Performance Comparison of Maximum Sharpe MVP and Equally Weighted Portfolio for SOXX Top 10 Constituents

Changlin Yi

April 15, 2025

Abstract

This report examines the risk-adjusted performance of a Maximum Sharpe Ratio portfolio (MVP) versus an Equally Weighted Portfolio (EWP) constructed from the top 10 constituents of the SOXX ETF. A rolling window methodology is employed using a 3-year look-back period for in-sample optimization and a 1-month look-ahead period for out-of-sample evaluation over 5 years of daily data. Annualized return (in %), volatility (in %), and Sharpe ratios (unitless) are computed and statistically compared. In addition, the evolution of significant portfolio weights (those exceeding 5%) is examined. The composition of the top 10 holdings is shown in Fig. 1 (source: ETF.com).

1 Introduction

The SOXX exchange-traded fund (ETF) tracks the performance of the ICE Semiconductor Index, offering exposure to key players in the semiconductor industry. Figure 1 shows the composition of the top 10 holdings of SOXX together with their respective allocation weights. (Image source: ETF.com.) This composition is central to our analysis as we construct the Maximum Sharpe Ratio portfolio (MVP) and compare its performance to that of an Equally Weighted Portfolio (EWP).

SOXX Holdings [As of 04/10/2025]

TOP 10 WEIGHT	60.08%
ALL OTHER	39.95%
NUMBER OF HOLDINGS	31

Symbol	Holding	Allocation %
AVGO	Broadcom Inc.	8.81%
NVDA	NVIDIA Corporation	8.57%
TXN	Texas Instruments Incorporated	7.27%
AMD	Advanced Micro Devices, Inc.	7.22%
QCOM	QUALCOMM Incorporated	6.84%
KLAC	KLA Corporation	4.56%
AMAT	Applied Materials, Inc.	4.43%
LRCX	Lam Research Corporation	4.25%
MPWR	Monolithic Power Systems, Inc.	4.1%
ASML	ASML Holding NV Sponsored ADR	4.03%

Figure 1: Top 10 holdings of SOXX with respective allocation weights.
Source: ETF.com

Portfolio optimization seeks to improve risk-adjusted returns by dynamically adjusting asset weights. In this study, the MVP is obtained by maximizing the Sharpe ratio using a 3-year in-sample period (approximately 756 trading days), while the EWP assigns a constant 10% weight to each asset. The analysis considers 5 years of daily price data.

2 Methodology

2.1 Data and Portfolio Construction

Historical daily prices for the top 10 SOXX ETF constituents were downloaded from Yahoo Finance. From these, daily returns were computed. The portfolio specification was defined with full-investment and long-only constraints. Two strategies were constructed:

1. **MVP:** Optimized by maximizing the Sharpe ratio using a 3-year (approximately 756 trading days) look-back period.

2. **EWP:** An equally weighted portfolio with each asset assigned a fixed weight of 10%.

2.2 Rolling Window Analysis

A rolling window approach was applied where:

- The in-sample (look-back) period spans 3 years.

- The out-of-sample (look-ahead) period covers 1 month (approximately 21 trading days).
- The window is moved forward by 1 month at a time until the end of the dataset.

For each window, annualized returns (in %), annualized volatilities (in %), and Sharpe ratios (unitless) were computed for both strategies.

2.3 Statistical Testing and Weight Evolution

The distributions of the annualized Sharpe ratios for the MVP and the EWP were statistically compared using a paired t-test and the Wilcoxon signed-rank test. Additionally, the evolution of the MVP weights is tracked by counting, in each window, the number of assets for which the weight exceeds 5%.

3 Results

3.1 Sharpe Ratio Distributions

The distribution of the annualized Sharpe ratios for both strategies is presented in Figs. 2 and 3. In Fig. 2 the x-axis denotes the annualized Sharpe ratio (unitless) and the y-axis indicates frequency. The same applies to Fig. 3.

