

Faculté des Sciences et Ingénierie
Sorbonne Université

Projet Cryptoptimisation : Plate-forme d'optimisation de portfolio de cryptomonnaies

JULIEN DENES, MICHAËL TRAZZI
Sous la supervision de THIBAUT LUST

Année 2017-2018 – Master 1, Semestre 2



Introduction

Les monnaies numériques, dites cryptomonnaies, sont aujourd'hui en plein essor. L'année 2017 a vu apparaître des centaines de nouvelles monnaies sur le marché et le bitcoin, la plus connue d'entre elles, ne cesse d'accumuler les records. En 2016, sa valeur a bondi de plus de 120%, dépassant les 1 000 dollars. En 2017, elle pulvérise ses propres exploits et s'évalue à près de 20 000 dollars. La progression fulgurante de son prix attire ainsi toujours plus d'émulation autour d'elle, et ceux qui souhaitent profiter de cette croissance sont toujours plus nombreux. L'investissement dans les cryptomonnaies présente néanmoins un risque important, et est réservé aux investisseurs présentant une excellente tolérance au risque. La haute volatilité des cours des cryptomonnaies exige en effet des nerfs d'acier : la valeur du bitcoin a par exemple été divisée par 3 entre décembre 2017 et janvier 2018.¹ Paradoxalement, ces nouvelles formes d'actifs financiers sont toujours plus accessibles, grâce au développement de plate-formes d'échanges faciles d'utilisation et sécurisées.

Partant de ce constat, on souhaite dans ce projet réaliser une plate-forme d'optimisation de portfolio de cryptomonnaies, basée sur différentes théories modernes de l'optimisation de portfolio d'actifs. On souhaite ainsi fournir une aide à la décision et à la gestion de portefeuille pour des utilisateurs de tous types, aussi bien habitués de la spéculation que novices en la matière.

1 Revue de la littérature existante

1.1 Les cryptomonnaies comme actifs financiers

Elendner et al. [1] proposent, dans leur publication « **The Cross-Section of Cryptocurrencies as Financial Assets** », une étude approfondie de la possibilité de considérer les cryptomonnaies comme des actifs d'investissement alternatifs. Ils évaluent en particulier leurs propriétés, en les comparant à celles des actifs standards. Leurs premières conclusions sont celles que l'on pourrait attendre : bien qu'en moyenne légèrement positifs (entre 0 et 1%), les rendements quotidiens des principales monnaies sont très volatiles : le maximum pour l'éthereum est par exemple de 55% et son minimum de -48% en une journée. Leur volatilité est comprise entre 3.3 et 10.0. Les cryptomonnaies présentent donc à première vue un intérêt du point de vue de leur rapide croissance, mais présentent un risque élevé. Les auteurs s'intéressent donc dans un second temps aux possibilités de diversifications qu'elles présentent. Leurs conclusions sont cette fois-ci moins instinctives : les cryptomonnaies sont deux à deux assez peu corrélées (par exemple 0.08 entre le bitcoin et l'éthereum), et en tous cas bien moins que les actifs boursiers traditionnels entre eux, même lors des phases de grande croissance ou de forte décroissance sur leurs marchés où les corrélations augmentent quelque peu. Même leur volatilité respectives sont très peu liées. Les auteurs enquêtent enfin sur la corrélation entre cryptomonnaies et d'autres actifs traditionnels, comme les monnaies nationales, l'or, ou les bons du Trésor américains. Leurs conclusions montrent une très forte indépendance, la plus forte corrélation observée n'étant que de 0.09. Les auteurs concluent donc sur le grand intérêt que présentent les cryptomonnaies pour la diversification de portfolio, que ceux-ci en soient uniquement constitués ou qu'ils soient mixés avec des actifs plus traditionnels.

1.2 Algorithmes d'optimisation de portefeuilles

Dans l'article de Li et Hoi [2] « **Online Portfolio Selection : A Survey** », les auteurs proposent un panorama des techniques d'optimisation de portfolio à l'heure actuelle (mai 2013), et proposent une analyse des principes mathématiques sur lesquelles elles reposent. Ils commencent par proposer une distinction entre la "Mean Variance Theory" et la "Capital Growth Theory". La première, basée sur la théorie de Markovitz, ne s'intéresse qu'à une seule période de temps : on sélectionne un portfolio fixé en cherchant le meilleur compromis entre "return" (mean) et risque (variance). La seconde théorie se focalise sur une sélection de portfolio multi-périodiques

