2019 Spring

FINA4803

Group 5

Cyclic Factor Investment

Final Report

Andrew Ng (20354380)
Thomas Lui (20352588)
Julia Wu (20329048)
Prashant Khairpuri (20354495)
Isaac Tang (20349000)
Helen Wong (20196087)

Content Page

I. Introduction and Motivation

The cyclic factor investment model **optimally invests in a basket of factors that have outperformed in similar economic cycles**. The model is based on the hypothesis that economic cycles are closely tied to the financial markets, and in different cycles there are different style factors that perform better. The motivation behind such a model stems from overcoming the limitations of the traditional factor model investing. However, let's first define factor investing; it is a strategy that chooses securities on attributes that are associated with higher returns. Factor-based investment strategies has its benefits, it provides investors with targeted and streamlined access to factor exposures. It allows investors to gain a holistic understanding of the drivers of returns. Moreover, the impact and significance of factors can be examined rigorously using statistical tools.

However, factors significance and performance are subject to market conditions, no single factor works all the time and returns tend to be cyclical. For example, swift changes in market direction are typically detrimental to momentum strategies—such as in 2000, following the collapse of the tech bubble, and in 2009, following the rapid recovery from the financial crisis. Another caveat of the traditional factor is the difficulty in controlling the exposure to a single factor in portfolio construction given the complicated correlation amongst the factors. Lastly, factor investing is not a new concept, people have been using factors to invest for decades, the investment scene is heavily crowded. With so many people trading similar factors, the directional return will be diminished.

The motivation is to tackle the cyclicality of factor investing. Through decomposing the level of Chinese economic activity into different stages of an economic cycle, the model can time the market and invest in factors that have outperformed in similar economic cycles of past in an aim to replicate those superior returns.

II. Data

In order to construct factor baskets of stocks and economic cycle, mainly 2 types of data were collected throughout the project, (i) data of Hang Seng Composite Index (HSCI) constituents and (ii) China macroeconomic data.

HSCI is chosen as the investable universe of the project. It provides a standard investable universe of Hong Kong stocks as grouped by Hang Seng Indexes Company. Its 494 constituents cover 95th percentile of total market capitalisation of the Hong Kong stock market. The below graph shows the trend of HSCI in the past 30 years.

Factor baskets of HSCI constituents are constructed by fundamental data and price movement of each constituent. Therefore, daily stock returns and quarterly fundamental data (details will be covered in the methodology part) are collected during the project for factor basket construction.

For economic cycle, since HSCI is chosen and the project will invest in Hong Kong stock market, the cycle is constructed using China monthly macroeconomic data so as to reflect the economy of the investable universe.

In terms of scope, 20 years of monthly data was collected which allowed the model to cover at least 2 financial cycles driven by financial crises and regulatory changes, as well as economic cycles of 6 years in average.

Fundamental and price data are mostly collected through Bloomberg Terminal, Thomson Reuters, and companies' or regulators' official websites. Python is used to scrap the available data on the websites. On the other hand, through Bloomberg STAT function, China macroeconomic variables from National Bureau of Statistics of China are collected. Collected data are processed using Python and R for analysis and comparison.

III. Methodology

1. Factor Basket Construction

The factor backet construction involves two major stages including factor selection and formation of factor basket. For factor selection, besides the most commonly used factors such as value, growth and momentum, other factors were also adopted based on research findings and current market trend.

Factor	Metrics
Momentum	1M Price Reversal; 11M Price Momentum
Growth	12M EPS Growth; 12M Dividend growth
Value	Forward 12M EPS; Forward 12M Growth; Trailing 12M P/B
Strong Balance sheet	Altman Z-score; Net Debt-to-Equity Ratio
Dual Beta	2Y Beta to HSCI; 2Y Beta to China Industrial Production
Sharpe Ratio	Trailing 6M Sharpe Ratio
Buyback	1 Year Buyback Yield
Short Interest	Short Interest Days

To form the factor baskets, each HSCI index constituent was ranked by metric value. Next, 2 baskets of equal-weighted stocks are formed for each factor: High Basket with stocks at top 10% rank and Low Basket with stocks at bottom 10% rank. Each basket contains around 50 stocks.

2. Cycle State Identification

Four monthly macro variables, China Retail sales, China Industrial Production, China Fixed Asset Investment (excluding rural areas), Value Added of Industries and Export Values were collected to indicate the level of economic activity. Principal Component Analysis was used to combine all macro variables into one, the first Principal Component indicates the growth of economic activities in different time.

The Hodrick-Prescott filter was then introduced to separate the macro series into four different cycle periods. The HP filter decomposes the first Principal Component into two components: trend and cycle. By comparing the original PC series and the trend line, the cycle labels are obtained. For example, when the trend line is above the first PC and increasing during the period, it is in above-trend-accelerating state.

Labels	Above Trend	Below Trend
Accelerating	above-trend-accelerating (1)	below-trend-accelerating (3)
Decelerating	above-trend-decelerating (2)	below-trend-decelerating (4)

3. Fitting Factor Basket Returns to Cycle Model

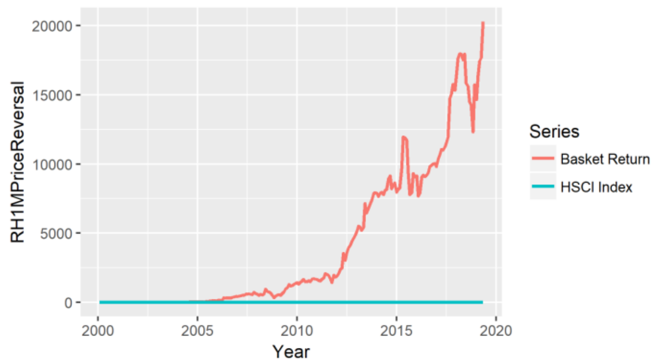
After forming different factor baskets and identifying four cycles, the average historical return for each basket under the four cycles is calculated. Hence, the top performing baskets in each cycle period can be identified. It is expected that the outperforming factors will generate the highest return under current cycle and they are the factor baskets that will be invested in.

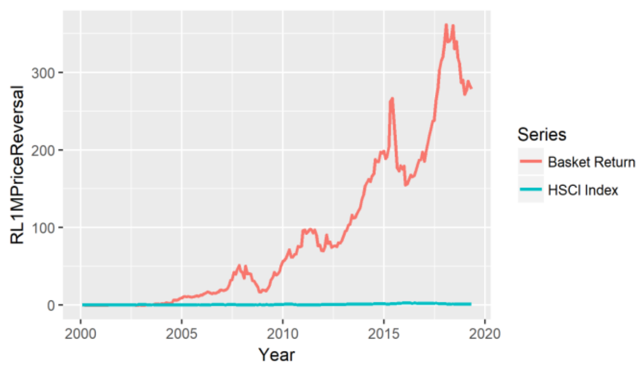
Finally, after deciding which basket will be invested in by fitting into cycle model, the weighting of the selected baskets needs to determined. The allocation of the three baskets will be optimised to generate the highest risk adjusted return. This will used to form the investing portfolio. The portfolio will be optimised and rebalanced on a monthly basis, to ensure that the optimal factors and corresponding stocks are being invested in.

IV. Empirical Results

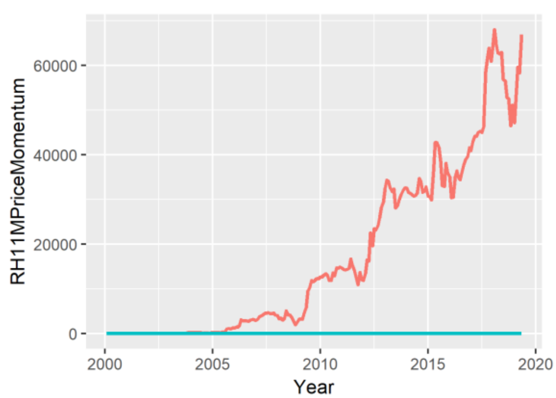
1. Factor Basket

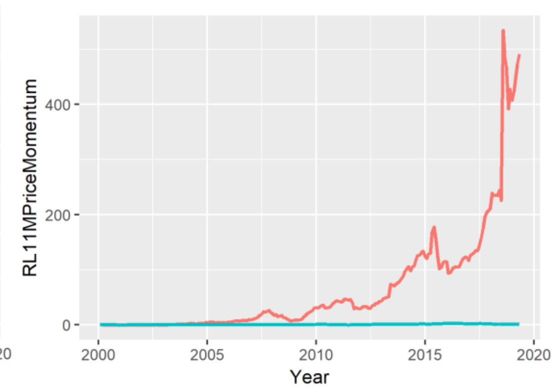
By ranking each HSCI index constituent by metric value, 2 baskets of equal-weighted stocks are formed for each factor: High Basket (RH prefix) with stocks at top 10% rank and Low Basket (RL prefix) with stocks at bottom 10% rank. Here is the trend of the 28 baskets created.



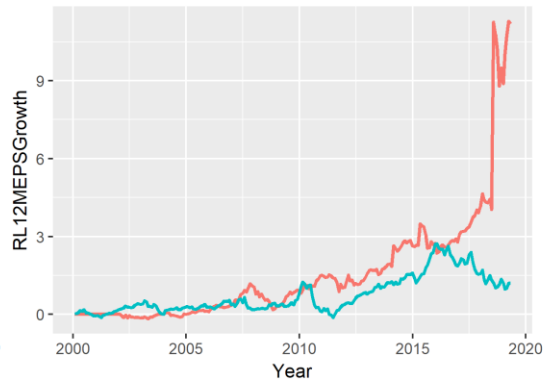


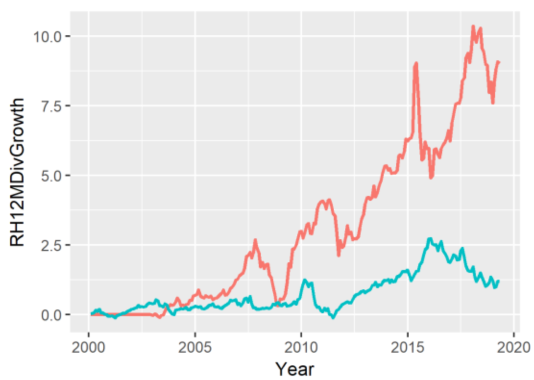
The series will remain the same for all the factor basket graphs below.

