Understanding the S&P 500

Miguel Bicas

2026-02-01

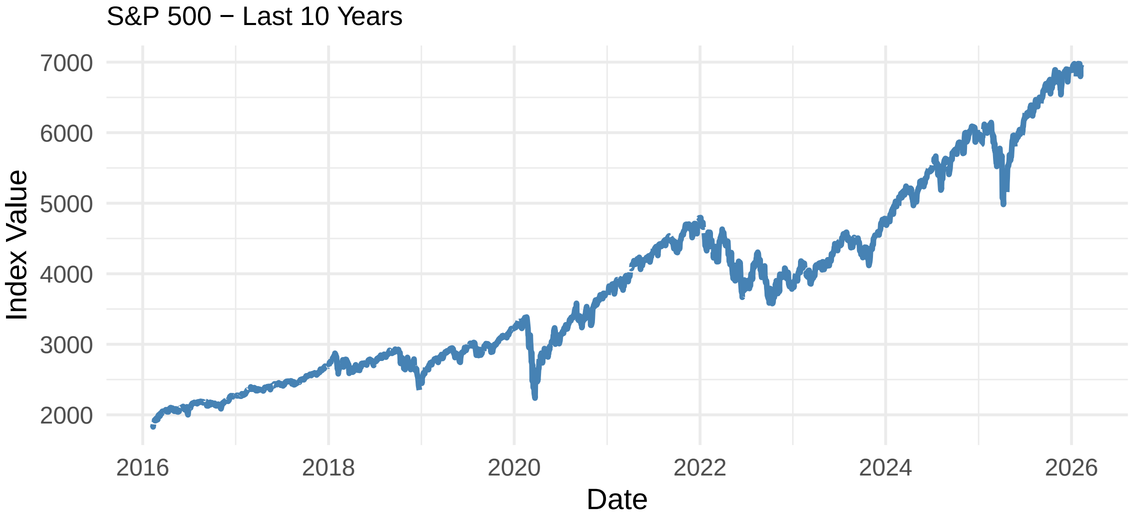
Table of contents

Project Outline

This project uses R to recreate and analyze the S&P500 using historical data from FRED and other public sources, with the goal of understanding what drives indicator movements. It examines the S&P 500 over the past decade by computing returns and growth rates and creating data visualizations. Additionally, it recreates the S&P500 using 11 sector focused ETFs and linear regression models.

1 The S&P500

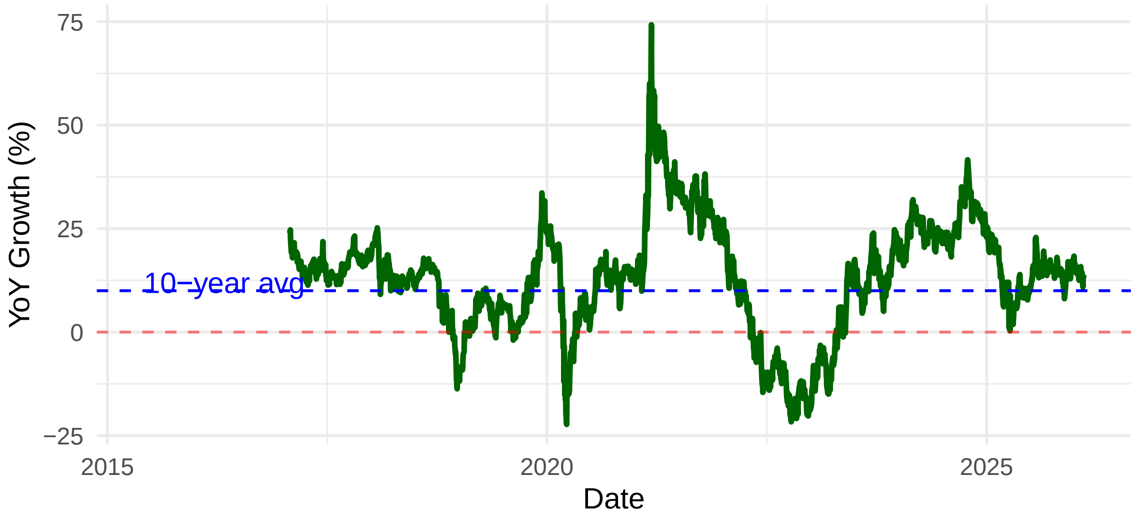
The *Standard and Poor's 500* (S&P500) is one of the most famous stock market indicators, it tracks the performance of the top 500 companies listed on stock exchanges within the United States. Today, the S&P500 is one of the most commonly traded indices and accounts for about 80% of the total market capitalization of U.S public companies [1]. This means that the 500 companies within the S&P500 account for 80% of the total value of all publicly traded U.S stocks. A list of all the current S&P500 companies can be seen here.