1. Source : <https://coinmarketcap.com/>

(ou séquentielle), c’est à dire en découpant une période en sous-séquences et en autorisant une modification du portfolio à la fin de chacune de ces périodes. On s’intéresse alors à sélectionner la meilleur rentabilité (ou taux de croissance). Les techniques de la “Capital Growth Theory” sont regroupées sous le nom de “Online Portfolio Selection”, sur lesquelles cet article se focalise. Les auteurs découpent ces techniques en 5 catégories : Benchmarks, Follow-the-Winner, Follow-the-Loser, Pattern-Matching Approaches, et Meta-Learning Algorithms. Les Benchmarks sont des algorithmes quasi-triviaux qui servent de repères : Buy and Hold (aucune modification du portfolio à chaque période), Best Stock (idem mais portfolio composé d’un unique stock, le meilleur), Constant Rebalanced Portfolios (le portfolio de la période suivante est toujours portfolio fixé). Les algorithmes Follow-the-Winner sont basés sur le principe de l’augmentation du poids des stocks avec la meilleur croissance à la période t pour former le portfolio du temps $t + 1$. Parmi eux, on pourra citer les Universal Portfolios (construit des portfolio composés d’un unique stock puis mixe les plus performants), Exponential Gradient (cherche à suivre le meilleur portfolio au temps t mais en restant proche du portfolio adopté précédemment), le très similaire Follow the Leader (qui ne tient pas compte de la proximité), ou Follow the Regularized Leader (idem mais avec un terme de régularisation, qui typiquement minimise la taille du vecteur de portfolio). Les approches Follow-the-Loser se basent quant à elles sur la théorie de la mean reversion, qui affirme en sorte que les stocks ayant eu de mauvaises performances à une période t seront ceux qui auront de bonnes performances au temps $t + 1$ (et vice versa). On citera l’algorithme Anticor (transfert le capital des bons performeurs au mauvais en proportion de la corrélation entre eux), ou encore Passive Aggressive Mean Reversion qui résout un programme linéaire de minimisation de la distance entre le portfolio de t et de $t + 1$ sous contrainte de respecter la “mean reversion property”. Online Moving Average Reversion fonctionne sensiblement sur le même principe mais en se basant sur plus d’une période en amont. On pourra noter que les algorithmes sont, d’après les tests des auteurs, les plus performants. Les approches de Pattern-Matching cherchent quant à eux d’abord à trouver un set C de prix similaires dans l’historique, puis à partir de celui-ci compose le portfolio le plus performant basé sur ce que ce set C laisse à prévoir. Enfin, les algorithmes de Meta-Learning vont chercher à combiner plusieurs portfolios performants à une période t pour former celui de la période $t + 1$, et ce selon différentes techniques.

Moody et Saffell [3] proposent, dans « **Learning to Trade via Direct Reinforcement** », un modèle d’algorithme de trading basé sur leurs précédentes publications, qui s’appuie sur une stratégie d’apprentissage renforcé récurrent (ou direct). L’algorithme qu’ils présentent s’affranchit de l’apprentissage d’une fonction évaluée, en se basant plutôt sur un feedback immédiat de performance lors de l’exécution d’une action. Ils cherchent ainsi à s’affranchir de deux biais majeurs des algorithmes d’apprentissage sur des données financières : la malédiction de la dimensionnalité de Bellman, due au fait que les précédents algorithmes s’appuyaient sur de la programmation dynamique, et la nécessité de s’appuyer sur des modèles de prédictifs, éliminée par la fonction de récompense immédiate. Ils utilisent pour celle-ci la différentiation d’utilité entre le portfolio au temps t et celle au temps $t + 1$. La fonction d’utilité utilisée est le ration de Sharpe, défini comme le gain divisé par la déviation standard (on ne maximise ainsi pas le profit mais le profit ajusté par le risque). Ces deux grandeurs ne pouvant être calculées au temps futur t , on se base plutôt sur une utilité espérée, basée sur les situations vues auparavant. Les résultats empiriques montrent que cet algorithme présente de meilleures performances que d’autres basées sur des fonctions évaluées (Q-Learning) sur le marché des échanges de monnaies US Dollar/British Pound. Il réussit en effet à dégager en moyenne sur 25 ans un bénéfice de 15% par an. Les auteurs insistent cependant sur la stabilité et la périodicité de ce marché : il est ainsi facile pour l’algorithme d’identifier des motifs réguliers pour lesquels il saura ainsi bien estimer l’utilité espérée.

D’autres travaux similaires font suite à celui-ci dans l’application des stratégies d’apprentissage pour le trading algorithmique. **Ban, El Karoui et Lim [4]** étudient ainsi un modèle d’apprentissage par renforcement très similaire, en y ajoutant deux techniques supplémentaires : la régularisation et la validation croisée. Ils cherchent de cette manière à contraindre la variance des estimations de gain et de risque, et ainsi réduire l’erreur d’évaluation de performance. Leurs évaluations théoriques démontrent selon les auteurs un avantage sur les algorithmes de référence ; les tests sont pour leur part moins concluant malgré l’enthousiasme affiché.