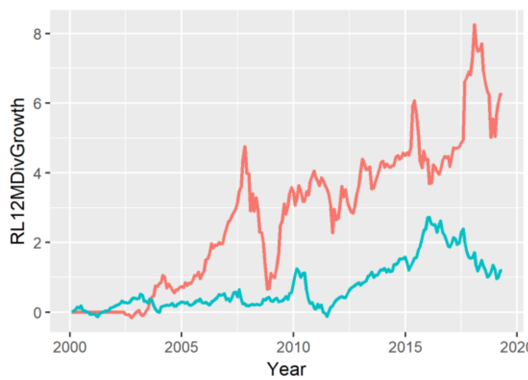


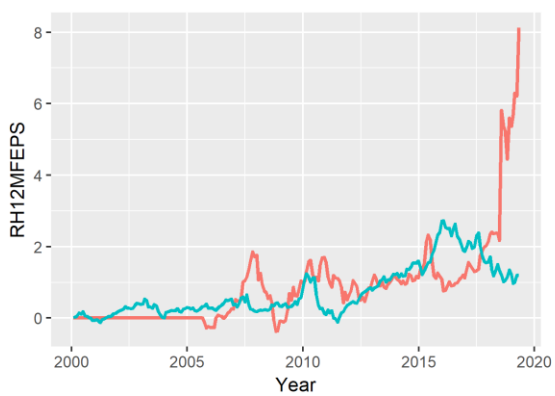


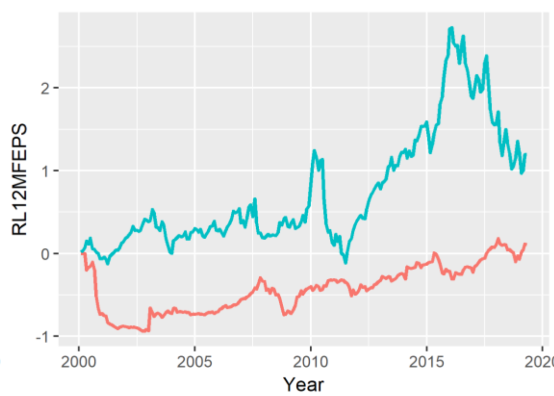

RH12MEPSGrowth
0
2
4
2000
2005
2010
2015
2020
Year


RL12MEPSGrowth
0
3
6
9
2000
2005
2010
2015
2020
Year

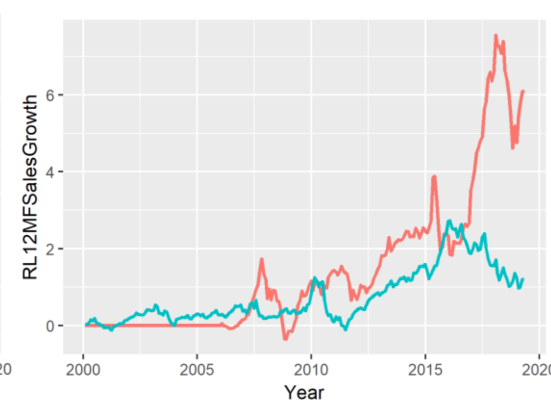

RH12MDivGrowth
0.0
2.5
5.0
7.5
10.0
2000
2005
2010
2015
2020
Year


RL12MDivGrowth
0
2
4
6
8
2000
2005
2010
2015
2020
Year

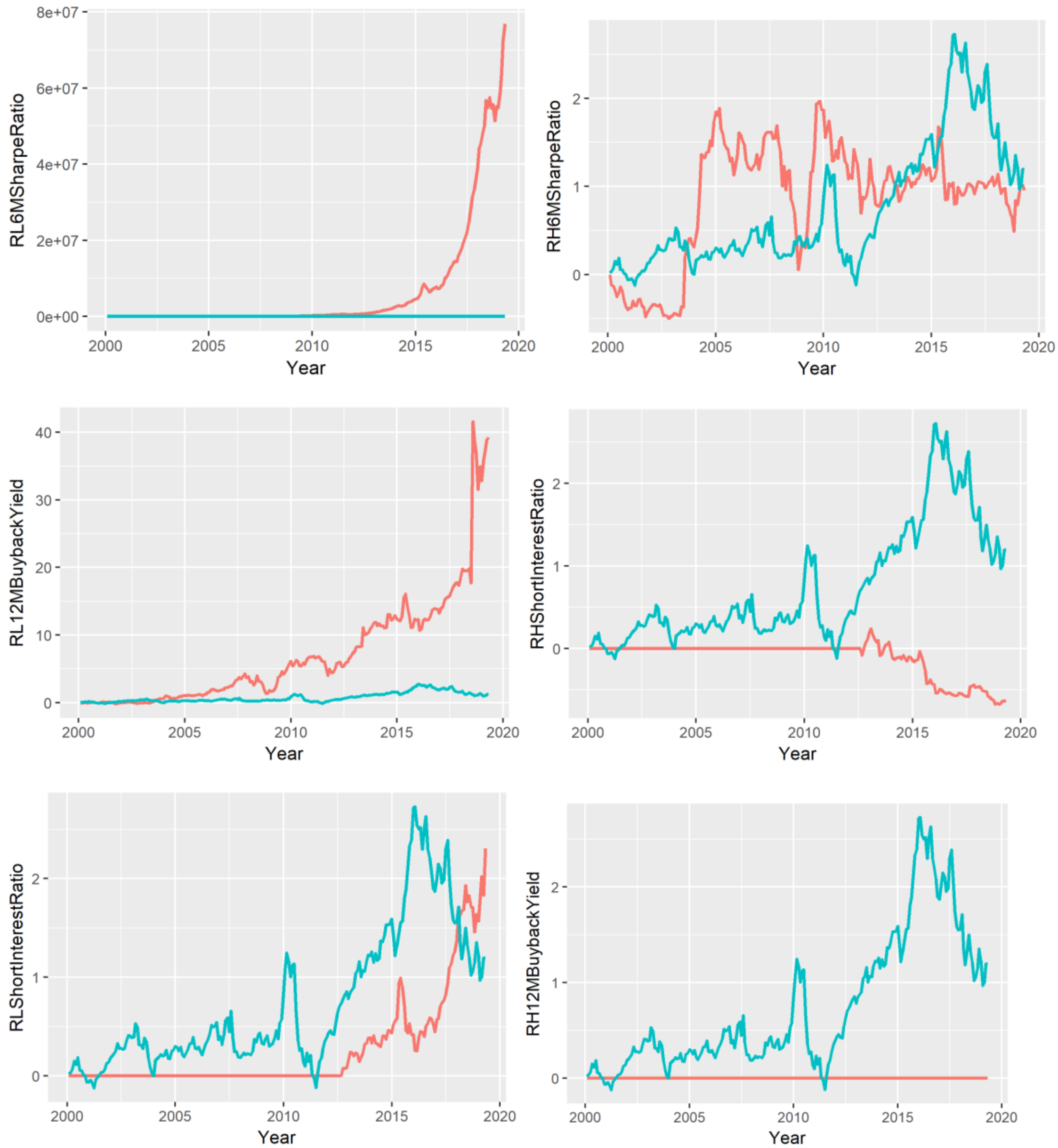

RH12MFEPS
0
2
4
6
8
2000
2005
2010
2015
2020
Year


RL12MFEPS
-1
0
1
2
2000
2005
2010
2015
2020
Year


RH12MFSalesGrowth
0
1
2
2000
2005
2010
2015
2020
Year


RL12MFSalesGrowth
0
2
4
6
2000
2005
2010
2015
2020
Year

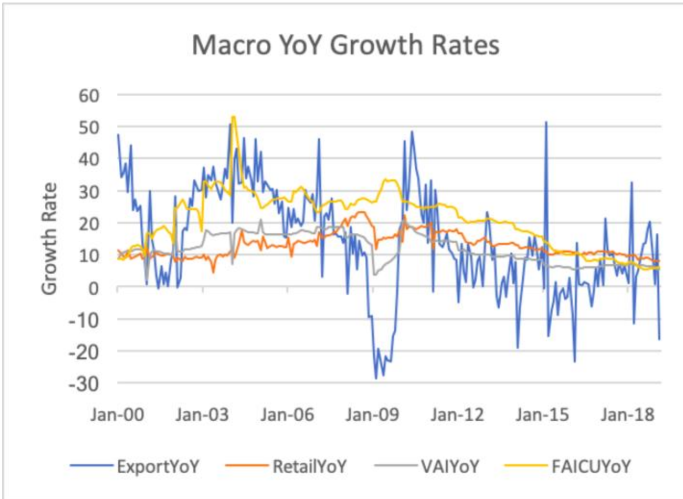

RL12MPB
RH12MPB
RHAltmanZScore
RLAltmanZScore
RL2YBetaHSCI
RL2YBetaHSCI
RH2YBetaCIP
RL2YBetaCIP
Year



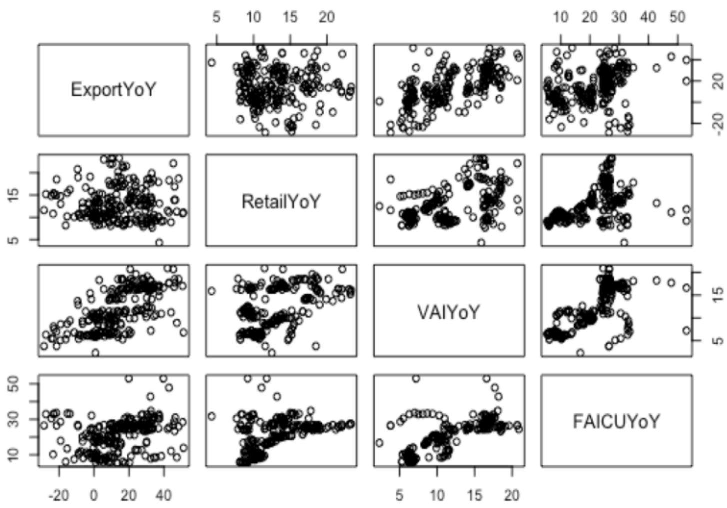
Note that some baskets are not characteristic at all (e.g. low buyback yield, due to most stocks have no buyback action in the previous 1 year) and are dropped during portfolio construction, and only retained here for purpose of symmetry. Some extreme spikes are noted in the return, and after investigation, are attributed to a stock allegedly engaged in price manipulation. Such stocks are removed from the portfolio construction as well, but are retained here as the factor basket construction procedure is algorithm in nature (see Appendix), while the anomaly checking is manual.

2. Cycle Model

As aforementioned, four monthly macro variables, China Retail sales, China Industrial production, China Fixed asset investment (excluding rural areas), Value Added of Industries and Export Values were used to indicate the level of economic activity. Since the original macro series are with high seasonality, the year-over-year growth rates of the series was used to remove seasonality. After removing seasonality, the components remaining in the series would be cycle index, trend and some random shocks.



The above chart shows the year-over-year growth rate series. Despite a higher volatility shown in the Export series, they share similar trend and pattern.

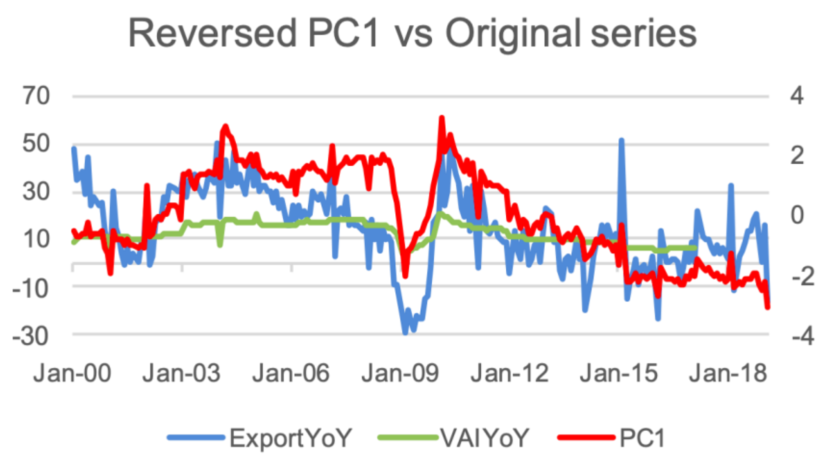


Besides, scatter plots between the pairs indicate that the series are roughly correlated. This is a desired feature as less information will be lost when we extract the main components using the PCA.

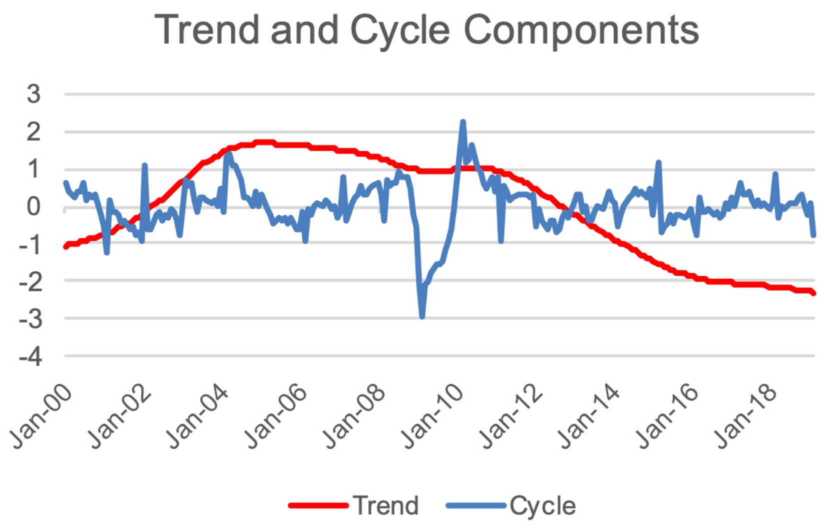
Next, each metric will undergo a Z-score normalization process as PCA requires each column to have zero mean and unit variance in order for the algorithm to find a correct first principal component. This is an essential step to ensure comparability across metrics with different units.

Fitting the scaled data to a Principal Component Analysis, the first component was generated and it accounts for 58.2% variance of the original series. Apart from that, Fixed Asset Investment and Value Added of Industries align more closely with PC1.

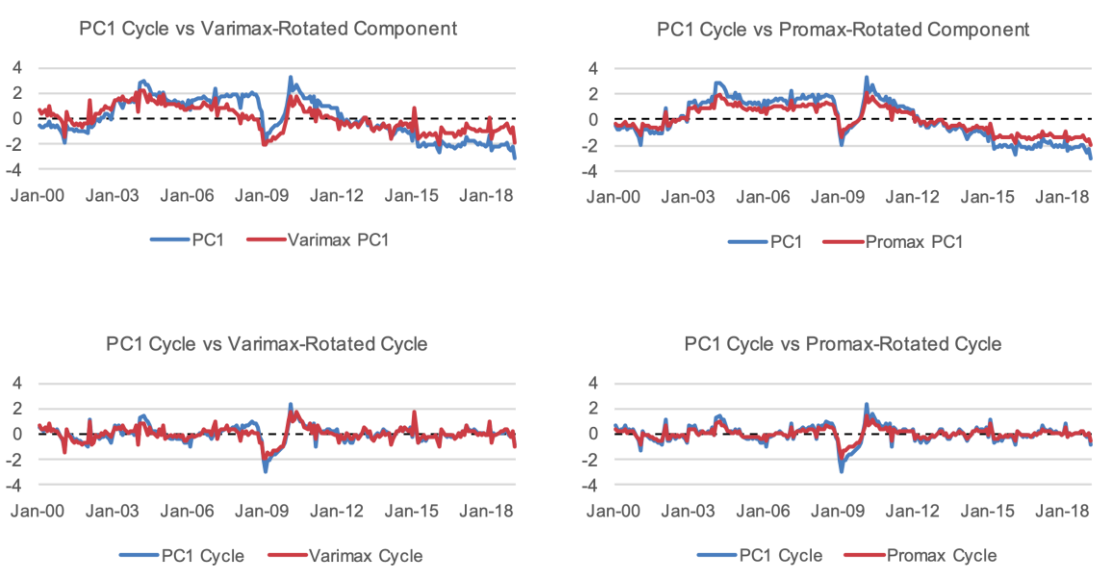
Since all the series are negatively aligned with PC1, its sign was reversed to make it more interpretable.



