

Filali Hamza
Sorba Marianne
Tinsi Laura

ENSAE
2015-2016

PROJET DE PROGRAMMATION:
Les algorithmes de trading



Sommaire :

Introduction	p3.
Partie 1 : Présentation des algorithmes et des stratégies	p4.
A. Le Trend Following	p4.
B. Le Pair Trading	p6.
C. Le Mean Reversion	p8.
Partie 2 : Mise en évidence des spécificités de chaque stratégie	p9.
A. Le Trend Following	p9.
B. Le Pair Trading	p10.
C. Le Mean Reversion	p11.
Conclusion	p15.
Annexe	p16.
Bibliographie	p19.

Introduction :

Nous avons décidé pour ce projet de nous orienter vers le monde de la finance en implémentant des stratégies de trading. Le but était donc d'élaborer un algorithme de gestion de portefeuille via différents modes de décisions. Étant un groupe de 3 élèves, nous avons naturellement décidé de présenter trois stratégies : le Trend Following, le Mean Reversion et le Pair trading. L'objectif ici n'était pas de recopier des stratégies déjà existantes mais de partir de leurs principes de base et d'en proposer notre version. Ainsi, chacun a pris en charge l'implémentation d'un des algorithmes présentés ci dessous.

En outre, si il est fréquent qu'un tel type de projet aboutisse à la mise en évidence d'une "meilleure stratégie", nous avons décidé d'envisager le problème sous un angle différent afin de bénéficier des avantages de chacune des stratégies et nous sommes donc posé le problème suivant :

Quelles sont les caractéristiques propres à chaque stratégie et en quoi la rendent elle plus performante que les autres dans certaines configurations à la fois de constitution de portefeuille et de conjuncture économique ?

En d'autres termes, quand utiliser quelle stratégie ?

Nous avons tenté d'y répondre en 2 parties avec, dans un premier temps, une présentation de chaque stratégie et une description des algorithmes que nous avons implémentés. Puis, dans une seconde partie, l'analyse des atouts et des caractéristiques de chaque stratégie, en nous appuyant notamment sur les résultats empiriques que les algorithmes nous ont permis d'obtenir.

I. Présentation des algorithmes et des stratégies

A) Le Trend Following

1. *Présentation de la stratégie*

a) Le Trend Following :

Le Trend Following est une stratégie financière qui consiste à suivre la tendance du cours d'une action considérée, qu'elle soit haussière ou baissière. Le principe est simple : lorsque le cours de l'action entre dans une tendance haussière, la stratégie nous dit d'acheter, en pariant sur le fait que le cours va continuer à augmenter pendant un certain temps. A l'inverse, lorsque le cours entre dans une tendance baissière, la stratégie nous dit de vendre, afin de pouvoir vendre l'action au prix le plus fort possible.

b) Les outils utiles

Cette stratégie nécessite l'implémentation d'un algorithme permettant de détecter les changements de tendances. Pour cela, nous avons eu recours à la moyenne mobile, et aux bandes de Bollinger. La moyenne mobile est la moyenne des n dernières valeurs journalières du cours passé:

$$MM(n, t) = 1/n * \sum_{k=0}^{n-1} P_{n-k}$$

L'avantage de la moyenne mobile est qu'elle ne prend pas en compte les variations quotidiennes, et permet de mieux visualiser la tendance globale de la courbe financière. La moyenne est en retard par rapport à la série elle-même, car elle prend seulement en compte les valeurs passées. Pour mettre en évidence le début d'une tendance, qu'elle soit haussière ou baissière, nous avons introduit la distance moyenne entre la série et la moyenne mobile :

$$dist(n, t) = \sqrt{1/n \sum_{i=0}^{n-1} (P_t - MM(n, t))^2}.$$

Plus la distance est grande, plus la tendance est marquée : la moyenne du cours de l'action sur les n jours passés est d'autant plus écartée de la valeur du cours actuel. Les bandes de Bollinger sont définies par l'intervalle:

$$[MM(n, t) - \alpha * dist(n, t), MM(n, t) + \alpha * dist(n, t)]$$

où α est un paramètre qui détermine la largeur voulue des Bandes de Bollinger. Pour $\alpha=2$, 95 % des valeurs observées se trouvent statistiquement situées entre les deux bandes extrêmes. Le fait que la série sorte de ce couloir indique un jour de plus forte variation, et très sûrement le début d'une tendance, haussière ou baissière. Pour l'implémentation de cette stratégie, nous avons choisi $\alpha=2$.

2. Présentation de l'algorithme

a) Structure

Ce programme a été implémenté en utilisant les modules *Quandl*, *pylab* et *pandas*. Il comporte une première partie intitulée « *Récupération des données* », dans laquelle nous avons utilisé le module *Quandl* pour récupérer les données qui nous intéressent (le cours de l'action voulu) à partir de la plateforme Quandl. Nous les avons ensuite mises sous forme de dictionnaire de façon à pouvoir les utiliser plus aisément. La deuxième partie est intitulée « *Graphique* ». Elle est composée de l'algorithme permettant de tracer le cours d'une action entre les dates t_1 et t_2 , ainsi que l'algorithme permettant de tracer le rendement de la stratégie entre les dates t_1 et t_2 . La troisième partie est intitulée « *Fonctions* » et est composée des diverses fonctions que nous avons utilisées pour implémenter la stratégie (calcul de la moyenne mobile, des bandes de Bollinger, etc...). La dernière partie est intitulée « *Décision et Rendement* » et est composée de l'algorithme de décision ainsi que l'algorithme de calcul du rendement.

b) Présentation de l'algorithme de décision :

L'algorithme de décision de la stratégie a été implémenté comme suit : lorsque la série dépasse sa bande supérieure de Bollinger, cela signifie que le cours de l'action a commencé à augmenter, et cela prédit donc une tendance haussière : On achète avant que les prix ne soient trop hauts. A l'inverse, quand la série devient plus faible que sa bande inférieure de Bollinger, on vend avant que les prix ne soient trop bas. L'algorithme nous dit donc à chaque date t de :

« Acheter » si la série a dépassé sa bande de Bollinger à la date t .

