AlphaCapture.

Author: Ashot Ordukhanyan (orduka@gmail.com)
Published Date: 09/30/2024

Abstract

Alpha Capture is an application created for submission to the Alphathon 2024 competition, organized by the Society of Quantitative Analysts (SQA).

More details about Alphathon 2024 can be found here.

AlphaCapture is a proof-of-concept application designed to illustrate portfolio optimization techniques that a portfolio manager in a hedge fund might use to build a long-short portfolio, given alpha, transaction cost, risk and shorting cost projections.

This document, as well as all of the source code associated with the application is made publically available under the terms of Apache 2.0 license https://github.com/ashotordukhanyan/AT2024Port

Introduction

Generating optimal portfolio selection strategy is a challenging and complicated problem.

Problem space is vast and potential solutions are plentiful. In addition to analyzing alpha projections, portfolio manager must derive a methodology of translating alphas into actionable trades, determine optimal portfolio rebalance schedules and balance out transaction cost, risk and expected return considerations in designing the portfolio.

In this submission I attempt to demonstrate a number of alternative techniques that can be utilized and their relative pros and cons.

Alpha Analysis.

Prior to delving into solving the stock selection problem, careful analysis of the strength of alpha signals available to the PM is in order. Competition organizers generously provided 6 years' worth of alpha signals and realized returns, as well as data associated with the Northfield risk model (factor loadings for our stock universe, factor definitions, variances and covariances).

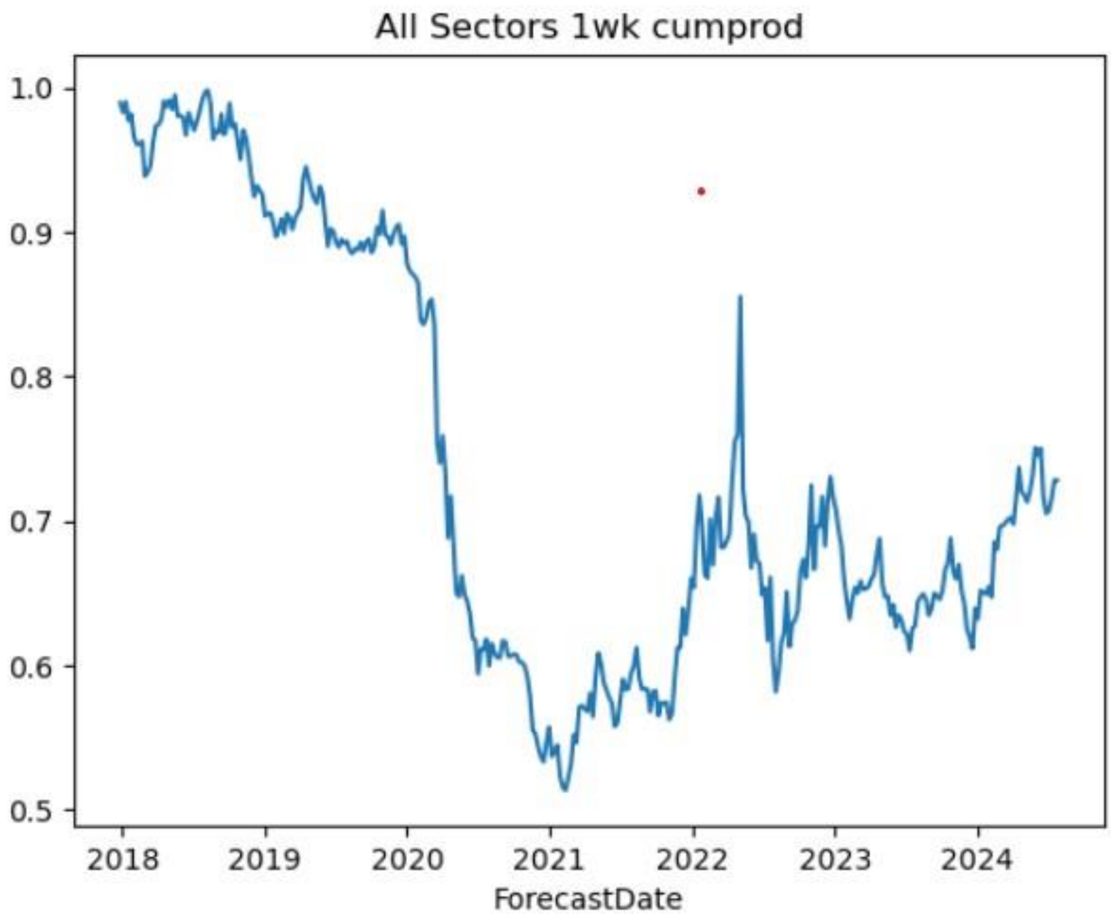
Using this data initial exploratory analysis was done on alphas and returns to determine the strength and optimal horizon for the alpha. It appears that the alpha signal is very weak (almost random).