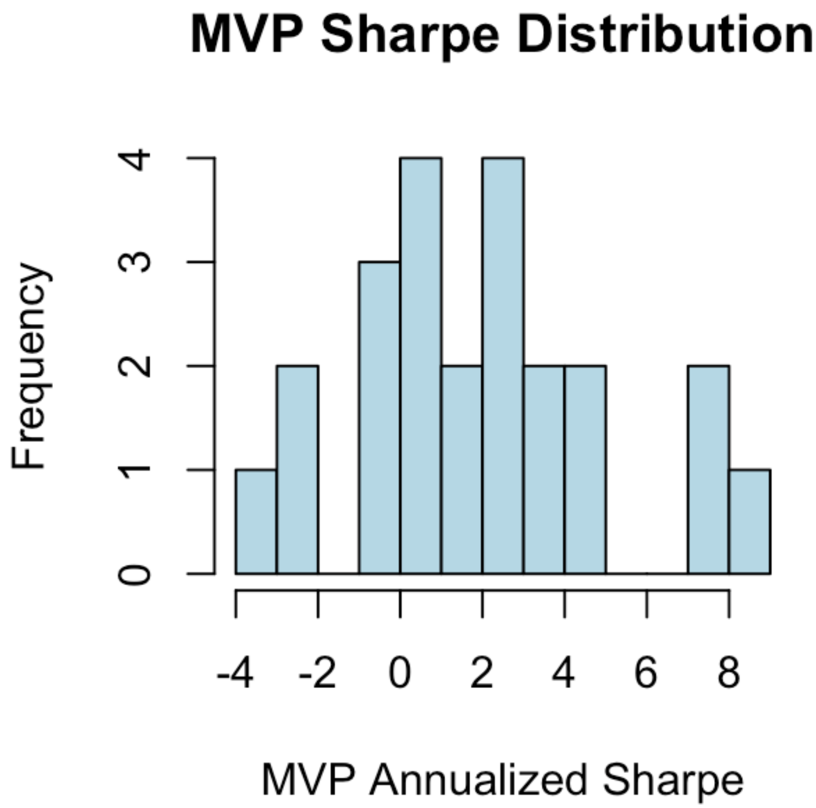

Figure 2: Histogram of the Annualized Sharpe Ratios for the Maximum Sharpe MVP. The x-axis shows Sharpe ratios (unitless), and the y-axis shows frequency.