« Vendre » si la série a dépassé sa bande inférieure de Bollinger à la date t .

« Ne rien faire » si la série est restée dans l'intervalle défini par les deux bandes.

c) Présentation de l'algorithme de rendement :

Pour calculer le rendement de la stratégie, nous disposons de deux compteurs : un compteur « cash » qui nous donne la quantité de cash que nous disposons à la date t , et un compteur « titre » qui nous donne la quantité de titres que nous disposons à la date t . A $t = 0$, nous disposons de 1€ (cash=1 et titre=0). Nous gardons cet euro jusqu'à la date où l'algorithme de décision nous dit d'acheter ($t = t_1$). Nous achetons donc une quantité équivalente à un euro de l'action considérée à la date t_1 . Si le prix de l'action est de P_1 , nous achetons donc une quantité $1/P_1$ de titres. Nous avons donc cash=0 et titre= $1/P_1$, jusqu'à ce que l'algorithme de décision nous dit de vendre en $t = t_2$. A la date t_2 , nous avons donc : cash = P_2/P_1 et titre = 0. Le processus se répète.

Pour chaque date t , l'algorithme de rendement nous renvoie la valeur liquide de nos biens :

$$\text{Valeur liquide} = \text{cash} + P_t * \text{titre}$$

B) Le Pair Trading

1. *Principe de la stratégie*

Le pair trading est une stratégie qui a la particularité de s'intéresser à l'observation des cours de deux actions relativement proches. Cette proximité des actions se traduit par des variations des séries temporelles des cours qui se « suivent » et se mesure par différents tests. Ces tests se basent sur la corrélation des actions mais aussi sur des méthodes de cointégration faisant intervenir la notion de mouvement stochastique que nous n'avons pas encore traité. Ainsi nous avons décidé de nous limiter ici au calcul du coefficient de corrélation entre les deux cours obtenu grâce à la fonction *corrcoef* du module *numpy* pour juger de la pertinence de l'utilisation de cet algorithme par rapport au Mean Reversion et Trend Following.

Cette stratégie se base sur l'observation du cours des rapports des prix des deux actions. L'algorithme que nous avons implémenté ici effectue la stratégie décrite ci dessous.

2. *Présentation et Description de l'algorithme*

a) Présentation du principe de algorithme

Dans leur ensemble, les algorithmes implémentés pour cette stratégie effectuent les actions suivantes. On dispose au départ de $N\text{€}$. On achète deux actions A_1 et A_2 de prix quotidien moyen (moyenne faite sur le prix d'ouverture, de fermeture, de cote la plus haute de la journée et de cote la plus basse) respectifs P_1 et P_2 en quantités $Q_1 = N/2P_1$ et $Q_2 = N/2P_2$. On observe le cours de P_1/P_2 . Tant que celui-ci reste entre les bandes de Bollinger on ne fait rien. Dès que ce cours dépasse la bande haute, on peut supposer que, ces deux actions étant liées; un retour autour de la moyenne mobile devrait s'effectuer de manière assez rapide. Il est à noter que l'on ne regarde pas ici les tendances mais les variations plus rapides ce qui justifie entre autre le choix $\alpha=1$ pour les bandes de Bollinger, contrairement au Trend Following où la contrainte de dépassement est beaucoup plus forte. Ainsi, si P_1/P_2 dépasse la bande haute on va supposer que le prix P_1 de A_1 augmente relativement par rapport à celui de A_2 . De ce fait, on aura que soit le prix de A_1 a augmenté, soit celui de A_2 a diminué. Il convient ici de vendre les actions dont le prix a temporairement et relativement augmenté et d'acheter celles à plus faible coût : le futur retour à la moyenne nous assure le gain. On vend donc toutes nos actions A_1 au prix en vigueur à la fermeture du jour où est constaté le dépassement et on achète alors une quantité $Q_2 = Q_1P_1/P_2$ d'actions A_2 . Tant que le rapport est au dessus de sa moyenne mobile : on ne fait rien. Quand celui-ci ré-atteint sa moyenne mobile on re-diversifie le portefeuille en vendant une quantité $Q_2/2$ d'actions A_2 et en achetant pour une quantité $Q_1 = Q_2P_2/2P_1$ d'actions A_1 . De même, lorsque le rapport descend en dessous de la bande basse de Bollinger, c'est que P_2 a relativement augmenté par rapport à P_1 . On effectue alors les mêmes opérations mais en inversant les rôles joués par A_1 et A_2 .

Les gains alors représentés dans la partie suivante constituent en fait l'évolution en chaque temps de la valeur de notre portefeuille. Ainsi il est facile d'observer la plus value

effectuée en chaque temps donnée par la différence entre la somme investie et la valeur actuelle déflatée (en € de la date d'investissement) de notre portefeuille.

b) Présentation technique des fichiers et d'un point du code

Ce programme a été implémenté grâce aux modules *pyensae*, *numpy*, *datetime* et *math*. Il comporte trois fichiers distincts : *fonctionsdebases*, *algodedecision* et *valeurportefeuille*.