Source: FRED

Figure 1: S&P 500 Index performance from 2016-2026

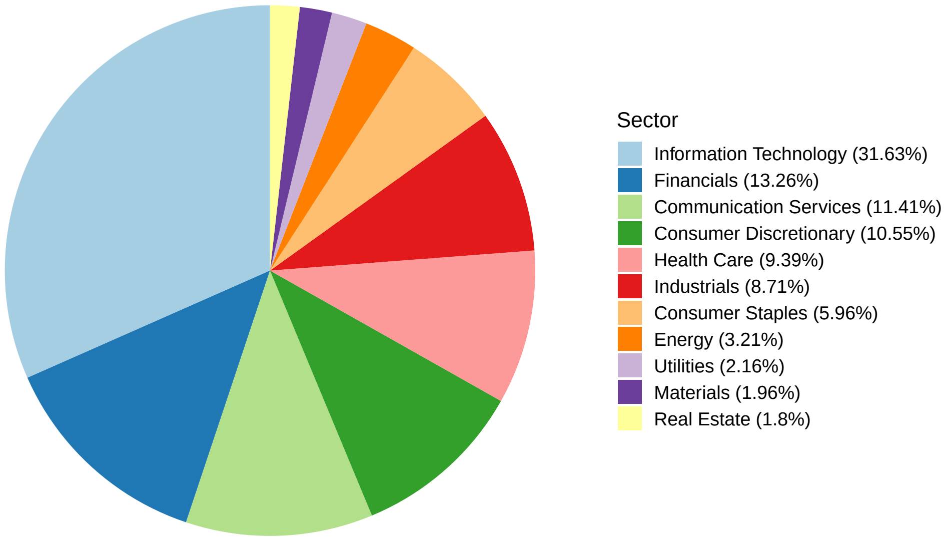
The value of the S&P500 is measured with an *index level*. The index level is a standardized score that moves up / down based on the combined stock prices of its companies. Figure 1 shows the S&P500's performance from 2016-2026; from 2016 to early 2026, the S&P 500 delivered exceptional performance, with a an investment of $100 the beginning of 2016 growing to about $393 by the end of 2025, a cumulative return of about 293%, or around 15% annualized with dividends reinvested [2]. The index reached the 7,000-point milestone for the first time on January 28, 2026, driven by strong gains in technology and AI-related stocks, though the period also included notable setbacks such as the the 2020 pandemic crash, and the 2022 bear market [3].



Source: FRED

Figure 2: S&P 500 Year-over-Year Growth 2016-2026

Figure 2 shows the S&P500's year-over-year growth rate from 2016 to 2026. Over this period, the index averaged an annualized return of approximately 15% (with dividends reinvested), well above the long-term historical average of around 10%. The chart has significant volatility, including a sharp dip below -25% during the COVID-19 crash in 2020, followed by a rapid recovery that pushed YoY growth above 70%. Despite these swings, the index spent the majority of the decade in positive territory, reflecting the overall strength of U.S. large-cap equities during this period. [4]



Source: S&P Dow Jones Indices, Feb 2026

Figure 3: S&P 500 Sectors

The S&P 500 is divided into 11 sectors classified under the Global Industry Classification Standard (GICS), with Information Technology having 31.63% of the index, followed by Financials (13.26%), Communication Services (11.41%), and Consumer Discretionary (10.55%). The remaining sectors Health Care, Industrials, Consumer Staples, Energy, Utilities, Materials, and Real Estate collectively make up the remaining third of the index, with Real Estate being the smallest at just 1.8%. The heavy concentration in technology reflects the large influence of companies like Nvidia, Apple, and Microsoft, whose combined market capitalization account for a significant share of the index's total value.

