Spring 2024 STAT 541 Final Project: Sequential Investment and Universal Portfolio Algorithms

Wenhao Pan
Department of Statistics
University of Washington
Seattle, WA, 98105
wenhaop@uw.edu

1 Introduction

1.1 Setting

We consider a sequential investment problem in a market with $d \geq 2$ stocked explained by Orabona [2019]. We observe a sequence of arbitrarily chosen nonnegative *market gains* vectors $w_1, \ldots, w_T \in \mathbb{R}^d_{\geq 0}$. For example, one way to define w_t for any time $t = 1, \ldots, T$ is that i-th coordinate of w_t is the ratio of the adjusted close price of i-th stock at time t to that at time $t-1$, i.e., $w_{t,i} = \frac{\text{adjust price at time } t}{\text{adjust price at time } t-1}$ for all $i = 1, \ldots, d$, $t = 1, \ldots, T$. An investment strategy for time t is specified by a vector $x_t \in \mathbb{R}^d$ such that $0 \leq x_t \leq 1$ and $||x||_1 = 1$. i-th coordinate of x_t specifies the fraction of the wealth allocated for i-th stock at time t. With the definition of w_t above for any time t, $x_{t,i}$ represents the fractions of the wealth at time $t-1$ to buy i-th as soon as the market opens and sell all the shares right before the market closes at time t. With initial health of \$1, after T rounds, our wealth Wealth_T is

$$\text{Wealth}_T = \sum_{i=1}^{d} \text{Wealth}_{T-1} w_{T,i} x_{T,i} = \text{Wealth}_{T-1} w_T^\top x_T = \prod_{t=1}^{T} w_t^\top x_t.$$

We aim to design or implement different *portfolio selection algorithms* that select $x_1, \ldots, x_t$ to maximize W_T.

1.2 Regret

Like bandit algorithms, we can define the regret of a portfolio selection algorithm by comparing its wealth accumulation against the *best constant rebalanced portfolio* (BCRP). *Constant* means that the allocation fraction of wealth at each stock is the same at each time, and *best* means that such fraction u_* maximizes $\text{Wealth}_T(u) = \prod_{t=1}^{T} w_t^\top u$.

Since wealth accumulation is multiplicative, we define the regret of any portfolio selection algorithm after T rounds as the ratio of BCRP's wealth to the algorithm's.

$$\text{Regret}_T = \frac{\text{Wealth}_T(u_*)}{\text{Wealth}_T} = \prod_{t=1}^{T} \frac{w_t^\top u_*}{w_t^\top x_t}.$$

2 Algorithms

Next, we describe the portfolio selection algorithms we implemented for numerical experiments.

Preprint. Under review.

Algorithm 1 F-Weighted Portfolio Selection

Require: $F : \Delta^{d-1} \to \mathbb{R}$ probability density function

Wealth$_0 = 1$
for $t = 1$ to T **do**
  Set $x_t = \frac{\int_{\Delta^{d-1}} x\text{Wealth}_{t-1}(x)dF(x)}{\int_{\Delta^{d-1}} \text{Wealth}_{t-1}(x)dF(x)}$
  Receive $w_t \in \mathbb{R}^d_{\geq 0}$
  Wealth$_t$ = Wealth$_{t-1} \cdot w_t^\top x_t$
end for

2.1 F-Weighted Portfolio Selection

Cover [1991] proposed the F-weighted portfolio selection algorithm described in Algorithm 1. Δ_{d-1} stands for a d-dimensional probability simplex. Example choices for F are uniform distribution and Dirichlet$_d\,(1, \ldots, 1)$, and we use the uniform distribution. The allocation $x_t$at each time t is the weighted average of all possible constant rebalanced portfolios by its generated wealth until time $t-1$ and its probability under F.

2.2 Constant Rebalanced Portfolio

Algorithm 2 Constant Rebalanced Portfolio

Wealth$_0 = 1$
for $t = 1$ to T **do**
  Set $x_t = (1/d, \ldots, 1/d)$
  Receive $w_t \in \mathbb{R}^d_{\geq 0}$
  Wealth$_t$ = Wealth$_{t-1} \cdot w_t^\top x_t$
end for

The constant rebalanced portfolio algorithm (CRP) is explained in Algorithm 2.

2.3 Best Constant Rebalanced Portfolio

BCRP is identical to CRP (Algorithm 2) except that $x_t = u_*$ as explained in Section 1.2.

2.4 Random Rebalanced Portfolio

The random rebalanced portfolio algorithm is identical to CRP (Algorithm 2) except that $x_t \sim$ Dirichlet$_d\,(1, \ldots, 1)$. Since Dirichlet$_d\,(1, \ldots, 1)$ is completely uninformative, we uniformly randomly select the rebalanced portfolio at each time.