Le fichier *fonctionsdebases* contient les fonctions permettant de calculer les indicateurs et les résultats de base : un prix moyen, les rapports des prix, la moyenne mobile ainsi que les bandes de Bollinger, des fonctions de modifications de dataframe et pour finir le calcul de la volatilité. Le fichier *algodedecision* contient les algorithmes permettant de prendre la décision de vendre ou acheter telle ou telle action de la paire : la fonction « *action* » retourne un *StockPrices* modifié contenant les rapports des prix à chaque date, la fonction « *position* » indique quelle action effectuer lors de la première date de dépassement des bandes de Bollinger qu'elle trouve, et la fonction « *gains* » retourne un tableau permettant de calculer les gains à chaque date. Le fichier *valeurportefeuille* contient la fonction qui retourne le calcul de la valeur du portefeuille à chaque période et l'affiche. Ainsi, l'algorithme principal est contenu dans le fichier *valeurportefeuille* mais fait appel aux fonctions préalablement construites dans les autres fichiers. Chaque instruction de l'algorithme principal est détaillée dans le code en commentaire ainsi que la marche à suivre pour exécuter le programme et pouvoir utiliser la stratégie.

En ce qui concerne le coût d'un des algorithmes du code nous avons choisi de décrire et calculer la complexité du calcul de la moyenne mobile. Afin d'alléger la rédaction du rapport, la description ligne par ligne du code ainsi que les étapes du calcul de la complexité ont été mise directement en commentaire du code et en annexe (Cf. fichier *fonctionsdebase* fonction « *moyenne_mobile* »).

c) Difficultés rencontrées lors de la mise en œuvre de l'algorithme

Un élément de réflexion important sur l'algorithme et qui nous a fait perdre du temps consistait dans le choix exacte de la stratégie (quelle action vendre, à quel moment et dans quelle quantité par rapport à la seconde ? quand fermer la position et revenir à un portefeuille diversifié pour que le gain soit le meilleur possible ?). Nous avons d'ailleurs laissé en commentaire pour les fichiers *algodedecision* et *valeurportefeuille* des versions précédentes des fonctions « *gains* » et « *gainsportefeuille* » que nous avons au fur et à mesure modifiées et améliorées pour arriver au résultat final. Un autre élément a été source de perte de temps ; c'est bien sûr de déboguer le programme notamment la prise en compte du fait que certains jours les prix des actions ne sont pas relevés et donc exclure ces dates des calculs. Ceci explique d'ailleurs l'affichage « trouver une autre date de fin » qui peut apparaître lorsque l'on saisit certains intervalles de temps.

Le principal problème que nous pouvons relever dans l'algorithme à proprement parler (et non la stratégie dont les rendements seront analysés en troisième partie) est sans

doute la lenteur du programme du fait des nombreuses fonctions auxquelles il fait appel et des nombreuses boucles (à titre d'exemple il a fallu plus de 20 minutes pour calculer et afficher les gains de la partie 2). Même si il reste assez lent, nous avons tout de même tenté d'en réduire le temps d'exécution par divers moyens comme par exemple l'allocation des variables en dehors des boucles, l'utilisation d'instruction telles que « s+= ».

C) Le Mean Reversion

1. *Principe de la stratégie*

La stratégie que nous avons établi se base sur un portefeuille d'actions déterminé (d'au moins 3 actions). Chaque mois le programme analysera les performances des différentes actions du portefeuille des « iperformance_window »¹ derniers jours. Le programme décide alors d'acheter les « iselection » actions qui ont les moins bonnes performances lors de ce mois-ci², ces actions sont alors revendues à la fin du mois. Les « iselection » actions qui ont les meilleures performances sont « shortées ».

Par exemple, en choisissant le portefeuille d'actions constitué des 3 actions suivantes: BNP Paribas, Crédit Agricole et Société Générale. En prenant iselection égal à 1 et iperformance_window égal à 20, l'algorithme analyse chaque début de mois l'action qui a eu la meilleure performance durant les 20 derniers jours. Cette action sera "shortée" (on vend 100 unités de prix de cette action et on achète ce montant à la fin du mois). L'action qui a eu la moins bonne performance durant ces 20 derniers jours est achetée pour être vendue le mois suivant.

2. *Justification*

L'intuition de cette stratégie vient d'un constat simple. Si l'on constate que le cours des actions est stationnaire et donc que le cours va osciller autour d'une position d'équilibre, la meilleure stratégie à suivre est d'acheter lorsque le cours vient de baisser et de vendre lorsque le cours monte. En réalité les conditions d'utilisation de notre algorithme sont différentes, nous verrons dans la partie 2 ces conditions. Il est à noter qu'ici contrairement à la stratégie mean reversion classique³, nous n'avons pas cherché à établir nos décisions à l'aide des bandes de Bollinger mais par une comparaison des performances au sein des actions du portefeuille. Cette stratégie est particulièrement pertinente pour des actions appartenant au même secteur économique comme nous le verrons dans la partie 2.

3. *Difficultés rencontrées*

Hormis le débogage du programme, il a été difficile de dégager les facteurs influençant le rendement d'un portefeuille. Il a fallu essayer de nombreuses combinaisons de portefeuille, faire varier de nombreux paramètres pour s'assurer de la véracité de nos conditions mises en évidence en seconde partie.

