plusieurs phases pour garantir la pertinence et la robustesse du modèle :

- **Phase 1 - Approche initiale (rejetée) :** Une première cible binaire simple classait les actions selon que leur **rendement annuel** était positif ou négatif. Cette approche a été jugée inadéquate car elle n'intègre pas **la performance globale du marché**. Une entreprise peut très bien afficher un rendement positif, mais bien en dessous de la performance de son indice.
- **Phase 2 - Benchmark annuel générique (rejeté) :** La cible a été affinée pour comparer le rendement de chaque action au rendement annuel de l'indice (NDX). Bien qu'introduisant la notion de performance relative, cette méthode souffrait d'un **défaut majeur** : les dates de calcul du benchmark (du 1er janvier au 31 décembre) ne correspondaient pas aux périodes de publication fiscale des entreprises, qui varient tout au long de l'année.

- **Phase 3 - Benchmark dynamique et personnalisé (retenue) :** Pour résoudre ce problème, une solution sur mesure a été développée (OPA#9, OPA#10). Une matrice de rendement de l'indice NASDAQ-100 a été calculée pour toutes les combinaisons de dates possibles. Pour chaque entreprise et chaque année, le rendement de l'indice sur la fenêtre temporelle exacte entre deux publications de résultats consécutives a été extrait et utilisé comme benchmark. Cette étape a été **fondamentale**, assurant que chaque prédiction est évaluée par rapport à un benchmark temporellement cohérent, reflétant ainsi la **génération d'alpha** sur la période pertinente. (**L'alpha** correspond à la partie du rendement surpassant l'indice de référence)

Il est à noter que suite à la **création de la variable cible** et comme lors de la création de la variable de croissance, un **cinquième de la volumétrie originelle des données est perdue**. En effet, le rendement est calculé grâce à la valeur close entre deux années. **Nous n'avons donc pas pu calculer le rendement pour 2024**, l'année 2025 n'étant pas terminée.

2. Sélection et Optimisation du Modèle

La sélection et l'optimisation du modèle ont suivi une approche itérative.

2.1 Sélection du Modèle

Exploration via AutoML : L'outil TPOTClassifier a été employé (OPA#7, OPA#8) pour explorer automatiquement un vaste espace de pipelines de modélisation. Les résultats ont clairement convergé vers les **modèles d'ensemble basés sur les arbres de décision**, en particulier le `RandomForestClassifier`.

Adoption du `RandomForestClassifier` : Fort de cette validation et des conseils de notre tuteur, le `RandomForestClassifier` de `scikit-learn` a été retenu comme modèle principal pour les raisons suivantes :

- **Performance et Robustesse :** Sa nature d'ensemble (bagging) le rend **robuste au surapprentissage** et capable de capturer des relations non linéaires complexes.
- **Interprétabilité :** Il est compatible avec des bibliothèques avancées comme `shap`, essentielle pour comprendre les facteurs de prédiction.

2.2 Optimisation des Hyperparamètres

Pour affiner le modèle, une recherche d'hyperparamètres a été conduite via `RandomizedSearchCV` (OPA#11). Cette méthode a été préférée à `GridSearchCV` pour son efficacité à **explorer un large espace de paramètres** (par exemple, `n_estimators`, `max_depth`, `min_samples_split`).

3. Ingénierie et Sélection des Caractéristiques

Le jeu de caractéristiques a fait l'objet d'un **travail itératif d'amélioration**, comme détaillé dans la partie Preprocessing et OPA#12 :

- **Création de caractéristiques de croissance :** Des indicateurs de croissance annuelle (par exemple, `...YoYGrowth`) ont été calculés pour la plupart des métriques fondamentales afin de capturer la dynamique de l'entreprise.
- **Transformation de caractéristiques :** Une transformation clé a été le remplacement du ratio `peRatio` par son inverse, `earningsYield` **(E/P)**. Cette métrique est plus stable et continue, gérant **naturellement les cas de bénéfices nuls ou négatifs** qui rendent le P/E Ratio non interprétable.
- **Sélection itérative de caractéristiques :** Après une première modélisation, une analyse de l'importance des variables a révélé que certaines caractéristiques synthétiques ajoutent **plus de bruit que de signal**. Un processus d'élimination itératif (OPA#12) a été mené, **résultant en un modèle final plus performant** et un jeu de données plus cohérent.

4. Cadre d'Évaluation et Métriques de Performance

Un cadre d'évaluation a été mis en place pour garantir la validité des résultats (OPA#11, OPA#12).

4.1 Validation

En raison de la nature chronologique des données, la **Validation Croisée Temporelle** (`TimeSeriesSplit`) a été systématiquement utilisée. Cette technique garantit que le modèle est **toujours entraîné sur des données passées** et validé sur des données futures, fournissant une estimation réaliste de sa performance en conditions réelles, **évitant ainsi la fuite de données du futur vers le passé**.

4.2 Métriques de Performance

Le **Rapport de Classification** (`precision`, `recall`, `f1-score`) et la **Matrice de Confusion** ont été les outils principaux d'analyse des performances.

> La Précision **de la classe 1 (surperformance)** a été identifiée comme la métrique la plus importante du point de vue métier, car elle mesure la capacité du modèle à identifier les opportunités d'investissement gagnantes.

Une métrique `average_precision` a été choisie pour l'optimisation des hyperparamètres, car elle est particulièrement adaptée aux jeux de données où l'on souhaite **optimiser pour une classe sans défavoriser l'autre**.

5. Résultats et Interprétation du Modèle

Cette section retrace le cheminement de la modélisation, depuis les tentatives initiales jusqu'au modèle final optimisé, en détaillant les performances obtenues et les enseignements clés.

5.1 Performances Initiales

Le parcours de modélisation a débuté par une **tentative de régression** pour prédire le rendement numérique exact (`%Return_NY`) à l'aide d'un `RandomForestRegressor` (OPA#6). Cette phase a rapidement mis en évidence l'inefficacité de cette approche, avec un **score R^2 négatif**, confirmant la difficulté à prédire l'amplitude des mouvements boursiers.

---

	precision	recall	f1-score	support
0	0.29	0.14	0.19	35
1	0.52	0.73	0.60	44
accuracy			0.47	79
macro avg	0.41	0.44	0.40	79
weighted avg	0.42	0.47	0.42	79

Suite à cet échec, le problème a été reformulé en une **première classification binaire** (rendement positif/négatif). Les résultats étaient globalement inefficaces, avec une précision de 0.29 et un rappel de 0.14 pour la classe 0, et une précision de

0.52 pour la classe 1, pour une accuracy globale de 47%. La matrice de confusion initiale révélait 30 Faux Positifs et 12 Faux Négatifs.

Classe prédite	0	1
Classe réelle		
0	5	30
1	12	32

5.2 Performance du Modèle Final Optimisé

Après les étapes d'ingénierie de la variable cible, de sélection du modèle, d'optimisation des hyperparamètres et de raffinage des caractéristiques, le modèle **RandomForestClassifier** a présenté les performances suivantes sur le jeu de test (OPA#11 et OPA#12) :

	precision	recall	f1-score	support
0	0.76	0.75	0.76	60
1	0.53	0.55	0.54	31
accuracy			0.68	91
macro avg	0.65	0.65	0.65	91
weighted avg	0.68	0.68	0.68	91

La matrice de confusion associée était la suivante :

Classe prédite	0	1
Classe réelle		
0	45	15
1	14	17

Ces résultats montrent une **nette amélioration générale**, surtout sur la classe 0, avec une accuracy passant à 68%. Le modèle est particulièrement performant pour la **détection de la classe 0 (sous-performance)**, avec une précision de 0.76 et un rappel de 0.75, ainsi qu'une réduction significative des Faux Positifs (passant de 30 à

15). Les métriques pour la **classe 1 (surperformance)** se sont également améliorées (Précision 0.53, Rappel 0.55), passant d'une performance médiocre à modérée.

