COLUMBIA UNIVERSITY

Optimal Sector Trading Portfolio Construction using Quadratic Programming

Group: Kebing Li, Juyeon Kang

Class: MATH5010 Intro to Math of Finance 2019 Fall/ Pf. Mikail Smirnov

Date: Dec 9th, 2019 (Rehearsal: at Butler Library on Dec 8th, 2019)

Used Python & Excel

Introduction and Motivation

- Goal:

  i. To achieve low risk portfolios by allocating optimal sector weights

  ii. To observe if the optimal sector weights allow us to have better market returns than benchmark

- Objective:
  - To provide optimal industry weights
  - To compare expected return of the portfolio with the benchmark

Methodology: Quadratic Programming

- $\min \omega^T \sum \omega$ subject to

  (i) $\omega^T * 1 = 1$

  (ii) $\omega^T \mu^p = \mu^p$

  → solve the quadratic programming

- $\min \frac{1}{2} \omega^T \sum \omega - d^T \omega$ subject to

  (i) ${A_{eq}}^T * b_{eq}$

  where D=2$\sum$, d=0, $A_{eq} = \left(\binom{1}{1} \mu_p\right)$, $b_{eq} = \binom{1}{\mu_p}$

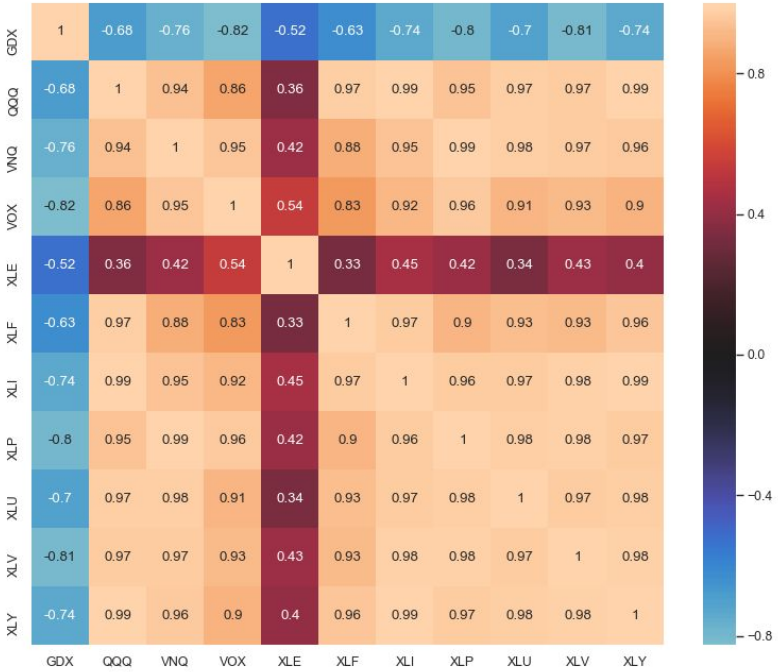
Data

- ETF data:
  - Yahoo Finance