¹ cf. le code python de l'algorithme mean reversion, , nous avons choisi dans nos exemples iperformance_window=20 jours, cela a en réalité peu d'importance dans le rendement final

² On en achète pour 100 unités de prix

³ http://www.xavierdupre.fr/site2013/documents/reports/finance_autostrat.pdf

II. Mise en évidence des spécificités de chaque stratégie

A) Le Trend Following

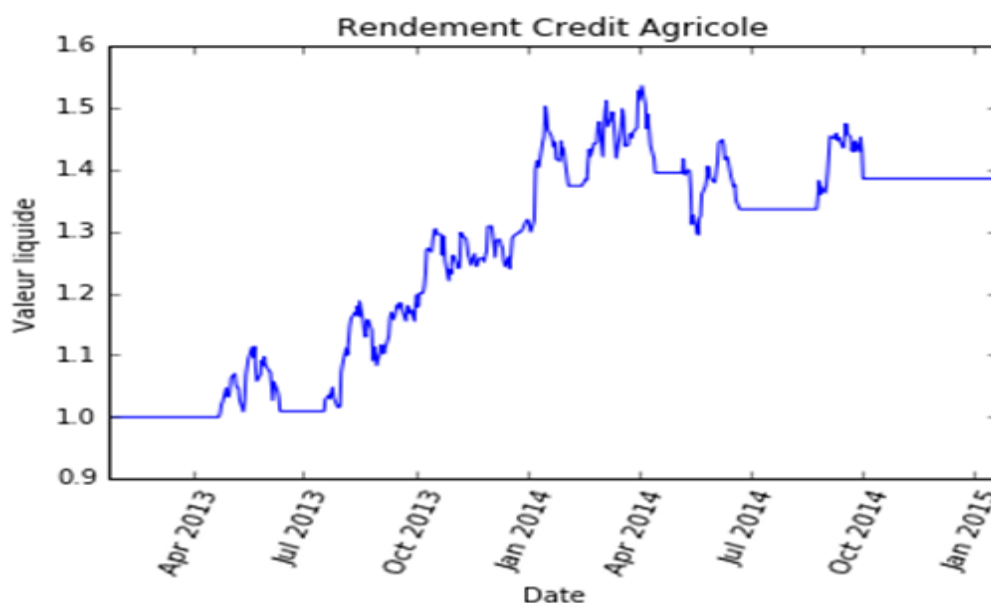
1. *Influence de la volatilité*

L'implémentation de la stratégie s'appuie sur une moyenne mobile de 20 jours. Ainsi, la stratégie détecte bien les tendances qui se situent au dessus de cette valeur. Elle peut toutefois prendre de mauvaises décisions si les variations au sein de la série se font sur une durée proche, ou plus courte. La stratégie Trend following est donc plus adaptée à une action à faible volatilité, puisqu'elle parie sur le fait qu'une tendance haussière ou baissière va continuer si celle-ci est détectée. Pour illustrer l'influence de la volatilité sur la stratégie, nous avons calculé le rendement obtenu sur l'action Apple entre 2012 et 2015, et sur l'action Crédit Agricole entre 2012 et 2015. Les volatilités de ces deux actions sont respectivement de 0.0072 et de 0.0103. On observe un meilleur rendement pour l'action Apple (On arrive à un rendement de 1.7, contre un rendement maximum observé de 1.5 pour Crédit Agricole), ce qui tendrait à confirmer l'influence négative qu'a la volatilité sur le rendement de la stratégie.

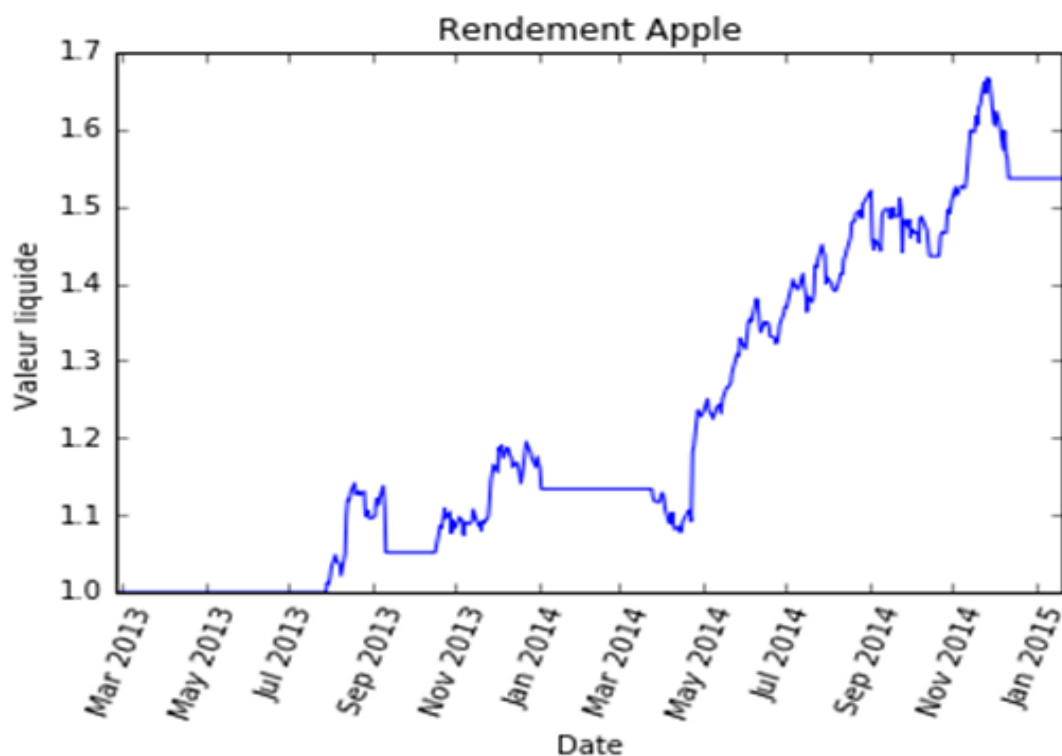
2. *Une stratégie indépendante de la corrélation des actions*

L'analyse des stratégies Mean Reversion et Pair Trading a mis en évidence le fait que ces deux stratégies sont plus adaptées à un portefeuille d'actions corrélées (Cf. parties II.B et II.C). La stratégie Trend Following, quant à elle, ne travaille pas sur un portefeuille d'actions mais sur une unique action. La corrélation des actions n'influe pas sur le rendement d'un portefeuille d'action, dans la mesure où le calcul du rendement se fait individuellement sur chaque action.