2.5 All-in Rebalanced Portfolio

The all-in rebalanced portfolio algorithm is identical to CRP (Algorithm 2) except that x_t is uniformly randomly drawn from $\{e_1, e_2, \ldots, e_d\}$ where e_i is i-th standard basis vector for $i = 1, \ldots, d$.

3 Experiments

To test and compare the performance of all five algorithms on real-world data, we implement them in Python and run them on the real stock market data with significant help from `https://github.com/Marigold/universal-portfolios` and `yfinance` package. The code is attached at the end of the report.

3.1 Stock Market Data

We use the adjusted close prices of FAANG companies (Meta, Amazon, Apple, Netflix, Google) in the last five years (from 06/02/2019 to 06/02/2024). The stock prices are plotted in Figure 1.

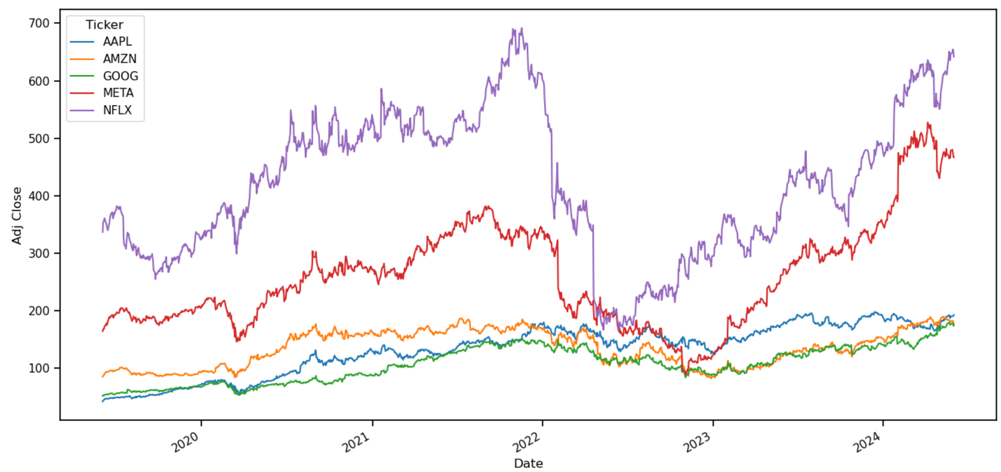

Figure 1: Adjusted close prices of FAANG companies from 06/02/2019 to 06/02/2024

3.2 Result and Analysis

The result of running five algorithms explained in Section 2 on the data plotted in Figure 1 is shown in Figure 2. The most interesting observation might be that the all-in algorithm, which seems nonsense, manages to double our initial wealth after five years. This observation could be explained by the general uprising trend of FAANG's stock price, except for the well-known decline in 2022, as observed in Figure 1. The next interesting observation is that the random algorithm performs at least as well as CRP and F-weighted algorithms. Again, this could be explained by the fact mentioned before. Thus, a natural future extension is to run the algorithms on the stocks that are more stable.

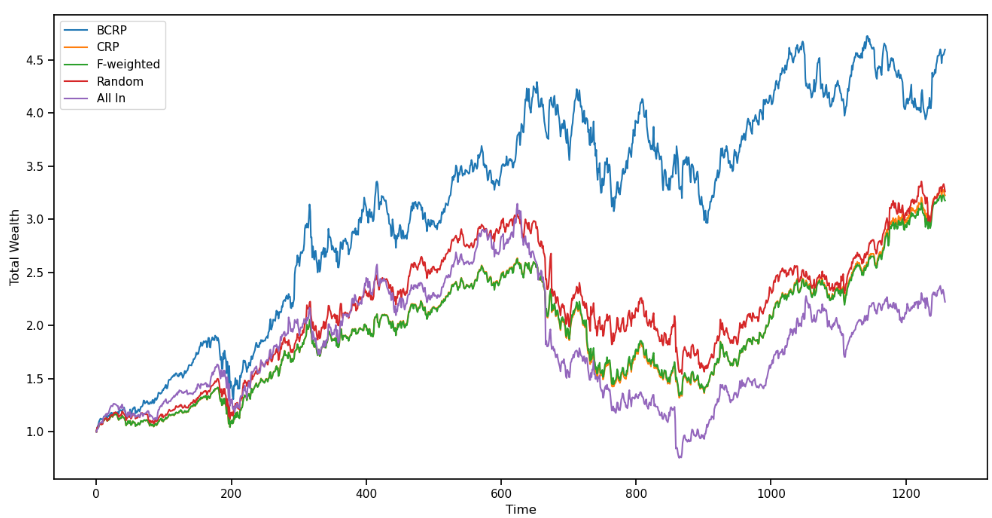

Figure 2: Wealth accumulation of five algorithms.

References

Thomas M Cover. Universal portfolios. *Mathematical finance*, 1(1):1–29, 1991.

Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.