

ECE 592 – Topics in Data Science

Project 3: Optimization

Due: October 30, 2019

1 Administrative Instructions

1. For any clarification or doubts, the TA Hangjin Liu (hliu25 AT ncsu DOT edu) is in charge of homeworks and projects. She should be your first point of contact on homework- and project-related issues.
2. The project can be submitted individually, in pairs, or triples.
3. You should submit electronically through Moodle by midnight the day that the homework is due.
4. Your report should describe any mathematical derivations, responses to questions, and results including any plots. Please justify your answers carefully, including providing interesting insights when relevant.

2 Motivation

A typical investment strategy that the finance community often recommends to long term investors, for example people saving for retirement, involves (i) diversifying between several assets, (ii) holding the same (or similar) proportion or percentages of your wealth in each asset over time.

Diversification means owning multiple assets, and this is advantageous if their price changes are weakly (or even negatively) correlated. To see why diversification can be useful, recall from class that the average of two independent and identically distributed (i.i.d.) random variables (RVs), each with mean μ and variance σ^2 , has the same mean μ but with a smaller variance, $\sigma^2/2$. If the two original RVs represent price changes of two financial assets, then the new averaged RV represents the price changes of a portfolio that invests half in each of them. This averaged new RV has the same expected profit (μ), yet its variance is smaller, meaning that price changes are typically smaller. *All other things being equal, people prefer a portfolio whose price is less volatile.* (In practice, the RVs will have different means and variances, and there may be correlations or dependencies between them. In this case, we will likely not hold equal amounts in each financial asset.)

Rebalancing involves holding the same proportion in each asset over time. As the market moves you away from your target, you (re)balance your portfolio periodically to maintain the desired proportions.

Why is rebalancing helpful? Many individuals partition their financial assets primarily between stocks and bonds, with some cash often held for liquidity purposes. Because stock prices typically fluctuate more than bond prices, recent performance in stocks tends to bias people's viewpoints. Some people tend to get over-confident after stocks are markedly up. For example, in January 2018 the stock market had gone up over 50% since February 2016 and over 300% since March 2009. Some investors became optimistic, thinking that prices will keep going up, and they increased their stock positions. However, these investors were handed multiple declines of 10-20% later in 2018. On the other hand, after stocks are down sharply (for example, early in 2009), many people are pessimistic and sell stocks. You can see how trying to keep a fixed allocation makes you more disciplined.

A mathematical reason for believing that rebalancing is helpful arises if we model asset price changes as i.i.d. With i.i.d. price changes, it can be shown rigorously that at the start of each trading period we want to hold the same proportions in different assets; perhaps more in assets whose μ is larger, and less if σ^2 is larger. Note that the assumption that price changes are independent appears, for example, in the **efficient market hypothesis**. This hypothesis is very well known in the finance community; it assumes that all current information is reflected in the prices of financial assets, implying that any dependencies between price changes are modest. While the hypothesis is somewhat imprecise, and there are some inefficiencies (profit opportunities) in financial markets, substantial effort and expertise are required to profit from these opportunities.

Finally, a recent blog post discussed how rebalancing can be considered as analogous to noise cancelation. Recall that noise canceling headphones generate a new acoustic signal whose values are negatively correlated with the noise, and superposition between the two acoustic signals results in the user enjoying a quiet flight. Similarly, rebalancing creates a superposition of multiple financial signals whose returns (price changes) are weakly or even negatively correlated, resulting in reduced volatility. Those of you who want to see how noise cancelation and financial returns were visualized can look at the blog post,

<https://portfoliocharts.com/2019/10/07/how-to-build-a-noise-cancelling-portfolio/>

The main idea in this project will be to optimize the proportions or weights that you will hold in different financial assets.

3 Investment Strategy Formulation

Portfolio return during one time period: Consider holding a portfolio comprised of D financial assets over N time periods, where time periods are typically trading days or months. For each time period n , the financial return of the entire portfolio is

$$r^n = \sum_{d=1}^D X_d^n \times h_d^n, \quad (1)$$

where X_d^n is the price change of asset d at time period n , and h_d^n is the proportion invested in asset d at time period n . Note that X_d^n is the relative price, not absolute price,

$$X_d^n = \frac{P_d^n}{P_d^{n-1}}, \quad (2)$$

where P_d^{n-1} and P_d^n are prices of asset d at time periods $n-1$ and n , respectively. For example, if the price of asset d is $P_d^n = 50$ at time n and $P_d^{n+1} = 48$ at $n+1$, then $X_d^{n+1} = P_d^{n+1}/P_d^n$ here is $48/50 = 0.96$. In practice, X_d^n will often be close to 1.

