Dynamic portfolio management with transaction costs

Alberto Suárez
Computer Science Department
Universidad Autónoma de Madrid
28049, Madrid (Spain)
alberto.suarez@uam.es

John Moody, Matthew Saffell
International Computer Science Institute
1947 Center Street. Suite 600
Berkeley, CA 94704, USA
moody@icsi.berkeley.edu, saffell@icsi.berkeley.edu

Abstract

We develop a recurrent reinforcement learning (RRL) system that directly induces portfolio management policies from time series of asset prices and indicators, while accounting for transaction costs. The RRL approach learns a direct mapping from indicator series to portfolio weights, bypassing the need to explicitly model the time series of price returns. The resulting policies dynamically optimize the portfolio Sharpe ratio, while incorporating changing conditions and transaction costs. A key problem with many portfolio optimization methods, including Markowitz, is discovering "corner solutions" with weight concentrated on just a few assets. In a dynamic context, naive portfolio algorithms can exhibit switching behavior, particularly when transaction costs are ignored. In this work, we extend the RRL approach to produce better diversified portfolios and smoother asset allocations over time. The solutions we propose are to include realistic transaction costs and to shrink portfolio weights toward the prior portfolio. The methods are assessed on a global asset allocation problem consisting of the Pacific, North America and Europe MSCI International Equity Indices.

1 Introduction

The selection of optimal portfolios is a central problem of great interest of quantitative finance, one that still defies complete solution. [1, 2, 3, 4, 5, 6, 7, 8, 9]. A drawback of the standard framework formulated by Markowitz [1] is that only one period is used in the evaluation of the portfolio performance. In fact, no dynamics are explicitly considered. Like in many other financial planning problems, the potential improvements of modifying the portfolio composition should be weighed against the costs of the reallocation of capital, taxes, market impact, and other state-dependent factors. The performance of an investment depends on a sequence of portfolio rebalancing decisions over several periods. This problem has been addressed using different techniques, such as dynamic programming [2, 5] stochastic network programming [3], tabu search [4], reinforcement learning [7] and Monte Carlo methods [8, 9].

A key problem with many portfolio optimization methods, including Markowitz, is finding "corner solutions" with weight concentrated on just a few assets. In a dynamic context, naive portfolio algorithms can exhibit switching behavior, particularly when transaction costs are ignored.

In this work, we address the asset management problem following the proposal of Moody *et al* [6, 7], and use reinforcement learning to optimize objective functions such as the Sharpe ratio that directly measure the performance of the trading system. A recurrent softmax architecture learns a direct mapping from indicator series to portfolio weights, and the recurrence enables incorporation of transaction costs. The softmax network parameters are optimized via the recurrent reinforcement learning (RRL) algorithm.

We extend the RRL approach to produce more evenly diversified portfolios and smoother asset allocations over time. The solutions we propose are to incorporate realistic transaction costs and

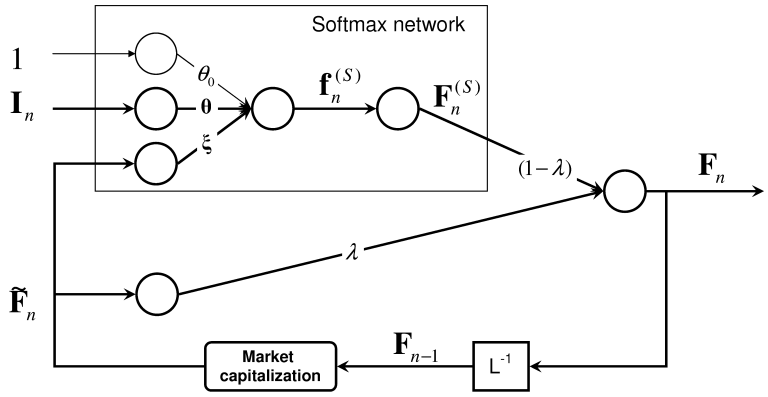

Figure 1: Architecture for the reinforcement learning system. The system improves on previous proposals by directly considering, when $\lambda \neq 0$, the current portfolio composition in the determination of the new portfolio weights.

to shrink portfolio weights toward the prior portfolio. The methods are assessed on a global asset allocation problem consisting of the Pacific, North America and Europe MSCI International Equity Indices.

