BBA EDHEC,

Programme de l'EDHEC

Accrédité EQUIS et AACSB

Reconnu par l'Etat à diplôme visé.

Mémoire de fin d'étude

4ème année

Promotion 2021

Fama & French : à la recherche de facteurs plus significatifs

Directeur de mémoire : DU JARDIN Philippe

Nom de l'étudiant : PAULUS Carl

Table des matières

Remerciements

Avant de rentrer dans le cœur du sujet j'aimerais remercier les personnes qui m'ont accompagné au cours de cette longue période de recherche académique. Ce mémoire a été une source d'apprentissage colossale où j'ai pu repousser mes limites à de multiples reprises.

En premier lieu, je tiens à remercier Monsieur Philippe Du Jardin pour m'avoir soutenu autant concernant la mise au point du sujet précis que des techniques de recherches et de l'élaboration du mémoire.

Monsieur Hugo Inzirillo, pour m'avoir guidé et inspiré lorsque que j'arrivais à bout de mes capacités. Ses conseils précieux distillés stratégiquement pour me faire progresser m'ont permis de me débloquer de certaines situations et ainsi d'aboutir sur un sujet sensiblement plus complet et intéressant. En effet, j'ai pu arriver au bout de mes compétences informatiques par moment et ce sujet m'a permis de me dépasser et de progresser considérablement.

L'équipe de trading de produits dérivés actions chez UniCredit Munich, pour m'avoir guidé dans la compréhension de modèles financiers divers et grandement utiles pour le développement et la rédaction de ce mémoire.

Le département Global Markets de BNP Paribas Londres où j'ai réalisé un stage dans l'équipe de structuration de stratégies quantitatives d'investissements (QIS) de Yannick Daniel, aux côtés de Andrea Bachian et Marouane Bouslama, pour m'avoir permis l'acquisition de compétences de développement informatique et de mathématique nécessaires.

Mes professeurs de la Frankfurt School of Finance & Management où je réalise mon Master pour m'avoir engagé dans des projets dont je ne me pensais pas capable et dont les enseignements m'ont nourri pour avancer sur le mémoire.

Pour finir, je souhaite remercier tous ceux qui m'ont permis de repousser mes limites car le sujet était ambitieux. Mais je suis aujourd'hui honoré de pouvoir vous présenter ce mémoire qui a été une source d'apprentissage immense en mathématique, statistique et informatique.

Introduction

Contexte et histoire

Rétrospectivement, on se rend compte que le risque n'a pas été souvent pris en compte jusque dans les années 1960, que ce soit en termes de théorie ou de preuves empiriques. Les marchés d'actions et d'options existent au moins depuis 1602, date à laquelle les actions de la Compagnie des Indes orientales ont commencé à être négociées à Amsterdam ; et les marchés d'assurance organisés étaient déjà bien développés au XVIIIe siècle. En 1960, les entreprises d'assurance s'appuyaient depuis des siècles sur la diversification pour répartir les risques. Mais malgré la longue histoire de la prise et du partage des risques sur les marchés financiers, le modèle d'évaluation des actifs financiers (MEDAF) a été élaboré à une époque où les fondements théoriques de la prise de décision dans l'incertitude étaient relativement nouveaux. Les théories rigoureuses des préférences des investisseurs en matière de risque et de prise de décision dans l'incertitude n'ont émergé que dans les années 1940 et 1950, notamment dans les travaux de John Von Neumann, Oskar Morgenstern et de Leonard Savage. La théorie du portefeuille, qui montre comment les investisseurs peuvent créer des portefeuilles d'investissements individuels pour arbitrer de manière optimale entre le risque et le rendement, n'a été développée qu'au début des années 1950 par Harry Markowitz.

Cela fait donc qu'une soixantaine d'années que beaucoup d'économistes se penche sur l'évaluation des actifs financiers et l'estimation de leurs rendements. Le premier modèle développé pour résoudre ces énigmes est le modèle à facteur unique appelé MEDAF. Aussi appelé CAPM (Capital Asset Pricing Model) en anglais. Il a été mis en place par Jack Treynor, William Sharpe, John Lintner et Jan Mossin au début des années soixante. Ceux-ci se sont appuyés sur les travaux antérieurs d'Harry Markowitz concernant la diversification et la théorie moderne de portefeuille. Par la suite est arrivé le modèle à trois facteurs proposé par Eugène Fama et Kenneth French en 1992, puis à cinq facteur en 2014.

Le MEDAF

Le modèle d'évaluation des actifs financiers (MEDAF) est un modèle économique qui explique le rendement des actions en fonction du rendement du marché. Les hypothèses derrière le modèle développé par Sharpe et Lintner sont les suivantes :

- C'est un modèle statique (une période)
- L'offre d'actifs est fixe
- L'offre d'actifs sans risque nette est nulle (emprunter et prêter au même taux r)
- Les rendements suivent une distribution normale.
- Les attentes sont homogènes concernant l'ensemble des opportunités d'investissement
- Les marchés financiers sont des marchés concurrentiels
- Il n'y a pas de coûts de transaction (taxes, frais, etc.)

A partir de ces hypothèses, on peut comprendre qu'il s'agira de l'équation suivante :

$$E(R_{actif}) = R_F + \beta_{actif} \cdot [E(R_M) - R_F]$$

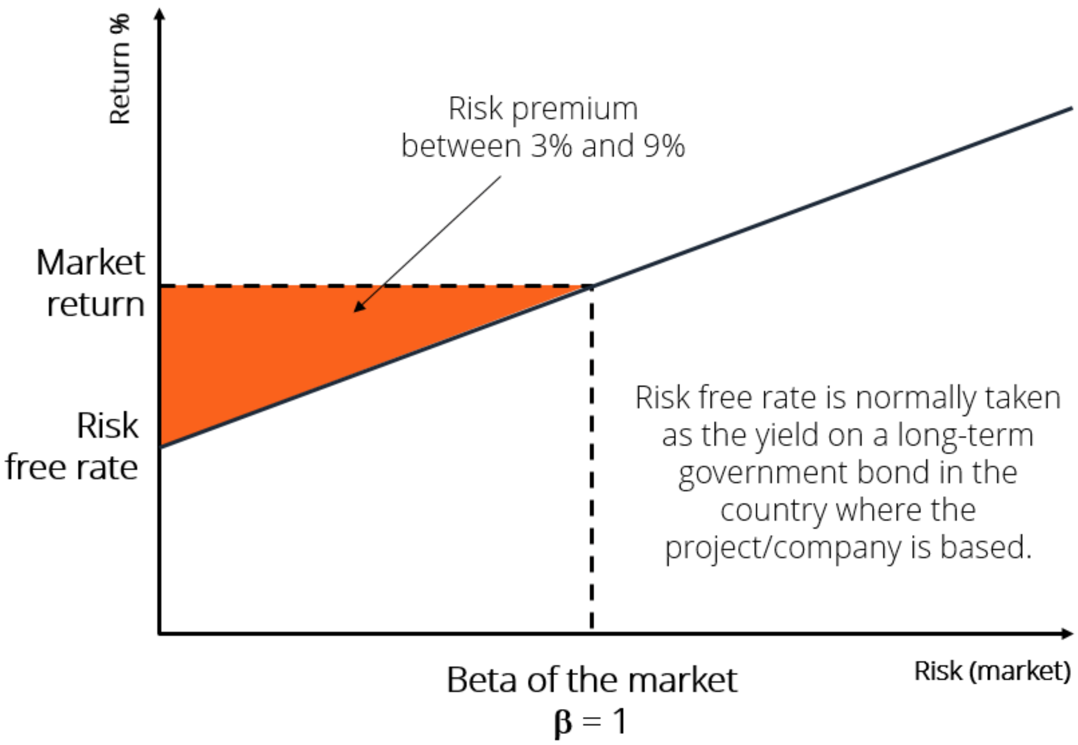
- $E(R_{actif})$ est l'espérance des rendements de l'actif.
- R_F est l'hypothèse d'emprunt et de prêt sans risque
- β_{actif} est le bêta de marché de l'actif. C'est la covariance de son rendement avec le rendement du marché divisé par la variance du rendement du marché :
$$\beta_{actif} = \frac{cov(R_M, R_{actif})}{var(R_M)}$$
- $E(R_M)$ est le rendement attendu des actifs qui ont des bêtas de marché égaux à zéro
- Le dernier terme $[E(R_M) - R_F]$ représente la prime de risque de marché

Le développement du CAPM est intimement lié à ce que l'on appelle « la théorie moderne de portefeuille » de Harry Markowitz et de sa frontière efficiente théorisé en 1952. Les hypothèses précédemment citées impliquent en effet que le portefeuille de marché M se situe sur la frontière de variance minimale pour que le marché des actifs s'équilibre.

Représentation du MEDAF

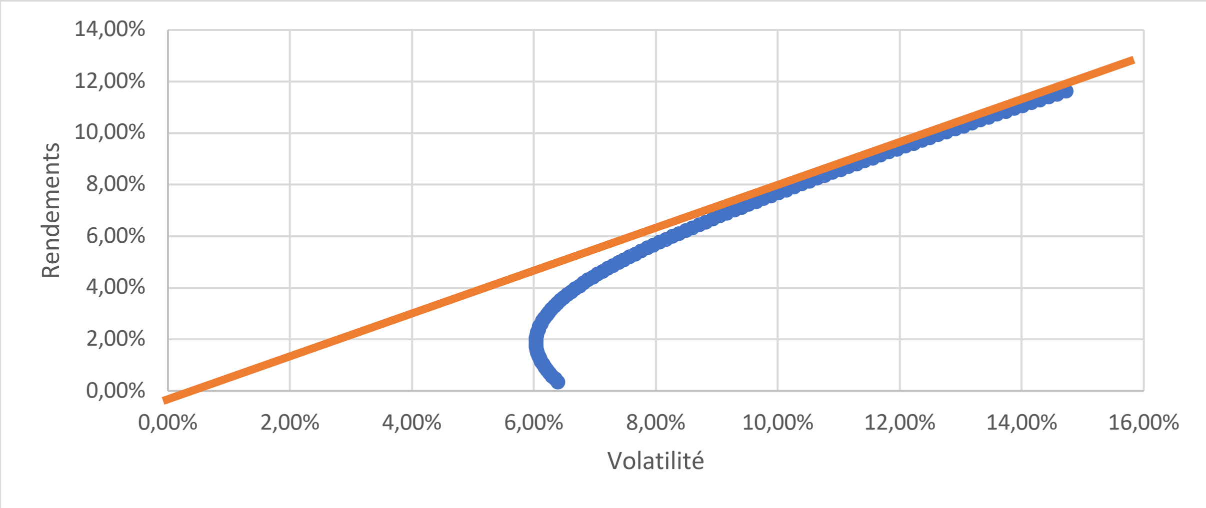
Les deux méthodes les plus communes pour représenter le MEDAF sont la Capital Market Line (CML) et la frontière efficiente. Ci-dessous la Capital Market Line où l'on distingue le surplus de rendement dû à la prise de risque (« risk premium ») par rapport au rendement d'un investissement non risqué (dit « risk-free »).

Figure 1 : Schématisation de la CML pour le MEDAF, source : Investopedia



Dans le graphique ci-dessous, la ligne orange est la Capital Market Line et la courbe bleue est la frontière efficiente. Dans ce cas, on voit que le portefeuille minimum variance dit « le plus efficient » selon le couple rentabilité-risque, obtient 2% de rendements pour une volatilité de 6%.

Figure 2 : Frontière efficiente



Limites du MEDAF

De nombreuses contradictions du MEDAF ont été démontrées au fil du temps. Suddhasatwa Basu démontre en 1977 puis 1983 que le MEDAF est un échec empirique. Il montre que les actions avec des ratios bénéfices/prix élevés ont obtenu des rendements significativement plus élevés que les actions avec des ratios bénéfices/prix faibles. Les études de Basu sont confirmées par Jeffrey Jaffe, Donald Keim et Randolph Westerfield. L'existence de cet effet met en échec le MEDAF : selon ce dernier, son bêta devrait être tout ce qui importe, or ce n'est pas le cas.

Nombreux sont ceux qui ont proposé des modèles alternatifs pour améliorer le MEDAF. Robert Merton développe en 1973 le modèle inter temporel d'évaluation des actifs financiers (ICAPM) pour capturer l'aspect multi-période de l'équilibre des marchés financiers. D'une manière différente, Stephen Ross propose en 1976 la théorie de l'arbitrage (Arbitrage Pricing Theory). Enfin, en 1979, Douglas Breeden propose le modèle basé sur la consommation. Mais la principale alternative au MEDAF reste le modèle de Fama et French, qui inclus des facteurs tailles (SMB) et book-to-market (HML), en plus de l'indice de marché $E(R_m - R_f)$, comme variables explicatives.

Le modèle d'évaluation à trois facteurs de Fama et French a été développé en réponse aux faibles performances du MEDAF pour expliquer les rendements réalisés. Fama et French soutiennent que les anomalies relatives au MEDAF sont capturées par le modèle à trois facteurs.

Fama & French

Eugène Fama et Kenneth French ont tout d'abord proposé deux nouvelles variables en 1993 après avoir observé que deux catégories d'actions avaient tendance à faire mieux que leur marché. Il s’agit des petites capitalisations, et des actions avec un ratio $Valeur\ Comptable/Valeur\ de\ march$é élevé.

$$R_{it} - R_{ft} = \alpha_{it} + \beta_1(R_{Mt} - R_{ft}) + \beta_2 SMB_t + \beta_3 HML_t + \epsilon_{it}$$

En ajoutant ces deux variables, Fama & French ont pu, avec leur nouveau modèle, expliquer 90% de la performance d'un portefeuille, contre 70% avec le modèle CAPM. Ces chiffres ont été obtenus après la réalisation de tests empiriques dans leur étude de 1993.

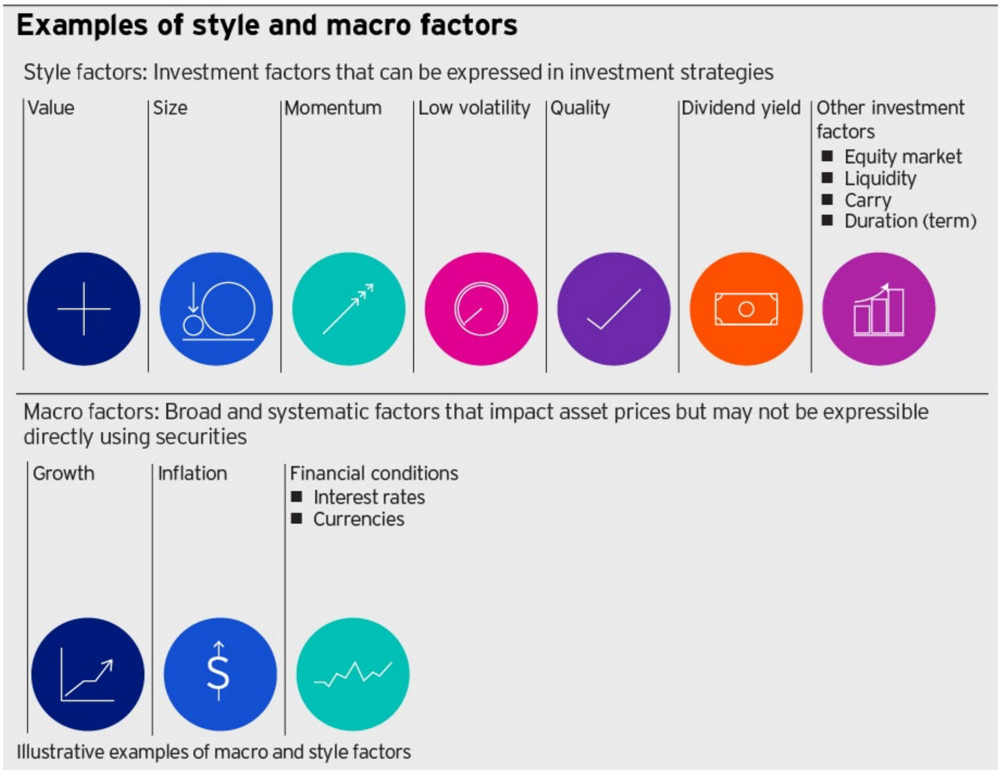
En 2015, Fama et French ont décidé d'étendre leur modèle en ajoutant deux nouveaux facteurs. Un facteur dit « momentum » nommé Robust Minus Weak (RMW) et le facteur d'investissement utilisé sous l'acronyme CMA (Conservative Minus Agressive).

$$R_{it} - R_{Ft} = a_i + b_i(R_{Mt} - R_{Ft}) + s_i SMB_t + h_i HML_t + r_i RMW_t + c_i CMA_t + e_{it}$$

Investissement factoriel

L’investissement factoriel s’est peu à peu démocratisé. Les investisseurs considèrent de plus en plus l'investissement factoriel comme un outil précieux pour atteindre leurs objectifs d'investissement. Aujourd'hui, près de la moitié des investisseurs institutionnels utilisent des stratégies factorielles pour au moins une partie de leurs portefeuilles. Les universitaires ont pu identifier plus de 600 facteurs susceptibles d'influencer les rendements et le risque, ce qui a incité les experts à parler de « factor zoo ».

Figure 3 : Classification des principaux facteurs, source : Invesco



Objectif de l'étude

L'objectif de cette étude va être de construire de nouveaux facteurs et de déterminer s'ils sont de bons indicateurs pour les gestionnaires de portefeuilles. Comme nous allons nous en rendre compte dans l'étude, les trois premiers facteurs de Fama & French sont très puissants et il n'y donc pas lieu de les remplacer. Nous allons développer de nouveaux modèles en remplaçant les deux derniers facteurs (RMW et CMA) par ceux que nous aurons créés. Nous souhaitons évaluer si ces nouveaux modèles permettent d'expliquer une plus grande part des rendements excédentaires d'un portefeuille.

Dans l'équation de régression détaillé plus haut, le rendement excédentaire est représenté par $(R_{it} - R_{ft})$. Ce sera notre variable dépendante. Une approche de régression des séries temporelles (valeurs quotidiennes des facteurs) est employée en utilisant la méthode des moindres carrés ordinaires (« Ordinary Least Squares »). Nous allons régresser ce rendement excédentaire des portefeuilles par rapport aux variables indépendantes que seront les cinq facteurs. Nous comparerons ensuite les coefficients de chacun des facteurs ainsi que leurs significativités. Nous pourrons enfin déterminer lesquelles sont les plus intéressant pour un gestionnaire de portefeuille.

Dans un premier temps, nous verrons les détails du fonctionnement du modèle original de Fama et French. S'en suivra une analyse de la littérature sur le sujet des modèles à facteurs, incluant le MEDAF et les modèles dérivés de Fama & French. Dans un second temps, les données et les méthodologies de création de portefeuilles et de nouveaux facteurs seront présentées. La dernière partie présentera et analysera les résultats statistiques et empiriques de la recherche.

Le modèle de Fama & French

I- Présentation du modèle

Une question fondamentale en finance est de savoir comment le risque d'un investissement doit affecter son rendement attendu. Le modèle d'évaluation des actifs financiers (CAPM) a fourni le premier cadre cohérent pour répondre à cette question. L'idée de ce modèle repose sur un seul facteur de risque, à savoir le rendement excédentaire du portefeuille sur le marché. Dans ce modèle, la covariance du rendement du portefeuille avec le rendement du marché joue un rôle important pour expliquer les variations du rendement excédentaire du portefeuille.

Comme exprimé précédemment, la version à trois facteurs de Fama et French a été développé en réponse aux faibles performances du MEDAF pour expliquer les rendements réalisés. Eugène Fama et Kenneth French démontrent dans leur étude empirique de 1992 que la covariance du rendement du portefeuille et du rendement du marché n'explique pas les variations du rendement excédentaire du portefeuille.