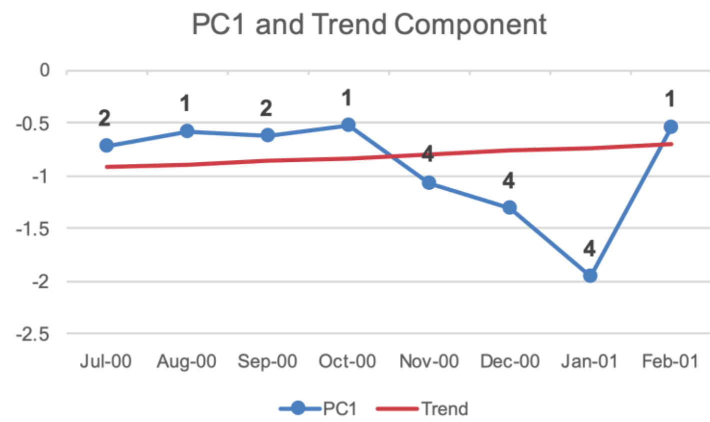
The PC1 is then fit to a HP filter with a frequency of 14,400, which decomposes PC1 into two different components. The first component is the trend line, the long-term smoothed level of the economic activities. The other component cycle is an index centred at 0. When the cycle index is above 0, the economy is in a recovery state. On the other hand, the economy is in the a recession state when the cycle index is below 0.



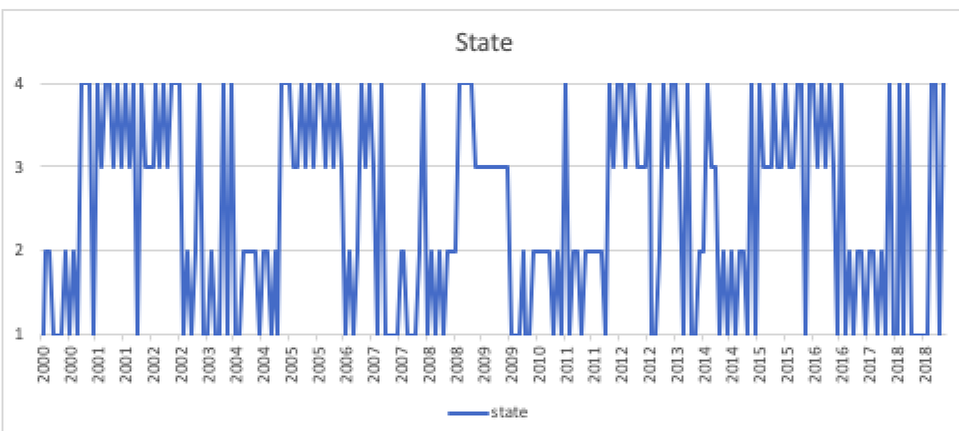
Aside from the backward shifting and forward shifting cases, rotation was also performed on the PCA to see how the PC and cycle components will be affected. By using rotation in factor analysis, a small number underlying common factors generating the observed variables were found. It is assumed that each observed variable is generated as a linear function of the underlying factors, plus a unique random error term. Here it can be noticed that the general trend of the rotated series are close to the one in the base case albeit with many discrepancies observed. Further test cases are included for the rotated series to check if they are more significant in identifying cycle.



The chart below illustrates how cycle states are identified; When the principal component is above the trend line and increasing, it is labelled as above-trend-accelerating (1); When the principal component is above the trend line and decreasing, it is labelled as above-trend-decelerating (2); When the principal component is below the trend line and increasing, it is labelled as below-trend-accelerating (3); When the principal component is below the trend line and decreasing, it is labelled as below-trend-decelerating (4).

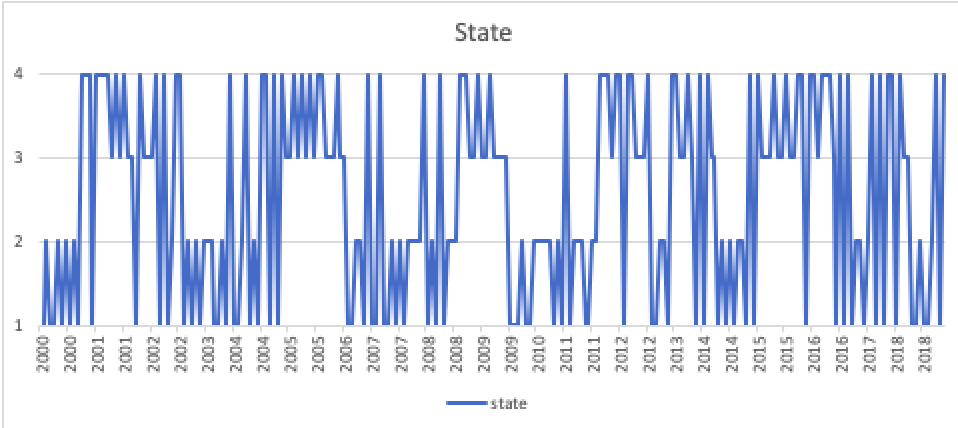


Here are the final results for base case of the cycle part, the state labels for the months. Form the graph, it can be observed that, in short term, the labels fluctuate between accelerating and decelerating while changes in above-trend and below-trend can be seen in the medium term. The frequent change of labels in the short term might not be a desired feature as this raises the need for rebalancing and thus increases the transaction costs. A smoother version for the labels was sought to tackle this issue.

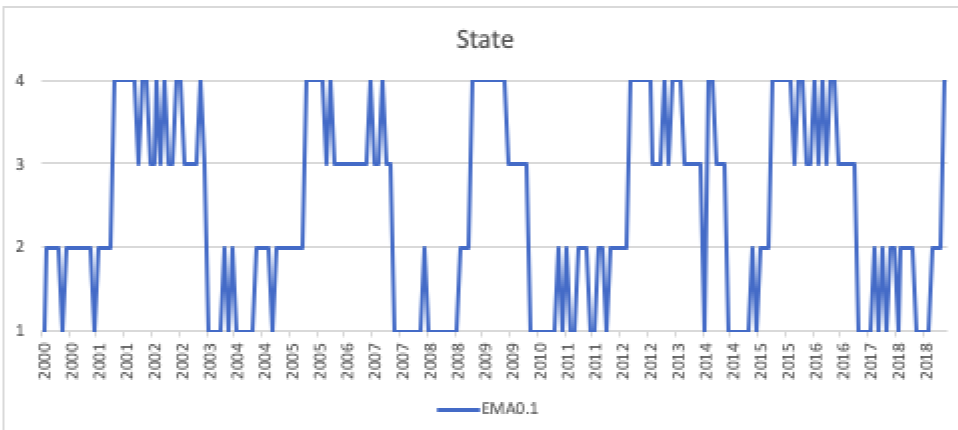


Below are the results for 5 extensions of the cycle model:

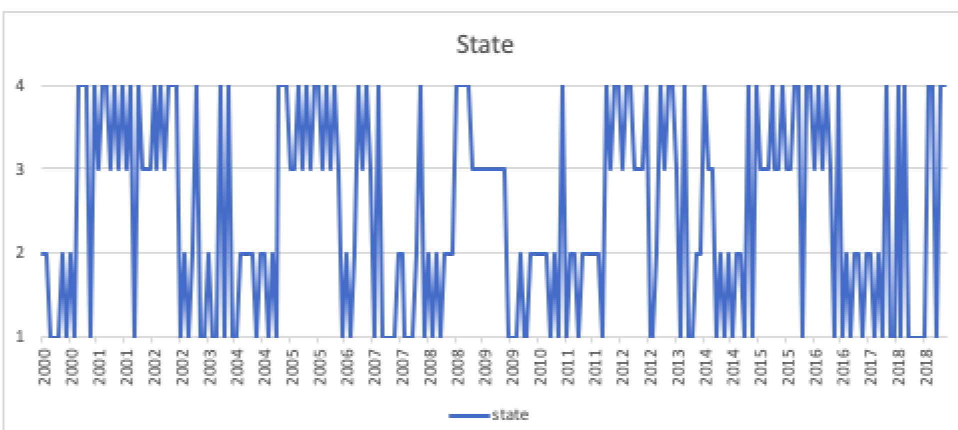
1. Varimax Rotation on the principal component



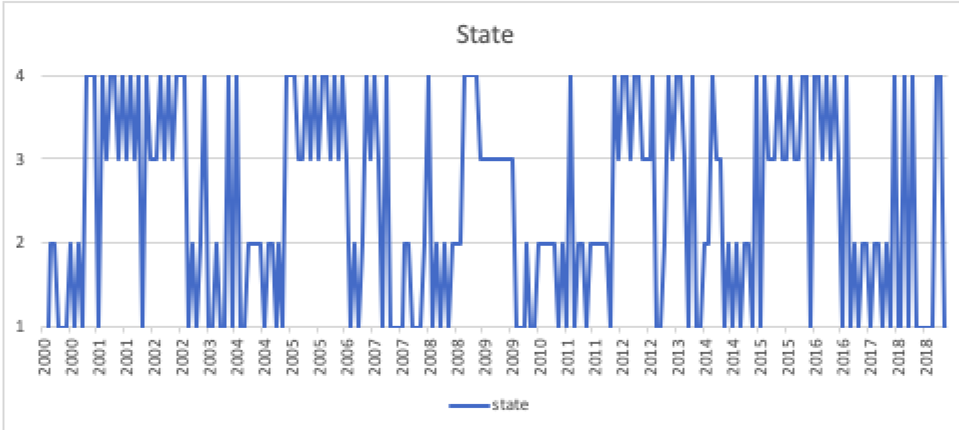
2. Exponential Smoothing on the PC1 with a smoothing parameter 0.15



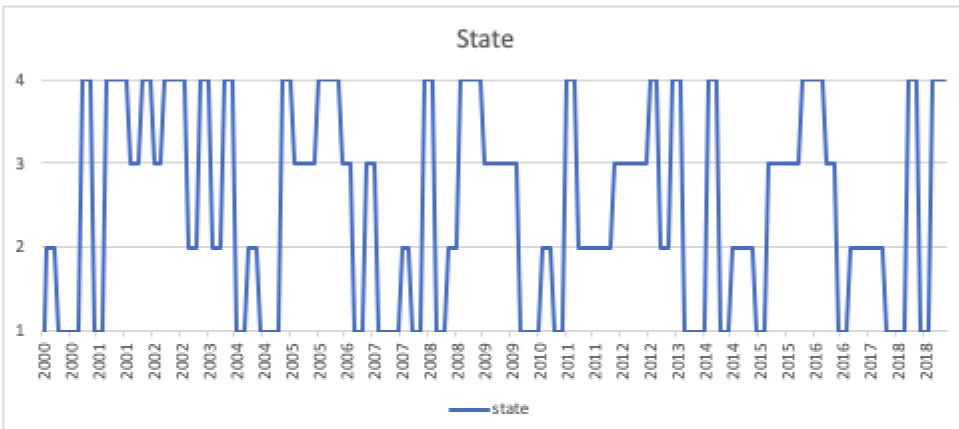
3. Lag 1 month Cycle Model



4. Forward 1 Month Cycle Model



5. Quarter Constraint Model: the labels for months in the same quarter are constrained to be the same.



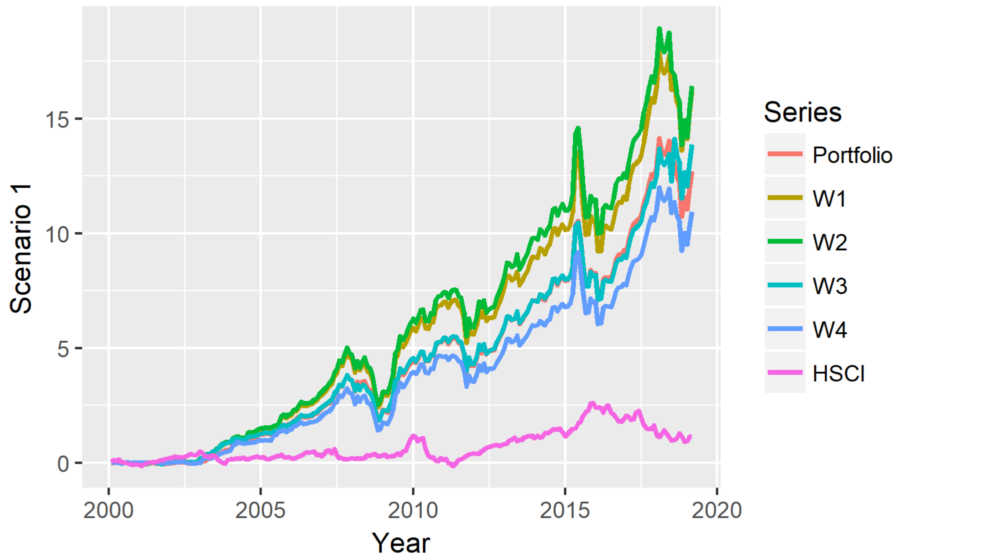
3. Portfolio Performance

Here are the optimal factor basket allocations for each state and overall portfolio returns under the 6 different cycle models.