Graphique 1: Rendement de l'action Crédit Agricole entre 2012 et 2015



Graphique 2: Rendement de l'action Apple entre 2012 et 2015



B) Le Pair Trading

1. *Les résultats*

Pour mettre en évidence l'utilité et la pertinence du Pair trading nous avons choisi deux paires d'actions différentes :

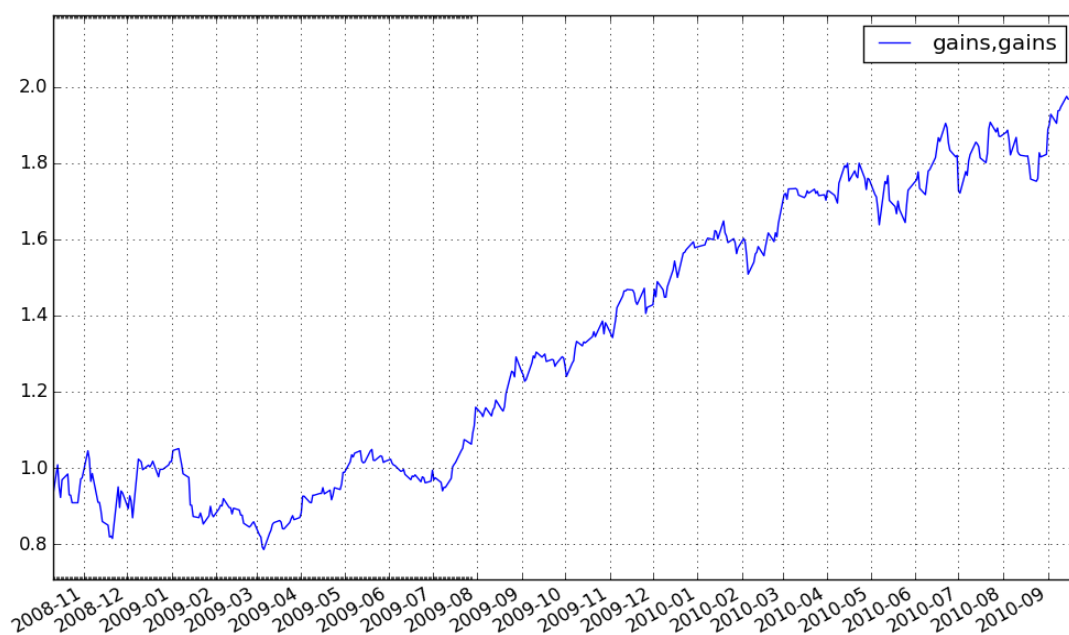
-L'Oréal-LVMH

-L'Oréal-Société Générale

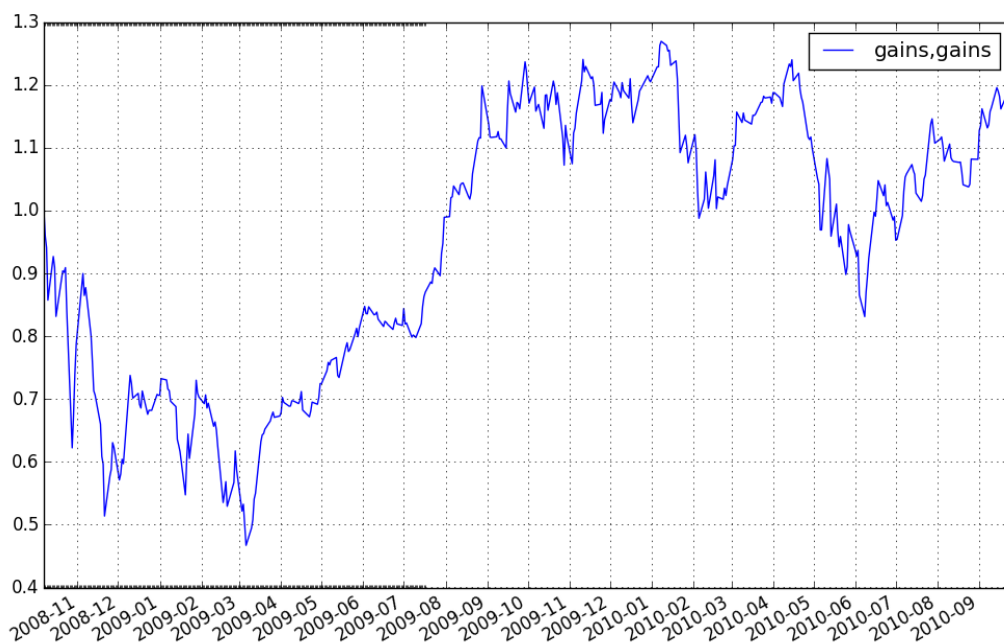
La première paire présentait un coefficient de corrélation de 0,92 alors que la seconde seulement de 0,55 sur la période 11/2008-09/2010. Pour chaque paire, nous avons investi 1€ en début de période. Le calcul des gains sur cette période nous a permis de confirmer l'efficacité de cette stratégie sur des paires fortement liées et sa moindre efficacité sur des paires faiblement liées comme le montrent les 2 graphiques ci-dessous.