2 Reinforcement learning architecture

We consider the problem of a creating a dynamically managed portfolio by investing in M assets. The composition of the portfolio can be modified at specified times $t_n = t_0 + n\Delta T, n = 0, 1, 2, \ldots, N-1$. The portfolio is evaluated in terms of accumulated profit at time $T = N\Delta T$ or of its risk-adjusted performance as measured by the Sharpe ratio. The architecture of the learning system is depicted in Figure 1. The portfolio weights predicted by the policy are a convex combination of $\tilde{\mathbf{F}}_n$, the composition of the portfolio at t_n^-, prior to rebalancing, and $\mathbf{F}_n^{(S)}(\mathbf{w})$, the output of a softmax network whose inputs are a constant bias term, the information set, $\mathbf{I}_n$ (either lagged information from the time series of asset returns or external economic and financial indices) and the current portfolio weights $\tilde{\mathbf{F}}_n$

$$\mathbf{F}_n(\lambda, \mathbf{w}) = \lambda \tilde{\mathbf{F}}_n + (1-\lambda)\mathbf{F}_n^{(S)}(\mathbf{w}) \tag{1}$$

The relative importance of these two terms in the final output is controlled by a hyperparameter $\lambda \in [0, 1]$. For $\lambda = 0$, the final prediction is directly the output of the softmax network. In the absence of transaction costs, a new portfolio can be created at no expense. In this case, the currently held portfolio need not be used as a reference, and $\lambda = 0$ should be used. If transaction costs are non-zero, it is necessary to ensure that the expected return from dynamically managing the investments outweighs the cost of modifying the composition of the portfolio. The costs are deterministic and can be calculated once the new makeup of the portfolio is established. By contrast, the returns expected from the investment are uncertain. If they are overestimated (e.g. when there is overfitting) the costs will dominate and the dynamic management strategy seeking to maximize the returns by rebalancing will have a poor performance. A value $\lambda > 0$ causes the composition of the portfolio to vary smoothly, which should lead to improved performance in the presence of transaction costs.

The parameters of the RRL system are fixed by either directly maximizing the wealth accumulated over the training period or by optimizing an exponentially smoothed Sharpe ratio [6]. The training algorithm is a variant of gradient ascent with learning parameter ρ extended to take into account the recurrent terms in 1. The hyperparameters of the learning system (ρ, η, λ) can be determined by either holdout validation or cross-validation.

Table 1: Performance of the portfolios selected by the different strategies. The values displayed between parentheses are performance measures relative to the market portfolio: The ratio of the profit for the corresponding portfolio to the profit of the market portfolio, and the difference between the Sharpe ratio of the portfolio and the Sharpe ratio of the market portfolio.

	Cost	**Market**	**Markowitz**	**RRL**
Profit	0 %	2.9084	3.1889 (1.0965)	3.4507 (1.1865)
	1 %		2.9094 (1.0003)	3.1825 (1.0942)
	2 %		2.6539 (0.9125)	3.1749 (1.0916)
	3 %		2.4205 (0.8322)	2.9176 (1.0031)
	5 %		2.0125 (0.6920)	2.8342 (0.9745)
Sharpe ratio	0 %	0.4767	0.5147 (0.0381)	0.5456 (0.0689)
	1 %		0.4804 (0.0037)	0.5110 (0.0350)
	2 %		0.4457 (-0.0309)	0.5080 (0.0314)
	3 %		0.4108 (-0.0658)	0.4793 (0.0027)
	5 %		0.3405 (-0.1362)	0.4682 (-0.0084)