Note that Wx cycles where $x \in \{1, 2, 3, 4\}$ denotes a hypothetical investment into the optimized portfolio under each economic cycle, disregard what the actual cycle is (i.e. W1 is the cumulative return where you invest into the optimized portfolio of stocks under macro cycle 1, even when the current cycle is cycle 2).

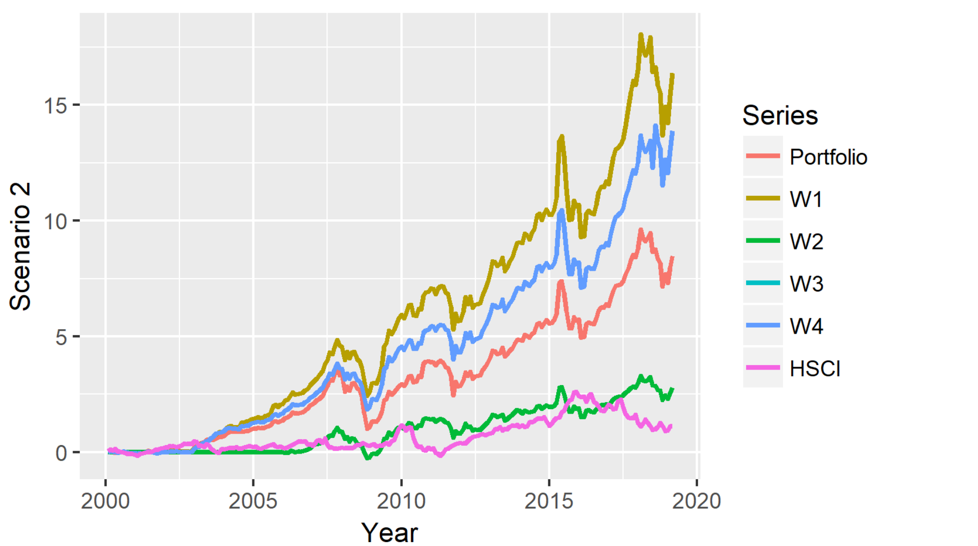
1. Base case

State 1		State 2		State 3		State 4	
RH 12M Buyback Yield	86%	RL Forward 12M EPS	99%	RL Net Debt-to-Equity Ratio	100%	RH 12M EPS Growth	70%
RL 12M EPS Growth	14%	RH 12M EPS Growth	1%			RL 12M EPS Growth	18%
						RL Net Debt-to-Equity Ratio	12%



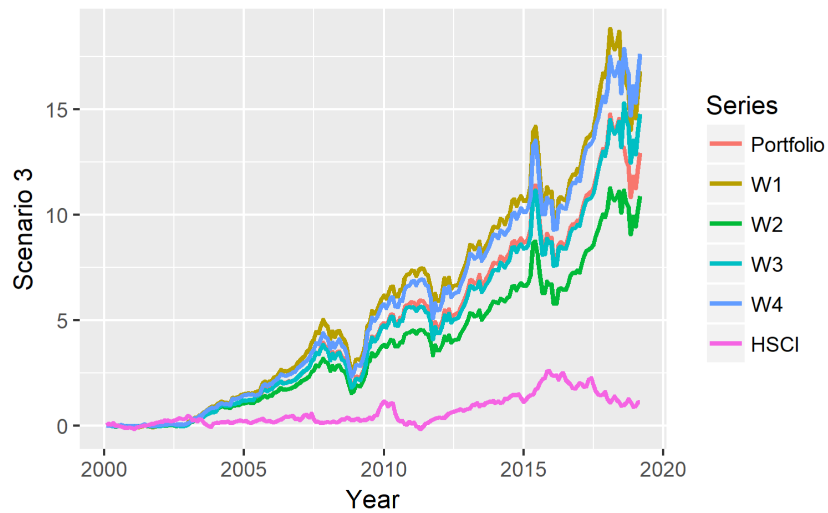
2. Varimax Rotation

State 1		State 2		State 3		State 4	
RH 12M Buyback Yield	88%	RL Forward 12M EPS	100%	RL Net Debt-to-Equity Ratio	100%	RL Net Debt-to-Equity Ratio	100%
RL 12M EPS Growth	12%						



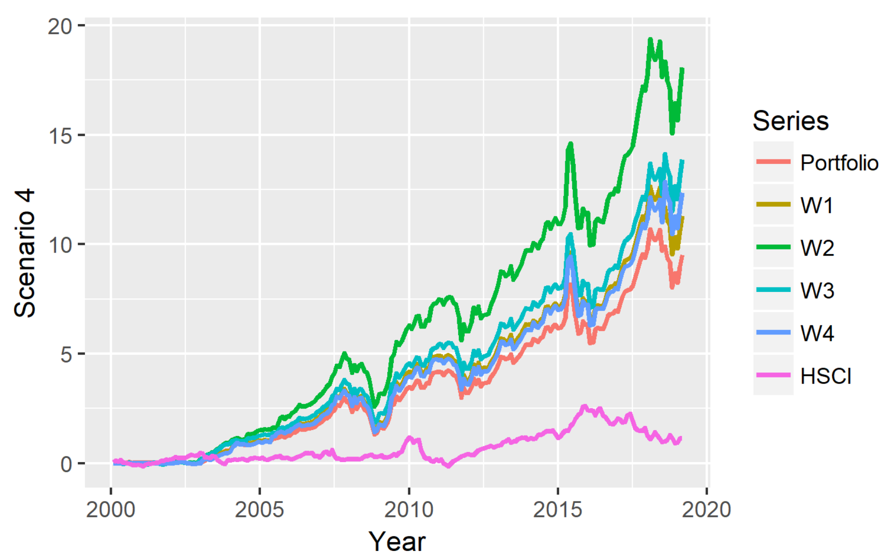
3. Exponential Smoothing on the PC1

State 1		State 2		State 3		State 4	
RH 12M Buyback Yield	99%	RL Net Debt-to-Equity Ratio	70%	RL 12M EPS Growth	57%	RH Trailing 12M P/B	93%
RL 12M EPS Growth	1%	RH 12M EPS Growth	23%	RH Net Debt-to-Equity Ratio	43%	RH 12M Buyback Yield	7%
		RH Short Interest Ratio	7%				



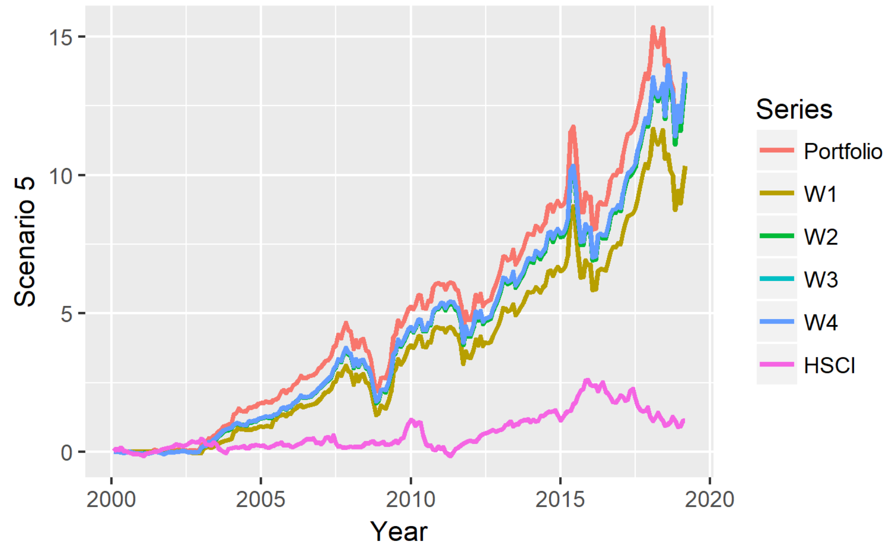
4. Lag 1 month Cycle Model

State 1		State 2		State 3		State 4	
RH 12M EPS Growth	73%	RH 12M Buyback Yield	69%	RL Net Debt-to-Equity Ratio	100%	RL 12M EPS Growth	100%
RL Net Debt-to-Equity Ratio	18%	RH Forward 12M EPS	31%				
RL Forward 12M EPS	9%						



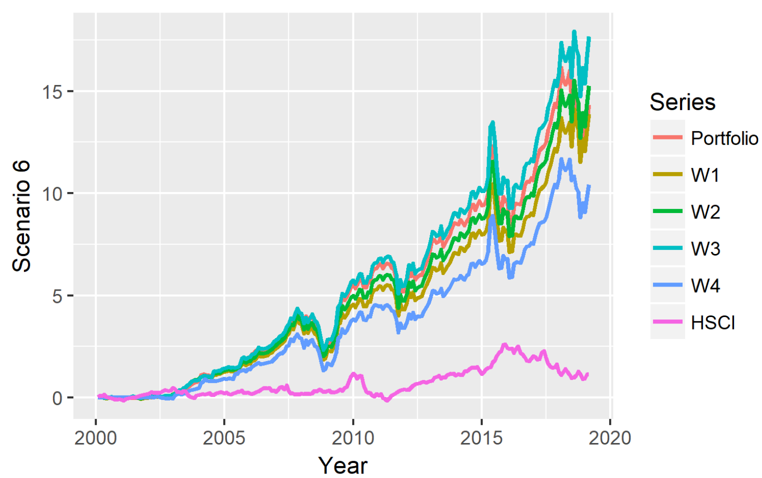
5. Forward 1 month Cycle Model

State 1		State 2		State 3		State 4	
RH 12M EPS Growth	88%	RL Net Debt-to-Equity Ratio	89%	RL Net Debt-to-Equity Ratio	100%	RL Net Debt-to-Equity Ratio	100%
RL 12M EPS Growth	12%	RH 12M Buyback Yield	11%				

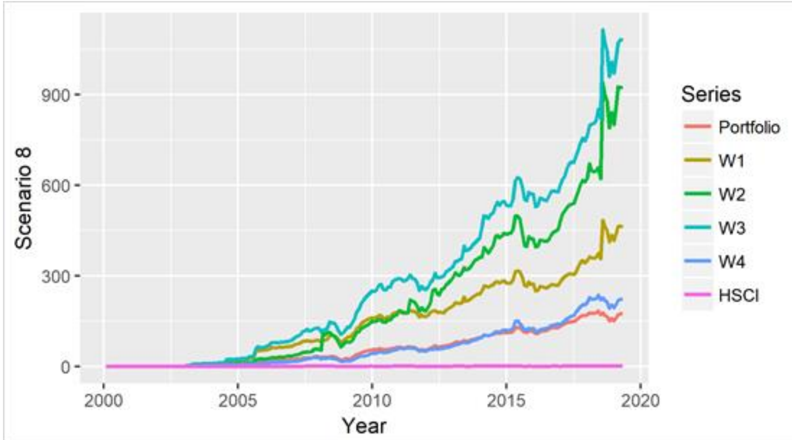


6. Quarter Constrain Model

State 1		State 2		State 3		State 4	
RL Net Debt-to-Equity Ratio	100%	RL Net Debt-to-Equity Ratio	38%	RL Net Debt-to-Equity Ratio	100%	RH 12M EPS Growth	84%
		RH Trailing 12M P/B	62%			RL 12M EPS Growth	16%

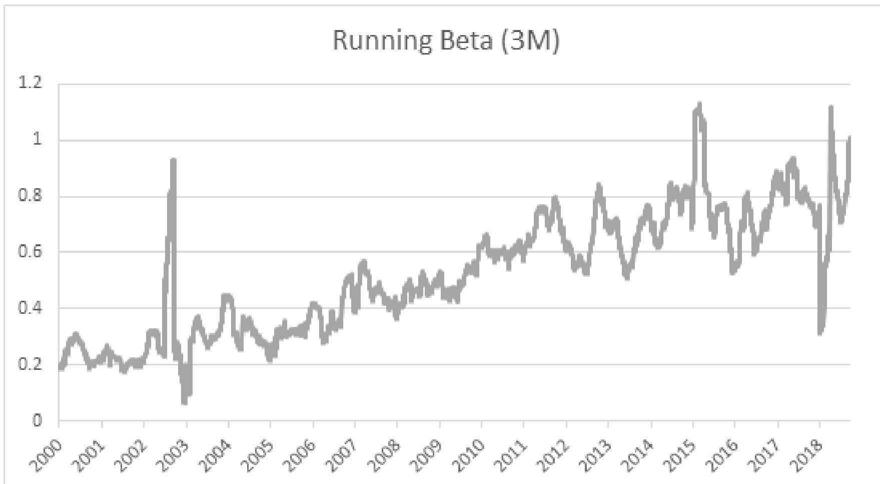


7. Combination of Lag 1 Month and Quarterly Constrained Model
After observing all the scenarios, it was concluded that it would be best to adopt a combination of Model 4 and 6, which is a Lag 1 Month and Quarterly Constrained Model.


