How to Survive the US Stock Market

Team: London Fog
Team Members: Angela Zhao, Zheng Ren, Yuzi Li, Yicheng Shen.
Final Product Delivery: An Interactive Shiny App

Overview of the Project

Investing in stocks can be quite overwhelming for beginner investors due to jargon-heavy tools and the lack of predictive statistical models. Existing platforms, like the Bloomberg Terminal, provide somewhat robust but expensive solutions, while most of the free options lack flexibility and statistical analysis. Another caveat of these platforms is the use of jargon, which may deter the users from understanding the data and making useful investment profiles. Our project aims to address this gap by offering an accessible, interactive, and beginner-friendly tool to help users optimize their investment strategies.

Objectives

The main goal of the project was to build a Shiny app that helps recreational investors make informed choices to improve their portfolio, particularly with respect to ETFs. We provide tools that can aid retail investors in understanding stock market performances, comparing stocks, and predicting portfolio outcomes. We hope to develop a user-friendly interactive Shiny app with real-time data and interpretive summary statistics, integrate Monte Carlo simulations for future portfolio predictions, and implement k-means clustering to group S&P500 stocks by their return and volatility.

Existing Work

The gold standard tool for investors is the Bloomberg Terminal, which is a powerful financial software platform that provides real-time data, analytics, news, and research documents. However, the subscription to Bloomberg Terminal is $30-50k/year. Free options are also available, such as Yahoo Finance and Stock Tracker, but they either lack statistical modeling or prediction capabilities. Finally, all of these are jargon-heavy and provide too much information for most recreational investors to sift through.

Data and Tools

Data for this project comes from the Yahoo Finance API via the `tidyquant` package in R. In particular, we are interested in pulling the daily stock price, trading volume, dividends, split ratio, and economic data. Data is updated every day after the market closes. In addition, we obtained lists of companies that make up popular ETFs (e.g. S&P500 and NASDAQ-100) by scraping from Wikitables using `rvest()`.

We used both functional programming and machine learning paradigms. Examples of functional programming include using high-order functions such as `sapply()` and `purrr::map()`. We also use anonymous functions and function composition for the portfolio calculations. An example of applying the machine learning paradigm is our unsupervised K-means clustering algorithm used on S&P500 data to categorize stocks. We also incorporated Monte Carlo simulations for portfolio prediction.

Challenges & Accomplishments

We addressed certain challenges and made the following accomplishments:

1. **Data Processing**: Combining multiple sources of data in a clean way.
2. **Interactivity**: Implementing interactive features with large, up-to-date data efficiently without compromising app performance.
3. **Prediction and Simulation**: Monte Carlo simulations for future portfolio performance.
4. **Machine Learning Integration**: Using K-means clustering for stock groupings.

Shiny App Overview

The developed app provides:

1. **Data Summary**: Tabular view of popular ETFs and stock market data.
2. **Stock Trends**: Comparative stock performance visualization (Figure 1).
3. **Market Distribution**: Sector-wise market performance analysis (Figure 2).
4. **Portfolio Analysis**: Historical and predicted stock performance of chosen portfolio using fast Monte Carlo simulations (Figure 3).
5. **Stock Clustering**: Clusters stocks based on volatility and return (Figure 4).

Functionality

The app is user-friendly, interactive, and well-documented, providing tools to analyze stocks and portfolios effectively. It allows non-professional investors to navigate and make informed decisions about their investments.

Usability and Documentation

The app prioritizes ease of use, offering customization and interactive data visualizations tailored for beginners. Code and data are well-documented and available for reproducibility.

Originality and Complexity

The originality of this project comes from finding an intuitive way to summarize and present stock information in a way that is more user-friendly for recreational/beginner investors than existing platforms. The design of our Shiny App is aimed at helping users beat benchmark ETFs, such as the S&P 500. We do this by allowing the user to compare their custom portfolios

to the performance of ETFs. We also use a k-means algorithm to identify groups of stocks in the S&P by their returns and volatility so the user can easily choose stocks based on their risk tolerance and desired returns. In brief, we try to creatively combine real-world stock market data, interactive visualization, and predictive statistical methods to help beginner investors optimize their portfolios.