Le modèle adopte deux facteurs de risque supplémentaires au MEDAF afin de remédier aux anomalies existantes. Fama et French fondent donc leur modèle sur le fait que les rendements excédentaires moyens des portefeuilles sont sensibles à trois facteurs. Le premier est le rendement excédentaire du portefeuille du marché, noté $(R_m - R_f)$. Le second est la différence entre le rendement excédentaire d'un portefeuille de petites actions et de grandes actions, noté SMB (Small minus Big). Le troisième est la différence entre le rendement excédentaire d'un portefeuille d'actions à fort ratio Valeur Comptable / Valeur de marché (VC/VM) et faible ratio VC/VM, noté HML (High minus Low).

Ils formulent leur modèle comme suit :

$$R_{it} - R_{ft} = \alpha_{it} + \beta_1(R_{Mt} - R_{ft}) + \beta_2 SMB_t + \beta_3 HML_t + \epsilon_{it}$$

- R_{it} : rendement totale d'un portefeuille à t.
- R_{ft} : taux de rendement sans risque à t.
- R_{Mt}: rendement total du marché de référence à t.
- $(R_{it} - R_{ft})$: rendements excédentaires du portefeuille par rapport au taux sans risque.
- $(R_{mt} - R_{ft})$: rendements excédentaires du marché par rapport au taux sans risque.
- SMB_t : aussi appelé « Small minus Big » ou « size premium », c'est l'espérance de la différence entre le rendement excédentaire d'un portefeuille de grandes capitalisations par rapport à un de petites capitalisations.
- HML_t : aussi appelé « High minus Low » ou « value premium », c'est l'espérance de la différence entre le rendement excédentaire d'un portefeuille de ratio VC/VM élevé par rapport à un portefeuille de ratio VC/VM faible.
- $\beta_{1,2,3}$: ce sont les coefficients des facteurs.
- L'alpha α_{it} représente l'excédent de rendement non expliqué par les facteurs.

L'alpha de Jensen est développé en 1968 par Michael C. Jensen. L'alpha évalue la performance d'un actif financier. A l'intérieur d'un modèle, il peut mesurer la surperformance ou la sous-performance d'un portefeuille par rapport à sa performance théorique dans le modèle.

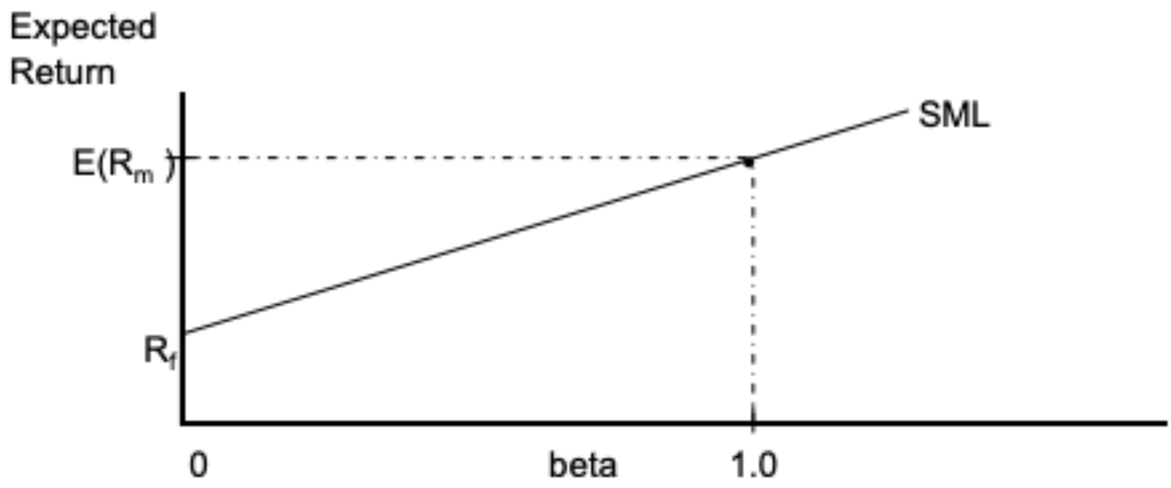
II- Revue Littéraire

Le plus connu des modèles d'évaluation, et donc le plus fourni en matière de recherche, est bien sûr le MEDAF. C'est celui à partir duquel beaucoup ont commencé à réfléchir pour développer d'autres facteurs. Les chercheurs se sont inspirés de la logique factorielle pour étudier les performances de portefeuilles en tout genre.

CAPM

Le modèle d'évaluation des actifs financiers tente de quantifier la relation entre le bêta d'un actif et le rendement attendu correspondant. Le modèle CAPM repose sur un certain nombre d'hypothèses, dont les plus pertinentes concernent le comportement des investisseurs et la présence d'un seul facteur de risque commun. Treynor, Sharpe et Lintner démontre essentiellement qu'un actif est censé rapporter le taux sans risque plus un premium (évoqué dans l'introduction) provenant de la prise de risque, et mesurée par le bêta de cet actif. Le graphique ci-dessous illustre cette relation entre le bêta et le rendement attendu. Cette ligne est appelée « Security Market Line ».

Figure 4 : Security Market Line



Bien que le MEDAF soit un outil extrêmement utile, l'efficacité globale du modèle suscite des inquiétudes. Plusieurs critiques importantes ont été formulées par la recherche universitaire au cours des dernières années. La principale critique étant qu'une bonne partie du rendement du portefeuille analysé, environ 30%, reste inexpliqué. De nombreux chercheurs estiment que d'autres facteurs de risque ont un impact significatif sur les rendements attendus sur le marché. Par conséquent, la simplicité de l'hypothèse du MEDAF d'un facteur de risque unique expliquant les rendements attendus a été remise en question.

En plus des démonstrations de Basu, Jaffe, Keim et Westerfield, dont nous avons parlé dans l'introduction, Werner De Bondt et Richard Thaler renforcent l'idée que d'autres facteurs pour expliquer le rendement excédentaire d'un portefeuille existent. Ils trouvent en 1985 que les actions qui ont eu de mauvais rendements au cours des trois à cinq dernières années ont des rendements moyens beaucoup plus élevés (que ceux ayant bien performé lors de cette même période) au cours des trois à cinq années suivantes.

Fama & French :

En 1993, Fama et French démontrent que les portefeuilles construits pour imiter les facteurs de risque liés au marché (donc SMB et HML) ont des effets importants sur les rendements des actions. Pour expliquer le facteur SMB, ils expliquent les différences entre les rendements du New York Stock Exchange (NYSE) et de la National Association of Security Dealers (NASD). Les actions sur le NYSE ont des rendements moyens plus élevés que les actions de taille similaire sur le NASD pendant la période de test. Leur analyse démontre que la raison de cette variation est la différence entre le risque des actions, qui est capturé par le modèle à trois facteurs. Page 37 est détaillé la logique du facteur HML. Il est expliqué que les actions à forte sensibilité ont tendance à être des entreprises dont les bénéfices sont constamment faibles, ce qui entraîne un faible cours des actions et donc des ratios de VC/VM élevés. Ces entreprises sont connues sous le nom de « Value ». Les actions peu sensibles à ce facteur de risque ont tendance à avoir des bénéfices constamment élevés, ce qui conduit à un faible ratio VC/VM. Ces valeurs sont dites « Growth ». Les auteurs concluent que le ratio VC/VM (Book Value / Market Value en anglais) est le facteur de risque le plus important qui explique la différence de rendement entre les actions du NYSE et celles du NASD.

En 1995, Fama et French ont testé si les variations des prix des actions, en fonction de la taille et du VC/VM, reflètent les variations des bénéfices. Ils montrent page 131 de leur étude que, conformément à la fixation rationnelle des prix, un VC/VM élevé signale la persistance de mauvais bénéfices et un VC/VM faible signale des bénéfices élevés. Le test de leur modèle a été effectué sur les marchés boursiers NYSE, AMEX et NASDAQ. Ils constatent que les facteurs marché et taille des bénéfices expliquent le niveau des rendements, mais ils n'ont trouvé aucune relation entre les facteurs VC/VM des bénéfices et les rendements.

Dans cette même étude, Fama et French vont plus loin. Ils affirment que les anomalies du CAPM disparaissent largement en utilisant le modèle à trois facteurs. Ce dernier peut expliquer les fortes tendances (« patterns ») des rendements observés lorsque les portefeuilles sont formés en fonction du rapport bénéfice / prix (Earnings / Price), du rapport flux de trésorerie/prix (Cash-Flow / Price) et de la croissance des ventes.

Modèle de Carhart

Le modèle de Carhart inclut un autre facteur de type « momentum » qui étend le modèle de Fama & French à trois facteurs. Mark Carhart présente en 1997 un modèle basé sur la performance des fonds communs de placement avec trois groupes d'actions et de portefeuilles ayant des rendements faibles, moyens et élevés. D'après les résultats, quatre facteurs ont permis d'expliquer la variation des rendements moyens des actions. Les quatre facteurs du modèle sont les suivants : le facteur de risque de marché (RMRF) dit « market premium », le facteur SMB, le facteur HML et enfin le WML (Winner minus Losers). Ce dernier évalue la différence de rendements entre les portefeuilles contenant des actions qui ont bien performé sur les marchés lors de la période précédente par rapport aux portefeuilles contenant des actions ayant peu ou pas performé lors de cette même période.

Études extérieures sur les modèles Fama French

En 1991, K. C. Chan et Nai-Fun Chen affirment que les petites et les grandes entreprises ont des caractéristiques de risque et de rendement différentes. Les petites entreprises de la Bourse de New York sont des entreprises qui n'ont pas eu de bons résultats, qui sont gérées de manière moins efficace et qui sont fortement endettées. Par conséquent, les petites entreprises ont tendance à être plus risquées que les grandes entreprises et ce risque n'est pas pris en compte par l'indice du marché (« market index »). Après avoir introduit de multiples expositions au risque dans l'indice de marché, un indice de levier (« leverage index ») et un indice de baisse des dividendes (« dividend-decrease index ») pour imiter les entreprises marginales, l'effet de taille perd son pouvoir explicatif. Les expositions au risque de ces indices sont aussi puissantes que la taille pour expliquer les rendements moyens des portefeuilles classés par taille.

Cependant, S. P. Kothari et Craig MacKinlay soutiennent en 1995 qu'une grande partie de la prime est due au "biais du survivant" et à la fouille des données (« data snooping »). La source de données pour les capitaux propres comptables contient un nombre disproportionné d'entreprises à haut niveau technicité qui survivent à la crise, de sorte que leur rendement moyen soit surestimé. L'hypothèse de « data snooping » des données suppose que les chercheurs qui se fixent pour objectif de rechercher des variables liées au rendement moyen, trouveront des variables, mais uniquement dans l'échantillon utilisé pour les identifier. Cependant, un certain nombre d'articles ont affaibli et même rejeté cette hypothèse, en plus de celle du biais de survie. Par exemple, Josef Lakonishok trouve en 1994 une forte relation positive entre le rendement moyen et le VC/VM pour les 20% plus grosses entreprises du NYSE-Amex, où le biais du survivant n'est pas un problème. De même, dans l'étude de Fama French de 1993, tout biais de survivant dans ces portefeuilles est insignifiant. En effet, les portefeuilles de valeur (« value-weight portfolios ») donnent plus de poids aux grandes valeurs et les chercheurs constatent que la relation entre le ratio VC/VM et le rendement moyen y est forte.

En 1998, Fama et French fournissent des preuves hors échantillon supplémentaires. Ils ont testé le modèle à trois facteurs sur treize marchés différents au cours de la période 1975-1995. Ils constatent que 12 des 13 marchés enregistrent une prime d'au moins 7,68 % par an pour les actions Value (VC/VM élevé). Sept marchés présentent des bêtas HML statistiquement significatifs. En 2002, Neal Maroney et Aris Protopapadakis ont testé le modèle FF à trois facteurs sur les marchés boursiers d'Australie, du Canada, d'Allemagne, de France, du Japon, du Royaume-Uni et des États-Unis. L'effet de taille et la prime de Value survivent pour tous les pays examinés. Ils concluent que les effets de taille et de VC/VM ont un caractère international.

En 2004, Clive Gaunt étudie le modèle à trois facteurs dans le contexte australien. L'étude couvre la période 1991 à 2000 des entreprises cotées à la bourse australienne. Il constate que le risque bêta tend à être plus important pour les petites entreprises et celles dont les ratios VC/VM sont plus faibles. Contrairement à l'étude de Fama & French, les bêtas sont ici en moyenne significativement inférieurs à 1. On observe également une augmentation monotone du coefficient du facteur SMB, lorsque l'on passe des plus grands aux plus petits portefeuilles. Ils trouvent des « intercepts » importants et positifs pour les petits portefeuilles. La variation expliquée, mesurée par les R^2 ajusté, est également beaucoup plus élevée que celle du MEDAF. L'auteur conclut que le modèle à trois facteurs fournit une meilleure explication des rendements observés des actions australiennes que le MEDAF.

Michael Drew et Madhu Veeraraghavan présentent en 2002 des preuves comme quoi le premium (excédent de rendement venant de la prise de risque) est expliqué par la taille et le ratio book-to-market (VC/VM) dans le cas de la Malaisie. Ils rapportent que les facteurs identifiés par Fama & French expliquent la variation des rendements boursiers en Malaisie et ne sont pas spécifiques à un échantillon. L'analyse a été limitée aux entreprises dont les données de rendement sont disponibles de décembre 1992 à décembre 1999. Les résultats montrent que les actions de petite taille et celles dont la valeur comptable par rapport au marché est élevée génèrent des rendements plus élevés que les actions de grande taille et celles dont la valeur comptable par rapport au marché est faible en Malaisie. La prime de taille et la prime de valeur génèrent des rendements annuels moyens de 17,70% et 17,69% par an respectivement. Le rendement annuel moyen généré par le marché n'est que de 1,92%. Les rendements de SMB et HML sont nettement supérieurs à ceux du marché. Leurs résultats montrent également que le pouvoir explicatif des variables est puissant tout au long de la période de l'échantillon et pas seulement en janvier. Ils rejettent donc la présence de l'effet du début de l'année.

Enfin, Belen Blanco obtient en 2012, elle aussi, des preuves empiriques en faveur du modèle à trois facteurs de Fama et French, par rapport au CAPM. Elle trouve des preuves de la manière dont les caractéristiques liées à la taille et au ratio BV/MV expliquent les rendements des actifs. Elle constate également que les résultats varient en fonction de la manière dont les portefeuilles sont formés.

Extensions de modèles à facteurs

Nous avons donc compris jusqu'ici que les entreprises sont confrontées à de nombreux facteurs de risque : le risque de marché, le risque de faillite, le risque de change, le risque lié aux fournisseurs etc. Étant donné que le MEDAF utilise un seul facteur pour décrire le risque global, il semble logique qu'un modèle comprenant plus de sous-facteurs puisse fournir un modèle plus descriptif et donc plus prédictif. En effet, des facteurs supplémentaires permettent une attribution plus spécifique des risques auxquels une entreprise est exposée.

En outre, d'un point de vue statistique, l'ajout de variables indépendantes à une régression améliore souvent le pouvoir explicatif d'un modèle. Pour ces raisons, les modèles multifactoriels relâchent l'hypothèse et la contrainte d'un facteur de risque unique et recherchent d'autres facteurs qui affectent le rendement attendu des actifs.

En raison des nombreuses hypothèses concernant les différents facteurs de risque et de l'abondance des données disponibles sur les actions cotées en bourse, de nombreuses recherches ont été menées dans le but d'identifier des facteurs de risque supplémentaires ayant une forte capacité de prédiction.

Kewei Hou, Chen Xue et Lu Zhang ont testé le « q-factor model » en 2015. Il met en œuvre le CAPM d'investissement via l'approche de portefeuille de Fama-French (1993). Le modèle à q facteurs dit que le rendement attendu d'un actif au-delà du taux sans risque est décrit par ses sensibilités au facteur marché, à un facteur taille, à un facteur investissement et à un facteur rendement des fonds propres (« ROE »).

Données et méthodes

Pour ce mémoire, nous nous sommes penchés sur trois marchés différents : européen, américain et japonais. La construction des différents portefeuilles nécessaire à l'étude sera donc mené sur la base des composants des indice Eurostoxx 50, Nasdaq 100 et Nikkei 225.

I- Data

Yahoo Finance

Pour l'extraction de prix des actifs, la donnée nous provient de Yahoo Finance. Une librairie Python permet de pouvoir en extraire la donnée très facilement grâce à la fonction yfinance.download(). Cette fonction permet d'extraire pour n'importe quel actif côté le prix d'ouverture, le plus haut journalier, le plus bas journalier, le prix de clôture, et le volume échangé chaque jour. Vous trouverez ci-dessous un aperçu de la donnée pour l'Eurostoxx 50. Nous avons choisi la période du 31/07/2021 au 30/07/2021 (2001 observations) car les volumes n'étaient pas disponibles à des dates plus anciennes.

Figure 5 : Aperçu de la donnée extraite de Yahoo Finance pour l'Eurostoxx 50

Date	Open	High	Low	Close	Volume
2013-07-31	2749.669922	2773.600098	2747.709961	2768.149902	43668400
2013-08-02	2816.139893	2819.139893	2796.530029	2811.000000	56128500
2013-08-05	2815.260010	2821.389893	2800.530029	2809.080078	33450100
2013-08-06	2809.179932	2820.620117	2777.159912	2790.780029	53265000
2013-08-07	2784.810059	2798.389893	2775.189941	2794.439941	51430500
...	...	...	...	...	...
2021-07-26	4103.500000	4106.399902	4072.860107	4102.589844	25844200
2021-07-27	4099.799805	4099.799805	4056.229980	4064.830078	22912400
2021-07-28	4067.840088	4103.450195	4065.250000	4103.029785	21437300
2021-07-29	4104.979980	4128.810059	4104.979980	4116.770020	22534200
2021-07-30	4113.109863	4114.279785	4078.330078	4089.300049	30295500

Cette partie « Data » relève du fichier Python input_data.py dont le code est visible dans l'annexe 1. Ce fichier inclut une classe nommée « Analysis » dans laquelle nous avons créé des fonctions permettant d'extraire chacun des éléments précédemment mentionnés afin de pouvoir les utiliser indépendamment plus tard. D'autres fonctions calculent les rendements logarithmiques (nécessaires à une évaluation statistique) et la volatilité journalière pour l'indice de référence ainsi que ses composants. Cela nous sera utile plus tard pour la construction des portefeuilles et des facteurs.

Kenneth Ronald French Data Library

Ensuite, nous récupérons les coefficients des facteurs existants. Le site internet de Kenneth French actualise régulièrement les fichiers de valeurs qui sont calculés sur des marchés différents et de plusieurs manières. Nous avons téléchargé les fichiers concernant les marchés européen, américains et japonais. A titre d'exemple, le fichier nécessaire pour le marché européen est intitulé « Europe_5_Factors_Daily.csv » où nous trouverons les cinq facteur. Pour le Momentum de Carhart, c'est « Europe_MOM_Factor_Daily.csv ». Voici le lien du site internet : https://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html

Figure 6 : Aperçu des valeurs quotidiennes des cinq facteurs de Fama & French

	Mkt-RF	SMB	HML	RMW	CMA
2013-08-02	0.84	0.06	-0.18	-0.15	-0.08
2013-08-05	0.10	0.58	-0.46	0.19	0.04
2013-08-06	-0.16	0.07	-0.45	0.07	-0.26
2013-08-07	0.13	0.15	0.52	-0.31	0.23
2013-08-08	0.94	0.02	0.63	-0.42	0.44
...	...	...	...	...	...
2021-07-26	0.34	0.55	1.15	-0.46	0.24
2021-07-27	-0.47	0.24	0.32	-0.02	0.26
2021-07-28	0.96	-0.06	-0.98	0.09	-0.71
2021-07-29	0.79	0.13	0.27	-0.19	-0.04
2021-07-30	-0.50	0.24	-0.55	0.47	-0.21

II- Construction des portefeuilles

Ensuite, nous devons créer des portefeuilles sur lesquelles nous calculerons la capacité explicative des facteurs. Trois portefeuilles sont construits dans le fichier portfolios.py que vous trouverez en annexe 2. En fin de compte, c'est le calcul des poids des composants qui différera d'un portefeuille à l'autre. A noté que les rebalancements ont lieux quotidiennement.