3 Preliminary results and ongoing work

The performance of the reinforcement learning system is assessed on real market data and compared to the market portfolio (optimal if the market were ideally efficient) and to the tangency portfolio computed using the Markowitz framework portfolio, which is optimal in terms of the Sharpe ratio, assuming zero transaction costs. The experiments are carried out using the MSCI International Equity Indices (gross level) that measure the performance of different economic regions (indices for the Pacific, North America and Europe) and of the global market (the World index) [10]. A total of 470 values of monthly data starting December 31, 1969 until January 30, 2009 are used. The objective is to learn optimal policies for the dynamic management of a geographically diversified portfolio. In particular, we consider the problem of improving the performance of the World index by investing in some of its constituents, the North America, the Europe and the Pacific indices. As inputs of the softmax network, we employ internal indices that measure the recent performance of each of the assets. The information set at time t_n, $\mathbf{I}_n$, consists of moving averages of the asset returns over the previous 3, 6, 12 and 24 months. Single month returns are not directly employed because they exhibit large fluctuations that make it difficult for the reinforcement learning system to distill any useful information from the data. Averages over periods longer than two years are probably not useful because of the changes in the economic conditions of the markets considered.

At each point in time the parameters of the model are determined employing data from the recent past. In our investigation, 10 years of data (120 points) are used. The weights of the softmax network are learned by optimizing the objective function (either profit or Sharpe ratio), using $2/3$ of the traing set data (80 points). Early stopping occurs during training at a maximum of the performance as measured on an independent validation set containing $1/3$ of the data (40 points). To make the policy more robust the process is repeated using 10 different random partitions of the data into training and validation sets. The final output is the average of the output of these 10 different learning systems. The hyperparameters of the model (the learning rate ρ, the time-scale of the moving average Sharpe ratio η, and the contribution of the the current portfolio to the rebalanced portfolio, λ) are determined based on the performance of the trained models on a holdout set composed of 100 points. Finally, the performance of the model is measured on the last 220 points of the series. Two performance measures are considered: the accumulated profit and the annualized Sharpe ratio, which is calculated as the quotient of the expected return and the standard deviation of the returns in the period considered. Policies are learned with different values transaction costs 0%, 1%, 2%, 3% and 5%. The performance measures for the different strategies are presented in Table 1. The ratio to market profit (shown between parentheses after the corresponding value of the profit) is greater than one when the wealth accumulated by the strategy considered is larger than the market portfolio. The difference to the Sharpe ratio of the portfolion (shown between parentheses after the corresponding value of the Sharpe ratio) is negative when the strategy underperforms, with respect to the mar-

ket. For zero transaction costs, both a Markowitz portfolio, and the reinforcement learning strategy perform better than the market portfolio. Since the market portfolio is never rebalanced, there is no cost associated to holding the market portfolio even when transaction costs are different from zero (other than the initial investment, no transactions are needed to implement this passive management strategy). In the presence of non-zero transaction costs, the performance of the Markowitz portfolio quickly deteriorates. Only for small transaction costs (1%), and according to the Sharpe ratio is it better than the market portfolio. By contrast, the reinforcement learning strategy improves the results of the market portfolio even when higher transactions costs are considered (up to 3%). However, for sufficiently high transaction costs (5%), the market portfolio outperforms the dynamic investment strategies considered.

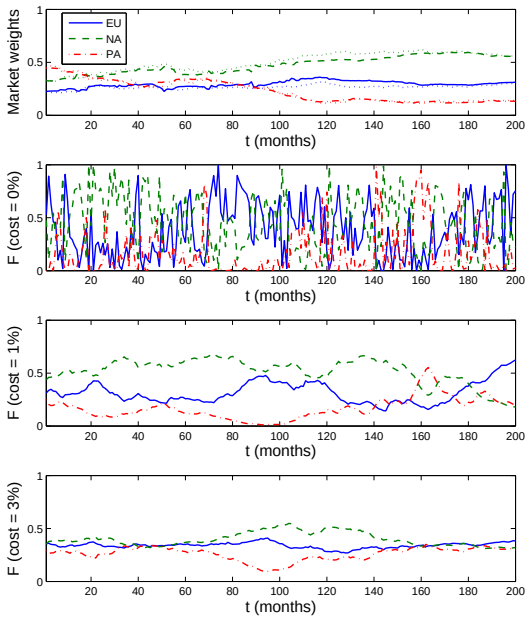

Figure 2: Evolution of portfolio weights for the market portfolio (top) and for the reinforcement learning systems for different transaction costs (0%, 1% and 3% from the top down). The figures show the sensitivity of the RL learners to vary their strategies with the level of transaction costs.