1.1 How is the S&P500 constructed?

The S&P 500 is a float-adjusted, market capitalization-weighted index, put simply, this means companies with larger market caps have a greater influence on the index's value. Each company's market capitalization is calculated by multiplying its share price by the number of publicly available (free-float) shares. The sum of all 500 companies' float-adjusted market caps is then divided by a proprietary divisor, maintained by S&P Dow Jones Indices, which is adjusted for corporate actions like stock splits, spinoffs, and share issuances to ensure continuity. The divisor is also adjusted when companies are removed or added from the index. The index level is calculated using the following formula:

$$\text{Index Level} = \frac{\sum_{i=1}^{500} P_i \times Q_i}{divisor}$$

Where P_i is the share price and Q_i is the is the float-adjusted number of shares outstanding.

1.2 Rebuilding the S&P500

A first instinct of reconstructing the S&P500 may be to follow the formula above, this would imply to sum the following for each stock:

1. Find the current price of the stock
2. Find the float-adjusted number of shares

Then we would need to estimate what the divisor would be and divide our sum by that divisor. This entire process would need to be done for each trading day that we are interested in.

Although this is not impossible, it is not the most efficient way to approximate the S&P500 and there are ways to get around my lack access to data (mostly the number of shares available on each day). Since we know the prominent sectors in the S&P500 (Seen in Figure 3) we can use Exchange-Traded Funds (ETFs) for each sector within the S&P500 and different statistical models to approximate the S&P500 index.

The ETFs we will use are: XLK (Technology), XLF (Financials), XLV (Healthcare), XLY (Consumer Discretionary), XLC (Communication Services), XLI (Industrials), XLE (Energy), XLP (Consumer Staples), XLRE (Real Estate), XLB (Materials), XLU (Utilities)

The majority of these ETFs are managed by State Street Investment Management and more information can be found here.

1.2.1 Loading Data Into R

Since data for each of these ETFs is available in the FRED data base, we can load in our data as follows [5]:

```r
sector_etfs <- c(
    "XLK",   # Technology
    "XLF",   # Financials
    "XLV",   # Health Care
    "XLY",   # Consumer Discretionary
    "XLC",   # Communication Services
    "XLI",   # Industrials
    "XLE",   # Energy
    "XLP",   # Consumer Staples
    "XLRE",  # Real Estate
    "XLB",   # Materials
    "XLU"    # Utilities
)

all_tickers <- c("SPY", sector_etfs)
```

Next we set our time frame and get store the daily prices for each of these ETFs:

```r
start_date <- Sys.Date() - years(10)
end_date <- Sys.Date()

prices <- tq_get(
    all_tickers,
    from = start_date,
    to   = end_date,
    get  = "stock.prices"
    ) %>%
    select(symbol, date, adjusted)

prices_wide <- prices %>%
    pivot_wider(names_from = symbol, values_from = adjusted) %>%
    drop_na()

# head(prices_wide)
```

1.2.2 Calculating Daily Returns & Prepare Data for Training

We then compute daily returns by using the formula:

$$\text{Daily Return} = \frac{\text{Today's Price}}{\text{Yesterday's Price}} - 1$$

```r
returns_wide <- prices_wide %>%
    arrange(date) %>%
    mutate(across(-date, ~ . / lag(.) - 1)) %>%
    drop_na()

# Separate into target (SPY) and predictors (ETFs)
spy_returns <- returns_wide$SPY
sector_returns <- returns_wide %>% select(all_of(sector_etfs)) %>% as.matrix()
dates <- returns_wide$date

# head(returns_wide %>% select(date, SPY, XLK, XLF, XLE), 5)
```

1.2.3 Making Our Model

Our goal is to reconstruct the S&P 500 using only the daily returns of the 11 sector ETFs. At a high level, if the S&P 500 is a weighted mix of sectors, then on any day SPY's return should be well-approximated by a weighted sum of sectors returns.

(*We use SPY because it's a highly liquid ETF that closely tracks the S&P 500 and provides reliable daily price data to model and replicate.*)

Let r_t^{SPY} be the SPY daily return on day t, and let $r_{t,1}, \dots, r_{t,11}$ be the daily returns of the sector ETFs on the same day.

We model:

$$r_t^{\text{SPY}} \approx \sum_{j=1}^{11} w_j \, r_{t,j},$$

where the coefficients w_j represent how much each sector ETF contributes to SPY's movement. In matrix form, this is:

$$\mathbf{y} \approx \mathbf{X}\mathbf{w},$$

where $\mathbf{y}$ is the vector of SPY returns over time, $\mathbf{X}$ is the matrix of sector returns (each column is one sector ETF), and $\mathbf{w}$ is the vector of weights.