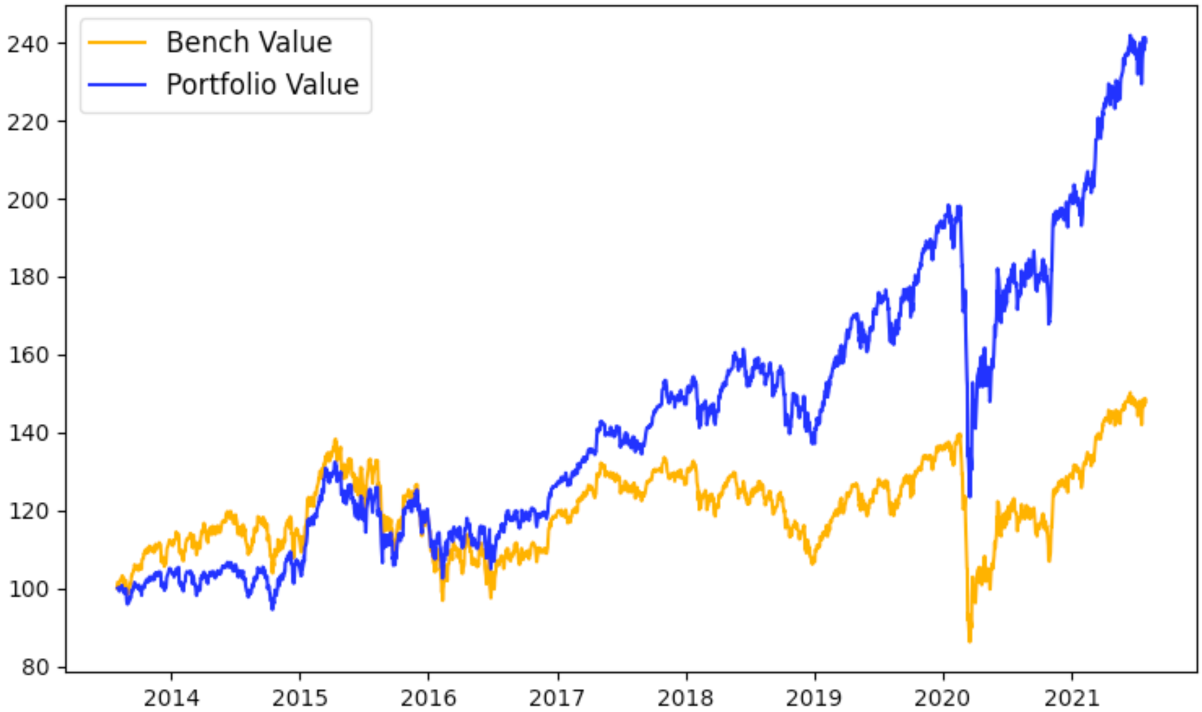
Le premier portefeuille créé est un portefeuille dit « équipondéré ». Le poids de chaque composant est le même.

Le deuxième portefeuille est un portefeuille dit « return momentum ». C'est-à-dire que les rebalancements sont effectués sur une logique de performances boursières passées. Concrètement, le code achètera les composants les plus performants de la période passée et vendra les composants les moins performants du portefeuille sur cette même période.

Le troisième portefeuille est un portefeuille dit « low volatility momentum ». C'est dire que les rebalancements sont effectué sur une logique de faible volatilité sur la période passée. Concrètement, le code achètera les composants les moins volatiles de la période passée et vendra les composants les plus volatile sur cette même période. L'objectif est d'éviter les composants risqués (donc volatile) et de favoriser le maintien du capital investit.

L'analyse de ce mémoire portera sur le seul portefeuille équipondéré. Le développement et le l'analyse plus complexe de nouveaux portefeuilles fera l'objet de recherches plus avancées dans la suite de mes études.

Figure 7 : Performance de l'Eurostoxx 50 et du portefeuille équipondéré sur base 100



III- Construction des nouveaux facteurs

Nous avons créé 18 facteurs qui peuvent être répartis en trois catégories. La première catégorie rassemble les facteurs calculés à base d'indicateurs de tendance. La deuxième catégorie concerne ceux calculés sur la base d'indicateurs plus connus sous le nom d'oscillateur. Et la troisième catégorie relève de ceux qui s'attèle à prévoir la volatilité. Beaucoup de ces facteurs sont basé sur des indicateurs techniques largement utilisé dans la création de stratégies de trading. Leur présentation sera divisée de la manière suivante : introduction, calculs, stratégie, visualisation graphique et construction des quotients quotidiens normalisés. Un travail d'optimisation a été fournis sur le calcules des facteurs dans le but d'atteindre des significativités optimisées.

Pour la normalisation des facteurs on divise la différence entre les valeurs X et la moyenne μ par l'écart-type σ:

$$X_{normalisé} = \frac{X - \mu}{\sigma}$$

Le développement des facteurs est disponible en l'annexe 3 et correspond au fichier factors.py.

Indicateurs de tendance

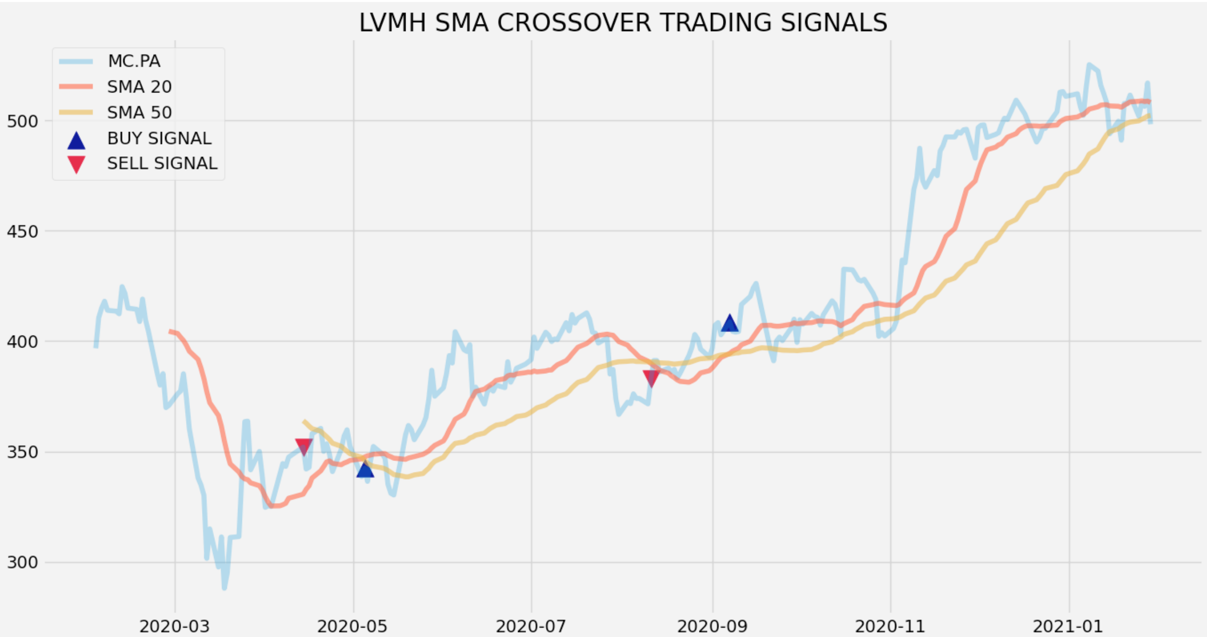
Ce sont des indicateurs de tendances qui aident les traders à identifier la direction du marché. Naturellement, les indicateurs connus pour cela sont les moyennes mobiles : moyenne mobile simple (SMA), moyenne mobile exponentielle (EMA), moyenne mobile pondérée (WMA) et enfin la convergence et divergence de moyennes mobiles (MACD). Dans ce travail, elles seront toutes calculées à partir des prix et des volumes.

Simple Moving Average (SMA)

La moyenne mobile simple (SMA) n'est rien d'autre que la valeur (prix ou volume) moyenne sur une période donnée. Ce facteur repose sur la stratégie de trading suivante. Deux SMA sont calculées, l'une avec une période courte et l'autre plus longue que la première. La stratégie achètera l'actif si la courbe de la SMA de période courte passe au-dessus de la courbe de la SMA de période longue. Et inversement, la stratégie vendra l'actif si la courbe de la SMA de période longue repasse au-dessus de la courbe de la SMA de période courte. En d'autres termes :

$$\text{SMA (période courte)} > \text{SMA (période longue)} \Rightarrow \text{achat}$$
$$\text{SMA (période longue)} > \text{SMA (période courte)} \Rightarrow \text{vente}$$

Figure 8 : visualisation de la stratégie menée sur les prix de LVMH



Pour la construction du facteur, nous avons choisi de ne retenir que la SMA de période courte que nous avons normalisée.

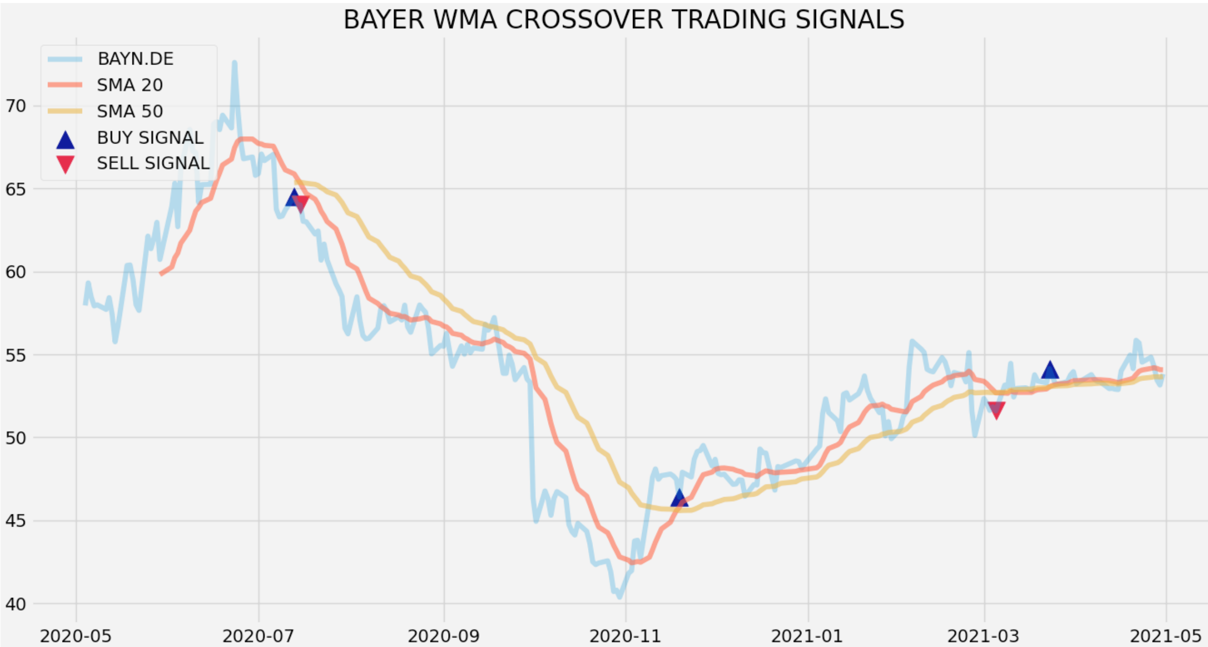
Weighted Moving Average (WMA)

L'une des limites de la SMA est qu'elle accorde un poids égal à chacune des valeurs quotidiennes inclus dans la période. Par exemple, dans une moyenne mobile de 10 jours, le jour le plus récent reçoit le même poids que le premier jour de la fenêtre : chacun reçoit une pondération de 10 %. Par rapport à la moyenne mobile simple, la moyenne mobile pondérée donne plus de poids au prix (ou volume) plus récent et progressivement moins à mesure que l'on remonte dans le temps.

Sur une moyenne pondérée de période 10, la valeur du 10ème jour sera multipliée par 10, celle du 9ème jour par 9, celle du 8ème jour par 8 et ainsi de suite. Le total sera ensuite divisé par la somme des pondérations (dans ce cas : 55). Dans cet exemple spécifique, le jour le plus récent reçoit environ 18,2% du poids total, le deuxième plus récent 16,4%, et ainsi de suite jusqu'au prix (ou volume) le plus ancien de la fenêtre qui reçoit 0,02% du poids.

Encore une fois, deux WMA de période différente vont être calculés et les signaux d'achat et de vente auront lieu aux croisements des courbes.

Figure 9 : visualisation de la stratégie menée sur les prix de Bayer



Pour la construction du facteur, nous avons choisi de ne retenir que la WMA de période courte que nous avons normalisée.

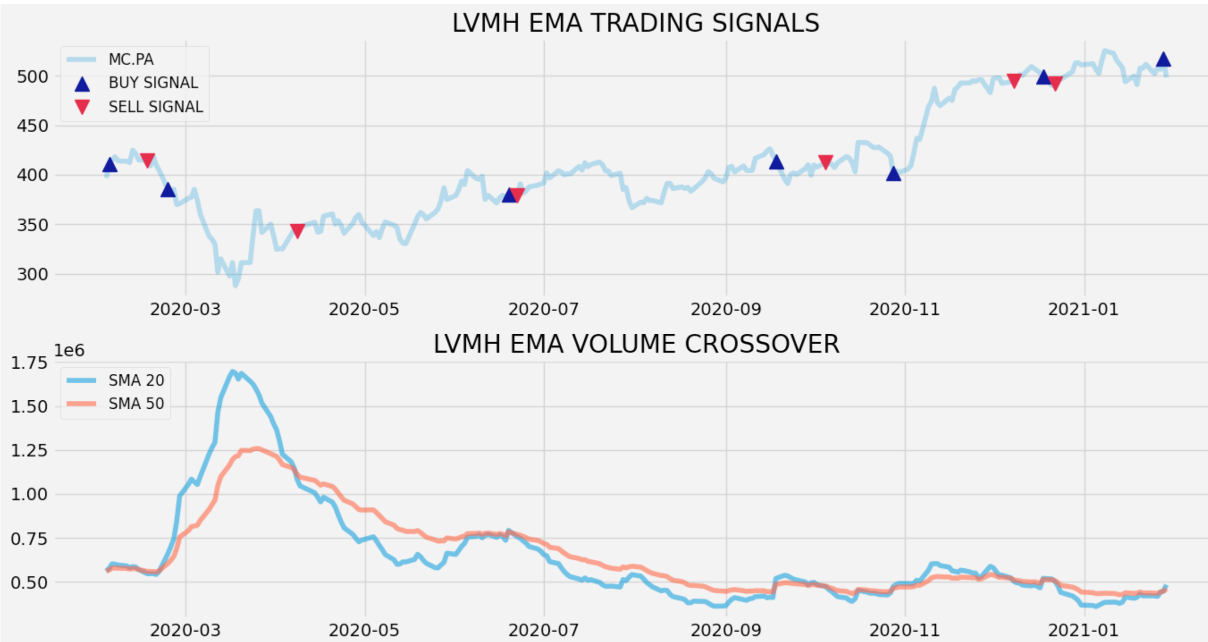
Exponential Moving Average (EMA)

Au même titre que la moyenne mobile pondérée, la moyenne mobile exponentielle (EMA) attribue un poids plus important aux observations les plus récentes. Bien qu'elle accorde un poids moindre aux données passées, elle est basée sur une formule récursive qui inclut dans son calcul toutes les données passées de notre série de prix. L'EMA au temps t est calculée comme le prix actuel multiplié par un facteur de lissage (« smoothing factor ») alpha (un nombre positif inférieur à 1) additionné à l'EMA au temps *t*-1 multiplié par 1 moins alpha. Il s'agit essentiellement d'une valeur comprise entre l'EMA précédente et le prix actuel :

$$EMA\,[t] = (\alpha \text{ x } Price\,[t]) + ((1 - \alpha) \text{ x } EMA\,[t - 1]) \text{ avec } \alpha = 2/\,(N + 1)$$

La stratégie de trading est la même : deux EMA sont calculées. L'une avec une période courte et l'autre plus longue que la première et les signaux auront lieu au croisement des courbes.

Figure 10 : visualisation de la stratégie menée sur les volumes de LVMH



Le facteur est construit à partir de la différence entre l'EMA de période courte et de période longue que nous avons normalisée.

Moving Average Convergence Divergence (MACD)

Le MACD est un indicateur de suivi de tendance qui est calculé en soustrayant deux moyennes mobiles exponentielles (une avec des périodes plus longues et l'autre plus courtes).

L'indicateur MACD comporte trois composantes importantes : la ligne MACD, la ligne de Signal, et l'histogramme des différences entre ces deux dernières.

Calcule de la ligne MACD = EMA (période courte) – EMA (période longue)

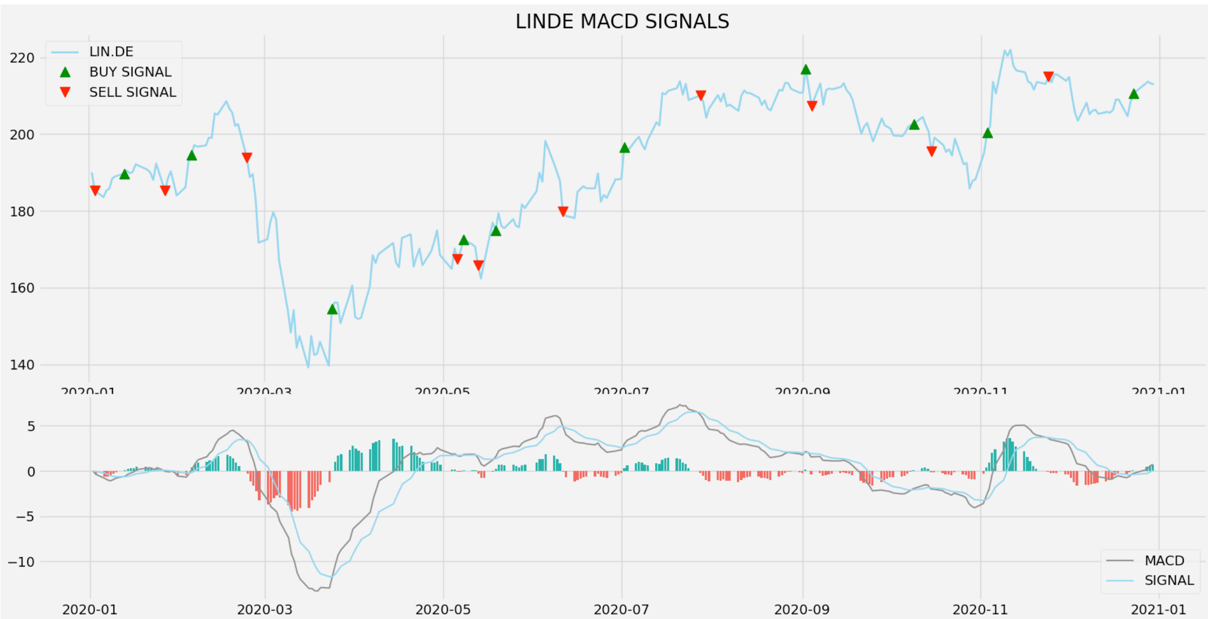
La ligne Signal est la moyenne mobile exponentielle de la ligne MACD elle-même pour une période donnée. La période la plus populaire pour calculer la ligne Signal est 9. Comme nous faisons la moyenne de la ligne MACD elle-même, la ligne Signal sera plus lisse que la ligne MACD. Typiquement, les périodes courtes et longues pour le calcul de la ligne MACD sont respectivement de 12 et 26 jours.