Lesson we learned

1. Since we are using constantly updated data, we must balance the amount of data we aim to retrieve and the time it takes to do so, ensuring the reliability of live data streams.
2. One of the most important lessons we learned was how to balance sophisticated statistical modeling and multi-sourced data with a user-friendly design. Creating a tool for users, especially beginners with limited knowledge of finance or statistics, required simplifying the presentation to ensure they could effectively utilize the tool.
3. We also learned how to coordinate as a team, which required us to delegate tasks effectively, align our goals, and integrate various aspects of the project seamlessly.
4. Throughout the project, our team utilized version control to streamline our collaboration.
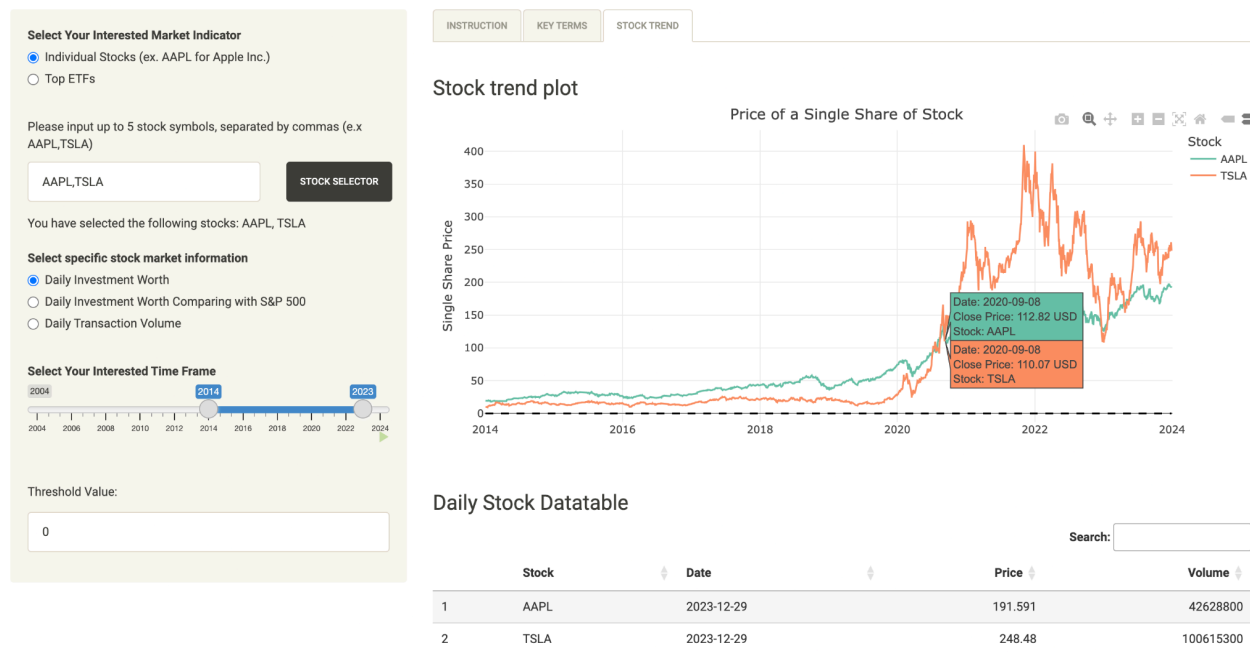
Limitations and Future Steps

1. **Better real-time data:** Our app pulls data from Yahoo Finance. The update is only daily and focuses on close prices instead of more accurate real-time data during the day.
1. **More functionalities**: While the app is beginner-friendly, more advanced investors may find its features insufficient compared to professional tools like Bloomberg Terminal. Also, unlike Bloomberg Terminal or Yahoo Finance, we do not consider any qualitative data, such as earnings reports and press releases from public companies.
2. **More Advanced Predictive Models:** The Monte Carlo simulations currently integrated into our app have several limitations. They possess limited predictive power, as they rely heavily on historical data and cannot account for unforeseen events, such as market crashes or black swan events. Additionally, their focus on past data increases the risk of overfitting, which may reduce their ability to generalize to future scenarios. As a result, they may oversimplify complex market dynamics and fail to capture unexpected changes. We plan to implement time series models, deep learning, or other advanced machine learning algorithms in the future to improve the accuracy of longer-term stock predictions and portfolio outcomes.
3. **Customization for Different Users:** We can also separate the tabs into tiers of functionality: basic for beginners and advanced analytics for experienced investors.

For more demonstrations of our product:

Link for our deployed Shiny app: https://zren0723.shinyapps.io/project4-london-fog/

Link for our codes: https://github.com/jhu-statprogramming-fall-2024/project4-london-fog

Below we also attach some plots from the demo to show app functionalities.