We start with ordinary least squares because it is the simplest and most standard way to estimate weights. OLS chooses $\mathbf{w}$ to minimize the total squared tracking error between the true SPY returns and the model's predicted returns:

$$\hat{\mathbf{w}}_{\text{OLS}} = \arg\min_{\mathbf{w}} \sum_t \left(r_t^{\text{SPY}} - \sum_{j=1}^{11} w_j r_{t,j} \right)^2 .$$

I selected this for three reasons:

1. We want to minimize the day-to-day discrepancy (tracking error) between SPY and our sector-based reconstruction
2. It is easy to implement and interpret; the fitted coefficients tell us which sectors explain more of SPY's variation.
3. OLS is a strong baseline to let us quantify how well sector returns alone can explain SPY returns (using R^2).

We make our model as follows:

```r
ols_model   <- lm(spy_returns ~ sector_returns - 1)
ols_weights <- coef(ols_model)
names(ols_weights) <- sector_etfs

cat(
  "=== OLS Regression ===\n",
  "R-squared: ", round(summary(ols_model)$r.squared, 6), "\n",
  "Sum of weights: ", round(sum(ols_weights), 4), "\n",
  "Weights:\n",
  sep = "", round(ols_weights, 4)
)
```

```
=== OLS Regression ===
R-squared: 0.992682
Sum of weights: 0.9693
Weights:
0.31270.10070.1110.11910.08870.08260.03590.05430.01060.01620.0374
```

Since OLS models can have negative coefficients we set up an optimization problem to find sector ETF weights that make the sector portfolio's returns track SPY as closely as possible, while forcing the weights to be non-negative and sum to 1.

```r
n_sectors <- ncol(sector_returns)

Dmat <- t(sector_returns) %*% sector_returns

Dmat <- Dmat + 1e-8 * diag(n_sectors)

dvec <- t(sector_returns) %*% spy_returns

Amat <- cbind(
  rep(1, n_sectors),
  diag(n_sectors)
)
bvec <- c(1, rep(0, n_sectors))
meq  <- 1

qp_result <- solve.QP(Dmat, dvec, Amat, bvec, meq)

constrained_weights <- qp_result$solution
names(constrained_weights) <- sector_etfs

cat(
  "\n=== Constrained Optimization ===\n",
  "Sum of weights: ", round(sum(constrained_weights), 4), "\n",
  "Weights:\n",
  sep = "", round(constrained_weights, 4)
)
```

```
=== Constrained Optimization ===
Sum of weights: 1
Weights:
0.30750.09530.12120.12130.09510.08670.03720.07470.00630.01460.04
```

```r
comparison <- data.frame(
  Sector      = sector_etfs,
  OLS         = round(ols_weights, 4),
  Constrained = round(constrained_weights, 4)
)
```

The graph below shows the difference in model coefficients before and after our optimization:

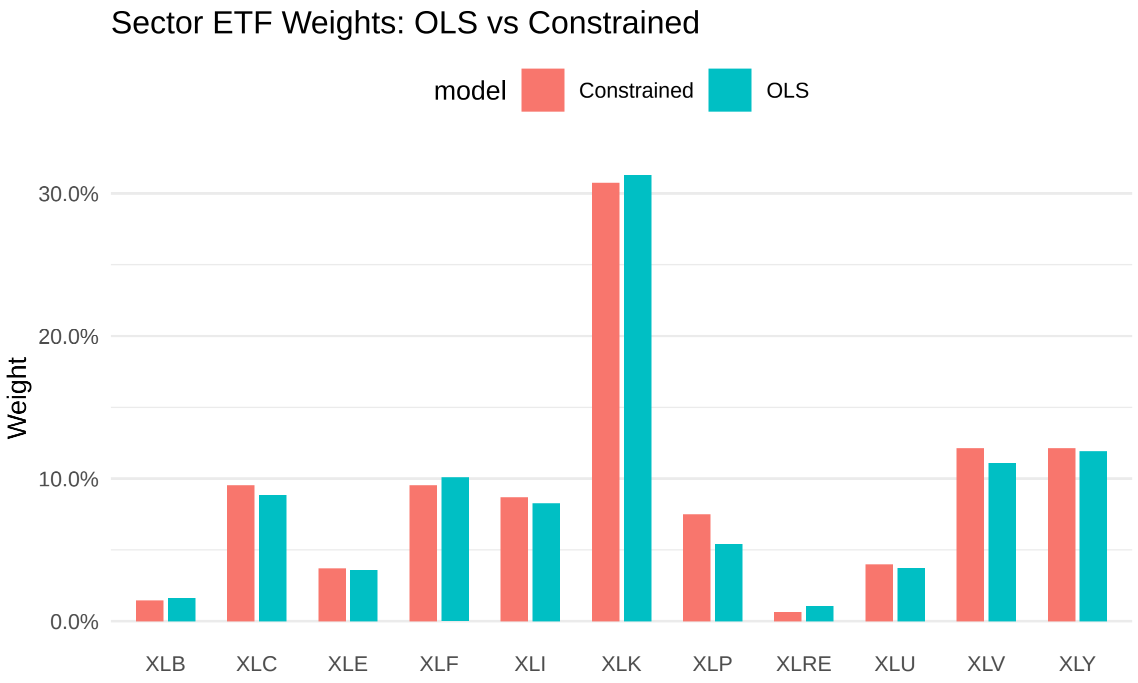


Figure 4: OLS vs. Constrained Model Coefficients

Next we normalize each ETF so day 1 = 1, then weight and sum, we also build a replica index and combine both of these into a data frame.

```r
sector_prices_matrix <- prices_wide %>% select(all_of(sector_etfs)) %>% as.matrix()
sector_normalized    <- sweep(sector_prices_matrix, 2, sector_prices_matrix[1, ], "/")

replica_index  <- (sector_normalized %*% constrained_weights) * 100
spy_normalized <- (prices_wide$SPY / prices_wide$SPY[1]) * 100

index_df <- data.frame(
  date     = prices_wide$date,
  SPY      = spy_normalized,
  Replica = as.numeric(replica_index)
) %>%
  pivot_longer(-date, names_to = "series", values_to = "value")
```