Réalisation de l'histogramme = MACD line – Signal line

La logique de la stratégie de trading basée sur le MACD est très similaire à celle d'une SMA :

$$\boxed{\begin{array}{l}\text{MACD line} > \text{Signal line} \Rightarrow \text{Achat} \\ \text{Signal line} > \text{MACD line} \Rightarrow \text{Vente}\end{array}}$$

Figure 11 : visualisation de la stratégie menée sur les prix de Linde



Pour le calcul des valeurs du facteur, nous avons choisi de ne retenir que la ligne MACD que nous avons normalisée. Cette ligne représente en effet le cœur de la logique de cet indicateur, les autres lignes ne servant qu'à générer des signaux de trading.

Oscillateur momentum

Dans cette section, nous allons développer des facteurs reposant sur des indicateurs dit « oscillateurs » et « momentum ». Oscillateurs, car ils oscillent entre deux bornes au-dessus et en dessous desquelles les actifs sont considérés comme respectivement surachetés et survendus. Momentum, car ils révèlent la force d'une tendance haussière ou baissière. On dit qu'une action est surachetée lorsque la tendance du marché semble être extrêmement haussière et qu'elle est destinée à se consolider (plutôt que de continuer à progresser). De même, une action atteint une zone de survente lorsque la tendance du marché semble extrêmement baissière et qu'on s'attend à ce qu'elle rebondisse.

Stochastic Oscillator

L'oscillateur stochastique est un indicateur utilisé pour déterminer si le marché est en état de surachat ou de survente. Les valeurs de l'oscillateur stochastique se situent toujours entre 0 et 100 en raison de sa fonction de normalisation. Les niveaux généraux de surachat et de survente sont considérés comme étant respectivement d'au moins 70 et d'au plus 30, mais ils peuvent varier selon la sensibilité de l'investisseur et de l'action en question.

L'oscillateur stochastique comprend deux composantes principales :

La ligne %K (« Fast Stochastic ») : le but de cette ligne est d'exprimer l'état actuel du marché (suracheté ou survendu). Elle est calculée en soustrayant du cours de clôture de l'action le prix le plus bas que l'action a atteint sur une période relativement courte et cette différence est ensuite divisée par la valeur calculée en soustrayant du cours le plus élevé de l'action le prix le plus bas que l'action a atteint sur cette même période. La valeur finale est obtenue en multipliant par 100 la valeur calculée à partir des étapes susmentionnées. On calcule la ligne %K avec comme convention une période de 14 jours :

$$\%K = 100 \times ((\text{Prix}[t] - \text{Prix le plus bas}) - (\text{Prix le plus haut} - \text{Prix le plus bas}))$$

Ligne %D (« Slow Stochastic ») : elle n'est rien d'autre que la moyenne mobile de la ligne %K pour une période plus longue que celle précédemment utilisée pour le calcul de la ligne %K. C'est une version lissée de la ligne %K, car le graphique linéaire de la ligne %D aura l'air plus lisse que la ligne %K. Le paramètre standard de la ligne %D est 3 comme longueur de période.

Un marché est considéré comme suracheté lorsque les courbe %K et %D dépassent un certain seuil (dont le niveau exact est à la discrétion du trader – généralement entre 70 et 90). Un marché survendu est mis en lumière lorsque les courbes sont en dessous d'un certain niveau (cela est aussi à la discrétion de l'investisseur – typiquement entre 10 et 30). Voici donc la stratégie retenue :

%K and %D < 10 indique un marché survendu ⇒ Achat

%K and %D > 90 indique un marché suracheté ⇒ Vente

Figure 12 : visualisation de la stratégie menée sur AXA

Pour la construction du facteur, nous avons conservé uniquement la ligne %K que nous avons normalisée. La ligne %D n'est qu'un outil pour la stratégie de trading, mais ne représente en rien une innovation.

Relative Strength Index

Développé par J. Welles Wilder en 1978, le RSI peut sembler proche de l'oscillateur stochastique en terme d'interprétation de la valeur, mais la façon dont il est calculé est assez différente. Le calcul du RSI s'effectue en trois étapes.

Tout d'abord il faut calculer les rendements quotidiens et séparer les gains des pertes. En utilisant ces valeurs séparées, on calcule les EMA des gains et des pertes séparément et pour une période généralement de 14 jours. Ensuite on calcule la force relative de l'actif en divisant l'EMA des gains par l'EMA des pertes. Elle peut être représentée mathématiquement comme suit :

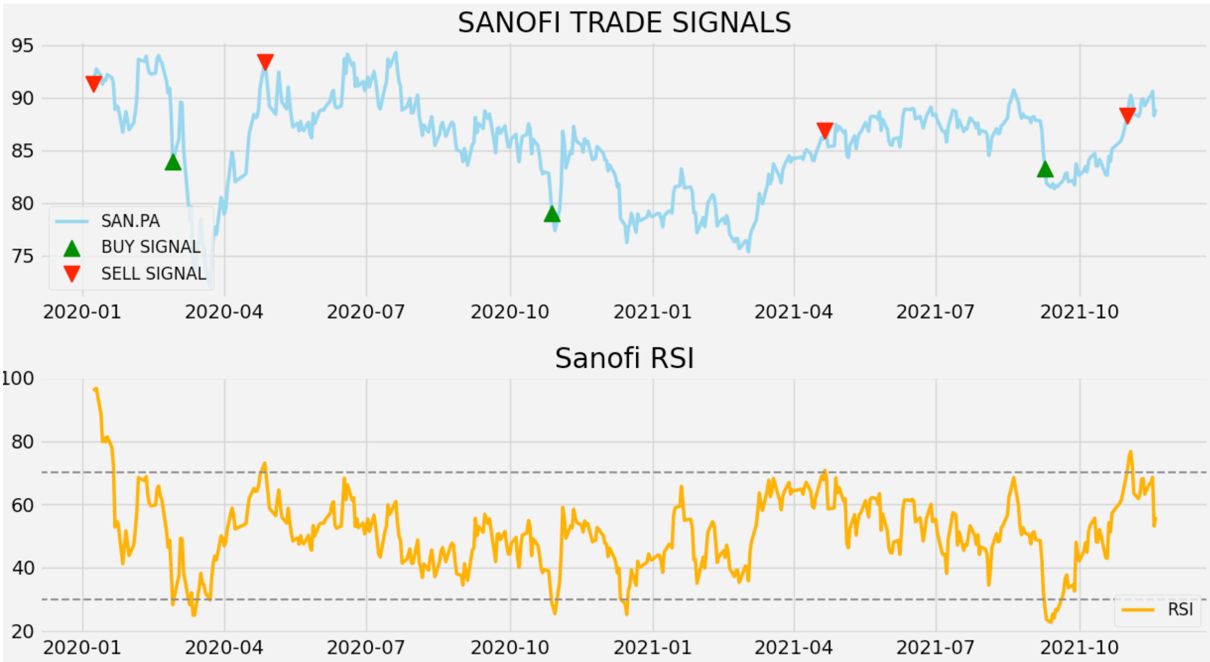
$$RS = EMA\ gains\ /\ EMA\ pertes$$

$$RSI = 100.0 - (100.0\ /\ (1.0 + RS))$$

Passons à présent à la stratégie de trading basé sur le RSI. Là encore, la logique est la même que pour l'oscillateur stochastique. C'est une stratégie de croisement simple avec les paramètres traditionnels de 30 et 70 comme niveaux de survente et de surachat respectivement, et avec 14 comme période de recul. Notre stratégie révèle un signal d'achat lorsque la valeur RSI précédente est supérieure au niveau de survente et que la valeur RSI actuelle passe sous le niveau de survente. De même, la stratégie révèle un signal de vente lorsque la valeur RSI précédente est inférieure au niveau de survente et que la valeur RSI actuelle passe au-dessus du niveau de survente. Notre stratégie de trading peut être représentée comme suit :

$$\boxed{\begin{array}{l} \text{RSI précédent} > 30 \text{ et RSI actuel} < 30 \Rightarrow \text{Achat} \\ \text{RSI précédent} < 70 \text{ et RSI actuel} > 70 \Rightarrow \text{Vente} \end{array}}$$

<u>Figure 13 : visualisation de la stratégie menée sur Sanofi</u>



Le facteur est le quotient RSI normalisé.

Williams %R

Au même titre que l'oscillateur stochastique et que le RSI, le %R de Williams (W%R) est un indicateur momentum qui oscille entre 0 et 100. Bien qu'ayant le même but de mettre en lumière la force d'une tendance, son calcul est différent que les deux précédents. Les seuils traditionnels pour les niveaux de surachat et de survente sont respectivement de 20 et 80, mais il n'est pas interdit de prendre d'autres valeurs.

Pour calculer les valeurs du W%R on détermine d'abord le plus haut et le plus bas sur une période de traditionnellement 14 jours. Ensuite, deux différences sont calculées. La première est la différence entre le cours de clôture par rapport au plus haut de la période. La seconde est la différence entre plus bas de la période par rapport au plus haut de la période. Enfin, la première différence est divisée par la seconde et multipliée par -100. Le calcul peut être représenté mathématiquement comme suit :

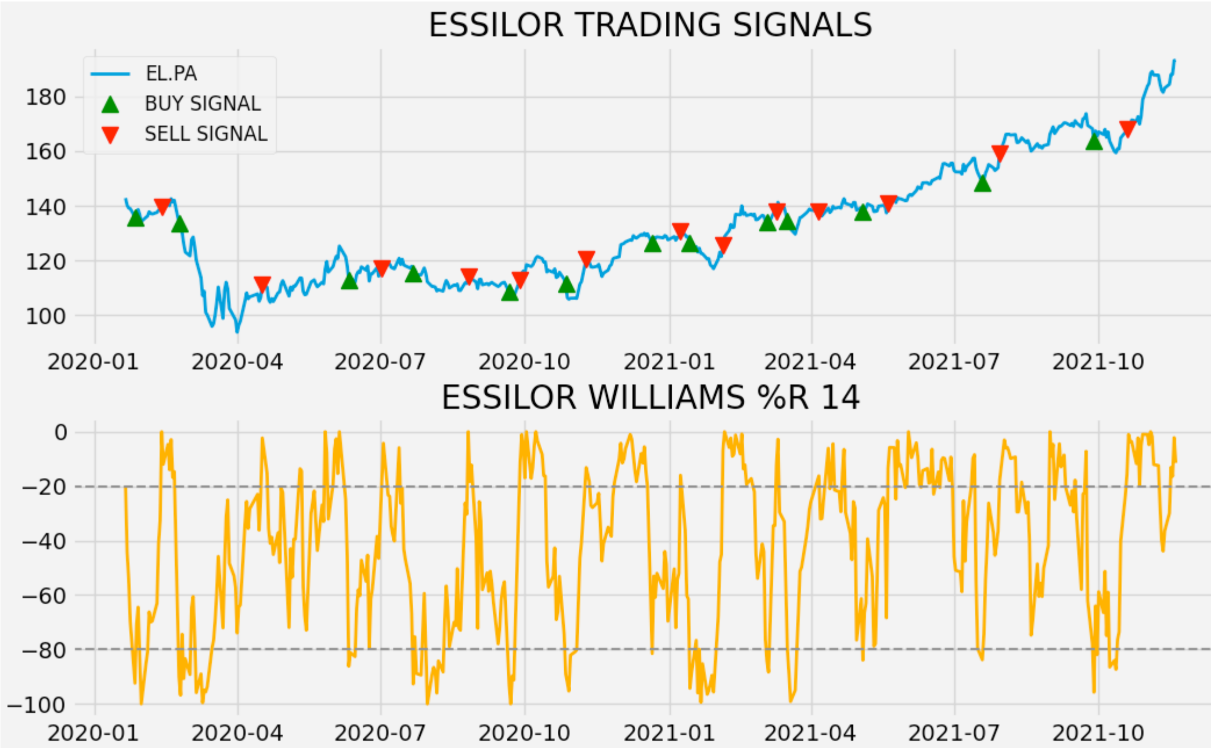
$$W\%R = [\text{Prix le plus haut} - \text{Prix}[t]] \,/\, [\text{Prix le plus haut} - \text{Prix le plus bas}] \times (-100)$$

L'idée sous-jacente de cet indicateur est que le titre va continuer à atteindre de nouveaux sommets lorsqu'il s'agit d'une forte tendance à la hausse et, de la même manière, le titre va atteindre de nouveaux bas lorsqu'il suit une forte tendance à la baisse.

La stratégie de trading que nous allons retenir est la suivante. Un signal d'achat sera révélé lorsque la lecture précédente du Williams %R est inférieure à -20 et que la lecture actuelle est supérieure à -20. De même, un signal de vente est généré lorsque la lecture précédente du Williams %R est supérieure à -80 et que la lecture actuelle est inférieure à -80. Notre stratégie de trading peut être représentée comme suit :

$$W\%R\,[t-1] < -20 \text{ et } W\%R[t] > -20 \Rightarrow \text{Achat}$$
$$W\%R\,[t-1] > -80 \text{ et } W\%R[t] < -80 \Rightarrow \text{Vente}$$

Figure 14 : visualisation de la stratégie menée sur Essilor



Les valeurs retenues pour le facteur seront donc uniquement les R de Williams normalisés.

On-Balance Volume

L'OBV utilise le flux de volume pour prédire les changements de prix des actions. Joseph Granville a développé pour la première fois la métrique OBV dans son livre Granville's New Key to Stock Market Profits sorti en 1963.

Granville voulait démontrer que le volume était la force clé des marchés et a conçu l'OBV pour prévoir le moment où les mouvements majeurs se produiraient en fonction des changements de volume. Dans son livre, il décrit les prédictions générées par l'OBV comme "un ressort que l'on enroule fermement". Il pensait que lorsque le volume augmente fortement sans changement significatif du prix de l'action, le prix finit par bondir à la hausse ou à la baisse.

L'OBV fournit un total courant du volume de négociation d'un actif et indique si ce volume entre ou sort d'un titre donné. L'OBV est un total cumulé du volume (positif et négatif).
Trois règles sont appliquées lors du calcul de l'OBV :

1- Si le prix de clôture d'aujourd'hui est supérieur au prix de clôture d'hier :

OBV [t] = OBV [t - 1] + Volume [t]

2- Si le prix de clôture d'aujourd'hui est inférieur au prix de clôture d'hier :

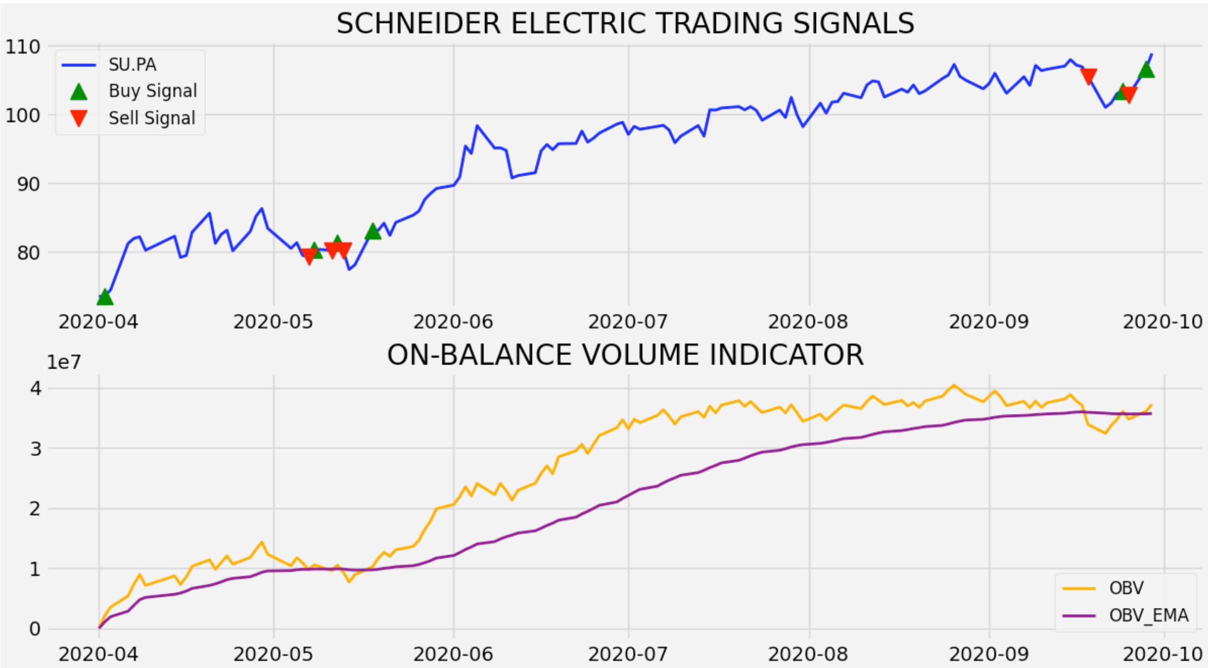
OBV [t] = OBV [t - 1] - Volume [t]

3- Si le prix de clôture d'aujourd'hui est égal au prix de clôture d'hier :

OBV [t] = OBV [t - 1]

Comme stratégie de trading nous avons choisi d'émettre des signaux d'achat et de vente à chaque croisement entre la courbe de l'OBV et de la courbe d'une moyenne mobile exponentielle calculée sur une période de 50 jours.

Figure 15 : visualisation de la stratégie menée sur Schneider Electric



Le facteur sera calculé avec des EMA de 20 périodes sur les valeurs de l'OBV. Le résultat sera normalisé.

Directional Movement Index

L’indice de mouvement directionnel, aussi connu sous l’appellation Average Directional Index (ADX), est un indicateur qui permet aux traders d’identifier la force de la tendance du marché. Ainsi, soit il doit être utilisé en complément d’une moyenne mobile, soit il doit inclure des indicateurs de tendance. Notre choix se porte sur la seconde solution pour la construction de notre facteur.

Pour identifier la direction de la tendance, l'ADX est combiné avec un indice directionnel positif (+ DI) et un indice directionnel négatif (- DI). Comme son nom l'indique, le + DI mesure la tendance haussière ou positive du marché, de même que le - DI mesure la tendance baissière ou négative du marché. Les valeurs de tous les composants sont comprises entre 0 et 100, et agissent donc comme un oscillateur. La période communément admise pour calculer l'ADX est de 14 jours.

Pour calculer les valeurs d'ADX avec 14 comme période de référence, il faut d'abord déterminer le mouvement directionnel positif (+ DM) et négatif (- DM). Le + DM est calculé en trouvant la différence entre le haut actuel et le haut précédent, et de même, le - DM est calculé en trouvant la différence entre le bas précédent et le bas actuel. Il peut être représenté comme suit :

$$+ DM = \text{haut actuel} - \text{haut précédent}$$
$$- DM = \text{bas précédent} - \text{bas actuel}$$

Ensuite, un Average True Range (ATR) avec 14 comme période de référence est calculé. L’ATR mesure la moyenne de la variation d'un actif sur les 14 dernières séances dans notre cas.

Maintenant, en utilisant le mouvement directionnel calculé et les valeurs ATR, l'indice directionnel positif (+ DI) et l'indice directionnel négatif (- DI) sont calculés. Pour déterminer les valeurs de + DI, la valeur obtenue en prenant la moyenne mobile exponentielle (EMA) du mouvement directionnel positif (+ DM) avec 14 comme période de référence est divisée par l'ATR de 14 jours calculé précédemment, puis multiplié par 100. La même chose s'applique à la détermination de (- DI), mais au lieu de prendre l'EMA de 14 jours de (+ DI), (- DI) est pris en compte.

