Report of Alg Trading Using Price Spreads

Author. 陳郁安 Student ID. B10703014

Strategy.

Here, I utilize the concept of mean-reversion strategy to trade; that is, the price spreads off the estimated mean is used as a trading signal, indicating to sell when the price is over the mean, while indicating to buy when the price is below the mean. Regularly, the price of a single stock follows random walk, and the return is mean-reverting to zero. In order to make the return profitable, I try to construct a portfolio and make the price to be as mean-reverting as possible. Weights for each stock in the portfolio is the hedge ratio found from the linear regression:

$$y = \mathrm{h}_1 y_1 + h_2 y_2 + \cdots + h_n y_n$$

y is the real value of price I want to estimate. By construction, the estimated y is a stationary time series; namely, y is constructed to be historically mean-reverting.

To simplify, I choose only two stocks to construct the portfolio. One is the stock of Electronic Arts, the other one is the stock of Take-Two Interactive. The two company are both in the video games industry and from USA. In this case, the regression to the will look like:

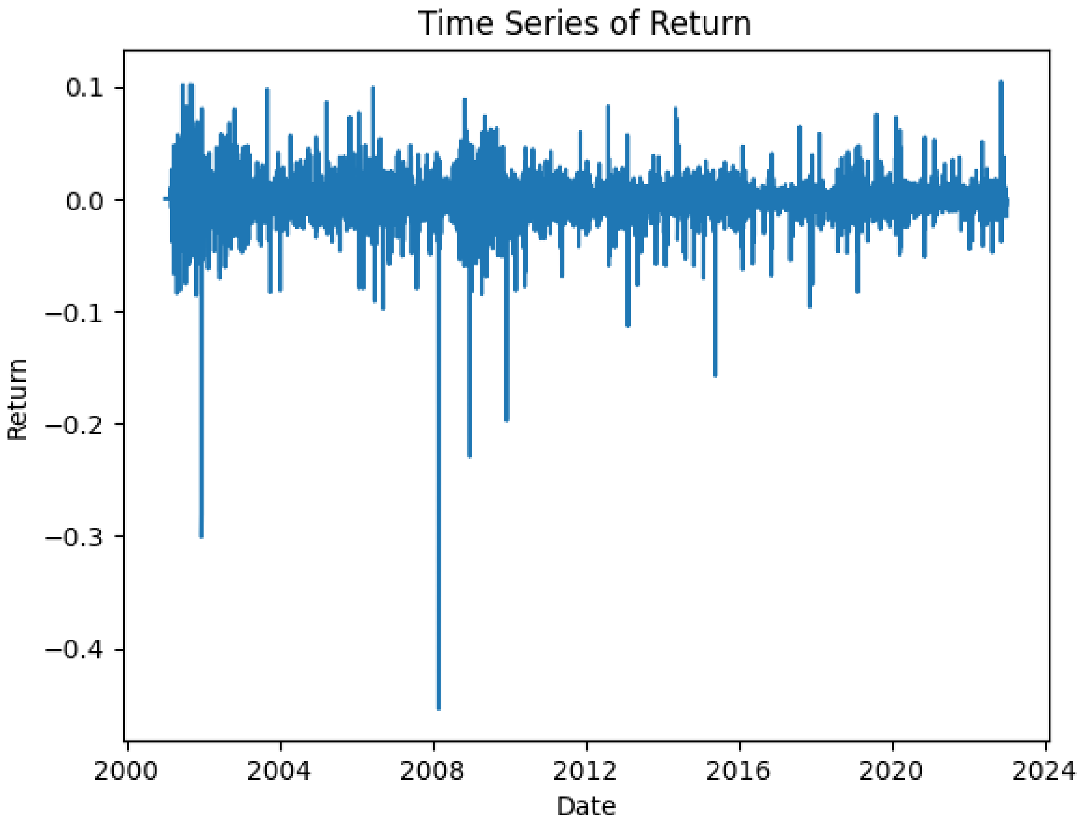
$$y = h_{\mathrm{EA}} y_{EA} + h_{TTWO} y_{TTWO}$$