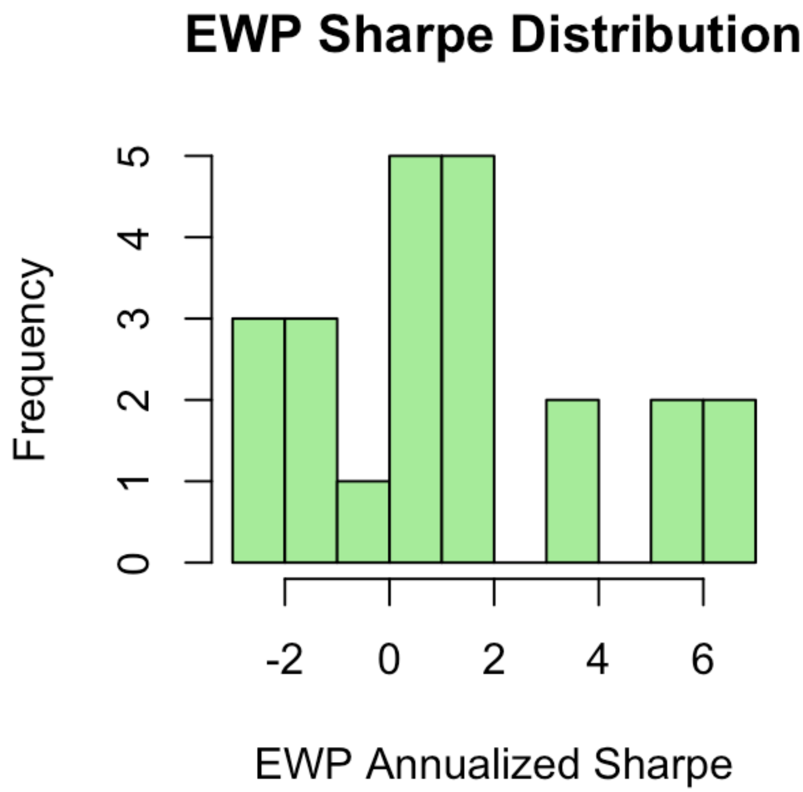

Figure 3: Histogram of the Annualized Sharpe Ratios for the Equally Weighted Portfolio. The x-axis shows Sharpe ratios (unitless), and the y-axis shows frequency.

The paired t-test resulted in a p-value of approximately 0.047, indicating that the difference in the mean Sharpe ratios between the two portfolios is statistically significant.

3.2 Evolution of Significant Portfolio Weights

Figure 4 displays the evolution of significant MVP weights, defined as the count of assets with an optimized weight exceeding 5%, across the rolling windows. The x-axis shows the date corresponding to the start of the out-of-sample period, and the y-axis shows the number of assets (unit: number of assets). Notably, most periods exhibit two assets above the threshold, with a brief spike to three assets around mid-2023.

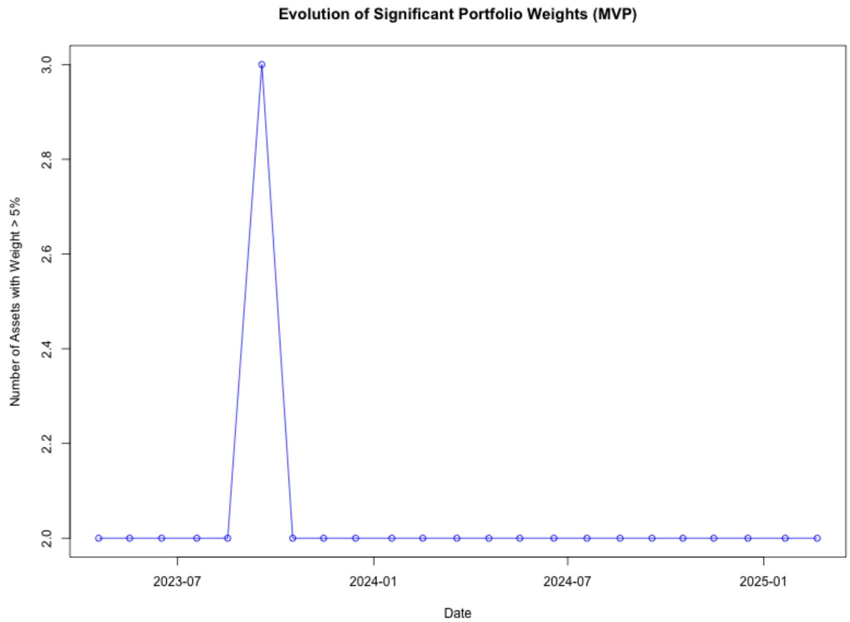

Figure 4: Evolution of the number of assets in the MVP with weights exceeding 5%. The x-axis represents the date (look-ahead period start) and the y-axis represents the number of assets.

4 Discussion

The results indicate that the Maximum Sharpe MVP strategy provides superior risk-adjusted performance compared to the Equally Weighted Portfolio, as corroborated by the statistically significant difference in their Sharpe ratios (p-value ≈ 0.047). However, the MVP sometimes exhibits higher volatility, reflecting potential trade-offs inherent in optimization.

The dynamic evolution of portfolio weights (Fig. 4) reveals that while most rolling windows include two key assets with weights above 5%, market conditions occasionally lead to a third asset becoming significant. This flexibility underscores the responsiveness of the optimization process to changing market dynamics.