Scenario 8
900
600
300
0
2000
2005
2010
2015
2020
Year
Series
Portfolio
W1
W2
W3
W4
HSCI

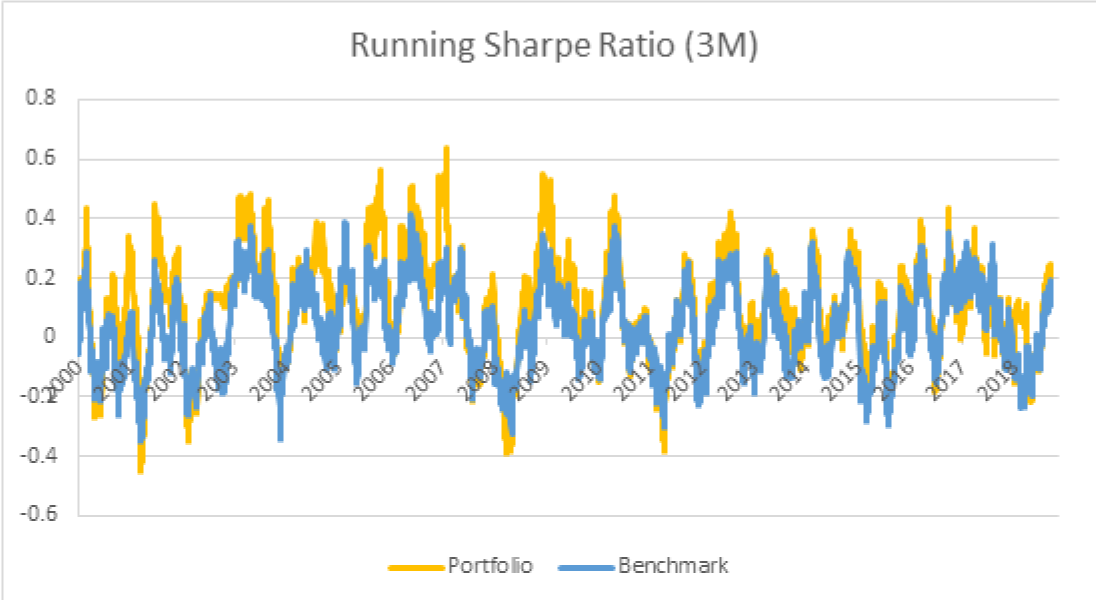
V. Risk Management

In our strategy, risk mainly comes from two aspects – traded risk and risk metrics.
Traded risk is risk that arise from the portfolio and the trading activities, and here our group look at the running beta of our portfolio to examine how risky it is compared to our market benchmark, which is the Hang Seng Composite Index.

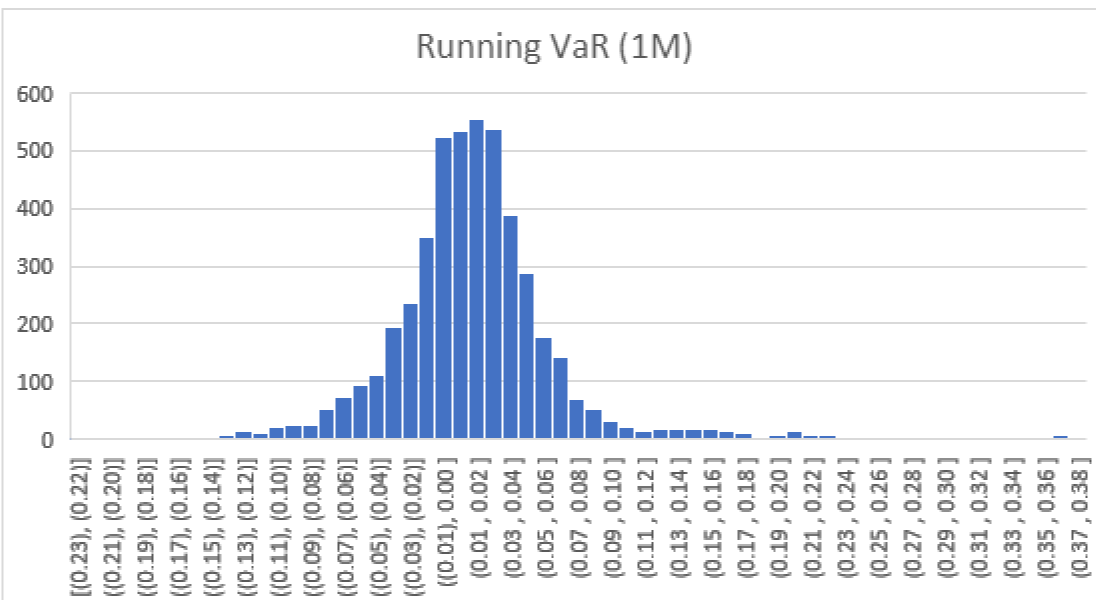


From the graph, we can see the running beta of our strategy fluctuates between 0.6-0.8 of the benchmark for the last decade. It is imperative that our strategy produces portfolio that is less sensitive to market-wide shocks.

Another metric that can measure our strategy's expected return with respect to its volatility is by examining the Sharpe ratio. Our strategy quite consistently outperform the benchmark in terms of risk-reward tradeoff, as captured by the running Sharpe ratio of the portfolio:

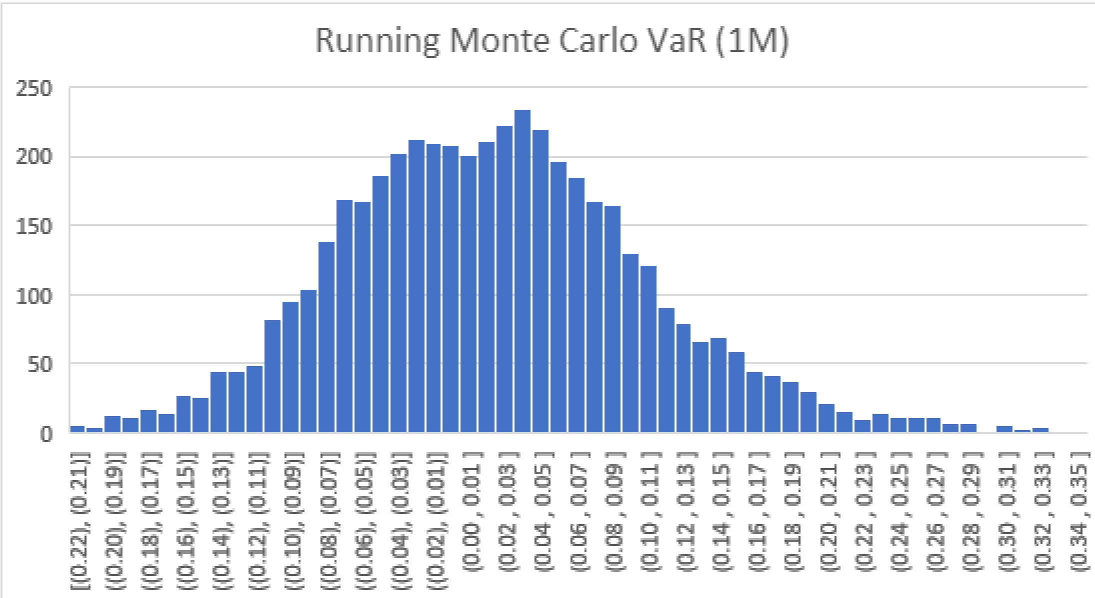


On the other hand, we can explore our strategy return distribution, which allows us to examine our strategy's value-at-risk (VaR). Our strategy has an approximately normal 1M cumulative return distribution with some extreme values on the right side (positive return). The 5% VaR on 1M horizon sits at -9% for the portfolio.



By running a Monte Carlo simulation over the optimized portfolio weights for repeated trials for a 1M investment horizon, with distribution given by recent 1 year return of our portfolio, we can simulate our future portfolio VaR.

The resulting distribution is heavy-tailed, with a 5% VaR on 1M horizon sitting at -14%.

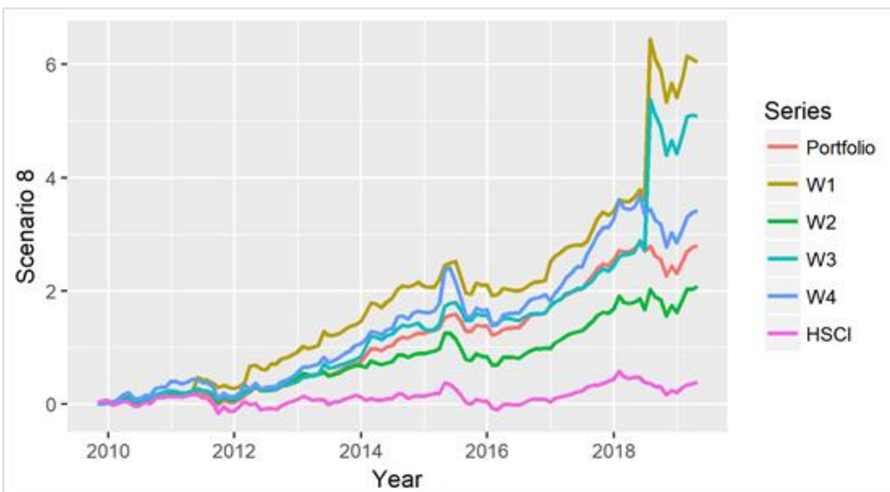


VI. <u>Robustness Tests</u>

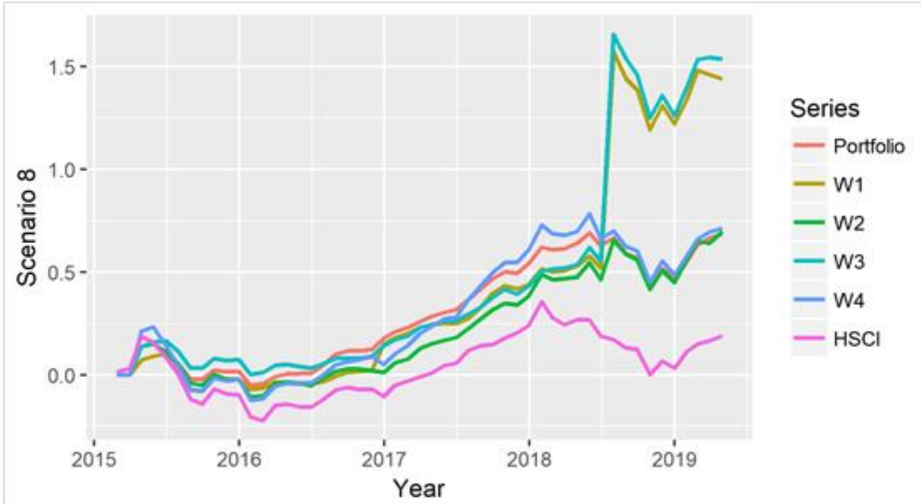
The model is based on the hypothesis that economic cycles are closely tied to the financial markets, and in different cycles there are different style factors that perform better. The backtesting results of the final selected model (Lag 1 Month Quarterly Constrained Cycles) indicated a 91.84x cumulative return over a 20+ years horizon.

Staying true to the hypothesis, there are many parameters in the model that can be tuned to give different results. The parameters include model training horizon (H), factor high/low basket cutoff (C), and maximum weight allowed in each basket (*w*). Note that our final selected model is trained based on (H = 2000:1 - 2019:4, C = 0.1, *w* = 0.25).