From the results obtained, several important observations can be made. As anticipated, the policy learned in the absence of transaction costs involves a large amount of portfolio rebalancing. At a given time, the investment is concentrated in the index that has had the best performance in the recent past. The switching observed for the portfolio weights is clearly undesirable i n real markets, where transaction costs make this type of behavior suboptimal. By contrast, the policies learned by the RRL system when transaction costs are considered to be smoother and require much less rebalancing. Furthermore, the portfolios selected are well-diversified, which is in agreement with financial good practices. The use of the current portfolio composition as a reference in the reinforcement learning architecture considered in (Fig. 1) is crucial for the identification of robust investment policies in the presence of transaction costs.

Current work includes extending the empirical investigation of the learning capabilities and limitations of the RRL system under different conditions. In particular, it is important to analyze its performance in the presence of correlations, autoregressive structure or heterocedasticity in the series of asset prices. Furthermore, the reinforcement learning system is being extensively tested using different financial data, and its performance compared with alternative investment strategies [11, 12]. Finally, it is also necessary to consider the possibility of investing in a risk-free asset so that strong decreases in profit can be avoided during periods in which all the portfolio constituents lose value.

References

[1] Harry Markowitz. Portfolio selection. *Journal of Finance*, 7(1):77–91, 1952.

[2] Paul A. Samuelson. Lifetime portfolio selection by dynamic stochastic programming. *The Review of Economics and Statistics*, 51(3):239–246, aug 1969.

[3] J. M. Mulvey and H. Vladimirou. Stochastic network programming for financial planning problems. *Management Science*, 38:1642–1664, 1992.

[4] F. Glover, J. M. Mulvey, and K. Hoyland. Solving dynamic stochastic control problems in finance using tabu search with variable scaling. In I. H. Osman and J. P. Kelly, editors, *Meta-Heuristics: Theory and Applications*, pages 429–448. Kluwer Academic Publishers, 1996.

[5] Ralph Neuneier. Optimal asset allocation using adaptive dynamic programming. In David S. Touretzky, Michael C. Mozer, and Michael E. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 952–958. The MIT Press, 1996.

[6] John Moody, Lizhong Wu, Yuansong Liao, and Matthew Saffell. Performance functions and reinforcement learning for trading systems and portfolios. *Journal of Forecasting*, 17(1):441–470, 1998.

[7] John Moody and Matthew Saffell. Learning to trade via direct reinforcement. *IEEE Transactions on Neural Networks*, 12(4):875–889, 2001.

[8] J.B. Detemple, R. Garcia, and M. Rindisbacher. A monte carlo method for optimal portfolios. *The Journal of Finance*, 58(1):401–446, 2003.

[9] Michael W. Brandt, Amit Goyal, Pedro Santa-Clara, and Jonathan R. Stroud. A simulation approach to dynamic portfolio choice with an application to learning about return predictability. *Review of Financial Studies*, 18(3):831–873, 2005.

[10] MSCI Inc. http://www.mscibarra.com/products/indices/equity/.

[11] Allan Borodin, Ran El-Yaniv, and Vincent Gogan. Can we learn to beat the best stock. *Journal of Artificial Intelligence Research*, 21:579–594, 2004.

[12] Amit Agarwal, Elad Hazan, Satyen Kale, and Robert E. Schapire. Algorithms for portfolio management based on the newton method. In *Proceedings of the 23rd international conference on Machine learning, ICML 2006*, pages 9 – 16, New York, NY, USA, 2006. ACM.