Heaton, Polson et Witte [5] tentent quant à eux d'appliquer un algorithme de deep learning pour la prédiction d'évolution de valeur d'un portfolio. Leur argument principal est que les réseaux de neurones pourraient avoir une capacité à déceler des interactions dans les données invisibles par les algorithmes fortement basés sur des modèles. La prédiction d'évolution est cependant souvent décriée dans la littérature (par exemple [3] et [6]), et ce, semble-t-il, à juste titre. Les résultats de cet article semblent en effet montrer une assez mauvaise précision, à cause d'une accumulation d'infimes imprécisions des estimations. Applicable donc, pour l'auteur, à des problèmes comme la limitation du passage d'ordre, la volonté de prédire l'évolution semble encore trop complexe mais surtout non nécessaire pour l'optimisation de portefeuilles.

1.3 Application des algorithmes aux portfolios de cryptomonnaies

Dans « **Application of Machine Learning Algorithms for Bitcoin Automated Trading** », Żbikowski [7] étudie en premier la possibilité d'appliquer un algorithme de machine learning pour le trading de cryptomonnaies. Il ne s'intéresse cependant qu'au bitcoin, et utilise 2220 points de données périodiques espacés de 15 minutes dans le temps. Il applique deux algorithmes : une machine à support de vecteurs (SVM) avec « Box Theory », et une SVM avec pondération par le volume. Ces deux algorithmes se montrent compétents dans la réalisation d'une stratégie de trading, le premier obtenant un retour sur investissement de 10.58%, le second de 33.58%. L'exposition au risque est également meilleure que la stratégie utilisée comme référence.

L'article de Jiang et Liang [6] « **Cryptocurrency Portfolio Management with Deep Reinforcement Learning** » s'intéresse à l'application des techniques de Machine Learning au problème de gestion du portfolio. Il existe des approches de type Deep Learning appliquées au marchés financier, mais celles-ci se focalisent sur la prédiction du prix, ce qui s'avère particulièrement difficile pour le cas des cryptomonnaies. Les travaux d'apprentissage concernant l'algorithmic trading n'essayant pas de prédire le prix, et n'utilisant pas de modèle pré-existant, utilisent des techniques de Reinforcement Learning (RL). Cependant, ces travaux se contentent d'échanger seulement un stock, ce qui ne nous intéresse pas pour le cas du portfolio. Les techniques de Deep RL récentes consistant à entraîner deux politiques en même temps peuvent s'avérer trop difficiles et instables. C'est pourquoi dans ce papier est adoptée une approche de RL adaptée au cas du portfolio. Cette approche utilise des réseaux de neurones (RNN, CNN et LSTM) prenant en entrée l'historique de stocks, pour prédire l'action adaptée à adopter à la fin de la période (e.g. acheter, vendre, etc.). Les résultats théoriques sont testés en pratique en effectuant un trading permanent sur Polonix.com, à intervalles de 30 minutes. Contrairement au marché habituel, le marché des cryptomonnaies est ouvert 24h/24 7j/7. Le trading continu est divisé sur des périodes de 30 minutes. On suppose qu'au début et à la fin de chaque période on peut acheter et vendre directement, sans influencer sur le marché global. Le bitcoin est considéré être notre « cash », ou cryptomonnaie de référence. A la fin de chaque période, en achetant/vendant on change notre vecteur de portefeuille, donné par la proportion de notre capital dans chacune des cryptomonnaies. Le but de l'algorithme est de générer une séquence de vecteurs de portefeuille permettant de maximiser le capital accumulé, en prenant en compte un coût constant de transaction de 0.25%. Les 12 cryptomonnaies avec la plus grande Coinmarketcap ont été utilisées. Cela permet d'assurer la liquidité totale à la fin de chaque période de trading, ce que nous avons supposé. En outre, cela garantit que nous ne traitons pas le cas de cryptomonnaies récentes qui pourraient avoir un comportement instable. Pour le cas où les données n'existent pas (la cryptomonnaie n'existe pas encore) on donne des (faux) prix décroissants pour inciter notre algorithme à ne pas la considérer. Pour définir notre problème de Reinforcement Learning il nous faut définir l'agent et l'environnement. Ici l'agent est notre algorithme, et l'environnement est l'ensemble des prix (accessibles) des stocks. Le flux de l'ensemble des prix étant trop difficile à traiter, on considère que tout ce dont dispose notre agent sont les prix les plus haut, les plus bas et le cours de clôture des actions, pour chaque période. Les états sont alors la donnée de cet environnement et du portfolio à la précédente période. Afin de déterminer une politique optimale on utilise une montée de gradient avec mini-batch et du online learning (les nouvelles données du marché arrivent en continu durant l'apprentissage). Les résultats tendent à montrer que cette approche surpasse toutes les méthodes traditionnelles. Des architectures utilisées pour les réseaux de neurones, la meilleure est CNN, suivie de près par un RNN basique, et bien

meilleure qu'un LSTM. Les hypothèses de liquidité totale et de non influence sur le marché sont cependant fortes. De plus, l'algorithme n'a pas été testé sur un marché réel plus traditionnel. Il faudrait dans ce cas étudier les corrélations entre le trading et la réaction du marché.