Essentiellement, le nouveau modèle **stabilise les performances sur la classe 1**, tout en **améliorant grandement les performances sur la classe 0**. Il convient également de noter la différence de répartition du support entre les deux jeux de données, le ratio (**34% surperformant** plutôt que **55%**) du dernier modèle correspondant bien plus à la réalité du marché.

6. Interprétabilité et Réorientation Stratégique

Comprendre les décisions du modèle était une priorité, menant à une réorientation stratégique.

6.1 Interprétabilité du Modèle

- **Importance Globale des Caractéristiques :** L'importance a d'abord été évaluée par la méthode native du Random Forest (`feature_importances_`) et par l'Importance par Permutation, sans réussir à en tirer de conclusions significatives.
- **Interprétation Globale et Locale avec SHAP :** La bibliothèque `shap` (SHapley Additive exPlanations) a été utilisée pour une analyse en profondeur (OPA#12). À l'aide de `shap.TreeExplainer` et `shap.force_plot`, chaque prédiction individuelle a pu être décomposée pour identifier les contributions de chaque caractéristique. Cette technique a été précieuse pour analyser les erreurs à haute confiance, révélant la performance sur la classe 0 du modèle.

6.2 Enseignements Clés et Réorientation Stratégique

Malgré les progrès constants, la **prédiction de la surperformance (classe 1) avec une très haute fiabilité est restée un défi**. Cependant, une analyse approfondie des prédictions du modèle ayant un **haut niveau de confiance pour la classe 0 (sous-performance)** (confiance > 0.7, OPA#12) a révélé un enseignement stratégique majeur : ces prédictions étaient extrêmement fiables, avec une réduction drastique des Faux Positifs.

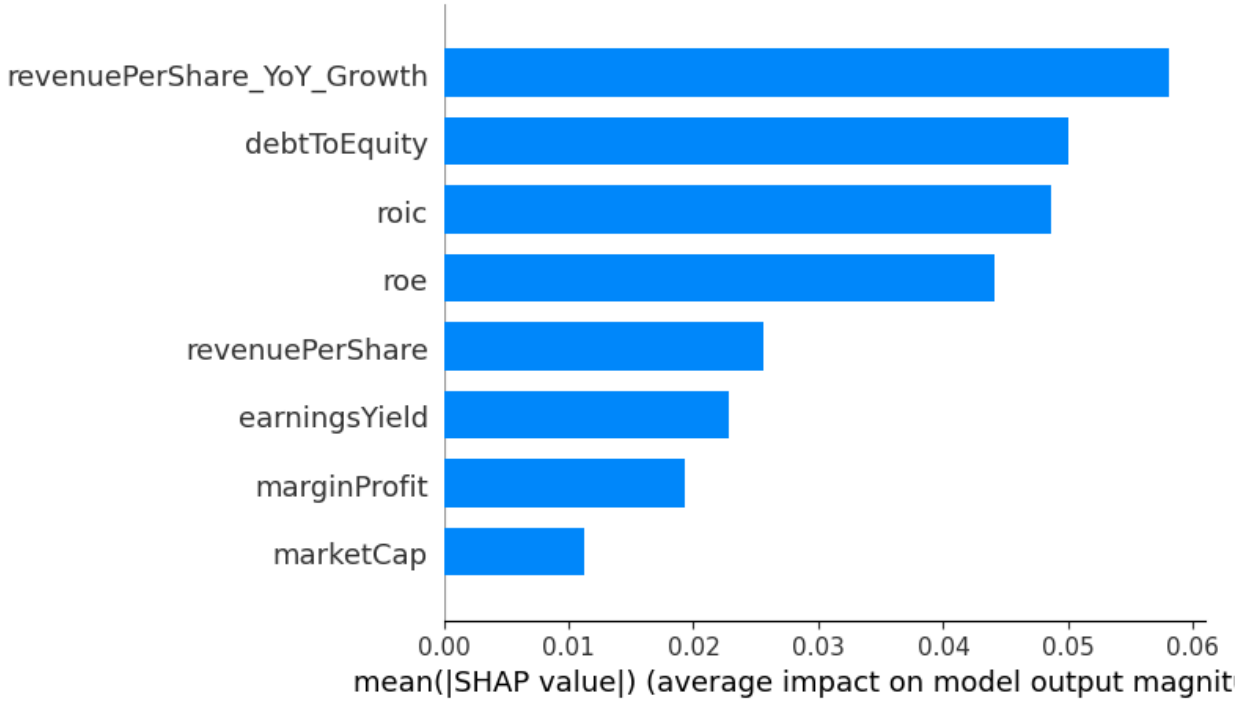
Classe prédite	0	1
Classe réelle		
0	13	1
1	2	2

La précision de haute confiance pour la classe 0 est de **92%**.

6.3 Interprétabilité du Modèle : Le Profil d'une Action à Risque

Pour comprendre pourquoi notre modèle identifie certaines entreprises comme étant à haut risque de sous-performance, nous avons combiné **l'analyse globale de l'importance des caractéristiques** (SHAP) avec une analyse comparative des distributions des variables clés. Cette approche nous permet de comprendre ce que le modèle a appris à considérer comme un "perdant" avec une haute confiance.

6.3.1 Analyse de l'Importance Globale des Caractéristiques (SHAP)



Le graphique de synthèse SHAP révèle la **hiérarchie des facteurs** qui influencent les prédictions du modèle. Il est clair que les décisions ne reposent pas sur un seul

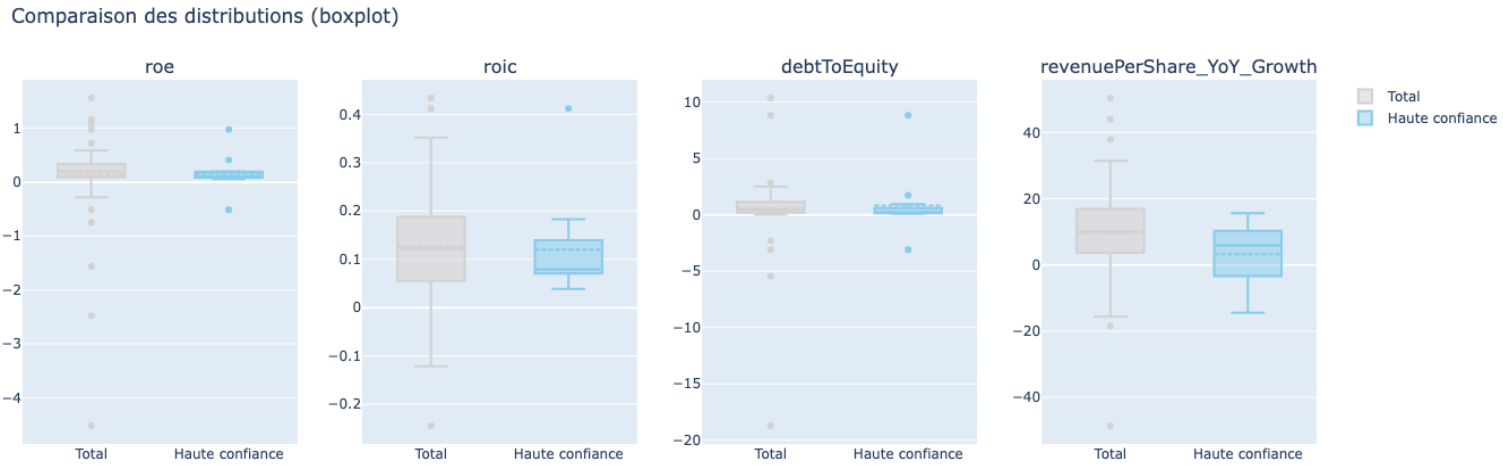
indicateur, mais sur un ensemble de preuves fondamentales. Les trois caractéristiques les plus influentes sont :

