Meta-Machine Learning: Automatic Programming of Trading Strategies

David Andre, PhD
Cerebellum Capital, LLC
dandre@cerebellumcapital.com
www.cerebellumcapital.com

Abstract

The field of machine learning (ML) has advanced at a rapid pace in the past decade, achieving feats long thought to be much further off, including defeating top Go players (Silver et al, 2016), learning to play Atari games (Mnih et al, 2013), beating human accuracy at image recognition (He, K., Zhang, X., Ren, S., & Sun, J., 2016), and language translation (Graves et. al. 2016).

Automatic programming is a branch of ML that produces predictive input-output mappings, learned from data, in the form of computer code rather than as parameters of a predetermined algorithm. Several approaches to automatic programming, including search (Koza, Bennett, Andre, Keane, 1999; Looks, Goertzel, Pennachin, 2005), gradient descent (Tran, Hoffman, Saurous, Brevdo, Murphy, and Blei, 2017), and Bayesian optimization of ML pipelines (Feurer, Klein, Eggensperger, Springenberg, Blum, and Hutter 2015) have achieved human-competitive solutions to complex problems in a variety of applications.

Quantitative researchers historically have been skittish of such complex techniques for fear of finding spurious correlations that will not generalize to the future. In fact, it is common to penalize the performance of one algorithm or model based on how many models were considered (Harvey and Lieu, 2015). This, to us, seems somewhat unscientific. It is not as if predictions of the theory of gravity are any less likely to be true if a researcher considers additional complex epi-cycle based models: proper scientific method prescribes additional tests to separate wheat from chaff. In this work, we describe and demonstrate a machine-learning based Bayesian approach to assessing the quality of machine-learned signals using features of the signal, the program/procedure that created that signal, the consistency of the signal, and the signal's performance both before and after discovery. We do not deny that many forms of bias exist – such data snooping bias (Lo, 1994), selection bias, survivor bias, and even signal combination bias (Novy-Marx, 2015) – we believe instead that these biases can be measured and mitigated. By automating more of the discovery process, better tests for overfitting are possible.

Our method does not follow a specific strategy but instead adapts to prevailing conditions. For demonstration purposes, we apply it to a simple single-stock prediction model where a stock is held long or short and beta-balanced with the SPY ETF. We represent signals as simple tree-based computer programs with terminals (inputs/features) of technical fundamental indicators and functions including arithmetic, simple statistics (boxcar mean, standard deviation, and z-score), Winsorization, and simple logical functions. The system searches over these programs using an evolutionary computation process and builds leaderboards of the best performers (according to an "in-sample" fitness function that penalizes for complexity and rewards for Sharpe ratio) that are not too correlated with other leaderboard members. This results in a leaderboard with hundreds of members per time-slice (a year in this experiment). To choose

what to trade in each time-slice, we look at all leaderboard candidates from previous time-slices and learn a classification function ("sieve") that predicts out-of-sample performance from features such as pre- and post-discovery performance, consistency of performance, and features of the learned program, including complexity. We then select and combine (using equal weighting) those signals that are predicted to do well out-of-sample, thus mitigating the signal combination bias, in agreement with (Novy-Marx, 2015).

The average out-of-sample performance of leaderboard candidates is only barely positive (52%) before the application of the sieve, but 75% after the sieve. Combining out-of-sample signals for 20 stocks results in a combined strategy with out-of-sample costed performance of about 10%/year and a Sharpe ratio of just over 1 for the years 2010-2015.

Feurer, Matthias, Aaron Klein, Katharina Eggensperger, Jost Springenberg, Manuel Blum, and Frank Hutter. "Efficient and robust automated machine learning." In *Advances in Neural Information Processing Systems*, pp. 2962-2970. 2015.

Graves, A., Wayne, G., Reynolds, M., Harley, T., Danihelka, I., Grabska-Barwińska, A., Colmenarejo, S.G., Grefenstette, E., Ramalho, T., Agapiou, J. and Badia, A.P., 2016. Hybrid computing using a neural network with dynamic external memory. *Nature*, *538*(7626), pp.471-476.

Harvey, Campbell R., and Yan Liu. "Backtesting." *The Journal of Portfolio Management* 42.1 (2015): 13-28.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770-778).

Koza, John R, Bennett, F., Andre, D., and Keane, M. *Genetic programming III: Darwinian invention and problem solving*. Vol. 3. Morgan Kaufmann, 1999.

Lo, A. W. Data-snooping biases in financial analysis. in *AIMR Conference Proceedings* 1994, 59–66 (CFA Institute, 1994).

Looks, Moshe, Ben Goertzel, and Cassio Pennachin. "Learning computer programs with the bayesian optimization algorithm." *Proceedings of the 7th annual conference on Genetic and evolutionary computation*. ACM, 2005.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Novy-Marx, R. *Backtesting strategies based on multiple signals*. (National Bureau of Economic Research, 2015).

Silver, David, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser et al. "Mastering the game of Go with deep neural networks and tree search." *Nature* 529, no. 7587 (2016): 484-489.

Tran, D., Hoffman, M. D., Saurous, R. A., Brevdo, E., Murphy, K., and Blei, D. M. (2017). Deep Probabilistic Programming. ArXiv e-prints. arXiv:1701.03757v1