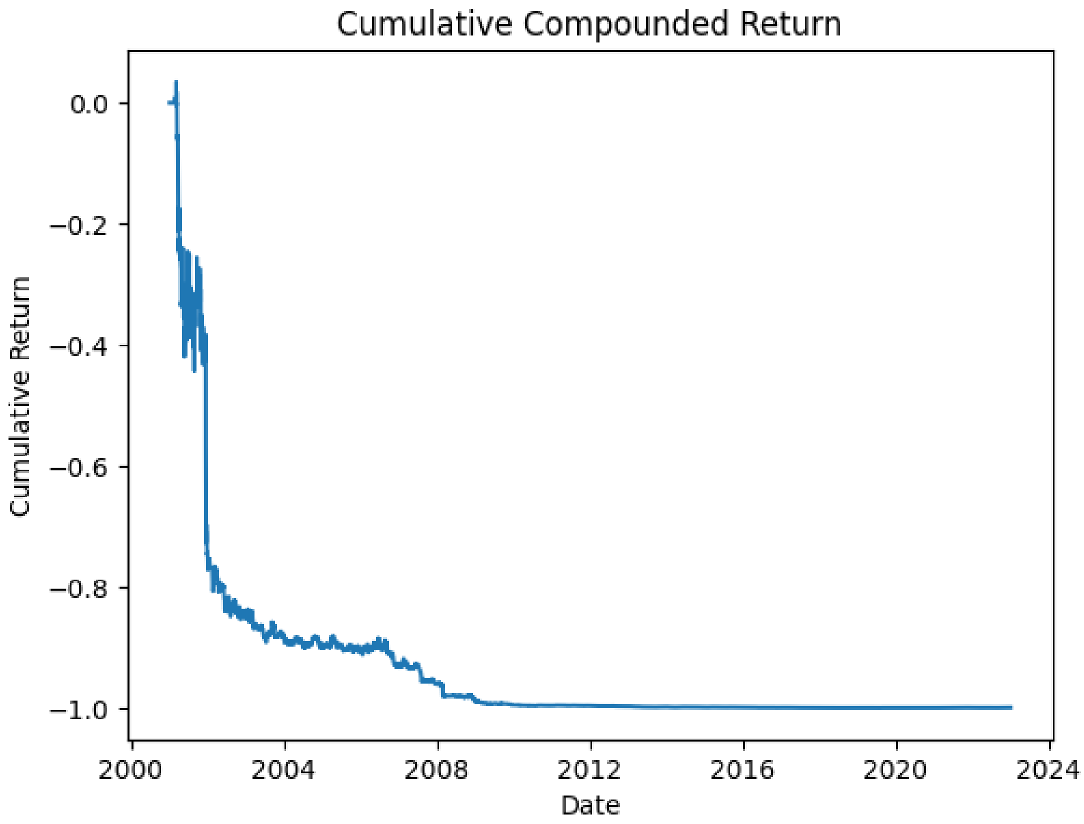
OLS will be used in estimating the hedge ratio. The sample for price y is taken from close price in the history. Coefficient h's should change dynamically, as there is always a new close price tomorrow. Here, the coefficient h's are re-estimated on a daily basis, with a lookback period of 20 days. The price spread is used as the trading signal. In execution, the number of share (unit of the portfolio) to own is the negative Z-score of the spread, and the hedge ratio determine to long or to short.

Backtesting and Performance Analysis.

In this backtesting, I simulate the execution of the aforementioned strategy from 2001 to 2023. The code is written in Python with IPython notebook, using yfinance to extract the information of stocks. The exhibition of the execution can be seen in github - https://github.com/cchdood/Mean-reversion-trading/blob/main/Trading_mean_reversion.ipynb .

The result of the strategy is, unfortunately, very awful. The average APR is -0.269741, with a sharpe of -0.846080. The following two graphs are the time series of the returns and of the cumulative returns with this strategy:





Therein lies an interesting fact: instead of lingering around zero, the cumulative return is often decreasing, which indicates that there's very likely to be a certain factor which makes the strategy destined to lose money for most of the time.

The reason of the continuous loss could probably be the long-lasting market sentiment in gaming industry. The release date of a new game could cross a long time from the announcement of the game. During the time, the company utilize a wide variety of ways to hype up the potential buyers. This kind of advertising could hype up or disappoint the gamers (or the investor), and the market sentiment could last until someone actually plays the game. So, I assume that the direction of the stock price changes more often after and near the release date, and less likely to change on the other time.

With this assumption, since the direction of prices, in long-term point of view, rarely change. I should expect the price will keep rising when it's overvalued (when people are hyped up), and expect it to keep declining when it's undervalued (when people are disappointed). So I try to reverse the strategy by making a change: the number of units to own is, instead of negative, the positive Z-score of the spread.

The result is satisfying. The average APR becomes 0.241369; and, the sharpe ratio is 0.846080. The following two graphs are the time series of the returns and of the cumulative returns, respectively.