1. **`revenuePerShare_YoY_Growth`** : La croissance (ou son absence) est le facteur le plus déterminant.
2. **`debtToEquity`** : La structure financière et le niveau d'endettement jouent un rôle crucial.
3. **`roic`** : L'efficacité avec laquelle l'entreprise utilise son capital est le troisième pilier de la décision.

Ces résultats confirment que notre modèle a appris à raisonner de manière similaire à un analyste fondamental, en se concentrant sur la dynamique de croissance, la solidité du bilan et la rentabilité opérationnelle.

6.3.2 Analyse Comparative des Distributions : Total vs. Haute Confiance

Pour affiner notre compréhension, nous avons **comparé la distribution** de ces variables clés **pour l'ensemble du jeu de données ("Total") par rapport au sous-ensemble des prédictions de sous-performance** (classe 0) faîtes avec une haute confiance (>0.7).

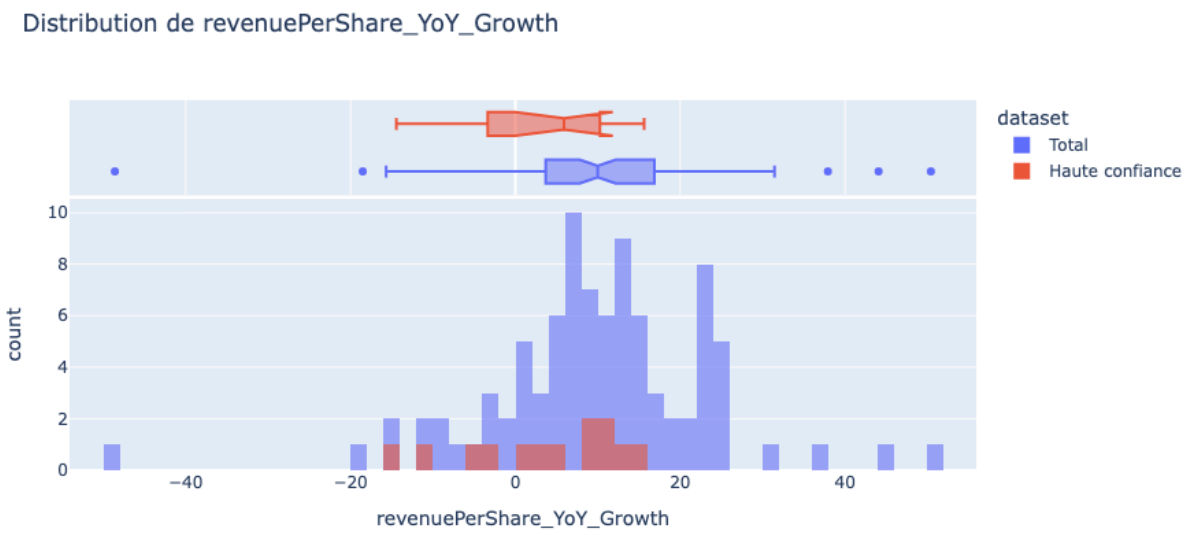


Ce graphique offre une bonne vue d'ensemble. Pour les entreprises classées à haut risque ("Haute confiance", en bleu clair), nous observons un déplacement systématique des distributions par rapport à la population totale ("Total", en gris) :

- Le `roe` et le `roic` **médians** sont nettement plus faibles.
- La croissance des revenus (`revenuePerShare_YoY_Growth`) **médiane** est inférieure.
- Le ratio `debtToEquity` **médian** est légèrement plus élevé et sa dispersion est plus contenue.

Analysons chaque facteur en détail pour en extraire la signification métier.

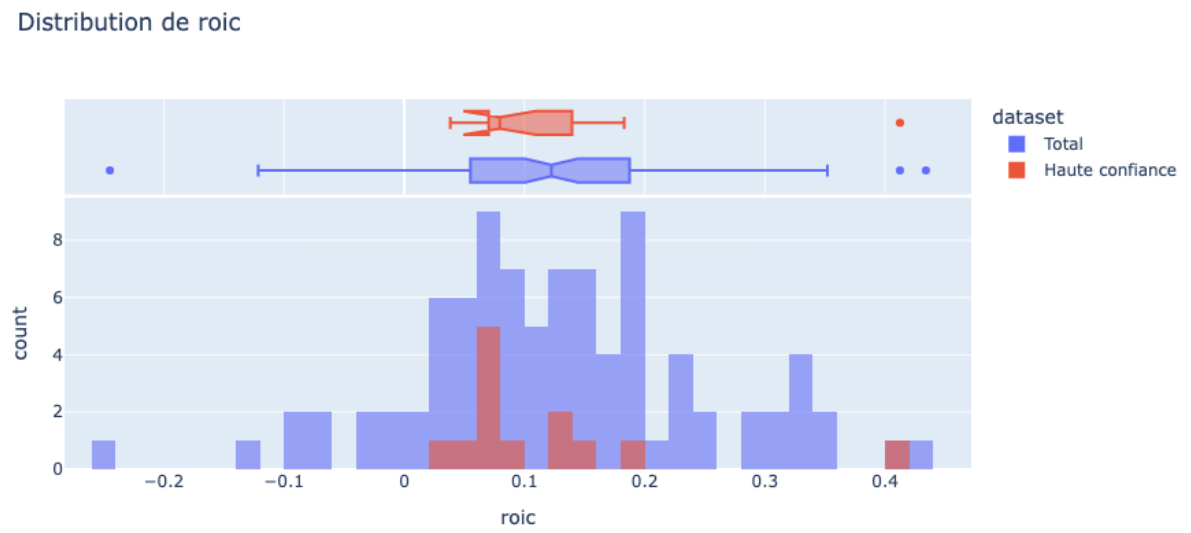
6.3.3 Le Signal Principal : La Stagnation de la Croissance



La visualisation est sans équivoque. Alors que la distribution globale (en bleu) montre une médiane de croissance des revenus positive, la distribution des entreprises à haut risque (en rouge) est **massivement décalée vers la gauche, avec une médiane proche de zéro ou légèrement négative**.

Interprétation : Le signal négatif le plus important pour notre modèle est la **perte de momentum**. Une entreprise qui ne parvient plus à faire croître son chiffre d'affaires, ou pire, qui voit ses revenus décliner, est immédiatement identifiée comme étant en danger. Le modèle a appris que l'absence de croissance est un précurseur de la sous-performance future.