Portfolio structure: The overall goal of the project is to optimize h_d^n . To do so, we add two constraints,

$$h_d^n \geq 0 \quad \text{and} \quad \sum_{d=1}^D h_d^n = 1. \quad (3)$$

The first constraint in (3) means that financial assets are only owned. In some markets, it is possible to borrow an asset from an investor who owns it, and return it to them later. This procedure is called shorting a financial asset, and it can be shown to produce a profit inversely proportional to the asset's price change, meaning that $h_d^n < 0$. Shorting is not available to all investors, and will not be considered in our project.

The second constraint in (3) is that we are investing all of our wealth (a proportion 1) in total over the D assets. We discussed earlier that X_d^n is often close to 1 in practice, and so r^n too is often close to 1. In particular, the values of X_d^n and constraints for h_d^n ensure that r^n is non-negative.

Optimal weights: Given these constraints, what should h_d^n be? Recall that we will invest over N time periods (more precisely, we will train our algorithm over N periods), and our overall return is

$$r = \prod_{n=1}^N r^n. \quad (4)$$

Because we are focusing on rebalanced portfolios with constant weights, $\mathbf{h} = [h_1, h_2, \dots, h_D]$, our goal is to compute the vector \mathbf{h} that maximizes the returns r (4).

In principle, the holdings could depend on d and n , and be arranged in a matrix \mathbf{H} , where h_d^n is an element of \mathbf{H} . However, because we require constant proportional holdings over time, we only consider a vector \mathbf{h} . Recall that r^n is non-negative, and we can modify the optimization of overall return, r , by taking the logarithm of this expression,

$$\log(r) = \sum_{n=1}^N \log(r^n) = \sum_{n=1}^N \log \left(\sum_{d=1}^D X_d^n \times h_d \right).$$

It can be shown that the logarithm operator is concave, and our maximization involves convex optimization.

4 Toy Example

Consider a simple scenario with only two stocks, $D = 2$. Additionally, their prices will follow a simple pattern. The price of Stock 1 remains constant over time, for example $1, 1, 1, 1, \dots$; and the price of Stock 2 goes up and down periodically, $1, 2, 1, 2, 1, 2, \dots$.

In this contrived example, a shrewd investor would alternate their holdings between Stocks 1 and 2, resulting in wealth doubling every 2 trading periods. If the returns sequences is rearranged, then X_2^n , the price change of Stock 2 in time period n , is either 0.5 or 2, each with probability 0.5. That constant vector \mathbf{h} should we use for these rearranged i.i.d. returns?

- (a) In the first scenario, you will either invest your wealth using 100-0 split. Using $\mathbf{h} = [1 \ 0]$, all the wealth is invested in Stock 1, r^n is always 1, and the product over n is $r = 1$ (4). Using $\mathbf{h} = [0 \ 1]$, all the wealth is invested in Stock 2, and it will fluctuate between \$1 and \$2 (or between \$0.5 and \$1).
- (b) In the second scenario, you will invest your wealth equally, i.e., $\mathbf{h} = [0.5 \ 0.5]$, at the start of each day. How will your wealth evolve?
 Day 1: you have \$1 in total to start with, and invest \$0.50 in each stock. Stock 1 went from 1 to 1 and you still have \$0.50 there; Stock 2 went from 1 to 2 and you have \$1; in total, your wealth is \$1.50.
 Day 2: you start with \$1.50, invest \$0.75 in each, Stock 1 maintains its value of \$0.75, Stock 2 goes from 2 to 1, resulting in \$0.375; in total, your wealth is \$1.125.
 You can continue to iterate this way and make various calculations. Because the stock prices are periodic, each 2 day period increases your wealth by 12.5%. For example, after 100 days your wealth will be $1 \times 1.125^{(100/2)}$.
- (c) In the third scenario, we again invest equally, $\mathbf{h} = [0.5 \ 0.5]$. However, instead of the deterministic price evolution 1, 2, 1, 2, ... of Stock 2, we assume that X_2^n is either 0.5 or 2, each with probability 0.5. In this case, $\mathbf{h} = [0.5 \ 0.5]$ will result in *average growth* of 12.5% per 2 days.