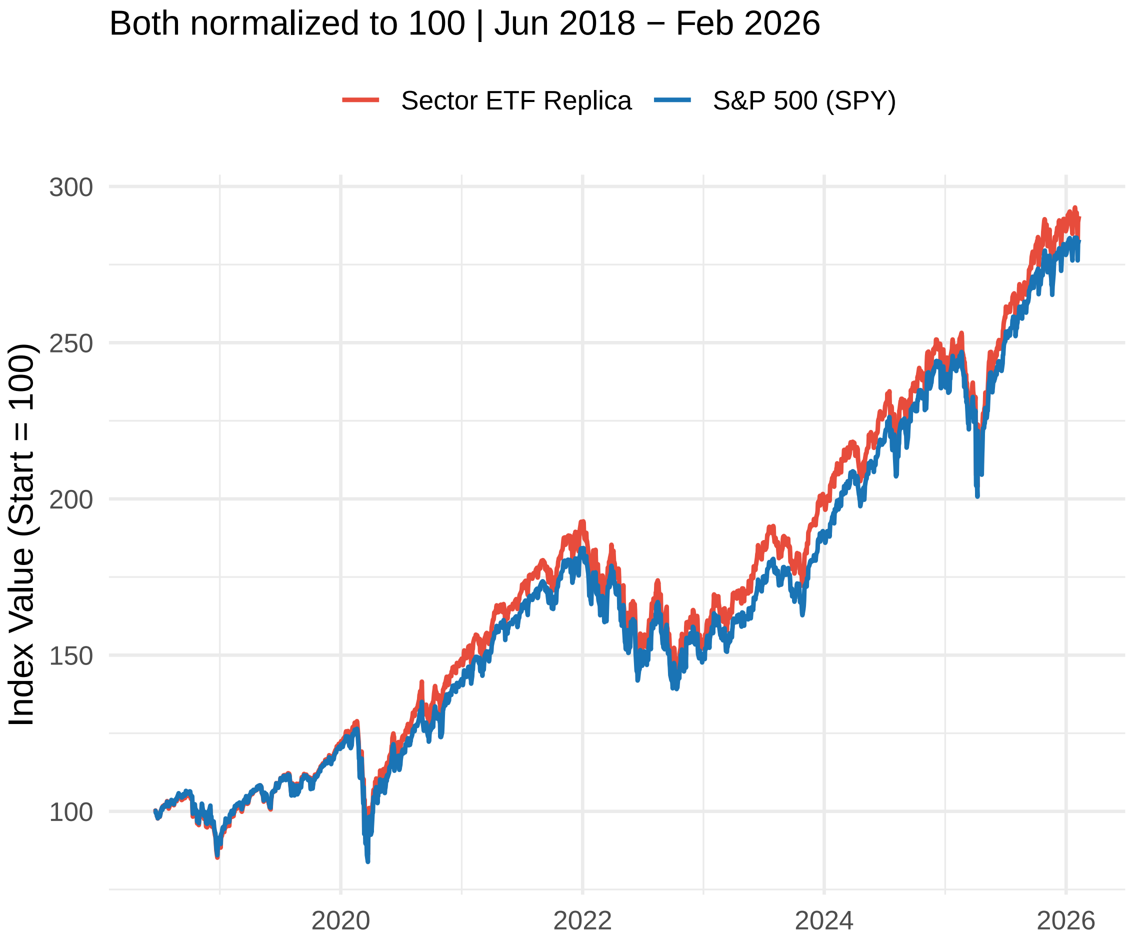


Figure 5: S&P 500 vs. Sector ETF Reconstruction

Figure 5 compares out S&P500 reconstruction to the actual S&P500, as you can see in the graph and from our R^2 of 0.992682 we were able to track the true value of the S&P500 very closely.