6.3.4 Le Signal de Rentabilité : L'Inefficacité Opérationnelle



Cette visualisation confirme le constat du boxplot initial. Pour les entreprises à haut risque (en rouge), le **ROIC médian est systématiquement plus bas** que pour la population générale.

Interprétation : Le modèle ne se contente pas de regarder la croissance ; il évalue l'efficacité. Une entreprise peut avoir des revenus stables, mais si elle est incapable de transformer efficacement son capital en profits, le modèle y voit un signe de faiblesse structurelle. Il ne s'agit pas seulement de détecter les entreprises qui perdent de l'argent, mais aussi celles qui sont **significativement moins rentables que leurs pairs**.

6.3.5 Le Signal de Solidité : La Structure Financière

L'analyse du ratio dette/fonds propres est plus subtile. La distribution pour le groupe à haut risque (en rouge) est plus concentrée et présente une médiane légèrement positive.

Interprétation : Le modèle semble pénaliser les extrêmes. Il identifie un risque non seulement dans un **endettement très élevé**, mais aussi potentiellement dans des situations de **fonds propres négatifs** (qui apparaissent dans la distribution totale mais sont moins présents dans le groupe à haute confiance). Le modèle a appris à reconnaître qu'une structure financière fragile, signalée par un ratio `debtToEquity` anormal, est un facteur aggravant qui, combiné à une faible croissance et une faible rentabilité, constitue un cocktail à haut risque.

6.2 Synthèse et Implications Métier : L'Archétype de l'Entreprise à Risque

L'analyse combinée de ces graphiques nous permet de définir l'archétype d'une entreprise que notre modèle identifie avec une haute confiance comme étant susceptible de sous-performer :

> **Il s'agit d'une entreprise qui a perdu sa dynamique de croissance (revenus stagnants ou en déclin), qui est devenue moins efficace dans l'utilisation de son capital (ROIC et ROE plus faibles que la moyenne), et dont la structure financière est une source de préoccupation (endettement significatif).**

Implications pour un analyste ou un gestionnaire de portefeuille :

- **Au-delà des indicateurs isolés** : Le modèle offre une vision synthétique. Une entreprise peut avoir un ROIC acceptable, mais si sa croissance est nulle et sa dette élevée, le modèle lèvera un drapeau rouge. Il détecte la **combinaison toxique** des facteurs.
- **Un système d'alerte précoce** : Le modèle peut identifier des signaux de faiblesse **avant qu'ils ne se traduisent par des pertes nettes massives**. La stagnation de la croissance et la baisse de l'efficacité sont souvent des signes avant-coureurs.
- **Aide à la décision factuelle** : Plutôt que de se fier à une intuition, un analyste peut utiliser le diagnostic du modèle pour justifier une analyse plus approfondie d'une position ou pour écarter un candidat à l'investissement. Le modèle fournit **un "second avis" quantifié.**

Ces observations nous ont conduits à une **réorientation de la stratégie d'application du modèle** :

- **Passer d'une stratégie de "picking des gagnants" à une stratégie de "filtrage des perdants" :** Plutôt que d'utiliser le modèle pour identifier des actions à acheter pour surperformer (où la précision n'est pas suffisante pour des décisions d'investissement directes et agressives), il devient un **outil de gestion des risques**.
- **Détection des signaux négatifs extrêmes :** Le modèle excelle à identifier les entreprises dont la sous-performance est prédite avec une très haute confiance, grâce à des fondamentaux détériorés ou des croissances négatives.
- **Système d'alerte pour la préservation du capital :** En conclusion, le modèle ne se positionne plus comme un prédicteur d'opportunités d'achat systématique, mais comme un **système d'alerte**. Il permet à un investisseur de **minimiser les pertes potentielles** en écartant les actions les plus susceptibles de sous-performer significativement le marché. Cette application capitalise sur la force et la fiabilité du modèle là où elle est la plus avérée, offrant une valeur concrète dans un contexte de gestion de portefeuille où la préservation du capital est primordiale.

Autre piste : l'analyse technique

1. Définition de l'analyse technique

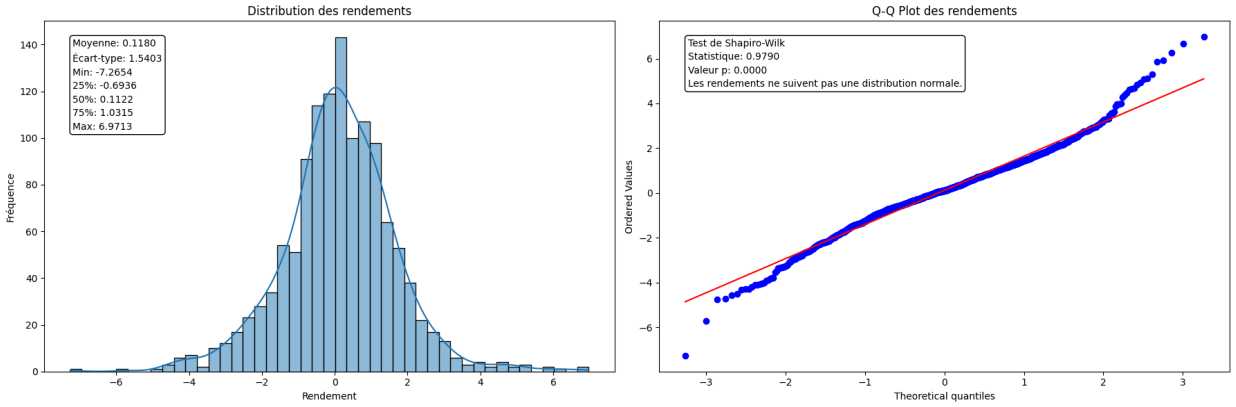
L'analyse technique est une méthode d'étude des marchés financiers qui consiste à **analyser les graphiques des prix et des volumes de transactions** pour anticiper les mouvements futurs.

Elle repose sur l'idée que **toute l'information disponible est déjà reflétée dans le prix**, et que les tendances passées tendent à se répéter.

Les analystes techniques utilisent des outils comme les moyennes mobiles, RSI, MACD, supports/résistances et figures chartistes pour identifier des signaux d'achat ou de vente.

Contrairement à l'analyse fondamentale, elle ne se base pas sur les résultats financiers ou l'économie de l'entreprise, mais **uniquement sur l'évolution des cours et du comportement du marché.**

Distribution de rendement de l'entreprise Apple



2. Création de feature

RSI (Relative Strengh Index)

Le RSI est un oscillateur de momentum qui mesure la vitesse et l'amplitude des variations de prix d'un actif. Développé par J. Welles Wilder, cet indicateur évolue entre 0 et 100 et permet **d'identifier les zones de surachat** (généralement au-dessus de 70) et de survente (généralement en dessous de 30).

$$RSI = 100 - \frac{100}{1 + RS}$$

où RS représente le ratio entre la moyenne des hausses et la moyenne des baisses sur une période donnée, **traditionnellement 14 jours**. Cet indicateur aide les traders à **anticiper les retournements de tendance** en signalant quand un titre pourrait être temporairement suracheté ou survendu.