5 Conclusion

This study compared the performance of a Maximum Sharpe MVP with that of an Equally Weighted Portfolio for the top 10 constituents of the SOXX ETF using a rolling window approach over 5 years of daily data. The MVP generally achieved higher annualized Sharpe ratios, a finding supported by statistical tests. Moreover, the analysis of portfolio weights shows that although the MVP typically focuses on two assets, there are periods when a third asset is incorporated due to shifting market conditions. These results suggest that dynamic portfolio optimization can lead to enhanced risk-adjusted returns relative to static allocation strategies.

A R Script

Listing 1: R Script for the Reprot

```r
##########################################################################################
# Changlin Yi
# cy2578@nyu.edu
# FRE 7831 HW-3
##########################################################################################

rm(list = ls())

# Core packages
library(quantmod)
library(PortfolioAnalytics)
library(ROI)
library(ROI.plugin.quadprog)
library(ROI.plugin.glpk)
library(xts)
library(lubridate)


# Top 10 SOXX ETF constituents
tkr_list <- c("AVGO", "NVDA", "TXN", "AMD", "QCOM",
              "KLAC", "AMAT", "LRCX", "MPWR", "ASML")

# Use 5 years of data: set endDate to today's date and startDate 5 years ago.
endDate   <- Sys.Date()                         # e.g., 2025-04-15 (today)
startDate <- endDate - years(5)                  # 5 years back

# Download historical daily adjusted close prices from Yahoo Finance
getSymbols(tkr_list, src = "yahoo", from = startDate, to = endDate)

# Merge the adjusted close prices into one xts object
prices <- do.call(merge, lapply(tkr_list, function(sym) Ad(get(sym))))
colnames(prices) <- tkr_list

# Compute daily returns (using the discrete return method)
returns <- ROC(prices, type = "discrete")[-1,]

# Use the column names from returns as asset names
fund.names <- colnames(returns)

# Create a portfolio specification with full investment and long-only constra
pspec <- portfolio.spec(assets = fund.names)
```

```r
pspec <- add.constraint(portfolio = pspec, type = "full_investment")
pspec <- add.constraint(portfolio = pspec, type = "long_only")
pspec <- add.objective(portfolio = pspec, type = "return", name = "mean")
pspec <- add.objective(portfolio = pspec, type = "risk", name = "StdDev")

# Look-back: 3 years    252 trading days/year * 3 = 756 days.
# Look-ahead: 1 month    21 trading days.
lookback_days  <- 252 * 3  # 756 days for optimization
lookahead_days <- 21       # 21 days to test out-of-sample performance

# Determine available observations
n_obs <- nrow(returns)
# Create a sequence of start indices for rolling windows.
# Here we roll forward by 21 days each time.
roll_starts <- seq(1, n_obs - (lookback_days + lookahead_days) + 1, by = look

# Pre-allocate vectors to store performance metrics (annualized)
n_rolls <- length(roll_starts)
mvp_ann_return <- rep(NA, n_rolls)
mvp_ann_vol    <- rep(NA, n_rolls)
mvp_ann_sharpe <- rep(NA, n_rolls)
ewp_ann_return <- rep(NA, n_rolls)
ewp_ann_vol    <- rep(NA, n_rolls)
ewp_ann_sharpe <- rep(NA, n_rolls)
# Pre-allocate a vector to store the count of significant weights (greater th
sig_counts <- rep(NA, n_rolls)

for(i in seq_along(roll_starts)) {

  # Define indices for look-back (in-sample) and look-ahead (out-of-sample) p
  lb_start <- roll_starts[i]