Figure 1. In our Stock Trend tab, users can choose any combination of up to five stocks and get visualized stock prices and transaction volumes for any timeframe in the past two decades. It can also be compared with S&P500 data as a way to see how it performs in the market.
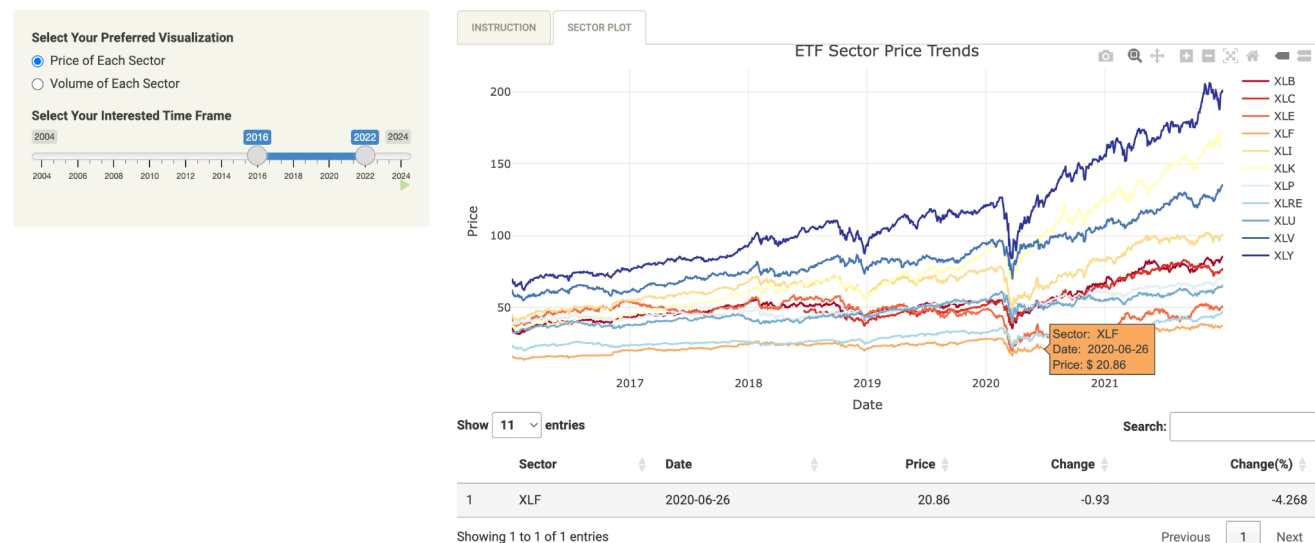
Figure 2. In our Market Distribution, the market performance is further broken down into 11 sectors. Major stock market events can be observed, such as the 2020 pandemic and the 2008 financial crisis.
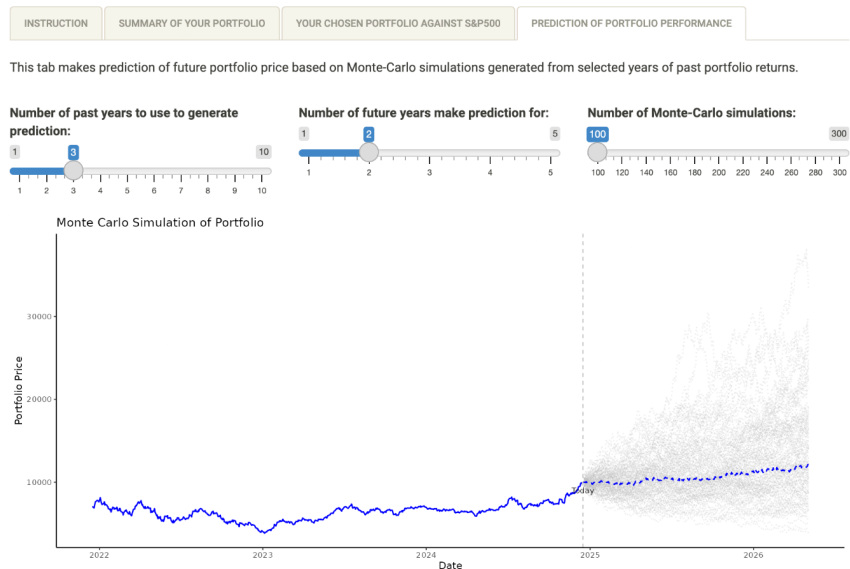
Figure 3. In the Understanding Your Portfolio tab, users can build and access their own portfolios, again with up to five stocks. The app runs Monte Carlo simulations to predict future portfolio performance based on existing data.
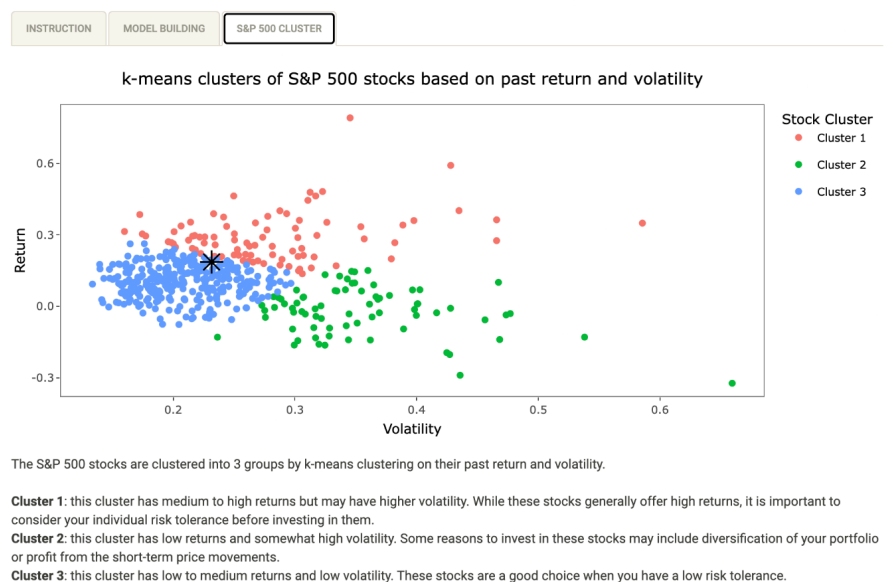

Figure 4. In the Stock Clustering tab, users can select a particular stock to see how it would be classified into one of the three stock clusters. This provides an intuitive way of categorizing stocks based on volatility and returns so that users can choose stocks based on their preferences.