Moyenne mobile

Les moyennes mobiles constituent l'**un des outils d'analyse technique les plus fondamentaux**, lissant les fluctuations de prix pour révéler la tendance sous-jacente d'un actif. La moyenne mobile simple (MMS) se calcule en **additionnant les prix de clôture** sur n périodes et en divisant par n :

$$MMS = \frac{P_1 + P_2 + \cdots + P_n}{n}$$

La **moyenne mobile exponentielle (MME)** accorde plus d'importance aux prix récents avec la formule :

$$MME = (\text{Prix actuel} \times \frac{2}{n+1}) + (\text{MME pr\#c\#dente} \times (1 - \frac{2}{n+1}))$$

Ces indicateurs servent de niveaux de support et de résistance dynamiques, et leurs croisements avec les prix ou entre elles-mêmes, et **génèrent des signaux de trading** largement utilisés par les analystes techniques.

MACD (Moving average convergence divergence)

Le **MACD** est un indicateur de suivi de tendance qui révèle les relations entre deux moyennes mobiles des prix d'un titre. Il se compose de trois éléments : la ligne MACD, la ligne de signal et l'histogramme.

$$MACD = MME_{12} - MME_{26}$$

La ligne de signal correspond à une MME de 9 périodes de la ligne MACD, tandis que l'histogramme représente la différence entre la ligne MACD et la ligne de signal. Les **croisements entre ces lignes génèrent des signaux d'achat ou de vente**, particulièrement utiles pour identifier les changements de momentum.

3. Objectif de notre programme.

1ere modélisation : XGBOOST

Ce programme utilise des **données boursières** pour **prédire si une action va surperformer le marché sur une période donnée**.

Il analyse **l'historique des prix et des indicateurs techniques** (RSI, MACD, moyennes mobiles, rendements). Les données sont triées chronologiquement afin de respecter la logique du temps.

Une partie des données anciennes (80%) sert à entraîner un modèle d'intelligence artificielle, et les plus récentes (20%) permettent de tester sa fiabilité.

L'algorithme utilisé, XGBoost, **identifie des schémas complexes** entre les indicateurs et la performance future.

Le programme évalue ensuite la qualité des prévisions grâce à des mesures statistiques et calcule un score global (ROC-AUC).

Il affiche aussi quels indicateurs techniques ont le plus influencé les décisions du modèle. L'objectif est de fournir un **outil d'aide à la décision pour l'investissement basé sur des analyses quantitatives**. Il peut être utilisé pour tester des stratégies sur l'historique avant application réelle.

Ce programme constitue une base pour développer un système prédictif automatisé en finance.

2e modélisation : Keras

Les **réseaux de neurones récurrents (RNN)** et particulièrement les **LSTM** (Long Short-Term Memory) sont particulièrement adaptés aux séries temporelles financières car ils peuvent capturer les dépendances à long terme dans les données séquentielles. Pour ce projet, nous avons essayé un **modèle hybride** combinant :

- **Couches LSTM** : Pour capturer les patterns temporels dans les prix
- **Couches Dense** : Pour l'intégration des indicateurs techniques
- **Dropout** : Pour éviter le surapprentissage

Les données incluent :

- Prix **OHLCV** (Open, High, Low, Close, Volume) de l'action
- Prix du **Nasdaq** (ETF QQQ)
- Prix du **Pétrole** (ETF USO)
- Prix de **l'or** (ETF GLD)

Plusieurs fenêtres de données ont été testées : de 10 à 60 jours.

Comme les différentes configurations du modèle donnent des résultats similaires, **un seul résultat sera présenté sur une seule action**.

3e Modélisation : TimesFM (modèle PyTorch pré-entraîné)

TimesFM est un modèle de fondation développé par Google Research qui présente plusieurs caractéristiques remarquables. Le modèle compte 200 millions de paramètres et a été entraîné sur plus de 100 milliards de points de données temporelles réelles. Il utilise une architecture de type décodeur uniquement, **similaire aux grands modèles de langage**, et emploie des mécanismes d'auto-attention pour apprendre les relations complexes entre différents points temporels.

Le modèle peut aussi fonctionner selon un principe d'auto-régression, **prédisant le segment suivant d'une série temporelle** puis utilisant cette prédiction pour les prévisions suivantes. Il supporte des longueurs de contexte et des horizons de prédiction variables, ce qui le rend particulièrement flexible pour diverses applications. Pour notre cas, et pour l'évaluation, nous n'utilisons pas cette fonctionnalité, et **regarderons uniquement la prédiction pour le jour suivant**.

L'entraînement de **TimesFM** s'est basé sur un corpus diversifié de séries temporelles incluant des données météorologiques, de trafic, de tendances de recherche et d'autres domaines présentant des patterns réguliers saisonniers. Cette diversité était censée permettre au modèle de **généraliser à de nouveaux types de données sans nécessiter de ré-entraînement spécifique**. Cependant **les données financières n'étaient pas incluses dans le corpus d'entraînement original de TimesFM**, qui se concentrait sur des séries temporelles présentant des patterns plus réguliers.

4. Analyse des Résultats

1ere modélisation : XGBOOST

📊 Rapport de classification (XGBoost) :

	precision	recall	f1-score	support
0	0.49	0.50	0.49	657
1	0.54	0.53	0.53	731
accuracy			0.51	1388
macro avg	0.51	0.51	0.51	1388
weighted avg	0.51	0.51	0.51	1388

ROC-AUC: 0.51

Le rapport montre que le modèle **n'arrive pas à faire mieux que le hasard**, avec une précision globale et un score ROC-AUC proches de 0,5.

Les indicateurs utilisés (RSI, MACD, moyennes mobiles, rendements) **ne semblent pas suffire pour détecter un signal clair**.

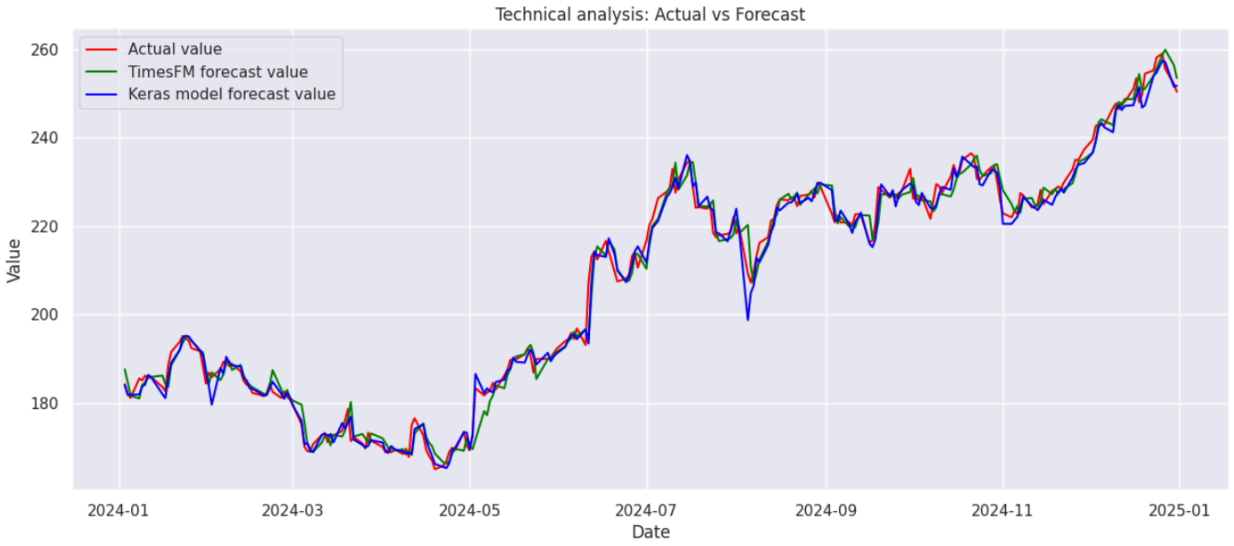
La performance très faible indique que le marché est **difficile à prédire avec les données actuelles**.