La formule pour calculer le + DI et le - DI peut être représentée comme suit :

$$+ DI\ 14 = 100 \times [EMA\ 14\ (+ DM) / ATR\ 14]$$

$$- DI\ 14 = 100 \times [EMA\ 14\ (- DM) / ATR\ 14]$$

L'étape suivante consiste à utiliser le + DI et le - DI pour calculer l'indice directionnel. Il peut être déterminé en divisant la valeur absolue de la différence entre le + DI et le - DI par la valeur absolue du total du + DI et du - DI multiplié par 100. La formule pour calculer l'indice directionnel peut être représentée comme suit :

$$DI\ 14 = |\ (+ DI\ 14) - (- DI\ 14)\ | / |\ (+ DI\ 14) + (- DI\ 14)\ | \times 100$$

L'étape finale consiste à calculer l'ADX lui-même en utilisant les valeurs déterminées de l'indice directionnel. L'ADX est calculé en multipliant la valeur précédente de l'indice directionnel par 13 (période de recul - 1) et en l'ajoutant à l'indice directionnel, puis en le multipliant par 100. La formule pour calculer les valeurs de l'ADX peut être représentée comme suit :

$$ADX\ 14 = [(PREV\ DI\ 14 \times 13) + DI\ 14] \times 100$$

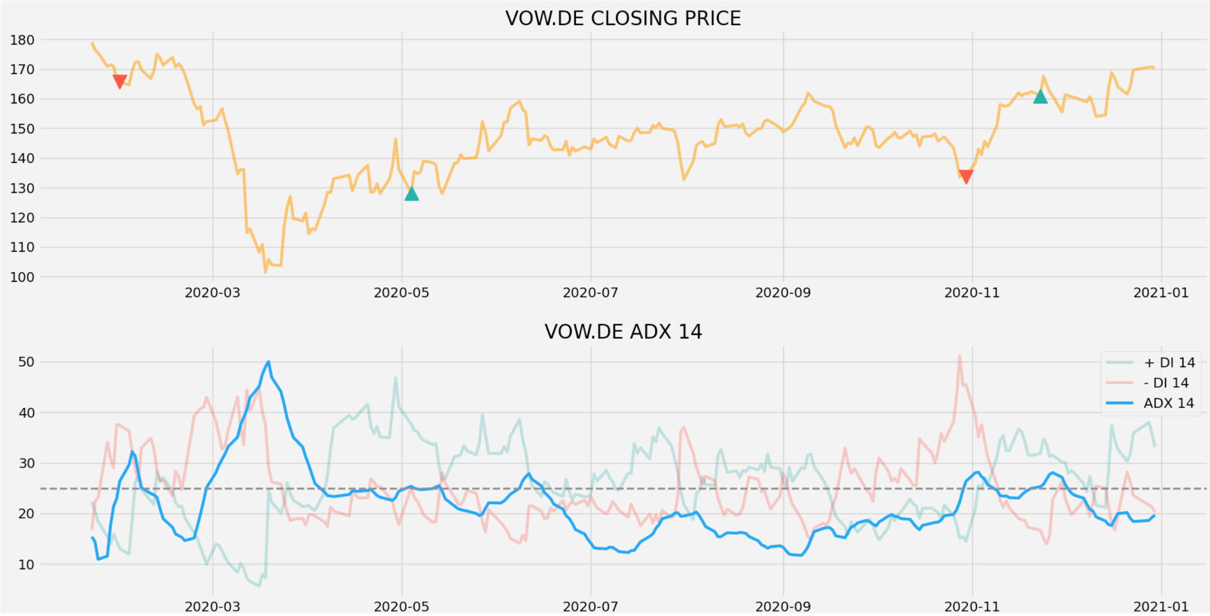
L'ADX ne peut pas être utilisé tel quel et doit être lissé. Depuis sa création par Wilder Wiles (le fondateur de l'ATR également), l'ADX est lissé par une moyenne mobile personnalisée dont nous avons parlé précédemment.

Dans le but de pouvoir visualiser cette stratégie, nous devons construire les signaux. Un signal d'achat est révélé chaque fois que la ligne ADX passe au-delà de 25 et que la ligne + DI est au-dessus de la ligne - DI. De même, un signal de vente est généré lorsque la ligne ADX passe au-delà de 25 et que la ligne - DI est au-dessus de la ligne + DI. Notre stratégie de trading peut être représentée comme suit :

$$ADX\ [t - 1] < 25 \text{ ET } ADX\ [t] > 25 \text{ et } + DI > - DI \Rightarrow \text{Achat}$$

$$ADX\ [t - 1] < 25 \text{ ET } ADX\ [t] > 25 \text{ et } + DI < - DI \Rightarrow \text{Vente}$$

Figure 16 : visualisation de la stratégie menée sur Volkswagen



Les valeurs de la ligne ADX normalisées composeront le facteur.

Facteurs de volatilité

Bollinger bands

Les bandes de Bollinger sont un type d'indicateur technique qui permet aux traders d'analyser la volatilité d'une action et de déterminer si le prix est relativement élevé ou bas. La bande supérieure se situe généralement à deux écarts-types au-dessus de la SMA et la bande inférieure à deux écarts-types en dessous de la SMA. Mathématiquement, on obtient :

$$\text{Bande supérieure} = \text{SMA} + 2 \times \text{écart-type}$$

$$\text{Bande supérieure} = \text{SMA} - 2 \times \text{écart-type}$$

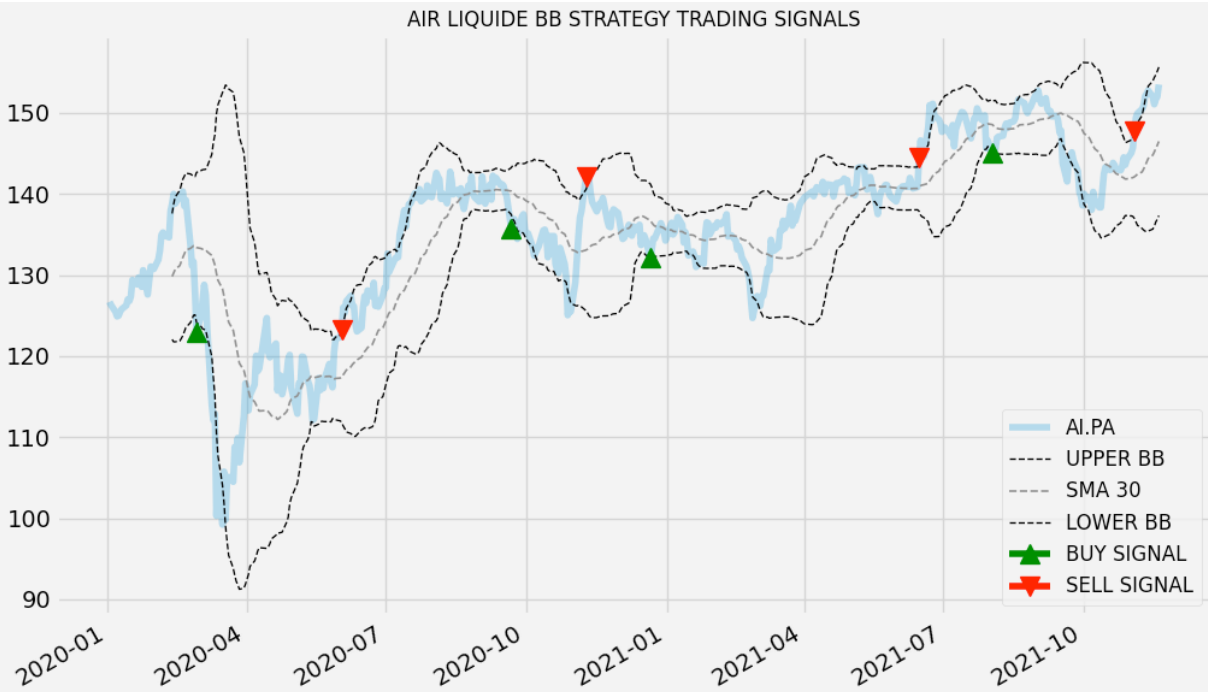
Pour mettre en place la stratégie de trading, nous identifions des signaux d'achat lorsque la ligne de prix touche la bande inférieure et des signaux de vente lorsque la ligne de prix touche la bande supérieure. Pour calculer les bandes, nous prendrons une période de recul de 30 jours.

Telle est la logique d'investissement au temps t :

Prix [t - 1] > bande inférieure [t - 1] et Prix [t] < bande inférieure [t] ⇒ Achat

Prix [t - 1] < bande supérieure [t - 1] et Prix [t] > bande supérieure [t] ⇒ Vente

Figure 17 : visualisation de la stratégie menée sur Air Liquide



Les quotients quotidiens du facteur seront les valeurs normalisées de la bande inférieure. Nous nous sommes rendu compte que cela lui conférait une significativité plus fréquente.

Fibonacci retracements

Les outils de trading Fibonacci sont utilisés pour déterminer les niveaux de support (aussi appelé niveau de résistance) ou pour identifier les cibles de prix. C'est l'existence des séries de Fibonacci dans le domaine mathématique qui a attiré l'attention des investisseurs pour le trading. Les nombres de Fibonacci fonctionnent pour trouver des niveaux clés dans n'importe quel titre boursier.

La séquence de Fibonacci est une série de nombres, commençant par zéro et un, dans laquelle chaque nombre est la somme des deux nombres précédents. La séquence commence donc par 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610. Elle s'étend à l'infini et peut être résumée par la formule suivante :

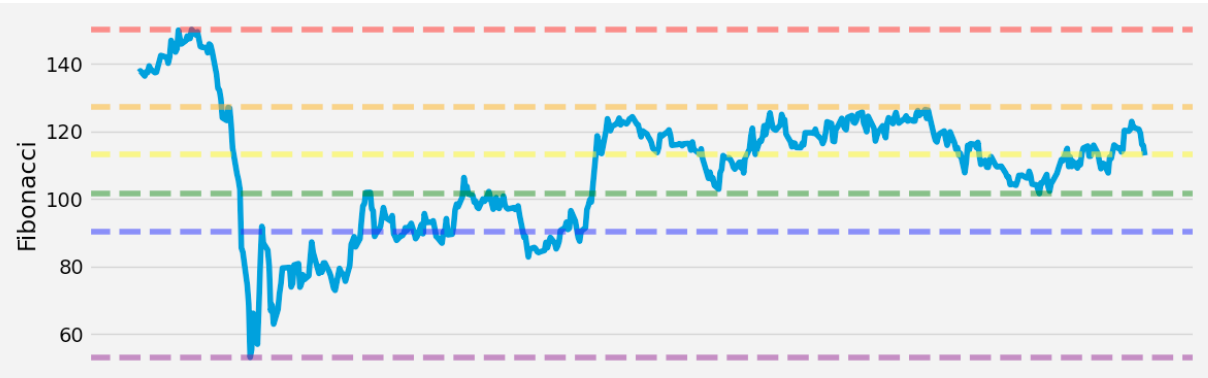
$$X_n = X_{n-1} + X_{n-2}$$

La suite de Fibonacci présente quelques propriétés intéressantes. Divisez un nombre quelconque de la séquence par le nombre suivant et vous obtiendrez souvent 0.618, dit le « golden number ». Une cohérence similaire est trouvée lorsque n'importe quel nombre de la séquence est divisé par un nombre situé deux places à droite de lui. Systématiquement, le résultat autour de 0.382.

Les ratios de Fibonacci à retenir pour l'analyse de series chronologies (« time series analysis ») sont 0, 23.6%, 38.2%, 50%, 61.8%, 78.6% et 1. Ils vont permettre de trouver un niveau de soutien. Chaque fois que le prix se déplace de manière importante vers le haut ou vers le bas, il a généralement tendance à être « corrigé » avant de continuer à se déplacer dans la direction initiale. Par exemple, si le cours de l'action est passé de 200 à 250 dollars, il est probable qu'il se rétracte jusqu'à 230 dollars avant de continuer à monter. Ce niveau de retracement à 230 $ est prévu à l'aide des ratios de Fibonacci.

Figure 18 : visualisation des retracements de Fibonacci sur Safran

Les niveaux de Fibonacci ci-dessous sont calculés sur toute l'année 2020 au lieu d'être seulement calculés sur une période de tendance haussière ou baissière comme cela devrait être le cas idéalement.



Pour la construction du facteur, nous recalculons chaque jour le niveau du « golden number » 0.618 sur les 20 séances précédentes, comme si les 20 séances précédentes représentaient une tendance haussière ou baissière. Les niveaux du nombre doré seront normalisés et formeront les valeurs quotidiennes de notre facteur.

Figure 19 : Récapitulatif des facteurs et de leurs sigles

NOM	**SIGLE**
FF’s Robust Minus Weak	RMW
FF’s Conservative Minus Aggressive	CMA
Carhart’s Winners Minus Losers	WML
SMA sur les prix	SMA_C
SMA sur les volumes	SMA_V
EMA sur les prix	EMA_C
EMA sur les volumes	EMA_V
WMA sur les prix	WMA_C
WMA sur les volumes	WMA_V
MACD sur les prix	MACD_C
MACD sur les volumes	MACD_V
Stochastic Oscillator	OSC
Relative Strength Index	RSI
Williams %R	W%R
Directional Movement Index	DMI
On-balance Volume	OBV
Bollinger Bands	BB
Fibonacci retracements	Fibo

IV- Régression et estimation

Pour rappel, notre but est de trouver quelles sont les quatrièmes et cinquièmes facteurs les plus significatifs. Pour cela nous allons créer un modèle pour chaque combinaison de facteurs. Ce travail est visible dans la classe « Combinations » que vous retrouverez dans l'annexe 3. Les modèles créés seront composés des trois facteurs conventionnels de Fama et French auxquels on ajoutera une paire de facteurs. La paire de facteurs sera une combinaison créée à partir des facteurs que nous aurons développé, des facteurs RMW et CMA, et enfin du facteur Momentum de Mark Carhart. En tout, il y a 153 combinaisons, donc 153 modèles.

Nous testerons ensuite les 153 modèles sur un indice. Cet indice sera le portefeuille équipondéré que nous avons créé plus tôt. Il sera noté PV^{ew} pour Portfolio Value Equally Weighted.

<u>Figure 20 : Résultats du modèle à trois facteurs sur le marché européen</u>

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.824
Model:                            OLS   Adj. R-squared:                  0.824
Method:                 Least Squares   F-statistic:                     3118.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        10:07:10   Log-Likelihood:                 7726.8
No. Observations:                2001   AIC:                        -1.545e+04
Df Residuals:                    1997   BIC:                        -1.542e+04
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0086      0.000     65.149      0.000       0.008       0.009
SMB           -0.0075      0.000    -24.605      0.000      -0.008      -0.007
HML            0.0018      0.000      8.036      0.000       0.001       0.002
Bias           0.0001      0.000      1.043      0.297      -0.000       0.000
==============================================================================
Omnibus:                      152.914   Durbin-Watson:                   2.011
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              822.825
Skew:                           0.033   Prob(JB):                    2.12e-179
Kurtosis:                       6.141   Cond. No.                         2.91
==============================================================================
```

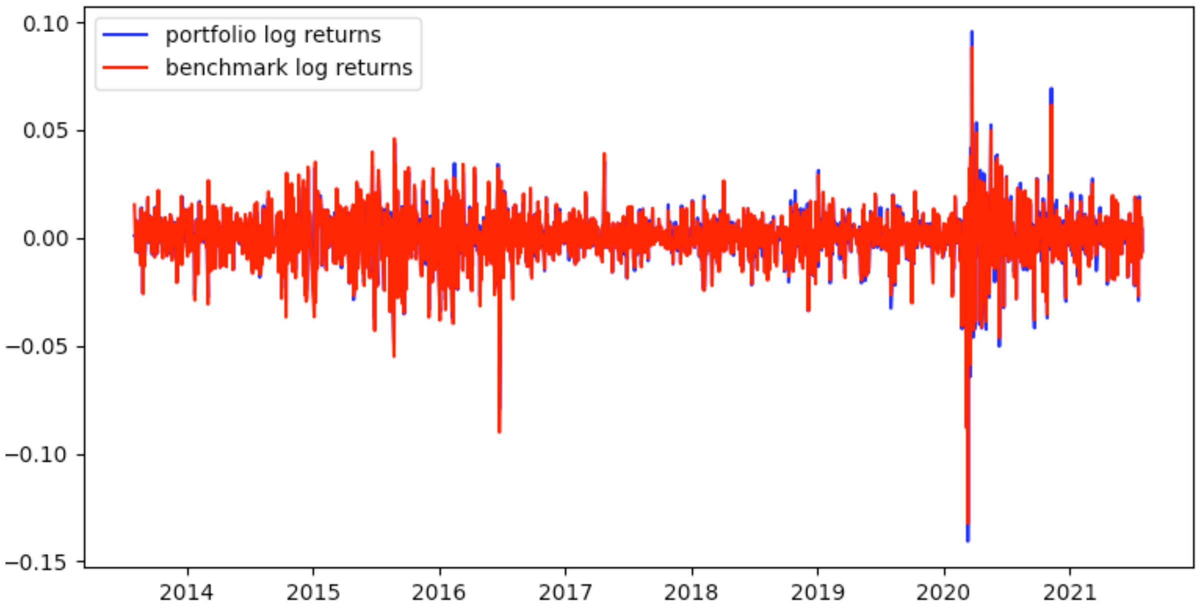
Pour une régression linéaire, nous devons vérifier que nos séries temporelles ne soit pas stationnaires. Elles sont stationnaires si leur moyenne et leur variance ne changent pas dans le temps. La non-stationnarité des données est également désignée par le terme de racine unitaire. Le test de Dickey-Fuller qui vérifie la non-stationnarité de la série est validé lorsque les p-value sont inférieur à 5%. L'objectif est d'annihiler toute tendance ou saisonnalité dans la série. Pour cela, nous calculons les rendements logarithmiques à partir desquelles la régression sera effectuée :

$$R_t^{ew} = \log(PV_t^{ew} / PV_{t-1}^{ew})$$

ou

$$R_t^{ew} = \log(PV_t^{ew}) - \log(PV_{t-1}^{ew})$$

Figure 21 : Rendements désaisonnalisés du portefeuille et de l'Eurostoxx 50



Ensuite, nous effectuons la régression. L'objectif est d'estimer les $\theta = \{\theta_0, \ \theta_1, \dots, \ \theta_q\}$ pour q-1 facteurs ; avec θ_0 la constante, également connu sous le nom de alpha, valeur qui contient les coefficients. Dans certaines études, ils peuvent aussi être appelé « intercepts » comme nous avons pu le croiser dans la revue de littérature. Il faut retenir que les alphas sont la différence entre le rendement du portefeuille et le rendement de l'indice de référence. C'est le « Bias » dans le modèle ci-dessus.

Nous estimons les alphas tel que :

$$\theta^* = argmin\left\{\frac{1}{2}\|R^{ew} - X\theta\|_2^2\right\}$$

$$f(X) = \frac{1}{2}\|R^{ew} - X\theta\|_2^2 = \frac{1}{2}\sum_{t=1}^{T}(R^{ew} - X_t\theta)^2$$

$\frac{df}{d\theta} \Leftrightarrow -\sum_{t=1}^{T}(R^{ew} - X_t\theta).X_t = 0$ Multiplié par -1 donne

$$\frac{df}{d\theta} \Leftrightarrow \sum_{t=1}^{T}(R^{ew} - X_t\theta).X_t = 0$$

$$\frac{df}{d\theta} \Leftrightarrow \sum_{t=1}^{T} R_t^{ew}.X_t - X_t.X_t^T.\theta = 0$$

$$\frac{df}{d\theta} \Leftrightarrow \sum_{t=1}^{T} R_t^{ew}.X_t = \sum_{t=1}^{T} X_t.X_t^T.\theta = 0$$

$$\frac{df}{d\theta} \Leftrightarrow \theta = \frac{\sum_{t=1}^{T} R_t^{ew}.X_t}{\sum_{t=1}^{T} X_t.X_t^T} = \frac{R^{ew}.X}{(X.X^T)^{-1}}$$

Dans le cas du modèle à 3 facteurs, la donnée sera $D = \{(R_1^{ew}, X_1), (R_2^{ew}, X_2), \dots, (R_T^{ew}, X_T)\}$ avec $X \in \mathbb{R}^q$ de tel manière que $X_t = [X_0, X_1, \dots, X_q]$ où X_0 sera la série pour calculer une constante avec $q = 3 + 1$, donc $\theta^* \in \mathbb{R}^{q+1}$. La logique est la même pour nos modèles à 5 facteurs.

Extension du modèle de Fama & French

Nous allons à présent analyser les résultats des 153 modèles pour chaque marché, en mettant un coup de projecteur sur les modèles d'origine (à trois facteurs FF3, et à cinq facteurs FF5) et le modèle le plus performant. Les facteurs les plus significatifs seront mis en lumière afin de mieux comprendre d'où proviennent les rendements.

I- Présentation des mesures statistiques

Nous analyserons les statistiques à partir des valeurs issues de la régression du modèle FF5. Le travail de modélisation du problème est effectué à l'aide de la librairie Python statsmodels. Le code de la régression est disponible en annexe 4 et correspond au fichier regression_py.

Figure 22 : Résultats du modèle FF5 sur le marché européen

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.827
Model:                            OLS   Adj. R-squared:                  0.826
Method:                 Least Squares   F-statistic:                     1906.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        10:21:07   Log-Likelihood:                 7742.8
No. Observations:                2001   AIC:                        -1.547e+04
Df Residuals:                    1995   BIC:                        -1.544e+04
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0084      0.000     61.489      0.000       0.008       0.009
SMB           -0.0078      0.000    -25.218      0.000      -0.008      -0.007
HML            0.0037      0.000      9.290      0.000       0.003       0.004
RMW            0.0023      0.001      4.085      0.000       0.001       0.003
CMA           -0.0021      0.001     -3.590      0.000      -0.003      -0.001
Bias        8.947e-05      0.000      0.788      0.431      -0.000       0.000
==============================================================================
Omnibus:                      129.868   Durbin-Watson:                   2.027
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              590.418
Skew:                          -0.005   Prob(JB):                    6.20e-129
Kurtosis:                       5.661   Cond. No.                         6.27
==============================================================================
```

L’information la plus importante que l’on doit observer est le résultat des R^2 (« R-squared » dans le tableau). C’est le pourcentage de variance expliqué par la droite. Dans ce travail, il représente l’excédent de rentabilité expliqué par les facteurs. Ici, c’est 82,7% de la variable dépendante qui est expliqué par les variables indépendantes, ce qui est déjà élevé. Dans le tableau de résultats du modèle à trois facteurs (voir plus haut), les R^2 s’élevaient à 82,4%. On gagne donc seulement 0,3% d’explication de la performance du portefeuille grâce aux deux derniers facteurs RMW et CMA.