In particular, a simple linear regression performed on returns at various horizons (1-4 business days and 1-4 weeks) showed very little statistical connection between predicted alphas and actual realized returns:

Horizon	slope	intercept	rvalue	pvalue	stderr
Return_1bd	0.0068	3.1886	0.0054	0.0308	0.0032
Return_2bd	0.0083	3.1877	0.0088	0.0004	0.0023
Return_3bd	0.0062	3.1879	0.0084	0.0008	0.0018
Return_4bd	0.0046	3.1875	0.0069	0.0055	0.0016
Return_1wk	0.0024	3.1882	0.0041	0.0996	0.0015
Return_2wk	0.0022	3.1877	0.0055	0.0273	0.0010
Return_3wk	0.0019	3.1874	0.0058	0.0208	0.0008
Return_4wk	0.0014	3.1874	0.0050	0.0451	0.0007

Figure below demonstrates a hypothetical cumulative return from a simple long-short strategy that would go long/short stocks based on the sign of the expected alpha in an equal weighted fashion.

As one can see, even without taking transaction and shorting costs into account this strategy will lose money over the term of the dataset. Given that the objective of this exercise is to demonstrate a long short strategy that uses the provided alpha to form the portfolio, this data exploration exercise set expectation of what we can expect the PNL of such a portfolio to be – it will be negative.



Lastly, I analyzed whether the magnitude of the alpha estimate (as opposed to the direction) is statistically meaningful. Alpha forecasts were divided into 4 equal sized quartiles and average realized returns for stocks in those quartiles were measured for the same representative set of horizons (1-4 business days + 1-4 weeks).

	Return_1bd	Return_2bd	Return_3bd	Return_4bd	Return_1wk	Return_2wk	Return_3wk	Ret
	avg	avg	avg	avg	avg	avg	avg	avg
Alpha								
Q1	0.0385	0.1239	0.1330	0.2766	0.2734	0.5000	0.7708	
Q2	0.0185	0.1138	0.0958	0.2522	0.2316	0.4374	0.6728	
Q3	0.0412	0.1533	0.1628	0.2822	0.2604	0.5042	0.7353	
Q4	0.0510	0.1847	0.2252	0.3515	0.3005	0.5680	0.8479	

Looks like the optimal holding period for a trade that would go long highest alpha quartile and go short the lowest one is 3 business dates (23 – 13 basis point spread). But given that in our simulation alphas are published weekly and we are constrained to hold all of the “capital” in stock and none in cash that effectively sets a lower limit for our holding period to be 1 week. The relative long/short returns of 3 basis points there will be wiped out by transaction and shorting costs.

Methodology

A number of different of different portfolio selection methodologies were analyzed and their relative performance was compared. Implementation of all the methodologies can be found in the provided trade.py file. Short description of what each methodology does:

- HighConviction – Trading strategy that places all of its capital into top/bottom n percentile of expected alpha returns, without any consideration of risk/transaction costs
- Indiscriminate – Trading strategy that trades every name in the alpha universe. Weights are allocated proportionally to the alpha forecasts.
- OptimizingTrader – Trading strategy that solves a classical Markowitz optimization problem and rebalances portfolio by considering portfolio variance, transaction costs, expected alphas and costs of carrying short inventory. This strategy we ultimately chosen as the "winner" (although I admit I was not clear whether the criteria by which the submission will be ultimately judged would agree with this choice).

Optimizing Trader.

Solves the problem of minimizing risk (portfolio variance) + transaction costs (linear function of traded notional by stock) + shorting cost (linear function of short notional by stock) – alpha (linear by stock).

Each term in the objective function has a multiplier. Through experimentation I ended up with risk mutliplier= 5, alpha_multiplier = 5 and other multipliers = 1.

Linear constraints are utilized to ensure that sum of all resulting weights is 1 for longs and -1 for shorts.

Concentration constraint is added to ensure that no single weight exceeds .25 in absolute value.

To facilitate transaction costs (function of absolute change in weights) and shorting costs (function of weight iff. weight < 0) I actually utilized 4 variables per asset.

They denote: BuyLong, SellLong, BuyShort and SellShort trades (changes in weights) respectively.

Implementation of this algorithm can be seen in portopt.py and trader.py python modules.

A Note on Data Quality.

Some of the data provided for this exercise was (intentionally?) corrupt and some data wrangling needed to be done to use it. Some of the data cleanups and checks can be seen in dataload directory as well as the associated pytest files (test_dataload.py).

In particular – the following choices were made by me in dealing with inconsistent data:

- All 9 digit cusips were converted to 8 digit cusips (check digit removed)
- In the case of files referencing duplicate cusips (alpha and risk model had these instances) – duplicated were removed with the last row being the effective one
- Securities that were included into the universe but did not have returns available on the date of the simulated rebalance were excluded from the universe