Pour améliorer, il faudrait ajouter d'autres variables, revoir la cible ou tester un autre horizon de temps.

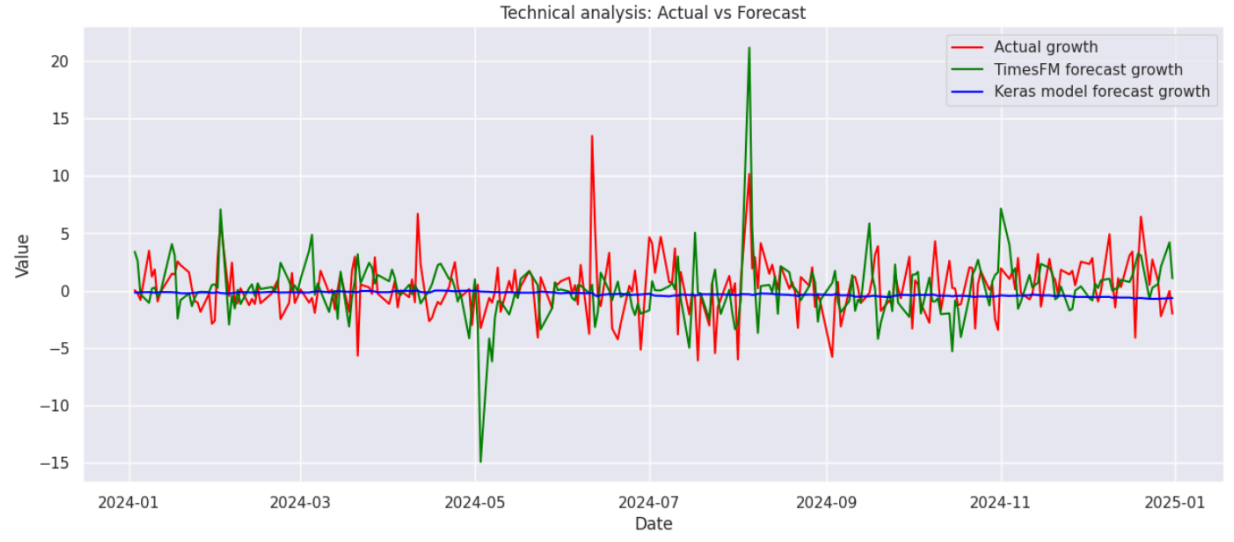
En l'état, **le modèle n'est pas encore exploitable** pour une stratégie d'investissement fiable.

2e et 3e modélisation : Keras et TimesFM

Le tracé de la valeur de l'action à la fermeture peut être trompeur. En effet, ce tracé peut **donner l'impression que la prédiction est fiable**. Cependant, un modèle extrêmement simple qui donnerait la valeur de fermeture de veille donnerait aussi cette impression, et il est évident qu'un tel modèle de prédiction est totalement inutile.



En traçant la croissance de l'action, on voit plus facilement la prédiction que fait le Modèle Keras : **juste une constante**.



Sur le modèle, TimesFM, nous essayons de voir si le modèle arrive au moins à prédire **le signe de la variation** (si l'action a augmenté de valeur ou diminué de valeur).

```
=== RAPPORT DÉTAILLÉ - Times FM ===
              precision    recall  f1-score   support

      Baisse       0.50      0.52      0.51       114
      Hausse       0.59      0.58      0.58       137

    accuracy                           0.55       251
   macro avg       0.55      0.55      0.55       251
weighted avg       0.55      0.55      0.55       251
```

Comme le modèle XGBoost, **le résultat n'est pas utile**. Avec environ 50% de prédiction correcte, le modèle ne fait pas mieux que le hasard.

Les différents modèles essayés n'ont pas de résultats satisfaisants.
Le jeu de données est **probablement insuffisant pour pouvoir profiter de la puissance du deep learning**.

Bien que TimesFM ait été entraîné avec des jeux de données temporelles très différents, il n'a pas été entraîné avec des données boursières.

Il serait intéressant de **ré-entraîner TimesFM avec un jeu de données boursières très conséquent** (au moins 10 ans de données, sur un très large panel d'actions et d'ETF).

Mise en Œuvre

Agent Analyste: "Stella", du Modèle à l'Application Interactive

Nous avons donc opéré une réorientation stratégique : notre modèle `RandomForestClassifier` n'est pas tant un outil pour **"choisir les gagnants"** qu'un puissant système de **"filtrage des perdants"** et sa fiabilité dans la **détection des actions à haut risque** constitue sa plus grande valeur.

Cependant, la puissance de ce modèle resterait **purement théorique** sans un moyen pour un **utilisateur non technique**, de l'interroger et d'en exploiter les résultats de manière simple et intuitive.

Pour combler ce fossé entre le modèle de Machine Learning et une application pratique, nous avons développé **Stella**, un agent conversationnel. L'objectif de **Stella** n'est pas de fournir des conseils d'investissement, mais de **rendre le modèle accessible à tous**, en traduisant une simple question en langage naturel en une chaîne complexe d'appels d'API, de traitement de données et d'inférence du modèle.

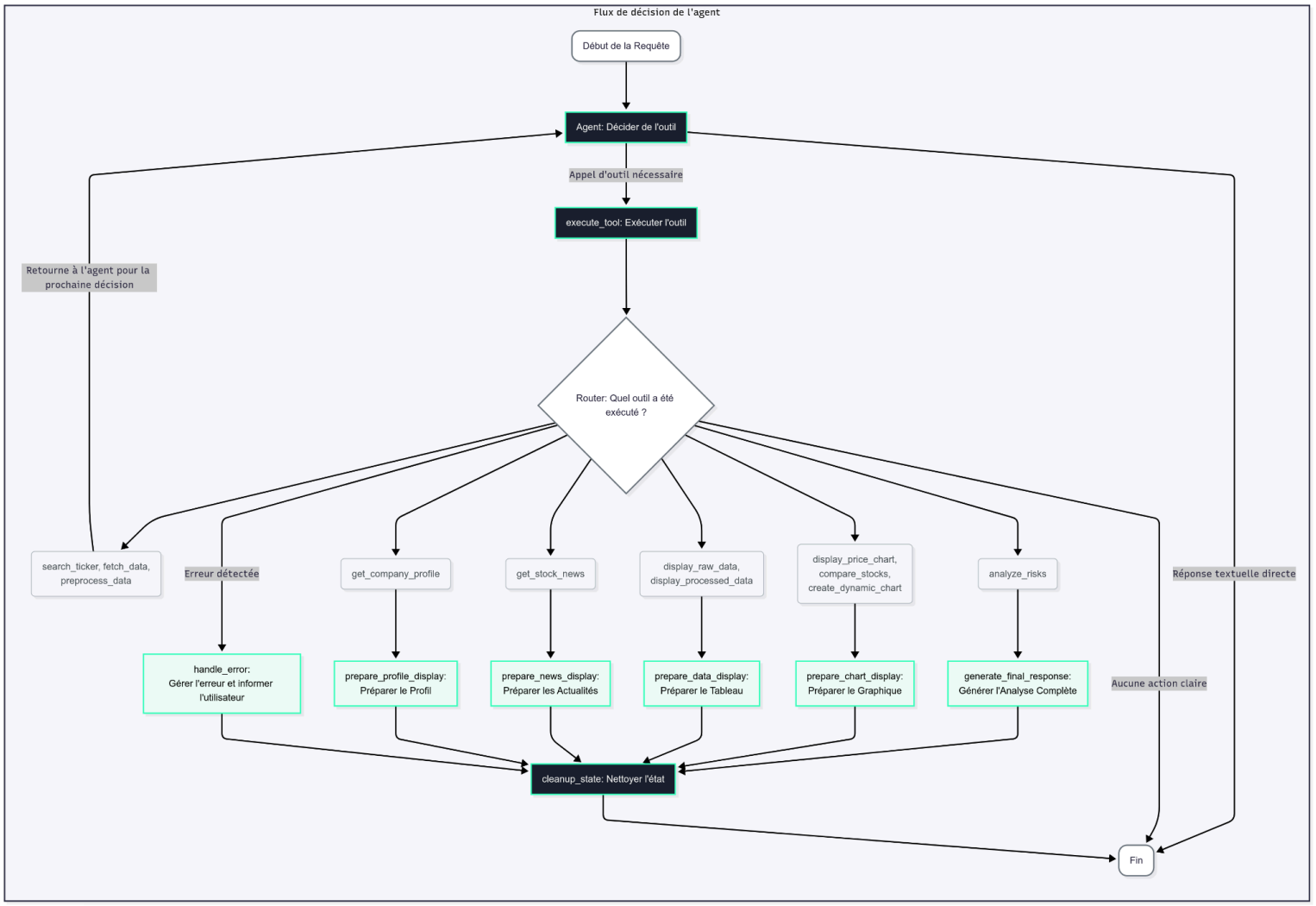
1. Architecture Technologique et Orchestration