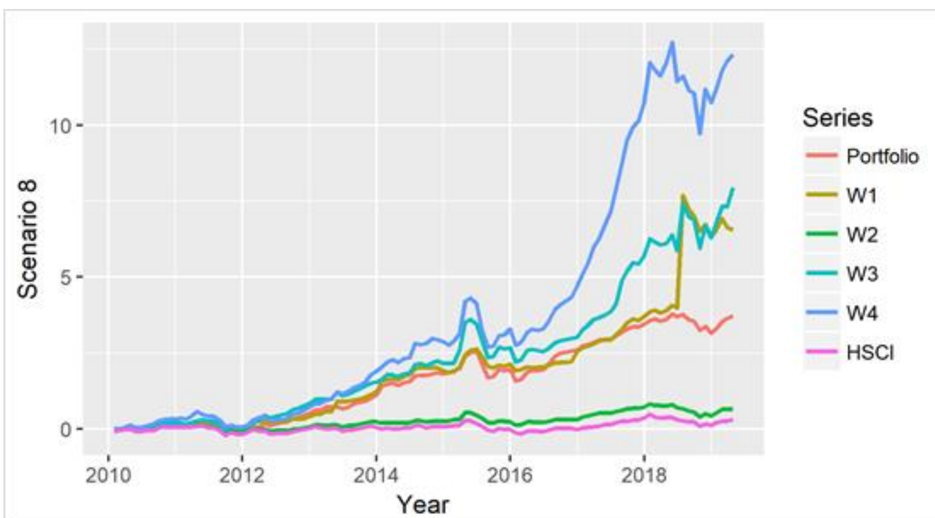
In the factor basket construction phase, since our original model is based on 20 year of market data. If we subset the data to retain only the most recent 10 year, thereby adjusting the parameter H, our strategy still gives a 2.8x of cumulative return:



Note that due to the high dimensionality of the data, the more reduced the training horizon is, the less characteristic the optimized portfolio weights under each cycle would be, due to having less available data to train our model. With a 5-year horizon model where (C = 2015:1 - 2019:4), the model still gives a 0.7x cumulative return. Note that the highly overlapped W*x* and portfolio return series mean our model cannot distinguish the outperformers in each macro state clearly:

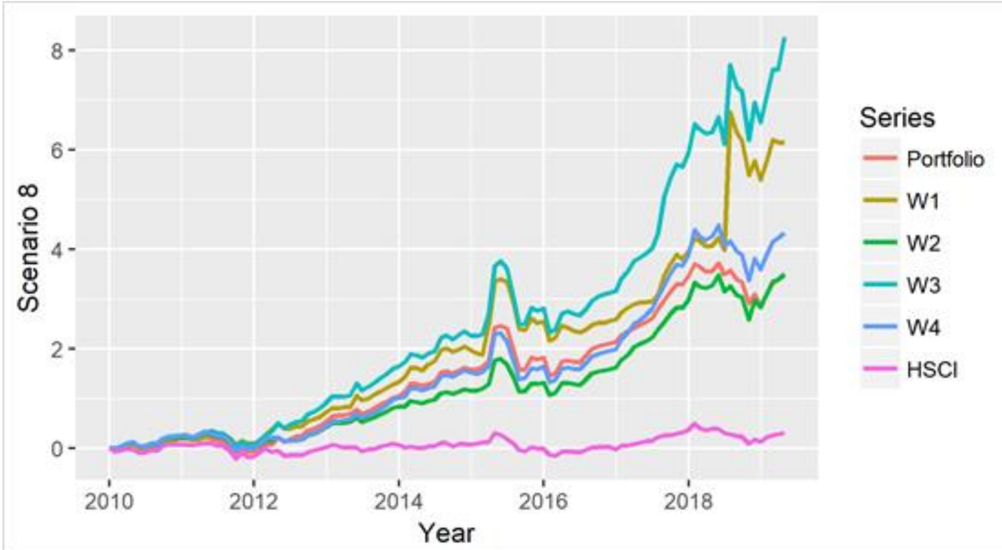


Reverting to the 10-year horizon, the factor cutoff can be further altered from the (C = 0.1), which means top and bottom 10% percentile of stocks in each factor will go into their respective high and low factor baskets, to (C = 0.05). Intuitively, this action strengthened our model's hypothesis, in a way that only stocks with more extreme factor values are included into the factor baskets, such that our strategy would be more sensitive to economic cycle changes. Under these parameters, the portfolio yielded a 3.72x Cumulative return:

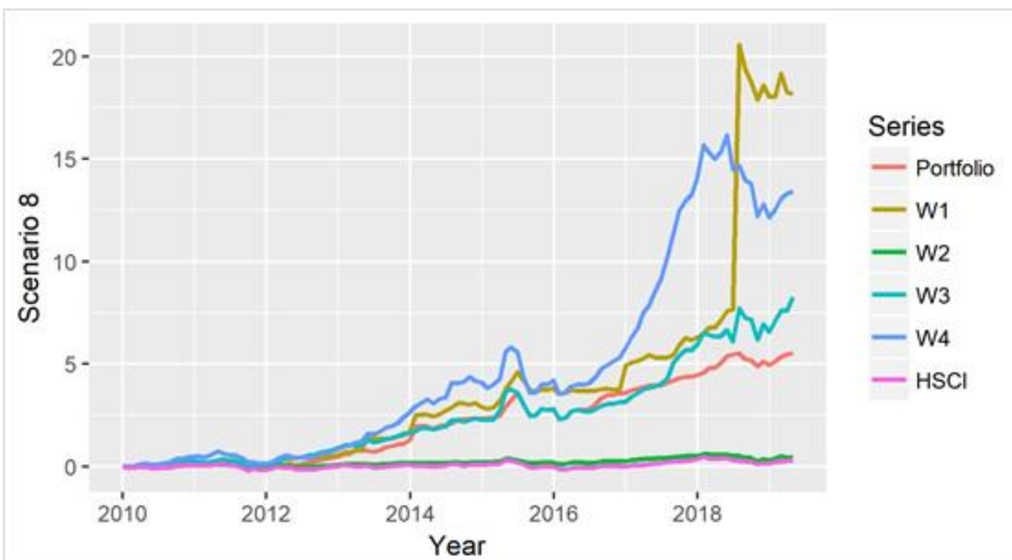


Specific to the portfolio optimization phase, the parameters can also be changed to create different bases for the optimizer to use. This will indirectly alter the number of stocks in our portfolio and therefore the extent of diversification. The case above where (H = 2010:1 - 2019:4, C = 0.05, w = 0.25), the portfolio on average invested into 81 different stocks in each cycle subjected to board lots constraints.

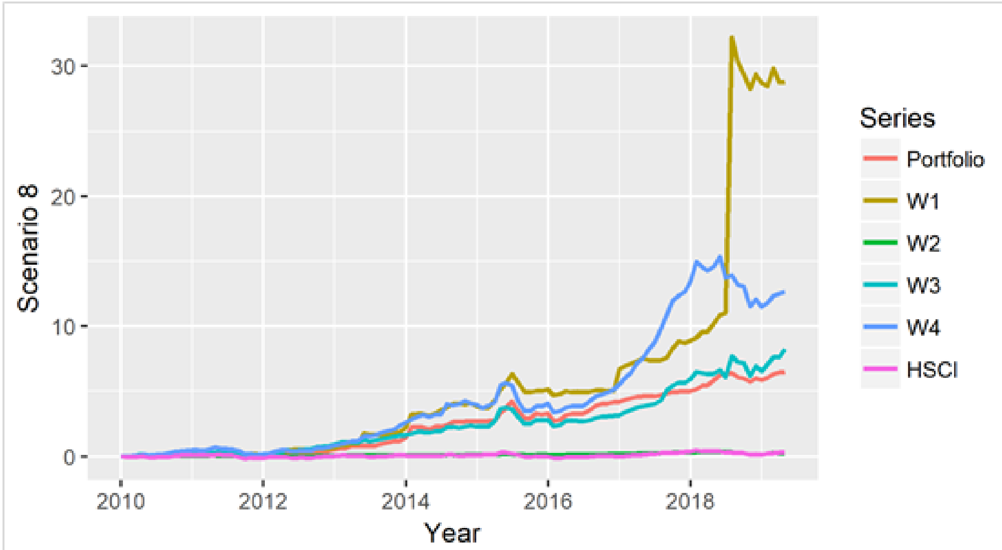
When (w = 0.10), the portfolio yielded a return of 3.46x, with an average of 124 stocks invested in each cycle:



When (w = 0.50), the portfolio yielded a return of 5.53x, with an average of 68 stocks invested in each cycle:



When (w = 0.75), the portfolio yielded a return of 6.50x, with an average of 49 stocks invested in each cycle:



Note that increasing (w) almost always lead to an increase in the return, due to only small positive and no negative correlation between baskets. This however will increase the volatility of portfolio, as the optimizer runs on optimizing the Sharpe ratio of our portfolio. To diversify the source of returns to different factors, a constraint of ($w = 0.25$) was chosen to try to enforce that no one factor could dominate in our optimized portfolio, such that the portfolio performance can be replicated in the future.

It shows that the results initially presented in the report are not forged, and while parameters being deliberately chosen based on intuition, hypothesis and strategy still fare well in the market under different assumptions and parameters.

VII. Thought Process

The biggest challenge encountered during the project was in the optimization of the portfolio construction process. As there are a total of 27 used factor baskets, the covariance matrix is complex and the results are infeasible to be executed if no constraints was applied, given a Lagrangian process was used. Initially, when long/short positions are both allowed, the numerical precision of the model always lies in an order of 10^-2 to 10^-6, which is infeasible to maintain and could subject us to large trading cost. To reduce the numerical complexity, factor baskets with below average Sharpe ratio in each state were removed from the optimization universe. However, the effect of that action on reducing numerical complexity was minimal, and increased the effort in data processing and analysis, as there was a loss of generality. Then, a long-only constraint was applied, which reduced the optimization space by half, and the results looked more feasible (refer to Scenario 1-6 in the Empirical Results section). At this point, most portfolio generated through our strategy invested only in 1 or 2 baskets out of the 27, such that the source of return and the associated risk from our strategy might not be diversified enough, so eventually it was decided to further apply a constraint to limit the maximum weight of each factor basket.

The group also came across the thought of using the stocks from the optimized portfolio as the universe, and further run an optimization on them to boost the Sharpe ratio of our portfolio. However, it was clear that such an action could actually diminish the significance of style factor in the strategy, and defies the hypothesis that economic cycles are closely tied to the financial markets, and in different cycles there are different style factors that perform better. This suggestion is eventually dropped as it also does not provide good backtesting results.

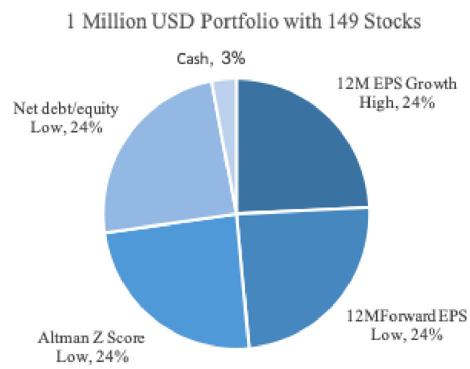
When constructing the economic activity line in the cycle model, it is noticed that there are already some alternatives in the market that track China economy, such as GS Current Activity Index. However, it is decided to stay with our own economic PCA line and not to use the index provided by other research houses as (1) the detailed methodology they used in constructing the index is unclear, which is a black box; (2)the adjustments of the index are usually not disclosed, which may affect the performance of our strategy when adjustment happens; (3) Our index already covers the four main aspect of China economic activity, export, industrial, consumption, and investment.

VIII. Live Trading

1. Live Trading and Rebalancing Details

The portfolio size was set at 1 million USD. The portfolio was first constructed in the Interactive Brokers platform during morning trading hours on 23rd April, 2019 and was rebalanced on 6th May, 2019. The planned investment horizon was set to be 10 year as the strategy is a long term macro driven investment strategy.

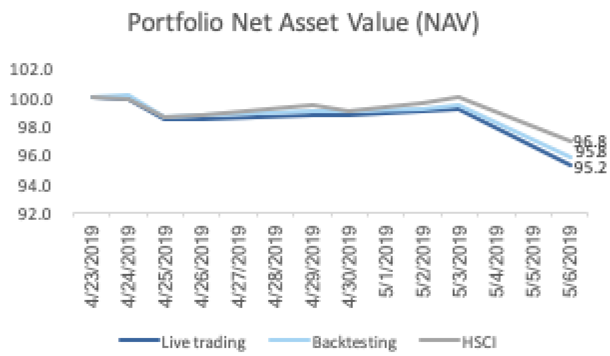
After first trade on 23rd April, the portfolio invested in 149 stocks, with a total market value of USD 966,998. The portfolio had a net asset value of USD 998,222 after paying a transaction fee of USD 1,778. The invested baskets include 12M EPS Growth High, 12M Forward EPS Low, Altman Z Score and Net Debt/Equity, each was invested with 24% of the money and the remaining 3% was left as cash.

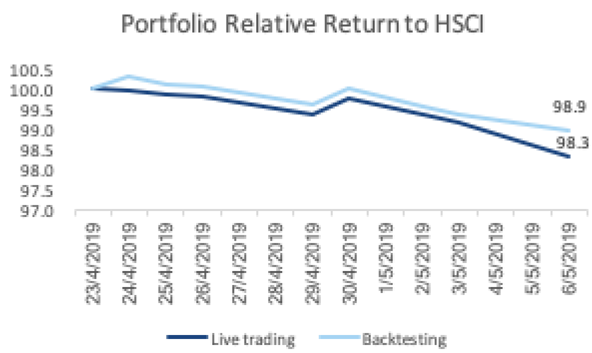


The portfolio will be rebalanced once per month, upon the monthly release of macro data. During the rebalance process, the signals will be first updated. The macro and factor data will be updated at the start of the trading day, which will take around 15 minutes. The factor baskets will then be constructed and favorable baskets indicated by the latest macro cycle will be selected, following by a optimization algorithm to produce a new portfolio with updated weights. This rebalance process will take another 15 minutes. Next, comparing the new portfolio with current portfolio, the number of shares needed to buy or sell can be calculated. Market orders are sent out to sell stocks that is no longer in the portfolio, buy stocks new in the portfolio. The update of weighting will be at least 30 minutes delay before trading commence, which diminish portfolio return against competitors adopting similar strategies.