  - Year: 2009-2019
  - Industry-ETFs

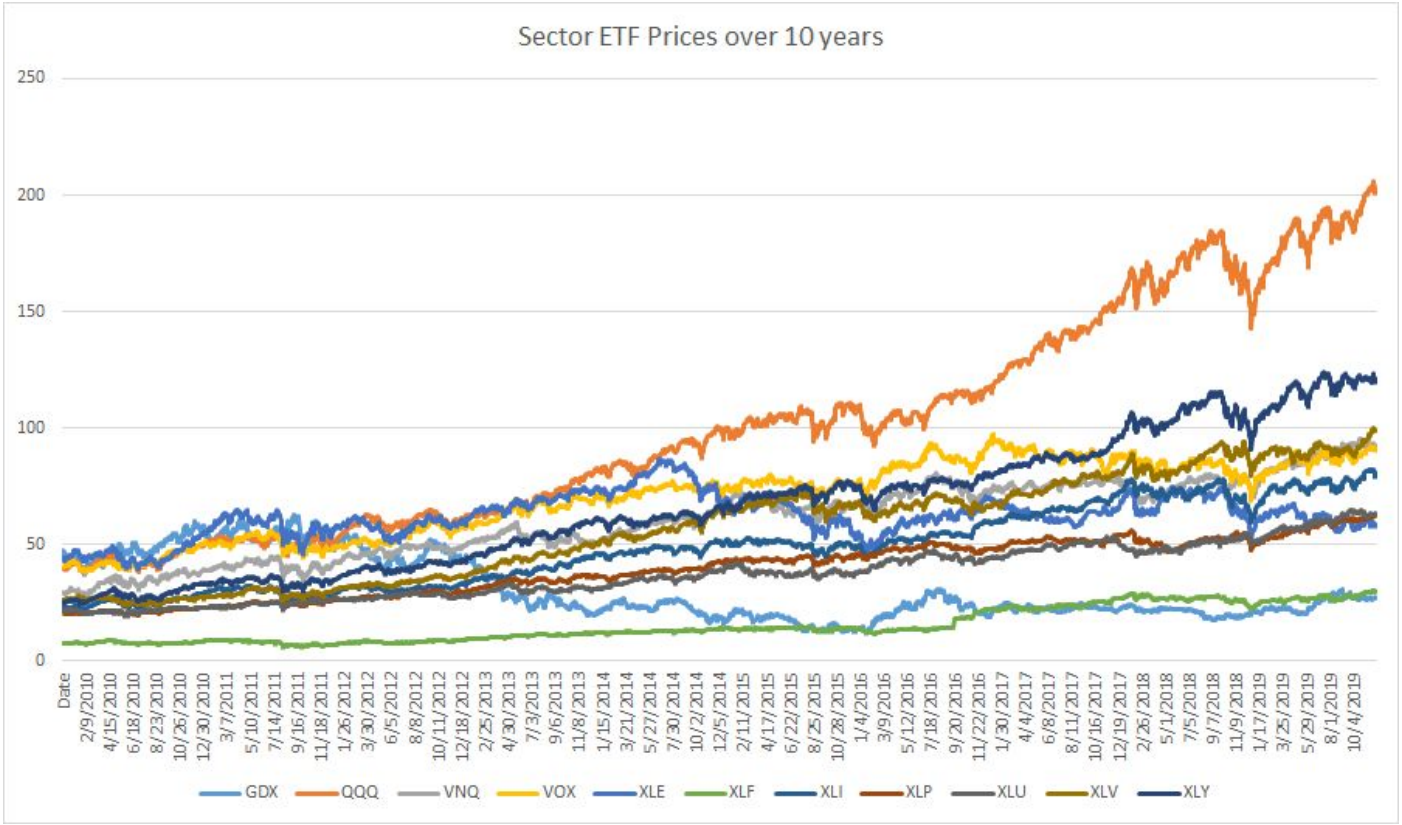
- Benchmark:
  - Yahoo Finance
  - Year: 2009-2019
    - S&P 500 Index

GDX	VanEck Vectors Gold Miners ETF
QQQ	Invesco QQQ
VNQ	Vanguard Real Estate Index Fund
VOX	Vanguard Communication Services ETF
XLE	Energy Select Sector SPDR Fund
XLF	Financial Select Sector SPDR Fund
XLI	Industrial Select Sector SPDR Fund
XLP	Consumer Staples Select Sector SPDR Fund
XLU	Utilities Select Sector SPDR Fund
XLV	Health Care Select Sector SPDR Fund
XLY	Consumer Discretionary Select Sector SPDR Fund

Data: Correlation Matrix between Sectors



Data: Sector ETF Prices over 10 Years



Strategy (Signal)

- Use Quadratic Programming to generate weights that minimize the risks while providing a greater-than-expected return.

$$R = \begin{bmatrix} r_{1,1} & \cdots & r_{1,11} \\ \vdots & \ddots & \vdots \\ r_{11,1} & \cdots & r_{11,11} \end{bmatrix} \text{ is the return matrix for the last 11 periods,}$$

$$\text{The objective function is to minimize } Var\left(\sum_{i=1}^{11} x_i * R_i\right) = X^T Cov(R) X,$$