Ensuite, notre première colonne informative est le coefficient. Pour chaque variable indépendante, il s'agit de la mesure de l'influence d'un changement de cette variable sur la variable dépendante. C'est l’équivalent du coefficient directeur a dans $y = ax + b$. Une unité de changement dans la variable dépendante affectera la valeur du coefficient de la variable indépendante. Si le coefficient est négatif, elles ont une relation inverse. Si l'une augmente, l'autre diminue.

L’erreur-type est une estimation de l'écart-type du coefficient, une mesure de la quantité de variation du coefficient dans l'ensemble de ses points de données. Le t est lié et constitue une mesure de la précision avec laquelle le coefficient a été mesuré. Une erreur-type faible par rapport à un coefficient élevé produit une statistique t élevée, ce qui signifie que votre coefficient est très significatif. Dans certains modèles, l’erreur-type est toujours à 0 mais le coefficient n’est pas toujours élevé. Ainsi, les coefficients ne sont pas systématiquement significatifs bien que l’erreur-type soit faible.

P>|t| est l'une des statistiques les plus importantes du résumé obtenu par le code. Elle utilise la statistique t pour produire la valeur p. La valeur p est une mesure de la probabilité que le coefficient soit mesuré par notre modèle par hasard. La valeur p de 0,000 pour le facteur SMB signifie qu'il y a 0% de chances que le facteur n'ait aucun effet sur la variable dépendante y (« Dep. Variable » dans le tableau), et que nos résultats soient le fruit du hasard.

Pour établir si un facteur est significatif ou non, il faut que la valeur p soit inférieur à 5% et que le t statistique soit supérieur à $|1{,}96|$. Ici, c’est le cas des cinq facteurs dont les « p-values » < 0,05 et dont les « t-statistics » $> |1{,}96|$.

Pour finir, nous devons interpréter les coefficients. Concernant le facteur SMB, un coefficient négatif indique que l'excédent de rentabilité du portefeuille provient principalement des grandes capitalisations. En gardant la même logique, un coefficient positif pour le facteur HML nous indique que ce même rendement provient plutôt des entreprises dites « Growth » (dont le ratio Book Value / Market Value est élevé) que des entreprises « Value » dont ce même ratio BV/MV est faible. Il est souvent intéressant d'analyser des portefeuilles plus équilibrés en termes de profil d'entreprise. Fama et French aboutissent d'ailleurs souvent à la conclusion que les entreprises « Value » performent mieux. Ainsi, dans un portefeuille équilibré, ce coefficient sera vraisemblablement négatif. Pour le coefficient RMW, s'il est positif, la performance est plutôt expliquée par les entreprises ayant une profitabilité forte que par celles dont la profitabilité est faible. Enfin, un coefficient CMA indique que l'excès de performance du portefeuille provient des entreprises investissant de manière conservative plutôt que de manière agressive.

II- Présentations des résultats

Il est intéressant d'observer le comportement des facteurs selon le marché. Certains peuvent être toujours significatifs (peu importe le second facteur avec lequel ils sont testés) dans un marché mais ne jamais parvenir à être significatif sur un autre marché.

Figure 23 : Statuts des facteurs selon le marché (cf. Figure 19 pour les sigles)

Statuts	*Eurostoxx 50*	*Nasdaq 100*	*Nikkei 225*
Toujours significatifs	Mkt-RF, SMB, HML, RMW, CMA, OCS, WR	Mkt-RF, SMB, HML, CMA	Mkt-RF, HML, RMW, CMA, WML, OCS, WR
Parfois significatifs	WML, SMA_C, SMA_V, EMA_C, EMA_V, WMA_C, MACD_C, MACD_V, RSI, DMI, BB, Fibo	EMA_V	SMA_C, EMA_C, WMA_C, MACD_C, RSI
Jamais significatifs	WMA_V, OBV	RMW, WML, SMA_C, SMA_V, EMA_C, WMA_C, WMA_V, MACD_C, MACD_V, OCS, RSI, WR, OBV, DMI, BB, Fibo	SMB, SMA_V, EMA_V, WMA_V, MACD_V, OBV, DMI, BB, Fibo

Ce tableau nous permet de mettre en lumière les spécificités des facteurs et des marchés. On remarque par exemple à quel point les facteurs de Fama & French sont puissants dans les trois marchés, sauf pour le SMB qui reste toujours assez éloigné du seuil de significativité sur le marché nippon. Sur l'Eurostoxx, la grande majorité de nos facteurs sont significatifs alors que très peu le sont sur le Nasdaq (WML de Carhart et RMW de FF compris). Grâce à la figure 24, on peut comprendre que le modèle à trois facteurs standard explique déjà la quasi-totalité de des rendements du Nasdaq et qu'il est donc difficile d'expliquer la très faible partie restante. Les figures 25 et 26 nous permettent d'observer cette absence de génération de R^2 supplémentaires par le modèle FF5 et par un modèle quelconque construit à partir des facteurs créés dans la partie 2.

Figure 24 : Résultats du modèle FF3 sur le marché américain

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.968
Model:                            OLS   Adj. R-squared:                  0.968
Method:                 Least Squares   F-statistic:                     4711.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        16:52:59   Log-Likelihood:                 2026.6
No. Observations:                 464   AIC:                            -4045.
Df Residuals:                     460   BIC:                            -4029.
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0100   8.69e-05    115.086      0.000       0.010       0.010
SMB            0.0007      0.000      3.615      0.000       0.000       0.001
HML           -0.0035      0.000    -26.620      0.000      -0.004      -0.003
Bias       -2.569e-05      0.000     -0.179      0.858      -0.000       0.000
==============================================================================
Omnibus:                       29.398   Durbin-Watson:                   2.073
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              104.472
Skew:                           0.031   Prob(JB):                     2.06e-23
Kurtosis:                       5.324   Cond. No.                         2.59
==============================================================================
```

Figure 25 : Résultats du modèle FF5 sur le marché américain

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.969
Model:                            OLS   Adj. R-squared:                  0.968
Method:                 Least Squares   F-statistic:                     2842.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        16:52:59   Log-Likelihood:                 2028.9
No. Observations:                 464   AIC:                            -4046.
Df Residuals:                     458   BIC:                            -4021.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0100   9.01e-05    110.659      0.000       0.010       0.010
SMB            0.0006      0.000      2.743      0.006       0.000       0.001
HML           -0.0032      0.000    -13.937      0.000      -0.004      -0.003
RMW        -5.429e-06      0.000     -0.014      0.989      -0.001       0.001
CMA           -0.0009      0.000     -1.997      0.046      -0.002   -1.48e-05
Bias       -1.436e-05      0.000     -0.100      0.920      -0.000       0.000
==============================================================================
Omnibus:                       31.287   Durbin-Watson:                   2.074
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              113.950
Skew:                           0.094   Prob(JB):                     1.80e-25
Kurtosis:                       5.421   Cond. No.                         6.30
==============================================================================
```

Figure 26 : Résultats du modèle avec les facteurs WMA_C et OBV sur le marché américain

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.969
Model:                            OLS   Adj. R-squared:                  0.968
Method:                 Least Squares   F-statistic:                     2818.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        16:52:59   Log-Likelihood:                 2027.0
No. Observations:                 464   AIC:                            -4042.
Df Residuals:                     458   BIC:                            -4017.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0100   8.71e-05    114.759      0.000       0.010       0.010
SMB            0.0007      0.000      3.609      0.000       0.000       0.001
HML           -0.0035      0.000    -26.457      0.000      -0.004      -0.003
Close_WMA      0.0004      0.001      0.427      0.669      -0.001       0.002
OBV           -0.0006      0.001     -0.552      0.581      -0.003       0.001
Bias           0.0002      0.000      0.645      0.519      -0.001       0.001
==============================================================================
Omnibus:                       29.383   Durbin-Watson:                   2.077
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              103.474
Skew:                           0.054   Prob(JB):                     3.40e-23
Kurtosis:                       5.311   Cond. No.                         24.7
==============================================================================
```

Enfin, il m'a paru intéressant de tester les facteurs de Fama et French voués au marché européen (extraits du fichier « Europe_5_Factors_Daily.csv ») sur le portefeuille Nikkei 225 afin de constater une différence structurelle entre les marchés.

Figure 27 : Résultats des facteurs européens de Fama & French sur le Nikkei 225

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.216
Model:                            OLS   Adj. R-squared:                  0.210
Method:                 Least Squares   F-statistic:                     34.46
Date:                Mon, 29 Nov 2021   Prob (F-statistic):           3.92e-31
Time:                        17:07:52   Log-Likelihood:                 1883.6
No. Observations:                 630   AIC:                            -3755.
Df Residuals:                     624   BIC:                            -3728.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0053      0.001     10.437      0.000       0.004       0.006
SMB            0.0045      0.001      3.588      0.000       0.002       0.007
HML            0.0006      0.001      0.447      0.655      -0.002       0.003
RMW            0.0013      0.002      0.517      0.605      -0.004       0.006
CMA            0.0050      0.002      2.055      0.040       0.000       0.010
Bias          -0.0001      0.000     -0.297      0.766      -0.001       0.001
==============================================================================
Omnibus:                       47.907   Durbin-Watson:                   2.369
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              208.371
Skew:                          -0.116   Prob(JB):                     5.66e-46
Kurtosis:                       5.808   Cond. No.                         7.47
==============================================================================
```

On constate que les facteurs européens expliquent 1/5 de l'excédent de rentabilité du portefeuille nippon. Seules les primes de risque de marché $\left(R_m - R_f\right)$ et de taille (SMB), qui sont pourtant calculés sur des données européennes, sont significatives. Une analyse de marché et une analyse des comportements des investisseurs poussées pourraient permettre d'expliquer ces résultats intéressants.

III- Résultats et sélection du meilleur modèle par marchés

Les trois premiers facteurs de Fama et French sont connus pour être très robustes et nous venons de le vérifier. Nous allons voir que ce n'est pas le cas des deux derniers. Il est intéressant de voir qu'ils peuvent être mis dans l'ombre par des facteurs loin d'être les plus complexes que nous ayons construit.

Figure 28 : Comparaison des résultats de Fama & French et de nos modèles

Vous pouvez observer tous les résultats détaillés ci-dessous dans les figures 29 à 37.

Modèles	*Eurostoxx 50*	*Nasdaq 100*	*Nikkei 225*
FF 3 facteurs	$R^2 = 0{,}824$	$R^2 = 0{,}968$	$R^2 = 0{,}822$
FF 5 facteurs	$R^2 = 0{,}827$	$R^2 = 0{,}969$	$R^2 = 0{,}832$
Meilleur modèle :	$R^2 = 0{,}852$	$R^2 = 0{,}969$	$R^2 = 0{,}853$
Facteurs 4 & 5 :	SMA_C & WMA_C	118 des 153 modèles permettent d'y arriver	CMA & WML

Sur l'eurostoxx, ce sont nos facteurs qui sont plus performants. Sur le Nasdaq, on peut rapidement faire le lien entre le fait que beaucoup de facteurs ne soient pas significatifs et le fait que la majorité des modèles créés ont le même résultat que le modèle original. Sur le Nikkei, c'est un mix entre le modèle de Carhart et de Fama & French qui l'emporte.

Figure 29 : Résultats du modèle FF3 sur le marché européen

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.824
Model:                            OLS   Adj. R-squared:                  0.824
Method:                 Least Squares   F-statistic:                     3118.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        17:13:11   Log-Likelihood:                 7726.8
No. Observations:                2001   AIC:                        -1.545e+04
Df Residuals:                    1997   BIC:                        -1.542e+04
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0086      0.000     65.149      0.000       0.008       0.009
SMB           -0.0075      0.000    -24.605      0.000      -0.008      -0.007
HML            0.0018      0.000      8.036      0.000       0.001       0.002
Bias           0.0001      0.000      1.043      0.297      -0.000       0.000
==============================================================================
Omnibus:                      152.914   Durbin-Watson:                   2.011
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              822.825
Skew:                           0.033   Prob(JB):                    2.12e-179
Kurtosis:                       6.141   Cond. No.                         2.91
==============================================================================
```

Figure 30 : Résultats du modèle FF5 sur le marché européen

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.827
Model:                            OLS   Adj. R-squared:                  0.826
Method:                 Least Squares   F-statistic:                     1906.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        17:13:11   Log-Likelihood:                 7742.8
No. Observations:                2001   AIC:                        -1.547e+04
Df Residuals:                    1995   BIC:                        -1.544e+04
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0084      0.000     61.489      0.000       0.008       0.009
SMB           -0.0078      0.000    -25.218      0.000      -0.008      -0.007
HML            0.0037      0.000      9.290      0.000       0.003       0.004
RMW            0.0023      0.001      4.085      0.000       0.001       0.003
CMA           -0.0021      0.001     -3.590      0.000      -0.003      -0.001
Bias        8.947e-05      0.000      0.788      0.431      -0.000       0.000
==============================================================================
Omnibus:                      129.868   Durbin-Watson:                   2.027
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              590.418
Skew:                          -0.005   Prob(JB):                    6.20e-129
Kurtosis:                       5.661   Cond. No.                         6.27
==============================================================================
```

Figure 31 : Résultats du modèle avec les facteurs SMA_C et WMA_C sur le marché européen

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.852
Model:                            OLS   Adj. R-squared:                  0.851
Method:                 Least Squares   F-statistic:                     2290.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        17:13:12   Log-Likelihood:                 7897.2
No. Observations:                2001   AIC:                        -1.578e+04
Df Residuals:                    1995   BIC:                        -1.575e+04
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0071      0.000     48.813      0.000       0.007       0.007
SMB           -0.0073      0.000    -25.933      0.000      -0.008      -0.007
HML            0.0014      0.000      6.751      0.000       0.001       0.002
Close_SMA     -0.0845      0.004    -19.251      0.000      -0.093      -0.076
Close_WMA      0.0844      0.004     19.247      0.000       0.076       0.093
Bias           0.0002      0.000      1.544      0.123   -4.38e-05       0.000
==============================================================================
Omnibus:                      176.837   Durbin-Watson:                   2.269
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1116.124
Skew:                           0.070   Prob(JB):                    4.33e-243
Kurtosis:                       6.656   Cond. No.                         83.8
==============================================================================
```

Figure 32 : Résultats du modèle FF3 sur le marché américain

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.968
Model:                            OLS   Adj. R-squared:                  0.968
Method:                 Least Squares   F-statistic:                     4711.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        17:18:02   Log-Likelihood:                 2026.6
No. Observations:                 464   AIC:                            -4045.
Df Residuals:                     460   BIC:                            -4029.
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0100   8.69e-05    115.086      0.000       0.010       0.010
SMB            0.0007      0.000      3.615      0.000       0.000       0.001
HML           -0.0035      0.000    -26.620      0.000      -0.004      -0.003
Bias       -2.569e-05      0.000     -0.179      0.858      -0.000       0.000
==============================================================================
Omnibus:                       29.398   Durbin-Watson:                   2.073
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              104.472
Skew:                           0.031   Prob(JB):                     2.06e-23
Kurtosis:                       5.324   Cond. No.                         2.59
==============================================================================
```

Figure 33 : Résultats du modèle FF5 sur le marché américain

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.969
Model:                            OLS   Adj. R-squared:                  0.968
Method:                 Least Squares   F-statistic:                     2842.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        17:18:02   Log-Likelihood:                 2028.9
No. Observations:                 464   AIC:                            -4046.
Df Residuals:                     458   BIC:                            -4021.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0100   9.01e-05    110.659      0.000       0.010       0.010
SMB            0.0006      0.000      2.743      0.006       0.000       0.001
HML           -0.0032      0.000    -13.937      0.000      -0.004      -0.003
RMW        -5.429e-06      0.000     -0.014      0.989      -0.001       0.001
CMA           -0.0009      0.000     -1.997      0.046      -0.002   -1.48e-05
Bias       -1.436e-05      0.000     -0.100      0.920      -0.000       0.000
==============================================================================
Omnibus:                       31.287   Durbin-Watson:                   2.074
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              113.950
Skew:                           0.094   Prob(JB):                     1.80e-25
Kurtosis:                       5.421   Cond. No.                         6.30
==============================================================================
```

Figure 34 : Résultats du modèle avec les facteurs RSI et Fibonacci sur le marché américain

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.969
Model:                            OLS   Adj. R-squared:                  0.968
Method:                 Least Squares   F-statistic:                     2817.
Date:                Mon, 29 Nov 2021   Prob (F-statistic):               0.00
Time:                        17:18:04   Log-Likelihood:                 2026.8
No. Observations:                 464   AIC:                            -4042.
Df Residuals:                     458   BIC:                            -4017.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0100   9.01e-05    111.019      0.000       0.010       0.010
SMB            0.0007      0.000      3.585      0.000       0.000       0.001
HML           -0.0035      0.000    -26.315      0.000      -0.004      -0.003
RSI        -5.278e-05      0.000     -0.331      0.741      -0.000       0.000
Fibo          -0.0001      0.000     -0.534      0.594      -0.000       0.000
Bias           0.0001      0.000      0.413      0.680      -0.000       0.001
==============================================================================
Omnibus:                       29.465   Durbin-Watson:                   2.076
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              105.358
Skew:                           0.008   Prob(JB):                     1.32e-23
Kurtosis:                       5.334   Cond. No.                         4.92
==============================================================================
```