2. Live Trading Results

The portfolio's absolute return for the trading period is -4.8%, relative return to benchmark HSCI is -1.7% while the backtesting portfolio's absolute return is -4.2% and relative return to benchmark is -1.1%.





The following are the summary of cash flows for the live trading portfolio and the backtesting(theoretical) portfolio.

Live Trading Results – Apr 23 to May 6

Changes in NAV	Total
Starting Value	1,000,000
Mark-to-Market	(44,375)
Dividends	751
Interest income (expense)	(481)
Commission and transaction fees	(3,623)

Other FX translations	2
Ending Value	952,274

Live Trading Results – Apr 23 to May 6

Changes in NAV	Total
Starting Value	1,000,000
Mark-to-Market	(42,642)
Dividends	751
Interest income (expense)	0
Commission and transaction fees	0
Other FX translations	0
Ending Value	958,109

Comparing the live trading portfolio and the backtesting portfolio, the trading cost is amounted to **USD 5,835**.

3. Trading Cost Analysis

Trading cost took up 12% of the loss as of May 6. There were four main factors that contributed to the trading cost.

First, commission and transaction fees of USD3,623 accounted for 62% of the total trading cost. The Interactive Broker platform charged 0.1827% of the total trading amount consisting of stock trading fee, stamp duty and SFC transaction levy.

Second, as the original portfolio was in USD, HKD was borrowed in advance to buy shares listed on the Hong Kong Exchange. The interest expense incurred is greater than the interest income gained from depositing USDs, and the net interest expense took up 8% of trading cost during the period.

Third, HK securities are traded in board lots, leading to the difference in intended purchase volume and actual purchase volume. The difference in trading volume caused the difference in mark-to-market value of the portfolio, making the portfolio not fully optimized and affecting the portfolio's performance in the long term.

The fourth source of trading costs was the difference in transaction price and last price, which also contributed to the difference in the mark-to-market value. There were several reasons behind the difference in price. As our order was market order, it could be traded at any price during the day, so it would potentially suffer loss when the market drops fast. And when the bid-ask spread widened, the implicit cost would become larger. In addition, when the quantity of trade is large, it could inflate the purchasing price and drag down the selling price, making the last transaction price unfavorable. Although this case would not happen in our simulated trading, it is worth being noted in real application of the trading strategy. The difference in intended and traded volume and price contributed to 30% of the total trading cost.

For the above factors, except the finance cost, several methods could be adopted to reduce the trading cost resulting from the other three factors. First, the trading amount in rebalance process could be reduced to cut and thus the commission fees. To achieve that, the rebalance frequency could be adjusted from monthly to quarterly and therefore lower the rebalancing cost by 66%, But the downside risk is that it may suffer from a greater loss when negative shocks last long and drag down our portfolio. Another possible way is to set equal weights for the selected basket, thus we could cut the rebalance cost by 20%. But the cons is that the portfolio sharpe ratio is no longer optimized. Back testing incorporating commissions could be run to see which solution generates a better result.

Second, the key issue for a 149-stock portfolio is that our number of shares for each stock is so small that the rounding error form the board lot restriction is large. Therefore, the number of stocks in the portfolio could be reduced, so that each stock takes up a larger weight. Fewer baskets could be chosen to invest in the portfolio. Currently, half of the baskets are chosen, and two-third of them could be reduced. However, this may compromise our optimization efficiency. Alternatively, the number of stocks in each factor basket could be reduced. Currently top 10% of the stocks in each basket are selected and it could be further reduced to top 5% in the future. In addition, if applicable, the trading portfolio could be scaled up from 1 million to 100 million to achieve high accuracy in trading volume. But the risk is that the larger trade value per order may impact the market and make the price unfavorable. And it is hard to buy or sell in huge amount, leading to the liquidity risk.

Third, for the trading cost arising from the price differences, three solutions could be used to tackle the three problems respectively. As previously mentioned, the uncontrollability of the market order would expose our trade to losses when price moves fast. Therefore, algorithm trade such as VWAP, TWAP could be adopted to obtain a more controllable price and thus a more controllable trading cost. For the cost related to bid-ask spread, the investable universe could be restricted to higher liquidity stocks and those low liquidity ones could be screened out. As for the trade impact on the market, this could mitigated by splitting the orders into more small pieces so that each trade has limited impact on the market. After the above methods, the trading frictions might be reduced.

IX. Conclusion

The hypothesis has shown to be correct, indeed economic cycles are closely tied to the financial markets, and in different cycles there are different style factors that perform better. The final backtested model is able to generate returns far superior to the return of the HSCI. Although live trading posted negative returns, the model is meant as a long term investment strategy where it can capture the changes along the economic cycle which takes time. Live trading however did allow for an understanding of how trading costs could eat away trading profits.

Provided more time, more leading macro indicators such as PMI and unemployment rate could have been used to improve the tracking of the macro environment. Better sources of fundamental data would have significantly improved the pace and efficiency of the model as lots of time was used to standardize and process fundamental data across the stocks ahead of the factor basket creation.

The possible shortcoming of the model lies in how it does not have the capability to capture and react to drastic changes in the economic environment that could possibly lead to a structural break (a permanent change in the economic and financial environment), as the prediction of the economic cycle is based on mean reversion. Events that bypass the economy and directly affect the financial system like multiple big selloffs, poor liquidity, financial system failure and likewise, could lead to the failure of our strategy.