Il est à noter que ces résultats ont été obtenus avec un calcul de la moyenne mobile effectuée sur 50 jours antérieurs et une largeur des bandes de Bollinger de $\alpha = 1$.

Graphique 3 : Gains ou valeur actuelle du portefeuille en € contenant la paire LVMH-L'OREAL en fonction du temps.



Graphique 4 : Gains ou valeur actuelle du portefeuille en € contenant la paire LVMH-Société Générale en fonction du temps.



Pour la première paire (celle fortement liée) la valeur du portefeuille a doublé à la fin de la période considérée alors que pour la seconde paire (celle faiblement liée), la valeur du portefeuille a tout juste été multipliée par 1.15 avec durant l'intervalle de temps considéré de larges périodes où le portefeuille avait une valeur inférieure à la valeur mise au départ et de fortes fluctuations.

De plus, il est à noter que la période choisie pour l'analyse met à rude épreuve les rendements puisqu'elle se situe juste après la crise. Ce choix n'est pas anodin puisqu'il met en valeur l'un des objectifs du pair trading qui est d'affaiblir le risque lié aux événements économiques sur les différents secteurs. En effet, pour les actions faisant parties d'un même secteur, le rapport de leurs cours est moins sensible à la conjoncture économique qu'à la différence entre les deux entreprises. Ceci est confirmé en observant comment varie le rendement des portefeuilles dans les 2 ans qui suivent la crise lorsque l'on prend des actions d'un même secteur et deux actions de secteurs très différents. Certes, la première paire subit quelques pertes entre fin 2008 et début 2009 mais elles restent très faibles et largement compensées par l'évolution du portefeuille par la suite à la différence de la seconde paire.

2. Les Pistes d'améliorations

Afin d'assurer une meilleure performance de la stratégie, on pourrait envisager d'employer des méthodes statistiques et de cointégration plus fines dans l'analyse des liens entre les deux actions.

C) Le Mean Reversion

Afin de mettre en évidence les facteurs influençant notre algorithme mean reversion nous allons considérer l'univers d'actions suivant :

ACTION	VOLATILITE historique entre 2006 et 2016
Apple	0.0094
Yahoo	0.0110
Microsoft	0.0077
Facebook	0.0114
Twitter	0.0160
BNP	0.0114
SocGen	0.0128
Crédit Agricole	0.0124
Portefeuilles d'action	numéro
Apple, Yahoo, Microsoft, Facebook, Twitter	1
BNP, Soc Gen, Crédit Agricole	2
BNP, Soc Gen,	3

Crédit Agricole, Facebook, Twitter, Yahoo	
Apple, Microsoft, Yahoo	4

1. *Influence de la volatilité*

La volatilité n'apparaît pas comme un paramètre important dans la constitution d'un portefeuille. En effet, nous constatons dans le graphique 5 de l'annexe que le portefeuille 1 à un fort rendement entre 2012 et 2016, or la volatilité des actions constituant ce portefeuille est inférieure à la volatilité globale entre 2006 et 2016 :

ACTION	VOLATILITE entre 2006 et 2016	Volatilité entre 2012 et 2016
Apple	0.0094	0.0070
Yahoo	0.0110	0.0088
Microsoft	0.0077	0.0067
Facebook	0.0114	0.0088
Twitter	0.0160	0.0156

De plus, nous avons constitué les portefeuilles 3 et 4 en fonction de la volatilité des actions de l'univers initial. Le portefeuille 3 est constitué des actions à plus forte volatilité tandis que le portefeuille 4 est constitué des actions à moins forte volatilité. Nous constatons dans les graphiques 7 et 8 que c'est le portefeuille 4 constitué d'actions à moins forte volatilité qui a le meilleur rendement. Ces deux arguments tendent à montrer qu'il faut privilégier les volatilités les moins élevées. En réalité, c'est avant tout le secteur économique qui joue un rôle important.

2. *Influence du secteur économique*

Les portefeuilles 1 et 2 ont été constitués en regroupant les actions selon leur secteur économique contrairement au portefeuille 3. Nous constatons que les portefeuilles d'un même secteur économique ont une tendance haussière ou baissière qui se dégage très clairement tandis que le rendement du portefeuille 3 oscille (voir graphiques 5, 6 et 7 de l'annexe). Ainsi regrouper les actions selon un même secteur économique permet de supprimer les perturbations et de dégager une tendance haussière ou baissière sur le long terme. Il est intéressant de remarquer que ce critère est similaire à celui de la stratégie Pair Trading. Une interprétation possible est la différence entre la stratégie Mean Reversion classique consistant à établir nos choix en fonction des bandes de Bollinger et la stratégie que l'on a implémentée consistant à établir nos décisions en fonction des performances des actions du portefeuille. Cette différence rapproche notre algorithme de trading du Pair Trading de part la comparaison des actions les unes avec les autres, ce qui peut expliquer l'importance de ce critère.

3. *Un postulat empirique*

Regrouper les actions selon leur secteur économique est une condition nécessaire mais non suffisante. Elle permet de dégager une tendance haussière ou baissière sur le long terme en éliminant au maximum les perturbations conjoncturelles liées à différents secteurs économiques. Néanmoins des perturbations persistent et peuvent induire en erreur sur la tendance haussière ou baissière du rendement d'un portefeuille. En effet on remarque sur le graphique 6 que la tendance du rendement est clairement baissière entre 2006 et 2016 alors qu'entre 2013 et 2015 le rendement est de 50%. Donc étudier le rendement sur deux ans n'est pas suffisant pour conclure sur le fait que cette stratégie soit gagnante ou non sur le long terme. Nous constatons en revanche empiriquement que si la stratégie est gagnante sur 5 ans (toujours pour des portefeuilles d'action d'un même secteur économique) elle le restera les années suivantes.