Figure 35 : Résultats du modèle FF3 sur le marché japonais

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.822
Model:                            OLS   Adj. R-squared:                  0.821
Method:                 Least Squares   F-statistic:                     963.4
Date:                Mon, 29 Nov 2021   Prob (F-statistic):          4.57e-234
Time:                        17:27:55   Log-Likelihood:                 2350.4
No. Observations:                 630   AIC:                            -4693.
Df Residuals:                     626   BIC:                            -4675.
Df Model:                           3
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0104      0.000     51.443      0.000       0.010       0.011
SMB            0.0002      0.000      0.543      0.587      -0.001       0.001
HML            0.0037      0.000     12.818      0.000       0.003       0.004
Bias          -0.0002      0.000     -0.848      0.397      -0.001       0.000
==============================================================================
Omnibus:                      118.910   Durbin-Watson:                   2.049
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              882.636
Skew:                          -0.610   Prob(JB):                    2.18e-192
Kurtosis:                       8.669   Cond. No.                         2.34
==============================================================================
```

Figure 36 : Résultats du modèle FF5 sur le marché japonais

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.832
Model:                            OLS   Adj. R-squared:                  0.831
Method:                 Least Squares   F-statistic:                     619.8
Date:                Mon, 29 Nov 2021   Prob (F-statistic):          3.03e-239
Time:                        17:27:55   Log-Likelihood:                 2369.4
No. Observations:                 630   AIC:                            -4727.
Df Residuals:                     624   BIC:                            -4700.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0100      0.000     48.563      0.000       0.010       0.010
SMB         9.833e-05      0.000      0.217      0.828      -0.001       0.001
HML            0.0066      0.001     11.798      0.000       0.006       0.008
RMW            0.0024      0.001      2.499      0.013       0.001       0.004
CMA           -0.0044      0.001     -4.740      0.000      -0.006      -0.003
Bias          -0.0002      0.000     -0.976      0.329      -0.001       0.000
==============================================================================
Omnibus:                      102.815   Durbin-Watson:                   2.066
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              591.097
Skew:                          -0.572   Prob(JB):                    4.41e-129
Kurtosis:                       7.605   Cond. No.                         5.58
==============================================================================
```

Figure 37 : Résultats du modèle avec les facteurs CMA et WML sur le marché japonais

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.853
Model:                            OLS   Adj. R-squared:                  0.852
Method:                 Least Squares   F-statistic:                     725.6
Date:                Mon, 29 Nov 2021   Prob (F-statistic):          3.07e-257
Time:                        13:05:25   Log-Likelihood:                 2411.3
No. Observations:                 630   AIC:                            -4811.
Df Residuals:                     624   BIC:                            -4784.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Mkt-RF         0.0097      0.000     49.415      0.000       0.009       0.010
SMB           -0.0004      0.000     -0.911      0.363      -0.001       0.000
HML            0.0031      0.001      5.881      0.000       0.002       0.004
CMA           -0.0043      0.001     -5.240      0.000      -0.006      -0.003
WML           -0.0034      0.000     -9.790      0.000      -0.004      -0.003
Bias          -0.0004      0.000     -1.661      0.097      -0.001    6.42e-05
==============================================================================
Omnibus:                      120.259   Durbin-Watson:                   2.158
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              860.162
Skew:                          -0.633   Prob(JB):                    1.65e-187
Kurtosis:                       8.583   Cond. No.                         5.43
==============================================================================
```

Conclusion

L'objectif de départ était de trouver des facteurs plus significatifs que ceux ajouter par Fama & French à leur modèle en 2015. Cet objectif a été réalisé. Par ailleurs, nous avons pu étudier la puissance marginale des facteurs RMW et CMA par rapport aux trois premiers. Ils expliquent tout au plus un pourcent de l'excédent de rentabilité supplémentaire, là où nos facteurs peuvent expliquer plus de trois pourcents supplémentaires.

Par la suite, il pourrait être intéressant de développer encore de nouveaux facteurs à tester sur de nouveaux portefeuilles. Nous pourrions créer des facteurs basés sur la liquidité ou encore sur d'autres indicateurs comptables dans la même logique que Fama & French. Aussi, il pourrait être désirable pour les investisseurs de connaître les raisons pour lesquelles certains facteurs fonctionnent mieux dans un marché qu'un autre. Il est probable que nous pourrions apprendre beaucoup sur les spécificités des marchés et leur structure.

Bibliographie

Perold, A. F. (2004). The capital asset pricing model. Journal of economic perspectives, 18(3), 3-24.

Eraslan, V. (2013). Fama and French three-factor model: Evidence from Istanbul stock exchange. Business and Economics Research Journal, 4(2), 11.

Yang, Q., Li, L., Zhu, Q., & Mizrach, B. (2017). Analysis of US sector of services with a new Fama-French 5-factor model. Applied Mathematics, 8(9), 1307-1319.

Blanco, B. (2012). The use of CAPM and Fama and French Three Factor Model: portfolios selection. Public and Municipal Finance, 1(2), 61-70.

Bundoo, S. K. (2008). An augmented Fama and French three-factor model: new evidence from an emerging stock market. Applied Economics Letters, 15(15), 1213-1218.

Womack, K. L., & Zhang, Y. (2003). Understanding risk and return, the CAPM, and the Fama-French three-factor model. Available at SSRN 481881.

Hou, K., Xue, C., & Zhang, L. (2017). A comparison of new factor models. Fisher college of business working paper, (2015-03), 05.

Feng, G., Giglio, S., & Xiu, D. (2017). Taming the factor zoo. Fama-Miller Working Paper, 24070.

Womack, K. L., & Zhang, Y. (2003). Understanding risk and return, the CAPM, and the Fama-French three-factor model. Available at SSRN 481881.

Harvey, C. R. (1993). Portfolio enhancement using emerging markets and conditioning information. World Bank Discussion Papers, 110-110.

Connor, G., & Korajczyk, R. A. (2010). Factor models in portfolio and asset pricing theory. In Handbook of portfolio construction (pp. 401-418). Springer, Boston, MA.

Jagannathan, R., & McGrattan, E. R. (1995). The CAPM debate. Federal Reserve Bank of Minneapolis Quarterly Review, 19(4), 2-17.

Sonubi, Y. (2019). Size and Value Effect: Application of the Fama French Three-Factor Model in the Irish Stock Market (Doctoral dissertation, Dublin, National College of Ireland).

Annexes

Annexe 1 : Fichier input_data.py récupérant les données et calculant les rendements logarithmiques et la volatilité des composants et de l'indice de référence.

```
from typing import Union, List
import yfinance as yf
import numpy as np
import pandas as pd


class Analysis:

    """ Choose the market: europe, america or japan """

    BENCHMARK_TICKER = None
    PORTFOLIO_COMPOSITION = None

    def __init__(self, market='japan'):

        if market == 'europe':
            Analysis.BENCHMARK_TICKER = "^STOXX50E"  # Eurostoxx 50
            Analysis.PORTFOLIO_COMPOSITION = list(['ASML.AS', 'MC.PA', 'SAP.DE', 'LIN.DE', 'SIE.DE', 'SAN.PA', 'OR.PA',
                                                   'SU.PA', 'ALV.DE', 'AIR.PA', 'AI.PA', 'BNP.PA', 'DAI.DE', 'BAS.DE',
                                                   'DTE.DE', 'SAN.MC', 'DPW.DE', 'DG.PA', 'ENEL.MI', 'IBE.MC', 'ADS.DE',
                                                   'EL.PA', 'CS.PA', 'IFX.DE', 'INGA.AS', 'BAYN.DE', 'KER.PA', 'ISP.MI',
                                                   'SAF.PA', 'ABI.BR', 'RI.PA', 'BN.PA', 'BBVA.MC', 'ITX.MC', 'PHIA.AS',
                                                   'STLA.MI', 'VOW.DE', 'MUV2.DE', 'CRG.IR', 'FLTR.IR', 'AD.AS',
                                                   'ENI.MI', 'DB1.DE', 'BMW.DE', 'KNEBV.HE'])

        elif market == 'america':
            Analysis.BENCHMARK_TICKER = "^NDX"  # Nasdaq 100
            Analysis.PORTFOLIO_COMPOSITION = list(['ATVI', 'ADBE', 'AMD', 'ALGN', 'GOOG', 'AMZN', 'AEP', 'AMGN', 'BIIB',
                                                   'ANSS', 'AAPL', 'AMAT', 'ASML', 'TEAM', 'ADSK', 'ADP', 'BIDU', 'ADI',
                                                   'CDNS', 'CDW', 'CERN', 'CHTR', 'CHKP', 'CTAS', 'CSCO', 'CTSH', 'EA',
                                                   'CPRT', 'COST', 'CSX', 'DXCM', 'DOCU', 'DLTR', 'KDP', 'EBAY', 'ILMN',
                                                   'EXC', 'FB', 'FAST', 'FISV', 'FOX', 'FOXA', 'GILD', 'IDXX', 'CMCSA',
```

```
                                                  'INCY', 'INTC', 'INTU', 'ISRG', 'JD', 'KLAC', 'KHC', 'LRCX', 'LULU',
                                                  'MAR', 'MRVL', 'MTCH', 'MELI', 'MCHP', 'MU', 'MSFT', 'MRNA', 'MDLZ',
                                                  'MNST', 'NTES', 'NFLX', 'NVDA', 'NXPI', 'ORLY', 'OKTA', 'PCAR',
                                                  'PYPL', 'PTON', 'PEP', 'PDD', 'BKNG', 'QCOM', 'REGN', 'ROST', 'SGEN',
                                                  'SIRI', 'SWKS', 'SPLK', 'SBUX', 'SNPS', 'TSLA', 'TXN', 'TMUS',
                                                  'VRSK', 'VRTX', 'WBA', 'WDAY', 'XEL', 'PAYX', 'VRSN'])

    elif market == 'japan':
        Analysis.BENCHMARK_TICKER = "^N225"  # Nikkei 225
        Analysis.PORTFOLIO_COMPOSITION = list(['4151.T', '4502.T', '4503.T', '4506.T', '4507.T', '4519.T', '4523.T',
                                                  '4568.T', '4578.T', '6479.T', '6501.T', '6503.T', '6504.T', '6506.T',
                                                  '6645.T', '6674.T', '6701.T', '6702.T', '6703.T', '6724.T', '6752.T',
                                                  '6753.T', '6758.T', '6762.T', '6770.T', '6841.T', '6857.T', '6861.T',
                                                  '6902.T', '6952.T', '6954.T', '6971.T', '6976.T', '6981.T', '7735.T',
                                                  '7751.T', '7752.T', '8035.T', '7201.T', '7202.T', '7203.T', '7205.T',
                                                  '7211.T', '7261.T', '7267.T', '7269.T', '7270.T', '7272.T', '4543.T',
                                                  '4902.T', '7731.T', '7733.T', '7762.T', '9432.T', '9433.T', '9434.T',
                                                  '9613.T', '9984.T', '7186.T', '8303.T', '8304.T', '8306.T', '8308.T',
                                                  '8309.T', '8316.T', '8331.T', '8354.T', '8355.T', '8411.T', '8253.T',
                                                  '8697.T', '8601.T', '8604.T', '8628.T', '8630.T', '8725.T', '8750.T',
                                                  '8766.T', '8795.T', '1332.T', '1333.T', '2002.T', '2269.T', '2282.T',
                                                  '2501.T', '2502.T', '2503.T', '2531.T', '2801.T', '2802.T', '2871.T',
                                                  '2914.T', '3086.T', '3099.T', '3382.T', '8233.T', '8252.T', '8267.T',
                                                  '9983.T', '2413.T', '2432.T', '3659.T', '4324.T', '4689.T', '4704.T',
                                                  '4751.T', '4755.T', '6098.T', '6178.T', '7974.T', '9602.T', '9735.T',
                                                  '9766.T', '1605.T', '3101.T', '3103.T', '3401.T', '3402.T', '3861.T',
                                                  '3863.T', '3405.T', '3407.T', '4004.T', '4005.T', '4021.T', '4042.T',
                                                  '4043.T', '4061.T', '4063.T', '4183.T', '4188.T', '4208.T', '4452.T',
                                                  '4631.T', '4901.T', '4911.T', '6988.T', '5019.T', '5020.T', '5101.T',
                                                  '5108.T', '5201.T', '5202.T', '5214.T', '5232.T', '5233.T', '5301.T',
                                                  '5332.T', '5333.T', '5401.T', '5406.T', '5411.T', '5541.T', '3436.T',
                                                  '5703.T', '5706.T', '5707.T', '5711.T', '5713.T', '5714.T', '5801.T',
                                                  '5802.T', '5803.T', '2768.T', '8001.T', '8002.T', '8015.T', '8031.T',
                                                  '8053.T', '8058.T', '1721.T', '1801.T', '1802.T', '1803.T', '1808.T',
                                                  '1812.T', '1925.T', '1928.T', '1963.T', '5631.T', '6103.T', '6113.T',
                                                  '6301.T', '6302.T', '6305.T', '6326.T', '6361.T', '6367.T', '6471.T',
                                                  '6472.T', '6473.T', '7004.T', '7011.T', '7013.T', '7003.T', '7012.T',
                                                  '7832.T', '7911.T', '7912.T', '7951.T', '3289.T', '8801.T', '8802.T',
                                                  '8804.T', '8830.T', '9001.T', '9005.T', '9007.T', '9008.T', '9009.T',
                                                  '9020.T', '9021.T', '9022.T', '9062.T', '9064.T', '9101.T', '9104.T',
```

```
                                                   '9107.T', '9202.T', '9301.T', '9501.T', '9502.T', '9503.T', '9531.T',
                                                   '9532.T'])

        else:
            raise AttributeError('not applicable market')

        self.market = market
        self.start = "2013-07-31"
        self.end = "2021-07-31"
        self.calendar = None

        self.component_data = None
        self.component_prices = None
        self.component_volumes = None
        self.component_log_returns = None
        self.component_volatilities = None

        self.benchmark_data = None
        self.benchmark_prices = None
        self.benchmark_volumes = None
        self.benchmark_high = None
        self.benchmark_low = None
        self.benchmark_log_returns = None
        self.benchmark_volatility = None

    """ Calendar """

    def __compute_calendar(self, series: List[Union[pd.Series, pd.DataFrame]]):
        """
        Parameters
        ----------
        series : List[Union[pd.Series, pd.DataFrame]]
            series is a list of pandas Series or DataFrame

        Returns
        -------
            a list of sorted timestamp
        """
        return sorted(set.intersection(*list(map(lambda x: set(x.index.tolist()), series))))
```

```python
    def set_calendar(self, series: List[Union[pd.Series, pd.DataFrame]]):
        self.calendar = self.__compute_calendar(series)

    """ Component Information """

    def get_component_data(self):
        all_stocks = ""
        for i in range(len(Analysis.PORTFOLIO_COMPOSITION)):
            all_stocks = str(all_stocks) + " " + str(Analysis.PORTFOLIO_COMPOSITION[i])
        temp_data = yf.download(all_stocks, start=self.start, end=self.end)
        self.component_data = temp_data.ffill().dropna().drop('Adj Close', axis=1)

    def get_component_prices(self):
        self.component_prices = self.component_data["Close"]

    def get_component_volumes(self):
        self.component_volumes = self.component_data["Volume"]

    def get_component_logreturns(self):
        if self.component_prices is not None:
            temp_log_returns = np.log((self.component_prices / (self.component_prices.shift(1)))).dropna()
            self.component_log_returns = temp_log_returns[temp_log_returns.index.isin(self.calendar)]
        else:
            raise TypeError(f'self.component_prices cannot be NoneType, it must be an instance of {type(pd.DataFrame)}')

    def get_component_volatilities(self):
        self.component_volatilities = (self.component_log_returns - self.component_log_returns.mean()) ** 0.5

    """ Index Information """

    def get_benchmark_data(self):
        self.benchmark_data = yf.download(Analysis.BENCHMARK_TICKER, start=self.start, end=self.end).drop('Adj Close', axis=1)

    def get_benchmark_level(self):
        self.benchmark_prices = self.benchmark_data["Close"]

    def get_benchmark_volumes(self):
        self.benchmark_volumes = self.benchmark_data["Volume"]

    def get_benchmark_highs(self):
```

```
        self.benchmark_high = self.benchmark_data['High']

    def get_benchmark_lows(self):
        self.benchmark_low = self.benchmark_data['Low']

    def get_benchmark_logreturns(self):
        if self.benchmark_prices is not None:
            temp_log_returns = np.log((self.benchmark_prices / (self.benchmark_prices.shift(1)))).dropna()
            self.benchmark_log_returns = temp_log_returns[temp_log_returns.index.isin(self.calendar)]
        else:
            raise TypeError(f'self.benchmark_prices cannot be NoneType, it must be an instance of {type(pd.DataFrame)}')

    def get_benchmark_volatility(self):
        self.benchmark_volatility = (self.benchmark_log_returns - self.benchmark_log_returns.mean()) ** 0.5
```

Annexe 2 : Fichier portfolios.py construisant des trois portefeuilles à partir des données du fichier précédent « input_data.py ».

```
from input_data import Analysis
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from statsmodels.tsa.tsatools import lagmat


class PortfolioConstruction(Analysis):
    BASIS = 100

    def __init__(self):
        super().__init__()
        self.get_component_data()
        self.get_component_prices()
        self.get_component_volumes()

        self.get_benchmark_data()
        self.get_benchmark_level()
        self.get_benchmark_volumes()
        self.get_benchmark_highs()
        self.get_benchmark_lows()

        self.set_calendar([self.benchmark_prices, self.component_prices])

        self.get_component_logreturns()
        self.get_component_volatilities()
        self.get_benchmark_logreturns()
        self.get_benchmark_volatility()

        self.weights = None
        self.portfolio_returns = None
        self.portfolio_basis_value = None
        self.benchmark_basis_value = None

    def benchmark_basis_calculation(self):
        bench_basis = (self.benchmark_prices / self.benchmark_prices[0])*PortfolioConstruction.BASIS
        self.benchmark_basis_value = bench_basis
```

```python
    def comp_weights(self):
        raise NotImplementedError

    def portfolio_ret(self):
        raise NotImplementedError

    def portfolio_basis_calculation(self):
        raise NotImplementedError

    def bench_vs_index(self):
        raise NotImplementedError

    def compute_levels(self):
        self.comp_weights()
        self.portfolio_ret()
        self.portfolio_basis_calculation()
        # self.benchmark_basis_calculation()
        # self.bench_vs_index()


class EquallyWeighted(PortfolioConstruction):

    def __init__(self):
        super().__init__()

    def comp_weights(self):
        self.weights = np.ones((len(self.component_log_returns), len(self.PORTFOLIO_COMPOSITION))) * 1 / len(
            self.PORTFOLIO_COMPOSITION)

    def portfolio_ret(self):
        self.portfolio_returns = (self.weights * self.component_log_returns.values).sum(axis=1)

    def portfolio_basis_calculation(self):
        final_calendar = self.calendar
        temp_final_mat = np.c_[
            self.portfolio_returns, lagmat(self.portfolio_returns, maxlag=len(self.portfolio_returns) - 1)]
        pf_basis_value = np.where(temp_final_mat == 0, 1, temp_final_mat).prod(axis=1) * PortfolioConstruction.BASIS
        pf_basis_value = pd.DataFrame(np.r_[PortfolioConstruction.BASIS, pf_basis_value], index=final_calendar,
                                      columns=["Index Value"])
```