Références

- [1] Hermann ELENDRER et al. « The Cross-Section of Crypto-Currencies as Financial Assets ». In : *Handbook of Blockchain, Digital Finance, and Inclusion, Volume 1*. Sous la dir. de David LEE KUO CHUEN et Robert H. DENG. Academic Press, 2018. Chap. 7, p. 145–170.
- [2] Bin LI et Steven C.H. HOI. « Online Portfolio Selection : A Survey ». In : *ACM Computing Surveys (CSUR)* 46.3 (2014).
- [3] J. MOODY et M. SAFFELL. « Learning to Trade via Direct Reinforcement ». In : *IEEE Transactions on Neural Networks* 12.4 (2001), p. 875–889.
- [4] Gah-Yi BAN, Nouredine EL KAROUI et Andrew E.B. LIM. « Machine Learning and Portfolio Optimization ». In : *Management Science* (2016).
- [5] J.B. HEATON, Nick POLSON et Jan WITTE. « Deep Learning for Finance : Deep Portfolios ». In : *Applied Stochastic Models in Business and Industry* 33.1 (2017), p. 3–12.
- [6] Zhengyao JIANG et Jinjun LIANG. « Cryptocurrency Portfolio Management with Deep Reinforcement Learning ». In : *Intelligent Systems Conference*. 2017.
- [7] Kamil ŻBIKOWSKI. « Application of Machine Learning Algorithms for Bitcoin Automated Trading ». In : *Machine Intelligence and Big Data in Industry*. Sous la dir. de Dominik RYŻKO et al. Springer International Publishing, 2016, p. 161–168.
- [8] James CUMMING. « An Investigation into the Use of Reinforcement Learning Techniques within the Algorithmic Trading Domain ». Mém.de mast. Imperial College London - Department of Computing, 2015.
- [9] Puja DAS et Arindam BANERJEE. « Meta Optimization and its Application to Portfolio Selection ». In : *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2011, p. 1163–1171.
- [10] Bin LI, Doyen SAHOO et Steven C. H. HOI. « OLPS : A Toolbox for On-line Portfolio Selection ». In : *The Journal of Machine Learning Research* 17.1 (2016), p. 1242–1246.
- [11] Bin LI, Peilin ZHAO et Steven C. H. HOI. « PAMR : Passive Aggressive Mean Reversion Strategy for Portfolio Selection ». In : *Machine Learning* 87.2 (2012), p. 221–258.
- [12] Tun-Jen CHANG, Sang-Chin YANG et Kuang-Jung CHANG. « Portfolio Optimization Problems in Different Risk Measures Using Genetic Algorithm ». In : *Expert Systems with Applications* 36.7 (2009), p. 10529–10537.
- [13] Shi CHEN et al. « Econometric Analysis of a Cryptocurrency Index for Portfolio Investment ». In : *Handbook of Blockchain, Digital Finance, and Inclusion, Volume 1*. Sous la dir. de David LEE KUO CHUEN et Robert H. DENG. Academic Press, 2018. Chap. 8, p. 175–206.
- [14] David LEE KUO CHUEN, Li GUO et Yu WANG. « Cryptocurrency : A New Investment Opportunity ? » In : *The Journal of Alternative Investments* 20.3 (2017), p. 16–40.
- [15] Harry MARKOWITZ. « Portfolio Selection ». In : *The Journal of Finance* 7.1 (1952), p. 77–91.
- [16] Alain P. CHABOUD et al. « Rise of the Machines : Algorithmic Trading in the Foreign Exchange Market ». In : *The Journal of Finance* 69.5 (2014), p. 2045–2084.
- [17] Stuart COLIANNI, Stephanie ROSALES et Michael SIGNOROTTI. *Algorithmic Trading of Cryptocurrency Based on Twitter Sentiment Analysis*. 2015.
- [18] Isaac MADAN, Shaurya SALUJA et Aojia ZHAO. *Automated Bitcoin Trading via Machine Learning Algorithms*. 2014.
- [19] Nils H. HAKANSSON et William T. ZIEMBA. « Capital Growth Theory ». In : *The Kelly Capital Growth Investment Criterion*. World Scientific, 2011. Chap. 41, p. 577–598.