3. *Guide d'utilisation de cet algorithme*

Pour résumer voici les étapes pour constituer un portefeuille gagnant :

- 1- Choisir des actions d'un même secteur économique
- 2- Privilégier les actions à faible volatilité
- 3- Analyser le rendement des 5 dernières années de votre portefeuille

Remarquons que cet algorithme est riche en paramètre, ceci est une méthode potentiellement gagnante mais n'est pas nécessairement l'unique méthode.

4. Pistes d'améliorations :

Le guide d'utilisation a été déduit empiriquement de nombreuses combinaisons de portefeuilles. Il serait intéressant de rajouter à cet algorithme de trading des tests statistiques comme un estimateur du rendement futur d'un portefeuille donné (constitué d'actions d'un même secteur économique).

On pourrait aussi rajouter un test Dickey Fuller Augmented pour tester la pertinence et le degré de stationnarité des actions d'un portefeuille.

Conclusion :

Si les différentes stratégies de trading qui existent dans le monde de la finance aujourd'hui demeurent encore, à notre sens, très complexes, ce projet nous a permis de développer un esprit critique vis à vis de l'utilité des différents modes de décision. Les trois stratégies implémentées dans ce mémoire ont été riches en enseignements de part leur diversité. Grâce à des analyses empiriques, nous avons pu constater que le Trend following préfère des actions à faible volatilité. Le Pair Trading, quant à lui, fonctionnera mieux pour des actions corrélées. Enfin, le Mean Reversion implémenté dans ce mémoire préfère des actions corrélées, à faible volatilité et a l'avantage d'offrir un plus large domaine des possibles avec de nombreux paramètres comme la constitution d'un portefeuille d'actions de plus de 3 actions.

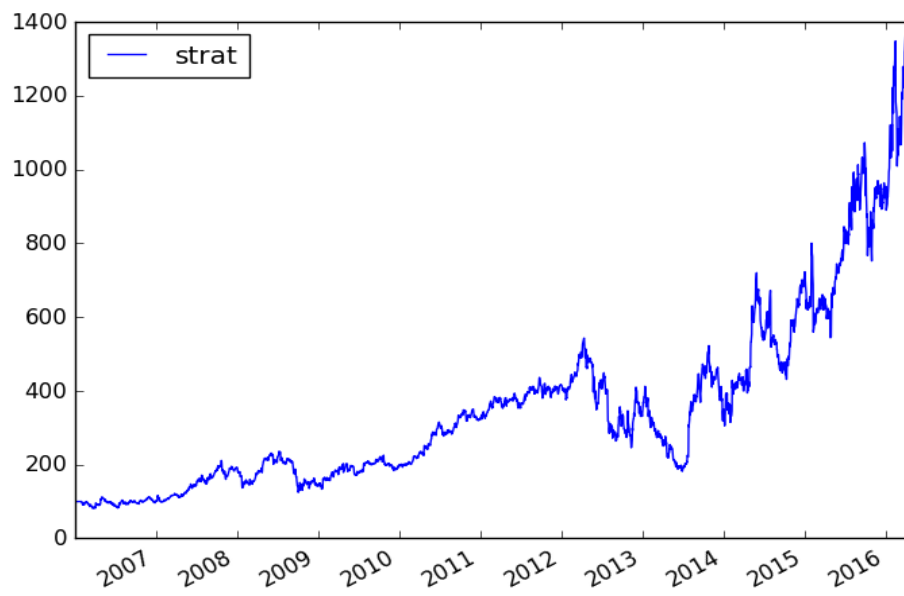
Ayant montré dans quel cadre général appliquer nos stratégies pour obtenir les meilleurs résultats, il serait alors intéressant d'approfondir l'étude en complexifiant les modes de décisions notamment par l'ajout de tests statistiques nous permettant d'évaluer les risques, de choisir de manière plus pertinente nos actions et d'optimiser nos gains.

Outre les conclusions financières que nous avons pu tirées, ce projet nous a surtout permis d'évoluer en terme de maîtrise, tant algorithmique, qu'en connaissance du langage python. De part l'autonomie que ce travail a requis et la liberté qui nous a été laissée, toutes les difficultés rencontrées concernant notamment l'implémentation durant la réalisation de ce projet, nous ont permis de développer notre esprit d'analyse, et notre inventivité pour détourner ou résoudre ces dernières.

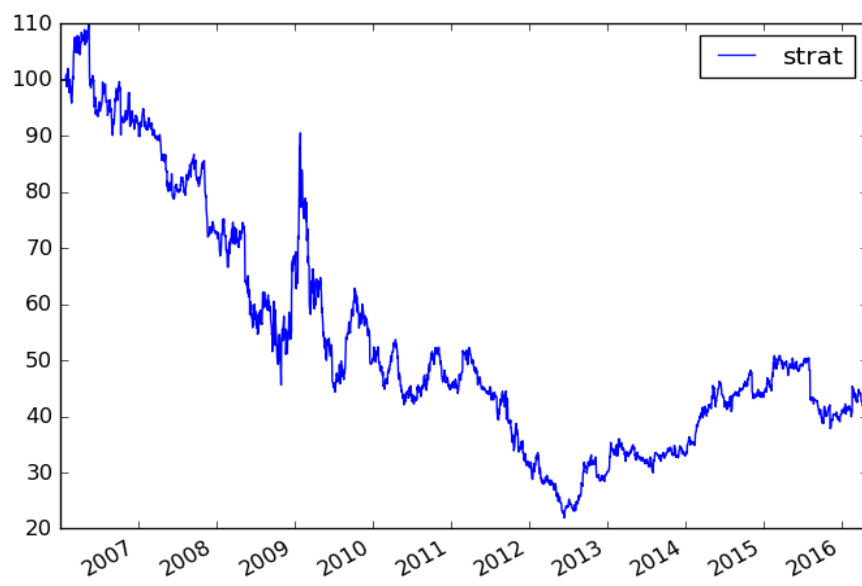
Annexe :