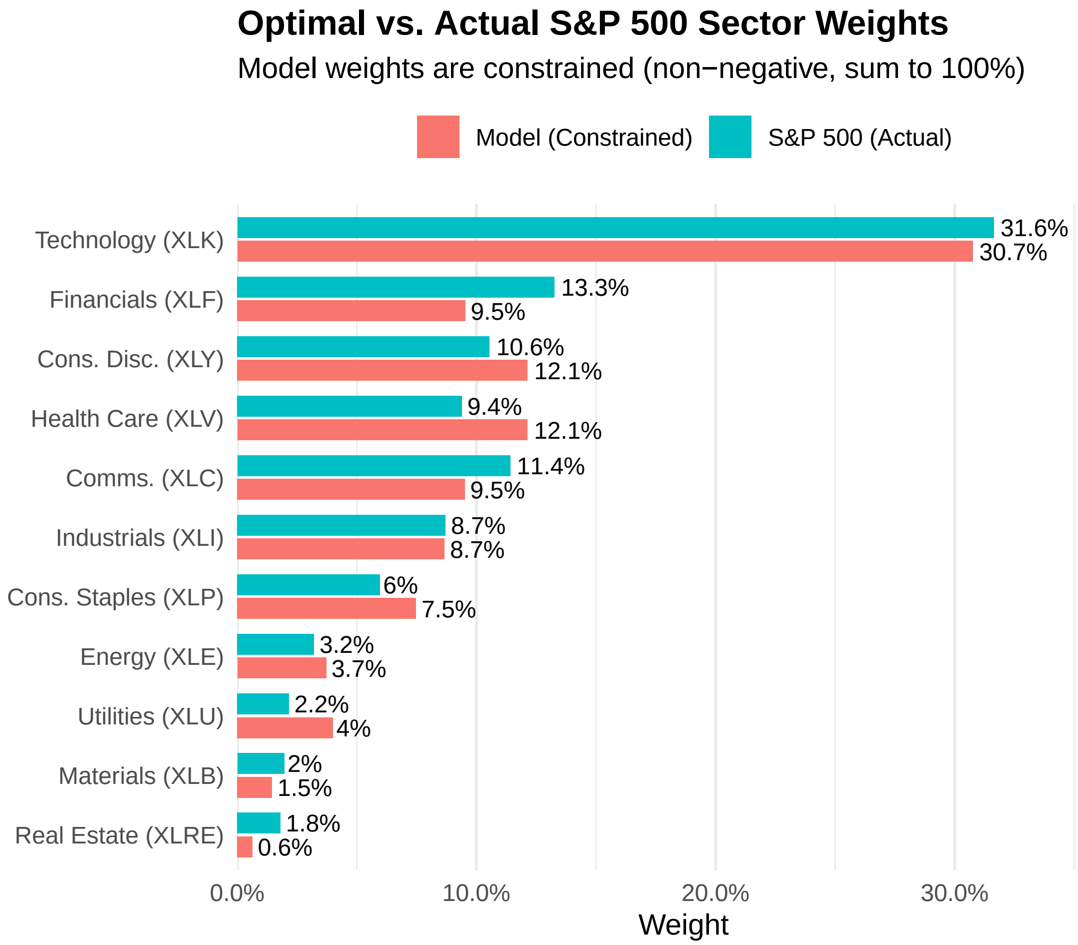


Figure 6: Optimal Sector Weights vs. Real S&P 500 Sector Weights

Figure 6 compares sector weights for our reconstructed model to the actual weights of the S&P500. Overall, the model matches the index reasonably well, especially in Information Technology, which remains the dominant sector in both portfolios. The most notable differences are that the model underweights Financials and Communication Services while overweighting Health Care and Utilities, suggesting these sectors better explain SPY's return movements over the sample period than their market-cap weights alone would imply. This is likely due to how those individual ETFs are constructed and their recent performance.

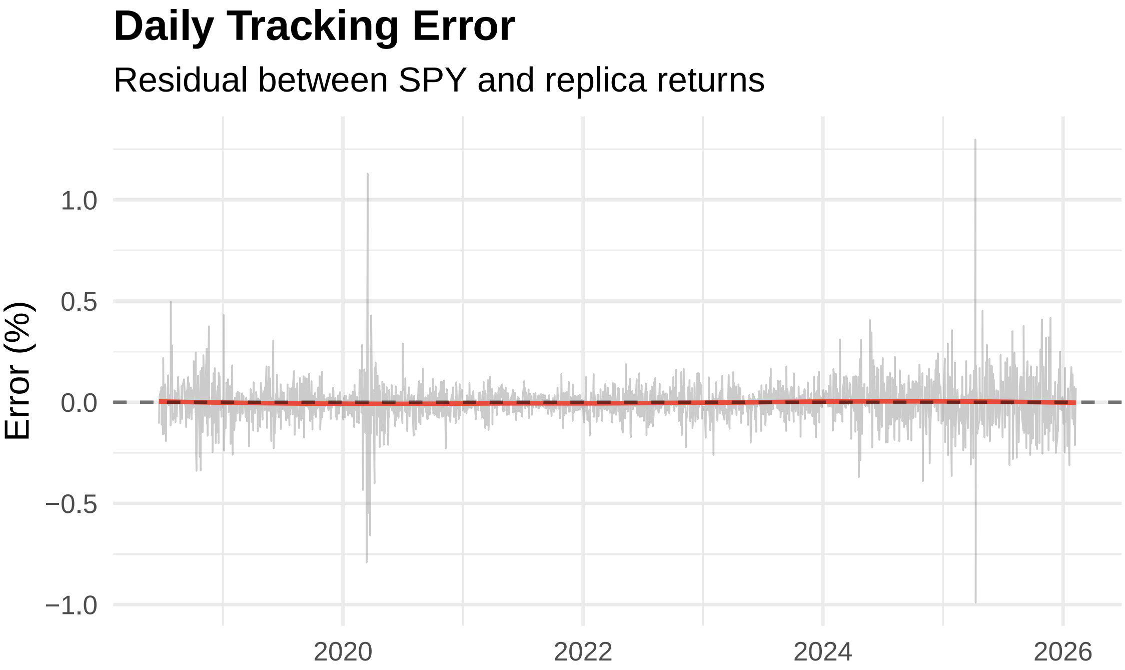


Figure 7: Daily Tracking Error Between Our Model and S&P500

Figure 7 shows the daily tracking error, (the difference between SPY's daily return and the return predicted by our sector-ETF replica). Most residuals cluster tightly around zero and the smoothed line stays near zero, indicating that the model is generally unbiased and tracks SPY fairly well on average. The occasional spikes, especially during more volatile periods highlight days when SPY moved for reasons not fully captured by sector ETFs alone (for example, Covid-19, index re-balancing effects, Supply Chain Issues, ETF-specific noise, or short-term factor shocks).