Stella est construite sur une architecture moderne qui combine l'intelligence des grands modèles de langage (**LLM**) avec la robustesse d'un framework d'orchestration.

A. **Interface Utilisateur (UI) :** L'interface de chat est développée avec Streamlit (fichiers `app.py` et `pages/1_👩_Stella, analyste.py`). Ce choix permet de créer une **application web interactive** et réactive, capable d'afficher du texte, des tableaux de données (`st.dataframe`) et des graphiques interactifs (`st.plotly_chart`), ce qui est essentiel pour présenter les résultats financiers.

B. **Cerveau de l'Agent (Orchestration) :** Le cœur de **Stella** est un graphe d'états construit avec la bibliothèque `langgraph` (`agent.py`). Cette approche structurée permet de définir une logique de décision complexe. Le graphe définit des **"nœuds"** (des étapes comme `fetch_data`, `analyze_risks`) et des **"arêtes"** (les transitions logiques entre ces étapes). **Un routeur** (`router`) intelligent évalue l'état de la conversation après chaque étape pour décider de la suivante, assurant que l'agent suit une séquence logique et cohérente.

Graph de décision de l'agent Stella



C. **Modèle de Langage (LLM) :** Pour comprendre les requêtes de l'utilisateur et décider quel outil utiliser, **Stella** s'appuie sur un grand modèle de langage, spécifiquement `moonshotai/kimi-k2-instruct` via **l'API Groq**. Le comportement de l'agent est guidé par un prompt système détaillé (`system_prompt` dans `agent.py`) qui lui assigne son rôle, ses capacités, et ses règles de fonctionnement strictes.

D. **Boîte à Outils (Tools) :** **Stella** dispose d'un arsenal d'outils définis dans `tools.py`. Chaque outil est une fonction Python qui exécute une tâche spécifique (par exemple, `search_ticker`, `fetch_data`, `analyze_risks`). **Le LLM ne fait que décider quel outil appeler** ; la logique métier réelle est encapsulée dans les scripts du répertoire `src/`.

2. Le Flux d'Analyse Fondamentale : Le Cœur du Projet

La fonctionnalité principale de **Stella** est d'exécuter la **séquence d'analyse fondamentale** aboutissant à l'utilisation de notre modèle de classification. Lorsqu'un utilisateur demande une "analyse complète" d'une entreprise, **Stella** déclenche le **flux suivant**, orchestré par LangGraph :

1. **Recherche du Ticker (`search_ticker`) :** Si l'utilisateur fournit un nom ("Apple"), **Stella** recherche d'abord le ticker correspondant ("AAPL") via l'API FMP.
2. **Collecte des Données (`fetch_data`) :** Elle récupère les données fondamentales annuelles pour le ticker identifié.
3. **Prétraitement (`preprocess_data`) :** Les données brutes sont nettoyées, et les caractéristiques nécessaires (comme `marginProfit` ou `revenuePerShare_YoY_Growth`) sont calculées, conformément à la pipeline définie dans notre phase de modélisation.
4. **Analyse des Risques (`analyze_risks`) :** C'est l'étape cruciale. Le DataFrame prétraité est passé à la fonction `analyse_risks` qui charge notre modèle entraîné (`rf_fundamental_market_classifier.joblib`) et effectue une prédiction.
5. **Synthèse et Réponse (`generate_final_response_node`) :** En fonction du résultat du modèle (**"Risque Élevé Détecté"** ou **"Aucun Risque Extrême Détecté"**), **Stella** génère une réponse textuelle claire et nuancée. Elle y joint un graphique de synthèse qui met en perspective la croissance et la valorisation de l'entreprise, offrant un contexte visuel à la prédiction.

Ce flux garantit que **chaque analyse soit reproductible**, cohérente et s'appuie directement sur le modèle de classification que nous avons développé et validé.

3. Fonctionnalités Étendues pour une Analyse Complète

Au-delà de l'analyse de risque, **Stella** a été dotée d'outils supplémentaires pour permettre une exploration financière plus large :

6. **Informations sur l'Entreprise :** Les outils `get_company_profile` et `get_stock_news` permettent d'obtenir respectivement une description détaillée de l'entreprise (secteur, CEO, description) et les dernières actualités la concernant.
7. **Visualisation des Prix :** L'outil `display_price_chart` offre la possibilité de visualiser l'historique du cours d'une action sur différentes périodes (de 1 mois à 5 ans).
8. **Analyse Comparative (`compare_stocks`) :** Une des fonctionnalités les plus puissantes de **Stella** est sa capacité à comparer plusieurs entreprises. L'utilisateur peut demander de comparer des actions sur une métrique fondamentale (ex: "compare le ROE de Google et Apple") ou sur la performance de leur prix sur une période donnée.
9. **Exploration Interactive des Données :** Grâce à l'outil `create_dynamic_chart`, un utilisateur avancé peut demander à **Stella** de générer des graphiques personnalisés (barres, lignes) en spécifiant les axes, permettant une exploration libre et dynamique des données financières.
10. **Recherche Documentaire (RAG)** : L'outil `query_research` permet à Stella de répondre à des questions en s'appuyant sur ce document PDF. En analysant le contenu via une base vectorielle, Stella retrouve les passages pertinents et les intègre dans ses réponses, rendant accessible une information complexe sans lecture manuelle du document.