X. Appendix

```
library(data.table)
library(zoo)
library(ggplot2)
setwd("C:\\Users\\user\\Downloads\\FINA4803_FYP\\Factor Data")
options(stringAsFactors = FALSE)

# Free Parameters
cutoff = 0.05 # top/bot X% of stocks included in the baskets

# Global Variables
factors = c("1MPriceReversal","2YBetaCIP","2YBetaHSCI","6MSharpeRatio",
                       "11MPriceMomentum","12MBuybackYield","12MDivGrowth",
                       "12MEPSGrowth","12MFEPS","12MFSalesGrowth","12MPB",
                       "AltmanZScore","NDE","ShortInterestRatio")
# not_in_used = c("6MSharpeRatioD", "1DReturnHSCI", "1DReturn")
returns = c("1MReturn")

bmk_returns = c("1MReturnHSCI")

init_variables = c(factors, returns, bmk_returns)

initalize_model <- function() {
  for (i in 1:length(init_variables)) {
        for (j in 1:length(init_variables[i])) {
        assign(init_variables[i][j],
               fread(paste0("FINA4803_",init_variables[i][j],".csv")),
               envir = .GlobalEnv)
        }
  }
}

# Initialization
initalize_model()
cat("Model Initialization Completed.\n")

# Extract Date and Securities
date = `1MReturn`$Date
rdate = date[length(date):1]
secs = colnames(`1MReturn`)[-1]

for (i in 1:length(init_variables)) {
  for (j in 1:length(init_variables[i])) {
        assign(init_variables[i][j],get(init_variables[i][j])[,1:=NULL])
  }
}
```

```r
# NA Cleaning in Returns
for (i in 1:length(returns)) {
  df = get(returns[i])
  df[is.na(df)] = 0
  assign(returns[i],df)
}

# Ranking of Factors
prank <- function(x) {return(rank(x, na.last = "keep", ties.method = "average"))}# , cols
= NULL, na.last = NA))}

for (i in 1:length(factors)){
  cat("Applying to",factors[i],"...\n")
  dt = get(factors[i])
  dt = data.table(t(apply(dt, 1, prank)))
  assign(factors[i], setnames(dt, secs))
}

cat("Ranking of Factors Completed.\n")

# Construction of H/L Baskets
basket_forming <- function(factor_name) {
  df = get(factor_name)

  for (i in 1:length(date)){
        x = df[i]
        rcount = (length(secs) - sum(is.na(x)))
        rparti = round(rcount * cutoff)
        lcutoff = rcount - rparti
        hcutoff = rparti
        lrow = x[,x > lcutoff & !is.na(x)]
        hrow = x[,x < hcutoff & !is.na(x)]
        assign(paste0("H",factor_name), rbind(get(paste0("H",factor_name)), hrow), envir
= .GlobalEnv)
        assign(paste0("L",factor_name), rbind(get(paste0("L",factor_name)), lrow), envir
= .GlobalEnv)
  }
}

for (i in 1:length(factors)){
  assign(paste0("H",factors[i]),data.frame(matrix(nrow = 0, ncol = length(secs))))
  assign(paste0("L",factors[i]),data.frame(matrix(nrow = 0, ncol = length(secs))))
  cat("Constructing H/L Baskets for",factors[i],"...\n")
  basket_forming(factors[i])
}
cat("Construction of Factor Baskets Completed.\n")
```

```r
# Applying Weights
cat("Application of Weights Completed.\n")

# Returns Calculation
options_HL = c("H","L")

cat("Calculating Returns of Factor Baskets.\n")
for (i in 1:length(factors)){
  for (j in 1:2){
        x = data.table(get(paste0(options_HL[j],factors[i])))
        vec_norm = rowSums(x == TRUE)
        basket_return_m = as.matrix(`1MReturn`) %*% t(as.matrix(x))
        basket_return = diag(basket_return_m)/vec_norm
        assign(paste0("R",options_HL[j],factors[i]),basket_return)
  }
}

cat("Calculation of Returns Completed.\n")

# Returns Aggregation
RMaster = cbind(RH11MPriceMomentum,RH12MBuybackYield,RH12MDivGrowth,
               RH12MEPSGrowth,RH12MFEPS,RH12MFSalesGrowth,RH12MPB,
               RH1MPriceReversal,RH2YBetaCIP,RH2YBetaHSCI,RH6MSharpeRatio,
               RHAltmanZScore,RHNDE,RHShortInterestRatio,RL11MPriceMomentum,
               RL12MBuybackYield,RL12MDivGrowth,RL12MEPSGrowth,RL12MFEPS,
               RL12MFSalesGrowth,RL12MPB,RL1MPriceReversal,RL2YBetaCIP,
               RL2YBetaHSCI,RL6MSharpeRatio,RLAltmanZScore,RLNDE,
               RLShortInterestRatio)

# Creating Periodic Return Master
PRMaster = RMaster[nrow(RMaster):1, ]
rownames(PRMaster) = date[length(date):1]
PRMaster = data.frame(PRMaster)

# Creating Cumulative Return Master
CRMaster = RMaster[nrow(RMaster):1, ]
CRMaster[is.nan(CRMaster)] = 0
CRMaster = CRMaster + 1
CRMaster = apply(CRMaster, 2, cumprod)
CRMaster = CRMaster - 1
rownames(CRMaster) = date[length(date):1]

# Zooing
ZRMaster = as.zoo(CRMaster)
time(ZRMaster) = as.Date(date[length(date):1], format = "%d/%m/%Y")

# Exporting Graphs
# for (i in 1:ncol(ZRMaster)){
```

```
#   g = autoplot.zoo(ZRMaster[,i])
#   g = g + xlab("Year") + ylab(colnames(ZRMaster)[i]) + geom_line(size = 1)
#   ggsave(paste0(colnames(ZRMaster)[i],".png"), plot = g, device = png(),
#       height = 8, width = 10, units = "cm")
# }

# Exporting PRMaster
fwrite(PRMaster,"FINA4803_PRMaster.csv", row.names = TRUE)

# Exporting HSCI Return
R1MReturnHSCI = `1MReturnHSCI`[nrow(`1MReturnHSCI`):1]
R1MReturnHSCI = R1MReturnHSCI + 1
R1MReturnHSCI = cumprod(R1MReturnHSCI)
R1MReturnHSCI = R1MReturnHSCI - 1

z.R1MReturnHSCI = as.zoo(R1MReturnHSCI)
time(z.R1MReturnHSCI) = as.Date(rdate, format = "%d/%m/%Y")
# g = autoplot.zoo(z.R1MReturnHSCI)
# g = g + xlab("Year") + ylab("HSCI Index") + geom_line(size = 1)
# ggsave("R1MHSCIReturn.png", plot = g, device = png(),height = 8, width = 10, units =
"cm")

# Max Sharpe Portfolio Construction
require(data.table)
require(quadprog)
require(Matrix)
require(zoo)
require(ggplot2)
setwd("C:\\Users\\user\\Downloads\\FINA4803_FYP\\Scenario Data")
options(stringAsFactors = FALSE)

# free parameters
subsample_range <- NULL # if subsample is to be used, changes @date variable
prec            <- 3    # number of decimals for stock weight
n_top           <- 14   # maximum number of baskets allowed (using = 1)
max_weight      <- 0.10 # maximum weight of each basket (using = 2)
using           <- 2    # 1 = stockwise, 2 = basket

# global parameters
scenario <- 8          # our chosen macro state

# helper functions
map <- function(w, slogi){
  r = data.frame(matrix(nrow = 1, ncol = 0))
  map_pos = 1

  for (i in 1:length(slogi)){
      if (slogi[i]) {
```

```
      r[,ncol(r) + 1] = w[map_pos]
      map_pos = map_pos+1
      } else {
      r[,ncol(r) + 1] = 0
      }
  }

  return(r)
}
unmap <- function(w, cycle_override = -1){
  if (cycle_override == -1) { cycle_override = macro[length(macro)] }
  get_list = substring(colnames(PRMaster), first = 2)

  invest_weight = data.frame(matrix(nrow = 1, ncol = length(secs)))
  invest_weight[is.na(invest_weight)] = 0

  for (i in 1:length(get_list)) {
      r = get(get_list[i])[1,]
      p = r/sum(r == TRUE)*w[cycle_override,i]
      # cat(i,": ",sum(is.nan(unlist(p))),"\n")
      if (sum(is.nan(unlist(p)))) { p = 0 }
      invest_weight = invest_weight + p
  }

  colnames(invest_weight) = secs
  return(invest_weight)
}
u.round <- function(w){
  w = round(w, prec)
  for (i in 1:4){
      w[i, ] = w[i, ]/sum(w[i, ])
  }
  return(w)
}

# ***************************************************
#* TODO: change definition of vnorm to regularize    *
# **************************************************
compute_return <- function(optimized_weight, ref_return, macro){
  ref_return.r = ref_return[nrow(ref_return):1]
  optimized_return = data.frame(matrix(nrow = 0, ncol = 1))

  for (i in 1:nrow(ref_return.r)){
      tmp = data.frame(as.matrix(ref_return.r[i,]) %*%
t(as.matrix(optimized_weight[macro[i],])))
      if (scenario == 5 || scenario == 7) {
      if (i == 1 || i == 2) {
      optimized_return = rbind(optimized_return, 0)
```

```
        colnames(optimized_return) = "Portfolio"
        }
        else {
        vnorm = 1 - sum(optimized_weight[macro[i], ][ref_return.r[i, ] == 0 &
optimized_weight[macro[i], ] != 0])
        if (!vnorm) { vnorm = 1 }
        optimized_return = rbind(optimized_return/vnorm, setNames(tmp,"Portfolio"))
        }
        }
        else {
        if (i == 1) {
        optimized_return = rbind(optimized_return, 0)
        colnames(optimized_return) = "Portfolio"
        }
        else {
        vnorm = 1 - sum(optimized_weight[macro[i], ][ref_return.r[i, ] == 0 &
optimized_weight[macro[i], ] != 0])
        if (!vnorm) { vnorm = 1 }
        optimized_return = rbind(optimized_return/vnorm, setNames(tmp,"Portfolio"))
        }
        }
  }
  return(optimized_return)
}

# load data
MacroCycle <- fread(paste0("FINA4803_S",scenario,"MacroCycle.csv"))
macro <- MacroCycle$state
macro <- macro[232-113:1]
PRMaster <- fread("..\\Factor Data\\FINA4803_PRMaster.csv")
date <- PRMaster[[1]]
PRMaster <- PRMaster[,-1]
PRMaster <- PRMaster[,-2] # Drop H12MBuyback due to quality

if (using == 1) {
  # select top factor return baskets
  n_top_baskets = data.frame(matrix(nrow = 0, ncol = n_top), stringsAsFactors = FALSE)

  for (i in 1:4){
        n_top_baskets.s = apply(PRMaster[macro == i,], 2, mean, na.rm = TRUE)
        n_top_baskets.top = n_top_baskets.s[rank(-n_top_baskets.s) <= n_top]
        n_top_baskets[nrow(n_top_baskets)+1,] =
        t(data.frame(names(n_top_baskets.top), stringsAsFactors = FALSE))
  }

  row.names(n_top_baskets) = c(1,2,3,4)
  cat("Retrieved top", n_top,"return baskets...\n")
  n_top_baskets
```

```
# construct portfolio from relevant stocks
optimized_weight = data.frame(matrix(nrow = 0, ncol = length(secs)))
optimized_pa = data.frame(matrix(nrow = 0, ncol = 1))

for (i in 1:4){
      stocks_picked = data.frame(matrix(nrow = length(date), ncol = length(secs)))
      stocks_picked[is.na(stocks_picked)] = 0

      for (j in 1:n_top){
      stocks_picked = stocks_picked + get(substring(n_top_baskets[i,j],2))
      }

      # Stocking picking based on whether a stock appear in the most recent
      # month's factor basket !! Will have forelooking bias
      slogi <- c(stocks_picked[1, ] > 0) # stock pick logical vector
      mData <- `1MReturn`[, slogi, with = FALSE]
      mData <- as.matrix(mData)
      nA    <- sum(slogi)
      cat("Total stocks relevant:", nA,"for state", i, "...\n")
      rf      <- 0    # riskfree rate (2.5% pa)
      mu    <- apply(mData, 2, mean, na.rm = TRUE) # means
      mu2   <- mu - rf                      # excess means

      # qp
      Dmat <- cov(mData, use = "complete.obs")
      # diagonal = diag(Dmat)
      # Dmat <- matrix(0, nrow = nA, ncol = nA)
      # diag(Dmat) = diagonal
      dvec <- array(0, dim = c(nA,1))

      # No Short Selling
      Amat <- matrix (1, nrow=nA)
      Amat <- cbind(1, diag(nA))
      bvec <- 1
      bvec <- c(bvec, rep(0, nA))

      meq  <- 1
      solQP <- solve.QP(nearPD(Dmat)$mat, dvec, Amat, bvec, meq = 1)

      # rescale variables to obtain weights
      w <- as.matrix(solQP$solution/sum(solQP$solution))
      optimized_weight <- rbind(optimized_weight, map(w, slogi))
      # colnames(optimized_weight) <- secs
      # View(t(optimized_weight))

      SR <- t(w) %*% mu2 / sqrt(t(w) %*% Dmat %*% w)
      optimized_pa <- rbind(optimized_pa, SR)
```

```
  }

  # Overfitting treatment if applicable
  optimized_weight = u.round(optimized_weight, prec)

  # Save weight
  colnames(optimized_weight) <- secs
  fwrite(optimized_weight, paste0("FINA4803_S",scenario,"Weight.csv"), row.names =TRUE)

  # Compute Benched Return
  cat("Computing Benched Return...\n")
  optimized_returns = data.frame(matrix(nrow = length(date), ncol = 0))
  optimized_return = compute_return(optimized_weight, `1MReturn`, macro)
  optimized_returns = cbind(optimized_returns, optimized_return)

  # What-If Scenarios of Nonsensical Investment
  for (j in 1:4){
        optimized_return = compute_return(optimized_weight, `1MReturn`, c(rep(j,
length(date))))
        optimized_returns = cbind(optimized_returns, optimized_return)
  }

}
if (using == 2) {

  # optimization by basket
  optimized_weight = data.frame(matrix(nrow = 0, ncol = 28))
  optimized_pa = data.frame(matrix(nrow = 0, ncol = 1))

  for (i in 1:4) {
        nA    <- ncol(PRMaster)
        mData <- PRMaster[macro == i,]
        mData <- as.matrix(mData)
        rf      <- 0    # riskfree rate (2.5% pa)
        mu     <- apply(mData, 2, mean, na.rm = TRUE)        # means
        mu2    <- mu - rf                                    # excess means

        # qp
        Dmat <- cov(mData, use = "complete.obs")
        dvec <- array(0, dim = c(nA,1))

        # No Short Selling
        Amat <- matrix (1, nrow = nA)
        Amat <- cbind(1, diag(nA), -diag(nA))
        bvec <- 1
        bvec <- c(bvec, rep(0, nA), rep(-max_weight, nA))

        meq  <- 1
```

```
        solQP <- solve.QP(nearPD(Dmat)$mat, dvec, Amat, bvec, meq)

        # rescale variables to obtain weights
        w <- as.matrix(solQP$solution/sum(solQP$solution))
        optimized_weight <- rbind(optimized_weight, t(w))

        # compute sharpe ratio
        SR <- t(w) %*% mu2 / sqrt(t(w) %*% cov(mData, use = "complete.obs") %*% w)
        optimized_pa <- rbind(optimized_pa, SR)
  }

  colnames(optimized_weight) <- colnames(PRMaster)

  # Rounding off weight
  optimized_weight = u.round(optimized_weight)
  fwrite(optimized_weight, paste0("FINA4803_S",scenario,"Weight",max_weight,".csv"),
row.names =TRUE)

  # Unmap basket weight to stock weight
  optimized_weight.s = data.frame(matrix(nrow = 0, ncol = length(secs)))
  for (i in 1:4) {
        optimized_weight.s = rbind(optimized_weight.s, unmap(optimized_weight, i))
  }
  fwrite(optimized_weight.s,
paste0("FINA4803_S",scenario,"WeightStock",max_weight,".csv"), row.names = TRUE)

  # Compute Benched Return
  optimized_returns = data.frame(matrix(nrow = length(date), ncol = 0))
  PRMaster[is.na(PRMaster)] = 0

  optimized_return = data.frame(matrix(nrow = 0, ncol = 1))
  for (i in 1:nrow(PRMaster)){
        tmp = data.frame(as.matrix(PRMaster[i,]) %*%
t(as.matrix(optimized_weight[macro[i],])))
        if (scenario == 5 || scenario == 7 || scenario == 8) {
        if (i == 1 || i == 2) {
        optimized_return = rbind(optimized_return, 0)
        colnames(optimized_return) = "Portfolio"
        }
        else {
        optimized_return = rbind(optimized_return, setNames(tmp,"Portfolio"))
        }
        }
        else {
        if (i == 1) {
        optimized_return = rbind(optimized_return, 0)
        colnames(optimized_return) = "Portfolio"
        }
```

```
        else {
        optimized_return = rbind(optimized_return, setNames(tmp,"Portfolio"))
        }
        }
  }

  optimized_returns = cbind(optimized_returns, optimized_return)

  # What-If Scenarios of Nonsensical Investment
  for (j in 1:4){
        optimized_return = data.frame(matrix(nrow = 0, ncol = 1))

        for (i in 1:nrow(PRMaster)){
        tmp = data.frame(as.matrix(PRMaster[i,]) %*% t(as.matrix(optimized_weight[j,])))
        if (scenario == 5 || scenario == 7|| scenario == 8) {
        if (i == 1 || i == 2) {
        optimized_return = rbind(optimized_return, 0)
        colnames(optimized_return) = "Portfolio"
        }
        else {
        optimized_return = rbind(optimized_return, setNames(tmp,"Portfolio"))
        }
        }
        else {
        if (i == 1) {
        optimized_return = rbind(optimized_return, 0)
        colnames(optimized_return) = "Portfolio"
        }
        else {
        optimized_return = rbind(optimized_return, setNames(tmp,"Portfolio"))
        }
        }
        }
        optimized_returns = cbind(optimized_returns, optimized_return)
  }
}

# Compute Periodic Returns and Perf Ana
optimized_returns = cbind(optimized_returns, `1MReturnHSCI`[length(date):1])
colnames(optimized_returns) = c("Portfolio","W1","W2","W3","W4","HSCI")

optimized_pa = rbind(optimized_pa,
mean(unlist(optimized_returns[,1]))/sd(unlist(optimized_returns[,1])))
row.names(optimized_pa) = c("1","2","3","4","P")
optimized_pa = cbind(c(sd(optimized_returns[macro == 1,1]),sd(optimized_returns[macro ==
2,1]),sd(optimized_returns[macro == 3,1]),sd(optimized_returns[macro ==
4,1]),sd(unlist(optimized_returns[,1]))),optimized_pa)
```

```
optimized_pa = cbind(c(mean(optimized_returns[macro == 1,1]),mean(optimized_returns[macro
== 2,1]),mean(optimized_returns[macro == 3,1]),mean(optimized_returns[macro ==
4,1]),mean(unlist(optimized_returns[,1]))),optimized_pa)
colnames(optimized_pa) = c("Mean", "Volatility", "SharpeRatio")
fwrite(optimized_pa, paste0("FINA4803_S",scenario,"PA.csv"), row.names = TRUE)

# Cumulative Return
optimized_returns = optimized_returns + 1
optimized_returns = apply(optimized_returns, 2, cumprod)
optimized_returns = optimized_returns - 1
z.optimized_returns = as.zoo(optimized_returns)
time(z.optimized_returns) = as.Date(date, format = "%d/%m/%Y")

g = autoplot.zoo(z.optimized_returns, facets = NULL)
g = g + xlab("Year") + ylab(paste0("Scenario ",scenario)) + geom_line(size = 1)
ggsave(paste0("FINA4803_S",scenario,"Return.png"), plot = g, # device = png(),
       height = 8, width = 14.64, units = "cm")
```