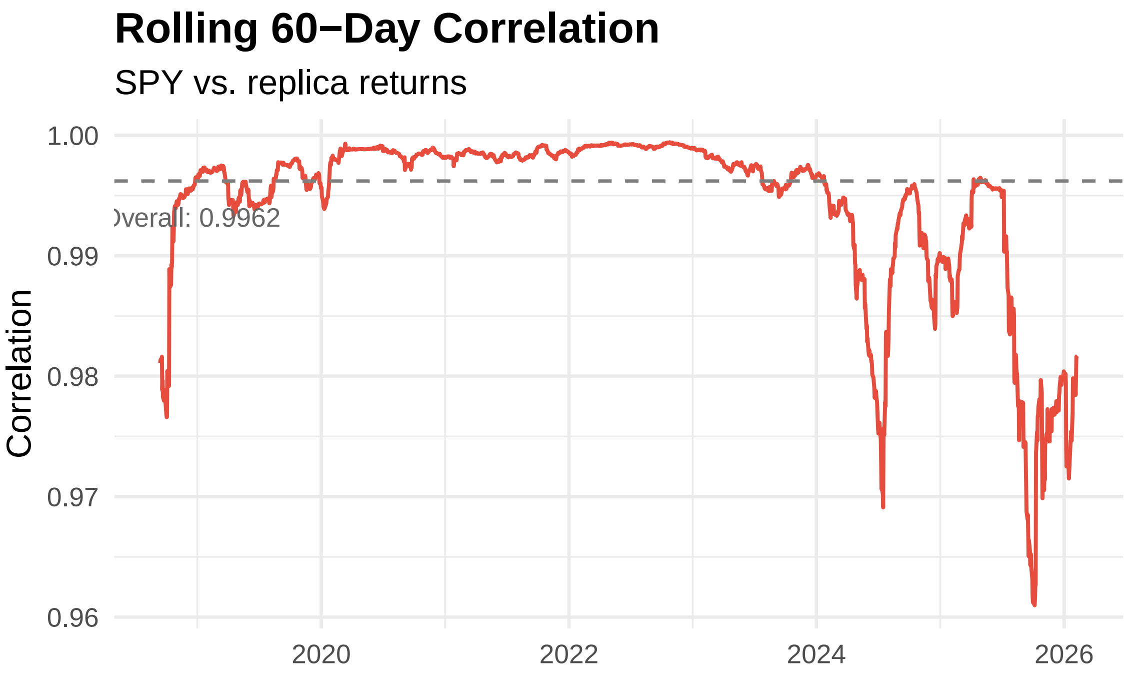


Figure 8: Rolling 60-Day Correlation

Figure 8 shows the rolling 60-day correlation between SPY returns and our replica returns, with the dashed line marking the overall correlation (about 0.996). For most of the sample the correlation stays extremely close to 1, meaning the replica captures SPY's day-to-day direction very well. The sharp dips late in the sample indicate short windows where the relationship temporarily weakened, likely during periods of unusual market dynamics when sector ETFs did not move in lockstep with the broad index.

1.3 Summary

Putting this all together, we created a model that uses individual sector ETFs to replicate the movement of the S&P500. Using OLS Regression and some optimization we were able to track the S&P500 with about 99% accuracy and derive similar sector weights compared to the actual S&P500. Looking into the future, I would like to add some smaller ETFs to see if it is possible to predict the S&P500 with better accuracy. I would also like to explore if it is possible to predict how the S&P500 will move by creating many smaller models to see how each ETF will move individually.

[1] S&P Dow Jones Indices, “S&p 500: The gauge of the u.s. Large-cap market.” 2025. Available: https://www.spglobal.com/spdji/en/documents/additional-material/sp-500-brochure.pdf

[2] I. Webster, “S&p 500: $100 in 2016 → today.” 2026. Available: https://www.officialdata.org/us/stocks/s-p-500/2016

[3] L. Shalett, “When will the market hit bottom? Watch these 2 metrics.” Morgan Stanley Wealth Management, Nov. 2022. Available: https://www.morganstanley.com/ideas/bear-market-2022-two-metrics

[4] Wikipedia contributors, “S&P 500.” Wikipedia. Accessed: Feb. 10, 2026. [Online]. Available: https://en.wikipedia.org/wiki/S%26P_500

[5] Federal Reserve Bank of St. Louis, “Federal reserve economic data (FRED).” Website. Accessed: Feb. 10, 2026. [Online]. Available: https://fred.stlouisfed.org/