Task: Is $\mathbf{h} = [0.5 \ 0.5]$ optimal? Does it maximize the rate of growth? Does it maximize r ? Please simulate this problem using any optimization method you choose, and find the optimal allocation.

5 Data Retrieval and Implementation

- (a) There are various online software packages for downloading data from various sources. One data set we identified is

https://stanford.edu/class/ee103/julia_files/asset_prices.csv

One option is for you to download this CSV file and run the rest of the project with it. Below we make some comments about the results we obtained for this file.

Another option is to download closing data prices for any of a variety of indices and / or financial assets. These prices can be daily over a period of several years. One free data source that you should be able to download from is Yahoo Finance.

- (b) Start with two assets, $D = 2$. In this case, $h_1 + h_2 = 1$, and we only need to optimize one parameter. This can be interpreted as optimizing over a 1D (1 dimensional) grid.

Task: Compute the optimal \mathbf{h} for $D = 2$.

- (c) Increase to $D = 3$ assets. You now have a 2D grid, and make sure that $h_1 + h_2 \leq 1$. With a small number of assets, D , it is often optimal to invest everything in the asset that grows the most over that time period. Our experiments with the Stanford file (above) provided non-trivial \mathbf{h} for some smaller problem sizes.

Task: Compute the optimal \mathbf{h} for $D = 3$.

- (d) So far, you could have used a brute force approach that searches over a finite grid of possible parameters. However, moving to $D = 4$ and beyond may require a more sophisticated approach. Recall that the logarithm function is concave, and our resulting convex optimization problem should be solvable with coordinate descent, gradient descent, or other approaches.

Task: Read more about any of these viable approaches for convex optimization, and implement them yourself in Python or Matlab. Something simple like coordinate descent is fine! Make sure to document your code.

- (e) Confirm your optimization code for $D = 2$ and $D = 3$ using a brute force approach. To do so, for $D = 2$ you can set $h_1 = 0 : 0.01 : 1$ (in Matlab) and $h_2 = 1 - h_1$, plug $\mathbf{h} = [h_1 \ h_2]$ into your algorithm, and verify that the resulting optimal \mathbf{h} is close to what your code outputs. For $D=3$, you could set $h_1 = 0 : 0.01 : 1$, $h_2 = 0 : 0.01 : 1 - h_1$, and $h_3 = 1 - h_1 - h_2$; it can be shown that this approach will examine roughly $0.5 \cdot 100^2$ \mathbf{h} vectors. Another way to confirm your results is that the gradient should be near-zero at the optimal point. However, that requires parametrizing the last number, h_D , to be defined as $1 - \sum_{d=1}^{D-1} h_d$, which may be complicated.

Task: Confirm your code using brute force for $D = 2$ and $D = 3$.

6 Possible Extensions

Below are possible directions for extending the project. These could be considered as part of a final project.

- (a) **Shorting:** Recall that shorting allows you to maintain a negative exposure to some financial asset by borrowing it from a current owner and returning it later. In this project, we assumed that $h_d^n \geq 0$, meaning that shorting was not allowed. How about changing the constraints in a way that allow shorting?
- (b) **Risk:** Some people define financial risk as the volatility of price changes. In our project, we only optimized returns r and ignored the volatility of price changes. However, this is somewhat naive, and you can optimize “risk adjusted returns.” That is, over a long duration of time, holding 100% in stocks will likely be profitable, but it is a volatile portfolio, and the price changes may be severe. Many individuals will get emotional and make mistakes as outlined above. Instead, risk adjusted returns involve optimizing

$$\text{returns} - \text{const} \times \text{volatility},$$

where the volatility could be the variance of returns, their standard deviation, or some other metric; the constant depends on the individual investor’s trade-off between risk and return. In words, many investors will prefer a slightly lower return if the fluctuations are markedly lower. Can you account for risk in your optimization?