4. Transparence et Visualisation du Processus de Décision

Un aspect innovant du projet est la **transparence du fonctionnement interne de l'agent**. La page `1_🎬 Visualisation de l'agent.py` est connectée à LangSmith, la **plateforme de traçabilité de LangChain**. Après chaque conversation avec **Stella**, l'utilisateur peut lancer une animation qui visualise, étape par étape, le **chemin de décision exact que l'agent a suivi** dans le graphe LangGraph.

Cette fonctionnalité, générée par la fonction `generate_trace_animation_frames`, offre une vision claire de la "pensée" de l'agent : quel nœud a été activé, quelle

décision le routeur a prise, et quelle a été la transition suivante. Cela renforce la confiance de l'utilisateur et constitue un outil de débogage et d'analyse.

5. Conclusion

Stella n'est pas une simple interface graphique. C'est un **agent intelligent** et autonome qui **encapsule la complexité de notre modèle de prédiction** et de tout l'écosystème de données qui l'entoure. Il réussit à traduire la stratégie de "filtrage des perdants", identifiée comme la plus pertinente pour notre modèle, en un outil interactif et accessible, réalisant ainsi l'objectif final de notre projet : **passer de la recherche académique à une solution pratique de gestion des risques**.

Difficultés rencontrées

Le parcours de ce projet, bien que fructueux, a été jalonné de défis.

Le principal **verrou scientifique** rencontré a résidé dans la **définition d'une variable cible temporellement cohérente et pertinente sur le long terme**. L'hypothèse initiale de comparer les entreprises à un benchmark calendaire générique s'est heurtée à la réalité des années fiscales décalées. La conception et la mise en œuvre de la **matrice de rendement dynamique** ont constitué une tâche complexe mais indispensable. Cependant, cette solution reste contrainte par une difficulté plus profonde liée aux **jeux de données**. L'accès gratuit à seulement **cinq ans de données** via l'API de Financial Modeling Prep a constitué une limitation majeure. Une véritable approche d'investissement fondamental se juge sur des horizons de 3, 5, voire 10 ans. Notre incapacité à construire une variable cible mesurant la performance 5 ans après la publication des données nous a contraints à une approche annuelle, **plus sensible au "bruit" du marché.**

Au-delà des données, un verrou méthodologique majeur résidait dans l'interprétation des décisions du modèle. Comprendre *pourquoi* il classifie une entreprise comme "à risque" était essentiel pour valider sa pertinence. C'est ici que l'acquisition de **compétences techniques** sur des outils d'interprétabilité avancés comme SHAP s'est révélée non pas comme une compétence additionnelle, mais comme le **véritable routeur du projet**. L'analyse approfondie des valeurs de Shapley, notamment sur les erreurs du modèle, a été le déclencheur de notre réorientation stratégique. C'est cette analyse qui a mis en lumière que les signaux de détérioration fondamentale étaient des précurseurs beaucoup plus fiables de la sous-performance que les signaux positifs ne l'étaient de la surperformance. Cette découverte, **issue directement de la résolution d'un défi d'interprétabilité**, a confirmé que la **pertinence** de l'approche finale a été trouvée en surmontant une difficulté technique et en pivotant d'une stratégie de prédiction vers une stratégie de filtrage.

Bilan

Le bilan de ce projet dépasse la livraison d'un modèle fonctionnel ; il aboutit à une conclusion technique : si la prédiction de la surperformance boursière reste une quête incertaine, **la détection fiable de la sous-performance à partir de données fondamentales est, elle, une problématique résolue par notre approche.** La valeur ajoutée de notre travail ne réside donc pas dans de la génération d'alpha, mais dans un mécanisme de **préservation du capital**.

Le modèle final, un `RandomForestClassifier` optimisé via `RandomizedSearchCV`, représente l'aboutissement de notre processus de modélisation. Cette conclusion stratégique est directement étayée par ses performances : alors que le modèle global atteint une **précision de 68%**, il excelle là où il est le plus utile. Sur les prédictions de sous-performance (classe 0) effectuées avec un haut niveau de confiance (>0.7), **sa précision atteint 92%**. Ce résultat valide son efficacité comme outil de filtrage des risques sur les entreprises américaines. Les quatre objectifs finaux du projet ont été atteints :

1. **Ingénierie d'une Variable Cible Robuste** : Atteint grâce à la matrice de rendement dynamique.
2. **Construction d'un Modèle Axé sur le Risque** : Atteint, le modèle étant spécifiquement performant pour identifier les actions à risque.
3. **Déploiement de l'Agent "Stella"** : Pleinement réalisé avec l'application Streamlit fonctionnelle.
4. **Enrichissement de l'Agent** : Atteint en dotant **Stella** d'une panoplie d'outils d'analyse complémentaires.

Tandis que l'objectif initial de génération d'alpha n'a pu être atteint, compte tenu de la complexité de la tâche. Le modèle peut ainsi s'inscrire dans le **processus métier d'un gestionnaire de portefeuille ou d'un analyste financier**. Il ne se substitue pas à la décision humaine mais agit comme un **système d'aide à la décision**.

Concrètement, il peut être utilisé pour :

- **Auditer un portefeuille existant** : Stella peut lever des "drapeaux rouges" sur les positions les plus à risque.
- **Filtrer une liste de surveillance (watchlist)** : Avant d'initier une position, un analyste peut écarter les candidats présentant des fondamentaux trop fragiles.

Suite du projet

Pour augmenter les performances et la portée du projet, plusieurs pistes d'amélioration peuvent être explorées :

1. **Enrichissement des Données** : L'amélioration la plus impactante serait d'obtenir un **historique de données fondamentales plus long** (10-15 ans). Cela permettrait de valider des stratégies d'investissement à plus long terme et de couvrir différents cycles économiques. L'ajout de données **macroéconomiques** (taux d'intérêt, inflation) ou de **données de sentiment** pourrait également apporter un signal prédictif supplémentaire.
2. **Élargissement du Périmètre d'Analyse** : Appliquer **la même méthodologie à d'autres indices** (S&P 500, EURO STOXX 50) permettrait de valider la robustesse de l'approche sur différents marchés.
3. **Création d'un ETF "filtré"** : L'approche fondamentale pourrait être utilisée pour créer un **ETF** (Exchange Traded Fund - produit financier reproduisant la performance d'un indice ou groupe de positions) **dynamique,** centré sur un indice et amélioré par le **"filtrage des perdants"**, éliminant une part des positions induisant de la sous-performance.

Ce projet apporte donc une contribution méthodologique. Il démontre une application de bout en bout du **Machine Learning à un problème financier**, en insistant sur la primauté de l'ingénierie de la variable cible. Surtout, il illustre comment des frameworks modernes (LangGraph), centrés autour de LLMs, peuvent **rendre accessible à des utilisateurs non-techniques** l'utilisation de modèles prédictifs et outils d'analyse financières, et ce, **en interagissant avec des agents IA par le langage naturel.**

Contributeurs au projet et séparation des tâches

- **Mathis GENTHON** : Recherche Fondamentale - Application Stella
- **Jerry PETILAIRE** : Recherche Fondamentale
- **Gilles LENY** : Recherche Technique
- **Samuel LEE KWET SUN** : Recherche Technique

Annexes et Ressources

- **Tableau des variables**
  https://docs.google.com/spreadsheets/d/1b92L9_Bpgajoo9ovzr07eJ7KXd8yfdyw/edit?gid=1488335766#gid=1488335766
- **Repository Github**

  https://github.com/DataScientest-Studio/nov24_cds_opa