A Monte Carlo Simulation-based Portfolio Risk Analyzer

Edward She

Advisor: Stephen Slade

Introduction

According to Gallup, roughly half of all Americans say they invest in stocks. In general, this is much better for the average American than storing money in low interest rate savings accounts, as the stock market has averaged 7% inflation-adjusted annual returns for the past sixty years. However, this 7% number is a long-term average over the entire market, meaning both that yearly returns may vary significantly and that a given portfolio may have much different returns and variation.

This fact has long been well understood by institutional investors, and it has given rise to an entire field of finance known as risk management. Risk is a factor of high importance in portfolio selection, and a large advantage of institutional investors over retail investors is their ability to manage and mitigate risk. Yet, many people choose to invest themselves, for a plethora of potential reasons. Some may have preferences towards certain stocks they want to hold, while others may want to avoid management fees. Regardless of reason, the average investor has much less of an idea of how risky a given portfolio may be, when this should be a factor of great importance. For my senior project, I hope to simplify the idea of risk for the average investor and even the playing field by creating a web application that will assess the risk and potential returns of a given portfolio. This will be accomplished through Monte Carlo simulations based on historical data and user-configurable inputs. The end goal is to create a tool that will be useful for investors to visualize variance in their potential returns and understand how different factors may affect variance, so that they may be better-informed investors.
Outline

Specifically, I hope to accomplish the following:

1. Data Sourcing – Find reliable historical data for stocks and economic indicators

2. Determine important economic indicators and their effect on the market through machine-learning methods
   a. These indicators will be made to be user-configurable inputs to the model with suggested default values
   b. Examples of possible economic indicators include GDP, consumer price index, unemployment rate, inflation target, etc.

3. Build backend for computing correlations and running Monte Carlo simulation of stock performances based on these correlations and economic indicator inputs
   a. Exact methodology behind the Monte Carlo simulations will be determined based on future analysis. The current idea is to simulate overall market movements, then generate individual stock movements based on this and historical betas between the stock and the market plus some inherent volatility of the stock.
   b. Economic indicators will be factored in either based on a learned model of how these indicators directly affect stock movements, or by using these indicators to determine expected market return. This will also be determined later based on analysis of which is a more accurate model.

4. Build website frontend for constructing a portfolio, inputting predicted economic indicators, and outputting results
a. Portfolio construction will involve inputting tickers and weights. Currently, I plan on supporting mid to large cap US equities across all sectors, though I may change this if necessary.

b. Predicted economic indicators will be configurable, and a default will be set based on average historical performance.

c. Results will include graphs of expected portfolio performance over time, as well as distributions of portfolio returns based on Monte Carlo simulation. Time horizon will be user-configurable, with 20 years being a reasonable time horizon.

d. Will also try to recommend alternative portfolio allocations with lower risk.

Tools

I plan on using Python for the backend of my application. Specifically, I hope to leverage the pandas, numpy, and scikit-learn packages within Python to analyze the large datasets associated with the market. In particular, scikit-learn offers many tools for machine learning in Python, which will be useful for both discovering significant features and how these factors might affect both market returns and volatility.

For the frontend, I will use HTML/CSS and JavaScript to build my web application. To generate graphs, I will either use Chart.js, a JavaScript graphing library, or matplotlib, a Python graphing module.

Timeline

The following illustrates my projected timeline for the semester, where objectives for parts one through four are described in greater detail under the outline section.
1. Source and clean data (now through the end of September)
2. Determine important economic indicators (early to mid October)
3. Build website backend (mid October to mid November)
4. Build website frontend (mid to late November)
5. Write the report (December)

Deliverables

1. A final report describing my process, results of any analysis, and demonstrating my final product.
2. The final product – a web application that allows users to construct a portfolio from mid to large cap US equities, specify optional projected economic indicators and a timeframe, and have the tool output distribution of potential returns based on Monte Carlo simulation and machine-learned correlations. The tool will also attempt to find and recommend similar, but potentially less risky portfolios.

References