- **Graphiques de la partie II.C) :**

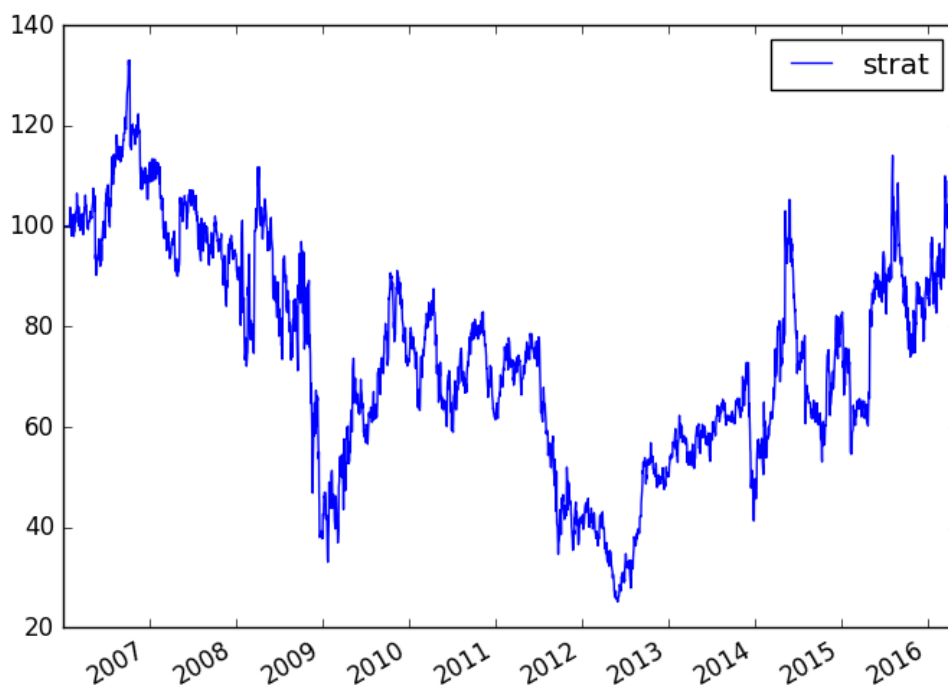
Graphique 5 : rendement du portefeuille n°1 entre 2006 et 2016



Graphique 6 : rendement du portefeuille n°2 entre 2006 et 2016



Graphique7: rendement du portefeuille n°3 entre 2006 et 2016



Graphique 8 : rendement du portefeuille n°4 entre 2006 et 2016



- **Code python : Calcul de la complexité de l'algorithme de la moyenne mobile.**

Calcule la moyenne mobile du jour date1 basée sur n jours antérieurs.

n représente l'intervalle de temps que l'on prend en compte pour le calcul de la moyenne mobile

Calcul du coût (Chaque chiffre en rouge représente le coût de la ligne d'instruction) :

```
def moyenne_mobile(date1,n,serie,name):
    dq=serie.dataframe (1)
    dt1=datetime.datetime.strptime(date1,"%Y-%m-%d")-datetime.timedelta(n) (1)
    dt2 = datetime.datetime.strptime(date1,"%Y-%m-%d") (1)
    delta = datetime.timedelta(1) (1)
    compt=0 (1)
    sum=0 (1)
    if dt1>dt2:(1)
        return "impossible" (1)
    while dt1<=dt2: # répète au maximum n fois ce qu'il y a dans la boucle (n)
        date = dt1.strftime("%Y-%m-%d") (1)
        if date not in dq["Date"]: # cette ligne parcourt à chaque fois tout le dataframe donc de
        cout m= la longueur du dataframe (m)
            while date not in dq["Date"]: # pareil que la ligne du dessus (l'opération de
            vérifier à chaque fois si la date est bien dans le dataframe est coûteuse mais nous ne voyons pas
            comment faire autrement) (m)
                dt1 = dt1 + delta (1)
                date = dt1.strftime("%Y-%m-%d") (1)
                if dt1>dt2: (1)
                    return "trouver une autre date de fin" (1)
            compt=compt+1 (1) #les lignes ci dessous font un simple calcul de moyenne à l'aide d'un
            compteur et d'une somme.
            sum=sum+dq.ix[date,name] (1)
            dt1 = dt1 + delta (1)
            moyenne=sum/compt (1)
    return moyenne
```

Au final la complexité de l'algorithme de calcul de la moyenne mobile est en **O(nmm)**.

Bibliographie :

_ mesure des risques de l'université de Lille

http://jb.desquilbet.pagesperso-orange.fr/docs/M2_GR03_Mesures_de_risque.pdf

_ Introduction aux algorithmes de Trading

http://www.xavierdupre.fr/site2013/documents/reports/finance_autostrat.pdf

_ Principes du PairTrading:

<http://biblos.hec.ca/biblio/memoires/2012NO57.PDF>