  lb_end   <- lb_start + lookback_days - 1
  la_start <- lb_end + 1
  la_end   <- lb_end + lookahead_days

  # Ensure we do not exceed the dataset
  if(la_end > n_obs) break

  # Subset the returns data for the look-back and look-ahead periods
  lb_returns <- returns[lb_start:lb_end, ]
  la_returns <- returns[la_start:la_end, ]

  mvp_opt <- optimize.portfolio(R = lb_returns,
                                portfolio = pspec,
                                optimize_method = "ROI",
```

```r
                                    maxSR = TRUE,
                                    trace = FALSE)
  mvp_weights <- mvp_opt$weights  # these are the optimized weights

  # --- Count the number of significant weights (> 5%) ---
  sig_counts[i] <- sum(as.numeric(mvp_weights) > 0.05)

  # Compute portfolio daily returns for the MVP portfolio
  mvp_daily <- as.numeric(la_returns %*% mvp_weights)

  # Compute performance metrics for MVP (annualized return, volatility, Sharp
  mvp_mean     <- mean(mvp_daily)
  mvp_sd       <- sd(mvp_daily)
  mvp_ann_return[i] <- mvp_mean * 252
  mvp_ann_vol[i]    <- mvp_sd * sqrt(252)
  mvp_ann_sharpe[i] <- ifelse(mvp_ann_vol[i] != 0, mvp_ann_return[i] / mvp_an

  # Compute the same for an Equally Weighted Portfolio (EWP)
  num_assets <- length(fund.names)
  ewp_weights <- rep(1/num_assets, num_assets)
  ewp_daily    <- as.numeric(la_returns %*% ewp_weights)
  ewp_mean     <- mean(ewp_daily)
  ewp_sd       <- sd(ewp_daily)
  ewp_ann_return[i] <- ewp_mean * 252
  ewp_ann_vol[i]    <- ewp_sd * sqrt(252)
  ewp_ann_sharpe[i] <- ifelse(ewp_ann_vol[i] != 0, ewp_ann_return[i] / ewp_an

  # Optionally, print progress:
  cat("Window", i, "of", n_rolls, ": MVP Sharpe =", round(mvp_ann_sharpe[i],
      "; EWP Sharpe =", round(ewp_ann_sharpe[i], 3), "\n")
}

# Create a results data frame using the date that marks the beginning of the
result_dates <- index(returns)[roll_starts + lookback_days]
results <- data.frame(Date        = result_dates,
                      MVP_Return = mvp_ann_return,
                      MVP_Vol    = mvp_ann_vol,
                      MVP_Sharpe = mvp_ann_sharpe,
                      EWP_Return = ewp_ann_return,
                      EWP_Vol    = ewp_ann_vol,
                      EWP_Sharpe = ewp_ann_sharpe)
print(results)

# Plot the time series of the number of assets with weight > 5%
plot(result_dates, sig_counts, type = "o", col = "blue",
```

```r
      xlab = "Date", ylab = "Number of Assets with Weight > 5%",
      main = "Evolution of Significant Portfolio Weights (MVP)")

# Compare Sharpe Ratio Distributions and Statistical Tests
par(mfrow = c(1, 2))  # Side-by-side plots
hist(mvp_ann_sharpe, main = "MVP Sharpe Distribution",
      xlab = "MVP Annualized Sharpe", col = "lightblue", breaks = 10)
hist(ewp_ann_sharpe, main = "EWP Sharpe Distribution",
      xlab = "EWP Annualized Sharpe", col = "lightgreen", breaks = 10)
par(mfrow = c(1, 1))  # Reset plotting layout

# Perform paired t-test and Wilcoxon signed-rank test comparing MVP vs. EWP S
t_test_result <- t.test(mvp_ann_sharpe, ewp_ann_sharpe, paired = TRUE)
wilcox_test_result <- wilcox.test(mvp_ann_sharpe, ewp_ann_sharpe, paired = TR

cat("\nPaired t-test on Sharpe Ratios:\n")
print(t_test_result)

cat("\nPaired Wilcoxon Test on Sharpe Ratios:\n")
print(wilcox_test_result)
```