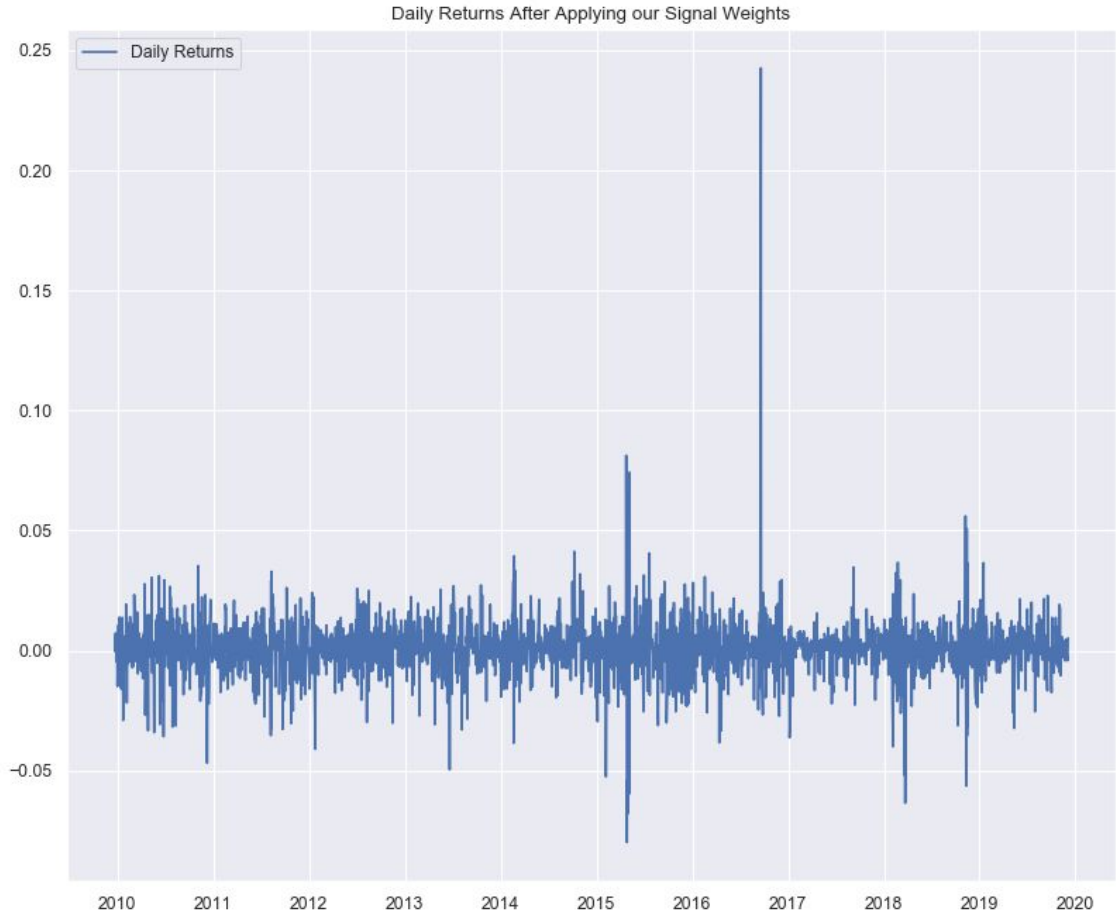
$$\text{while having } \sum_{i=1}^{11} X^T = 1 \text{ and } \forall_i: [r_{i,1} \dots r_{i,11}] * X \geq E[R]$$

Strategy (Construction)

- Key Assumption: The price movement (or the return movement) of the selected ETFs will roughly keep the same in the next n trading days as the previous n trading days. (In this case, n = 11)
- Trading Frequency: Compute the optimal weights every 11 trading days and rebalance the portfolio each time having a new weights
- Sizing: Signal weighting, using the weights computed
- No leverage in general
- Implicitly hedged already since every time we hold both long and short positions in our portfolio; market exposure is fixed as the sum of weights equals to 1

Strategy (Results)

- Daily Returns over the past ten years
- Mean: 0.0008 = 0.08%
- Std: 0.0123 = 1.23%
- Annualized Volatility = 0.1954 = 19.54%



Strategy (Results)

- Cumulative Returns over the past ten years
- From 0 - 2.0625 = 206.25%
- Annualized Return = 0.2304 = 23.04%
- Information Ratio = 1.18
- Sharpe Ratio = 1.08
- Benchmark (S&P 500) Sharpe Ratio over the past ten years = 0.87

Constraints & Concerns

- Can only look back 11 trading days due to the restriction of using Quadratic Programming
- Future security price movements are supposed to be unchanged from the past in a short period (essentially a momentum-kind strategy)
- Short-selling in ETFs can be quite competitive
- Costs: not take into account transaction costs
- Sometimes the strategy is sort of too risky when it has huge long/short positions (weights>1/weights<-1) (In our case, we have 902 long positions that have a weight>1 and 517 short positions that have a weight<-1) → may incur additional risks

Improvements

- Increase the number of securities in the portfolio
- Test over different trading frequencies and find out the best performer
- Longer time horizon in terms of data
- Add stress tests
- Other mean-variance optimization tools
- Other instruments

References

- https://www.mathworks.com/help/optim/examples/using-quadratic-programming-on-portfolio-optimization-problems.html
- http://support.sas.com/documentation/cdl/en/ormpug/63975/HTML/default/viewer.htm#ormpug_qpsolver_sect013.htm
- https://www.morningstar.com/indexes/spi/spx/risk
- https://scaron.info/blog/quadratic-programming-in-python.html
- https://ycharts.com/indicators/10_year_treasury_rate
- https://www.macrotrends.net/2488/sp500-10-year-daily-chart
- https://etfdb.com/
- https://finance.yahoo.com/

Thanks for listening!