```python
        self.portfolio_basis_value = pf_basis_value

    def bench_vs_index(self):
        comp_levels = self.component_prices[self.component_prices.index.isin(self.calendar)]
        pf_level = (self.weights * comp_levels[1:]).sum(axis=1)
        pf_basis = pd.DataFrame((pf_level / pf_level[0]) * PortfolioConstruction.BASIS)

        plt.plot(self.benchmark_basis_value, color='orange', label='Bench Value')
        plt.plot(pf_basis, color='blue', label='Portfolio Value')
        plt.legend(loc='upper left', fontsize=12)
        plt.title('Bench vs Portfolio')
        plt.show()


class ReturnMomentum(PortfolioConstruction):

    def __init__(self):
        super().__init__()

    def comp_weights(self, criteria: int = 3):
        self.weights = self.component_log_returns.apply(
            lambda x: (2 * (x > x.median()) - 1) * 1 / criteria, axis=0).shift(1).fillna(0)

    def portfolio_ret(self):
        self.portfolio_returns = (pd.DataFrame(self.weights) * self.component_log_returns).sum(axis=1) + 1

    def portfolio_basis_calculation(self):
        final_calendar = self.calendar
        temp_final_mat = np.c_[
            self.portfolio_returns, lagmat(self.portfolio_returns, maxlag=len(self.portfolio_returns) - 1)]
        pf_basis_value = np.where(temp_final_mat == 0, 1, temp_final_mat).prod(axis=1) * PortfolioConstruction.BASIS
        pf_basis_value = pd.DataFrame(np.r_[PortfolioConstruction.BASIS, pf_basis_value], index=final_calendar,
                                      columns=["Index Value"])
        self.portfolio_basis_value = pf_basis_value

    def bench_vs_index(self):
        comp_levels = self.component_prices[self.component_prices.index.isin(self.calendar)]
        # pf_level = (1+(self.weights * comp_levels).sum(axis=1)).cumprod()
        pf_level = (self.weights * comp_levels[1:]).sum(axis=1)
        pf_basis = pd.DataFrame((pf_level / pf_level[0]) * PortfolioConstruction.BASIS)
```

```python
        plt.plot(self.benchmark_basis_value, color='orange', label='Bench Value')
        plt.plot(pf_basis, color='blue', label='Portfolio Value')
        plt.legend(loc='upper left', fontsize=12)
        plt.title('Bench vs Portfolio')
        plt.show()


class LowVolMomentum(PortfolioConstruction):

    def __init__(self):
        super().__init__()

    def comp_weights(self, criteria: int = 3):
        self.weights = self.component_volatilities.apply(
            lambda x: (2 * (x < x.median()) - 1) * 1 / criteria, axis=1).shift(1).fillna(0)

    def portfolio_ret(self):
        self.portfolio_returns = (pd.DataFrame(self.weights) * self.component_log_returns).sum(axis=1) + 1

    def portfolio_basis_calculation(self):
        final_calendar = self.calendar
        temp_final_mat = np.c_[
            self.portfolio_returns, lagmat(self.portfolio_returns, maxlag=len(self.portfolio_returns) - 1)]
        pf_basis_value = np.where(temp_final_mat == 0, 1, temp_final_mat).prod(axis=1) * PortfolioConstruction.BASIS
        pf_basis_value = pd.DataFrame(np.r_[PortfolioConstruction.BASIS, pf_basis_value], index=final_calendar,
                                      columns=["Index Value"])
        self.portfolio_basis_value = pf_basis_value

    def bench_vs_index(self):
        comp_levels = self.component_prices[self.component_prices.index.isin(self.calendar)]
        pf_level = (self.weights * comp_levels[1:]).sum(axis=1)
        pf_basis = pd.DataFrame((pf_level / pf_level[0]) * PortfolioConstruction.BASIS)

        plt.plot(self.benchmark_basis_value, color='orange', label='Bench Value')
        plt.plot(pf_basis, color='blue', label='Portfolio Value')
        plt.legend(loc='upper left', fontsize=12)
        plt.title('Bench vs Portfolio')
        plt.show()
```

Annexe 3 : Fichier factors.py récupérant les facteurs de Fama & French et construisant les 18 nouveaux facteurs.

```
from portfolios import PortfolioConstruction

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
plt.style.use('fivethirtyeight')


class FactorConstructions(PortfolioConstruction):

    def __init__(self):
        super().__init__()
        self.get_benchmark_data()
        self.get_benchmark_level()
        self.get_benchmark_volumes()
        self.get_benchmark_highs()
        self.get_benchmark_lows()

        if self.market == 'europe':
            self.factors = "Europe_5_Factors_Daily.csv"
            self.carhart_momentum = "Europe_MOM_Factor_Daily.csv"

        elif self.market == 'america':
            self.factors = "North_America_5_Factors_Daily.csv"
            self.carhart_momentum = "North_America_MOM_Factor_Daily.csv"

        elif self.market == 'japan':
            self.factors = "Japan_5_Factors_Daily.csv"
            self.carhart_momentum = "Japan_MOM_Factor_Daily.csv"

        else:
            AttributeError('for which market do you need FF factors?')

    @staticmethod
    def normalize(x: pd.Series):
        return (x - x.mean()) / x.std()

    """ Fama & French factors """
```

```python
    def get_famafrench_factors(self):
        _raw = pd.read_csv(self.factors, index_col=0, parse_dates=True, skiprows=5).truncate(before=self.start,
                                                                                             after=self.end)
        if self.component_log_returns is None:
            raise TypeError("self.component_log_returns is None")
        index_ = _raw[_raw.index.isin(self.component_log_returns.index)]
        _ = index_.pop("RF")
        return index_

    def get_carhart_momentum(self):
        _raw = pd.read_csv(self.carhart_momentum, index_col=0, parse_dates=True, skiprows=5).truncate(before=self.start,
                                                                                                      after=self.end)
        if self.component_log_returns is None:
            raise TypeError("self.component_log_returns is None")
        index_ = _raw[_raw.index.isin(self.component_log_returns.index)]
        return index_

    """ Trend following factors Moving Averages: can be run on prices and volumes """

    # Simple Moving Average
    def sma(self, criteria='Close', slow=14, fast=3):  # the criteria can also be 'Volume'
        slow_table = pd.DataFrame(self.benchmark_data[criteria].rolling(window=slow).mean()).rename(
            columns={criteria: criteria + '_slowSMA'})
        fast_table = pd.DataFrame(self.benchmark_data[criteria].rolling(window=fast).mean()).rename(
            columns={criteria: criteria + '_fastSMA'})
        table = pd.concat([slow_table, fast_table], join='inner', axis=1)

        # Create the time series of the factor
        time_series = pd.DataFrame(table[criteria + '_fastSMA'])
        return time_series.apply(self.normalize, axis=0)[criteria + '_fastSMA'].fillna(0).rename(criteria + '_SMA')

    # Exponential Moving Average
    def ema(self, criteria='Close', slow=14, fast=3):  # criteria can be 'Volume' also
        slow_table = pd.DataFrame(self.benchmark_data[criteria].ewm(span=slow, adjust=False).mean()).rename(
            columns={criteria: criteria + '_slowEMA'})
        fast_table = pd.DataFrame(self.benchmark_data[criteria].ewm(span=fast, adjust=False).mean()).rename(
            columns={criteria: criteria + '_fastEMA'})
        table = pd.concat([slow_table, fast_table], join='inner', axis=1)
```

```python
        # Create the time series of the factor
        time_series = pd.DataFrame(table[criteria + '_fastEMA'] - table[criteria + '_slowEMA'])
        return time_series.apply(self.normalize, axis=0)[0].fillna(0).rename(criteria + '_EMA')

    # Weighted Moving Average
    def wma(self, criteria='Close', slow=14, fast=3):  # criteria can be 'Volume' also
        slow_weights = np.arange(1, slow+1)
        slow_table = self.benchmark_data[criteria].rolling(window=slow).apply(
            lambda prices: np.dot(prices, slow_weights) / slow_weights.sum(), raw=True)

        fast_weights = np.arange(1, fast+1)
        fast_table = self.benchmark_data[criteria].rolling(window=fast).apply(
            lambda prices: np.dot(prices, fast_weights) / fast_weights.sum(), raw=True)

        all_wma = pd.concat([slow_table, fast_table], join='inner', axis=1)
        all_wma.columns = [criteria + '_slowWMA', criteria + '_fastWMA']

        # Create the time series of the factor
        time_series = pd.DataFrame(all_wma[criteria + '_fastWMA'])
        return time_series.apply(self.normalize, axis=0)[criteria + '_fastWMA'].fillna(0).rename(criteria + '_WMA')

    def macd(self, criteria='Close', slow=21, fast=9, smooth=9):
        exp1 = self.benchmark_data[criteria].ewm(span=fast, adjust=False).mean()
        exp2 = self.benchmark_data[criteria].ewm(span=slow, adjust=False).mean()
        macd = pd.DataFrame(exp1 - exp2).rename(columns={criteria: criteria + '_macd'})
        signal = pd.DataFrame(macd.ewm(span=smooth, adjust=False).mean()).rename(columns={criteria + '_macd': 'signal'})
        hist = pd.DataFrame(macd[criteria + '_macd'] - signal['signal']).rename(columns={0: 'hist'})
        all_macd = pd.concat([macd, signal, hist], join='inner', axis=1)

        # Create the time series of the factor
        return all_macd.apply(self.normalize, axis=0)[criteria + '_macd'].fillna(0).rename(criteria + '_macd')

    """ Oscillating factors: stochastic oscillator, rsi, w%r, obv, dmi """

    def oscillator(self, k_lookback=14, d_lookback=3):
        high = self.benchmark_high
        low = self.benchmark_low

        lowest_low = low.rolling(k_lookback).min()
        highest_high = high.rolling(k_lookback).max()
```

```python
        k_line = ((self.benchmark_prices - lowest_low) / (highest_high - lowest_low)) * 100
        d_line = k_line.rolling(d_lookback).mean()
        all_osc = pd.concat([k_line, d_line], join='inner', axis=1).rename(columns={0: 'k_line', 1: 'd_line'})

        # Create the time series of the factor
        return all_osc.apply(self.normalize, axis=0)['k_line'].fillna(0).rename('OSC')

    def rsi(self, lookback=14):
        ret = self.benchmark_prices.diff()
        up = []
        down = []
        for i in range(len(ret)):
            if ret[i] < 0:
                up.append(0)
                down.append(ret[i])
            else:
                up.append(ret[i])
                down.append(0)
        up_series = pd.Series(up)
        down_series = pd.Series(down).abs()
        up_ewm = up_series.ewm(com=lookback - 1, adjust=False).mean()
        down_ewm = down_series.ewm(com=lookback - 1, adjust=False).mean()
        rs = up_ewm / down_ewm
        rsi = 100 - (100 / (1 + rs))
        rsi_df = pd.DataFrame(rsi).rename(columns={0: 'rsi'}).set_index(self.benchmark_prices.index)
        rsi_df = rsi_df.dropna()

        # Create the time series of the factor
        return rsi_df.apply(self.normalize, axis=0)['rsi'].fillna(0).rename('RSI')

    def william_r(self, lookback=14):
        high = self.benchmark_high
        low = self.benchmark_low

        highh = high.rolling(lookback).max()
        lowl = low.rolling(lookback).min()
        wr = -100 * ((highh - self.benchmark_prices) / (highh - lowl))

        # Create the time series of the factor
        return pd.DataFrame(wr).apply(self.normalize, axis=0)[0].fillna(0).rename('WR')
```

```python
def on_balance_volume(self):  # and obv_ema
    prices = self.benchmark_prices
    volumes = self.benchmark_volumes

    obv = [0]
    for i in range(1, len(prices)):
        if prices[i] > prices[i - 1]:
            obv.append(obv[-1] + volumes[i])
        elif prices[i] < prices[i - 1]:
            obv.append(obv[-1] - volumes[i])
        else:
            obv.append(obv[-1])

    obv = pd.Series(obv, index=prices.index)
    obv_ema = obv.ewm(span=20).mean()

    # Create the time series of the factor
    return pd.DataFrame(obv_ema).apply(self.normalize, axis=0)[0].fillna(0).rename('OBV')

def dmi(self, lookback=14):
    high = self.benchmark_high
    low = self.benchmark_low

    plus_dm = high.diff()
    minus_dm = low.diff()
    plus_dm[plus_dm < 0] = 0
    minus_dm[minus_dm > 0] = 0

    tr1 = pd.DataFrame(high - low)
    tr2 = pd.DataFrame(abs(high - self.benchmark_prices.shift(1)))
    tr3 = pd.DataFrame(abs(low - self.benchmark_prices.shift(1)))
    frames = [tr1, tr2, tr3]
    tr = pd.concat(frames, axis=1, join='inner').max(axis=1)
    atr = tr.rolling(lookback).mean()

    plus_di = 100 * (plus_dm.ewm(alpha=1 / lookback).mean() / atr)
    minus_di = abs(100 * (minus_dm.ewm(alpha=1 / lookback).mean() / atr))
    dx = (abs(plus_di - minus_di) / abs(plus_di + minus_di)) * 100
    # adx = ((dx.shift(1) * (lookback - 1)) + dx) / lookback
```

```python
        # adx_smooth = adx.ewm(alpha=1 / lookback).mean()

        # Create the time series of the factor
        return pd.DataFrame(dx).apply(self.normalize, axis=0)[0].fillna(0).rename('DMI')

    """ Volatility factors: bollinger bands, fibonacci"""

    def bollinger_bands(self, lookback=14):
        rolling_returns = self.benchmark_prices.rolling(window=lookback)
        sma = rolling_returns.mean()
        std = rolling_returns.std()
        # upper_bb = sma + std * 2
        lower_bb = sma - std * 2

        # Create the time series of the factor
        time_series = pd.DataFrame(lower_bb)  # upper_bb - lower_bb
        return time_series.apply(self.normalize, axis=0)['Close'].fillna(0).rename('BB')

    def fibonacci(self):
        data = self.benchmark_data
        ratios = [0, 0.236, 0.382, 0.5, 0.618, 0.786, 1]

        period_length = 20
        fibonacci_levels = [[np.nan] * len(ratios)] * period_length

        for i in range(period_length, len(data)):

            highest_swing = -1
            lowest_swing = -1

            daily_df = data[i-period_length:i]
            df_high = daily_df['High']
            df_low = daily_df['Low']

            for j in range(1, df_high.shape[0] - 1):
                if df_high[j] > df_high[j - 1] and df_high[j] > df_high[j + 1] and (
                        highest_swing == -1 or df_high[j] > df_high[highest_swing]):
                    highest_swing = j

                if df_low[j] < df_low[j - 1] and df_low[j] < df_low[j + 1] and (
```

```python
                        lowest_swing == -1 or df_low[j] < df_low[lowest_swing]):
                    lowest_swing = j

            daily_levels = []
            max_level = df_high[highest_swing]
            min_level = df_low[lowest_swing]

            for ratio in ratios:
                if highest_swing > lowest_swing:  # Uptrend
                    daily_levels.append(max_level - (max_level - min_level) * ratio)  # ratio*100 ?
                else:  # Downtrend
                    daily_levels.append(min_level + (max_level - min_level) * ratio)  # ratio*100 ?

            fibonacci_levels.append(daily_levels)

        # Create the time series of the factor
        time_series = pd.DataFrame(fibonacci_levels, columns=ratios, index=data.index)
        return time_series.apply(self.normalize, axis=0)[0.618].fillna(0).rename('Fibo')


# calculate factor combinations (by 2) + Momentum, RMW, CMA
class Combinations(FactorConstructions):

    def __init__(self):
        super().__init__()
        self.three_factors = self.get_famafrench_factors().drop(['RMW', 'CMA'], axis=1)

        # FF-based factors
        self.rmw = self.get_famafrench_factors()["RMW"]
        self.cma = self.get_famafrench_factors()["CMA"]
        self.wml = self.get_carhart_momentum()["WML"]  # mom

        # tailored trending factors
        self.sma_price = self.sma('Close')
        self.ema_price = self.ema('Close')
        self.wma_price = self.wma('Close')
        self.macd_price = self.macd('Close')

        self.sma_volume = self.sma('Volume')
        self.ema_volume = self.ema('Volume')
```

```
        self.wma_volume = self.wma('Volume')
        self.macd_volume = self.macd('Volume')

        # tailored oscillator factors
        self.osc = self.oscillator()
        self.rsi = self.rsi()
        self.wr = self.william_r()
        self.dmi = self.dmi()
        self.obv = self.on_balance_volume()

        # tailored volatility factors
        self.bb = self.bollinger_bands()
        self.fib = self.fibonacci()

    def five_factor_combinations(self):
        factor_df = []
        factor_list = [self.rmw, self.cma, self.wml, self.sma_price, self.sma_volume, self.ema_price,
                       self.ema_volume, self.wma_price, self.wma_volume, self.macd_price, self.macd_volume,
                       self.osc, self.rsi, self.wr, self.obv, self.dmi, self.bb, self.fib]  # self.w_volume

        factor_list_copy = factor_list.copy()

        couples = []
        dust_bin = []

        for factor in factor_list:
            for factor_copy in factor_list_copy:
                if factor.name != factor_copy.name and factor_copy.name not in dust_bin:
                    couples.append((factor.name, factor_copy.name))
                    factor_df.append(pd.concat([self.three_factors, factor, factor_copy], join='inner', axis=1))
            dust_bin.append(factor.name)

        print(len(couples))
        return factor_df
```

Annexe 4 : Fichier regression.py construisant le modèle de régression et effectuant la régression sur les 153 modèles pour les trois marchés.

```
import pandas as pd
from factors import Combinations

from portfolios import EquallyWeighted, ReturnMomentum, LowVolMomentum
from dataclasses import dataclass
from statsmodels.api import OLS
from statsmodels.regression.linear_model import RegressionResults, RegressionResultsWrapper
import numpy as np


class LinearEstimator(Combinations):
    _bias = "Bias"

    @dataclass(init=False)
    class LinearRegressionResult:
        t_stats: object = None
        p_values: object = None
        coefficients: object = None

        @classmethod
        def from_model(cls, result: RegressionResults):

            _obj = cls()
            object.__setattr__(_obj, "result", result)
            _obj.coefficients = result.params
            _obj.t_stats = result.tvalues
            _obj.p_values = result.pvalues
            return _obj

        def to_csv(self, file_name: str):
            path = "/Users/carlpaulus/OneDrive - EDHEC/Documents/travail/EDHEC/Cours BBA/BBA4/Mémoire/Regression results/"
            to_export = {key: value for key, value in vars(self).items() if key != "result"}
            pd.DataFrame(to_export).to_csv(path + file_name + ".csv")

        def print(self):
            try:
                res = self.__getattribute__("result")
```

```python
                if isinstance(res, RegressionResultsWrapper):
                    print(res.summary())
            except AttributeError:
                raise TypeError("No result in LinearResultRegression")

    def __init__(self, hasconst: bool = True):
        super().__init__()
        self._hasconst = hasconst
        self._model = None
        self._result = None

    @property
    def model(self) -> OLS:
        return self._model

    @property
    def result(self) -> LinearRegressionResult:
        return self._result

    def __call__(self, endog: np.ndarray, exog: np.ndarray):
        if len(endog) == len(exog):

            if self._hasconst:
                exog[self._bias] = 1
            self._model = OLS(endog, exog)
            self._result = self.LinearRegressionResult.from_model(self.model.fit())
        return self.result


# calculate results for each combination (all factors + carhart momentum + RMW + CMA) and FF3, FF4, FF5
class Calculator(LinearEstimator):

    def __init__(self):
        super().__init__()
        self.ew = EquallyWeighted()
        # self.rm = ReturnMomentum()
        # self.lv = LowVolMomentum()

        self.ew.compute_levels()
        # self.rm.compute_levels()
```

```
        # self.lv.compute_levels()

        self.estimator = LinearEstimator()
        self.factors = Combinations().five_factor_combinations()

    def compute_results(self):
        self.estimator(self.ew.portfolio_returns, self.three_factors).print()

        for factor in self.factors:
            print(str(self.market) + ' ' + str(factor.columns[3:5][0]) + ' ' + str(factor.columns[3:5][1]))
            result = self.estimator(self.ew.portfolio_returns, factor)
            # result.to_csv(str(self.market) + '_' + str(factor.columns[3:5][0]) + '_' + str(factor.columns[3:5][1]))
            result.print()

        # the way to run all regressions on all portfolios :
        # portfolios_list = [self.ew, self.rm, self.lv]
        # for portfolio in portfolios_list:
        #     for factor in self.factors:
        #         result = self.estimator(portfolio.portfolio_returns, factor)
        #         # result.to_csv("result " + portfolio.__name__ + " " + factor.__name__)
        #         result.print()
```