Python Backtesting System

14 Apr 2022

Sector Rotation Strategy – Hong Kong market

Interview Candidate:

Andy Chan
andyinter1@gmail.com
+852 9603 3105

Content

System Environment

External Libraries	Numpy, Pandas, PyQt5, SciPy, Matplotlib, itertools
Python Version	3.7

Please refer to the python system in below link for more details

https://github.com/ccfandy1/Backtesting---Sector-Rotation

Introduction of the back-test python system

The backtesting system aims to test the feasibility of sector rotation investment strategy. The strategy will use growth of basic earnings per share as an indicator to filter out outperforming sectors in specific market. Throughout of the testing period, the system will generate semiannually investment signals and weights based on sector growth and market cap respectively. Certain numbers of outperforming sectors will be invested, and the virtual portfolio performance calculated afterwards will eventually be compared with the benchmark index performance for strategy validation. The system is written with object-oriented programming structure.

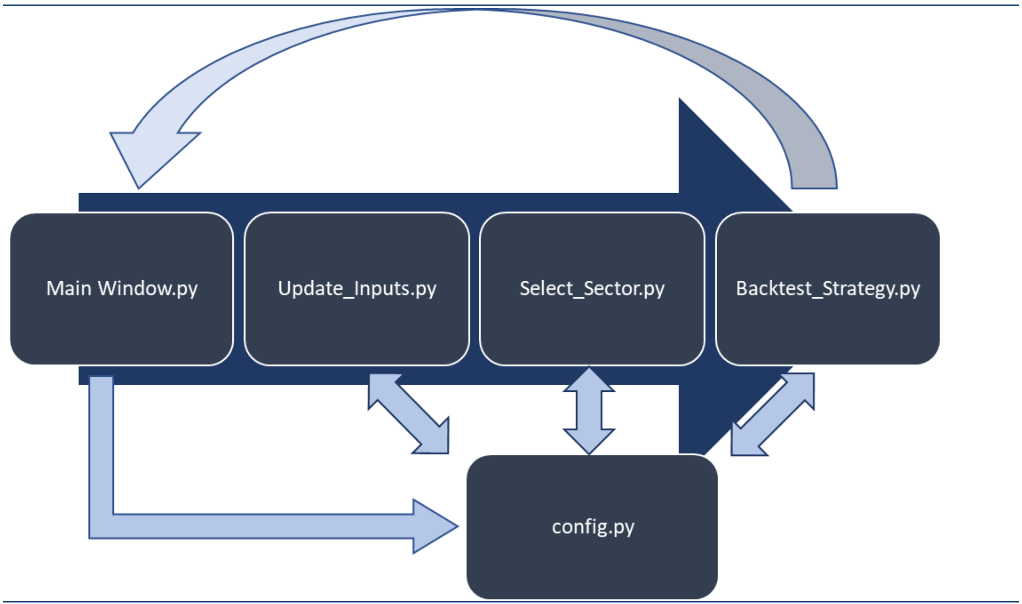
Strategy components:

Indicator	Growth of basic earnings per share
Signal	Sectors with high sector growth. For simplification, investment signal directly triggers investment actions without other considerations
Rule	Invest in outperforming sectors. Rebalance semi-annually Trade entry and exit follow the same signal-based rule
Initial data	• Stocks related: - earnings growth - stock price - market cap • Benchmark related: - market index value

Python program structure summary

The backtesting system contains 5 modules: 1) Main Window, 2) Update_Inputs, 3) Select_Sector, 4) Backtest_Strategy, and 5) Config (fig. 1). The Main Window module will transfer user inputs to config module. Then the remaining modules will conduct strategy backtest along with config module and feed the final outcomes back to Main Window module for display.

Figure 1 Python structure



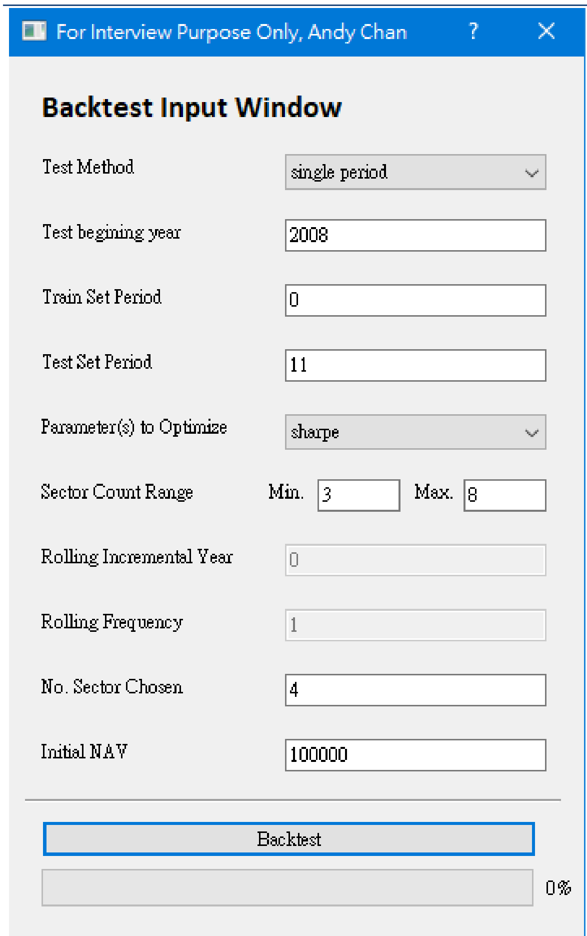
Brief introduction of 5 modules:

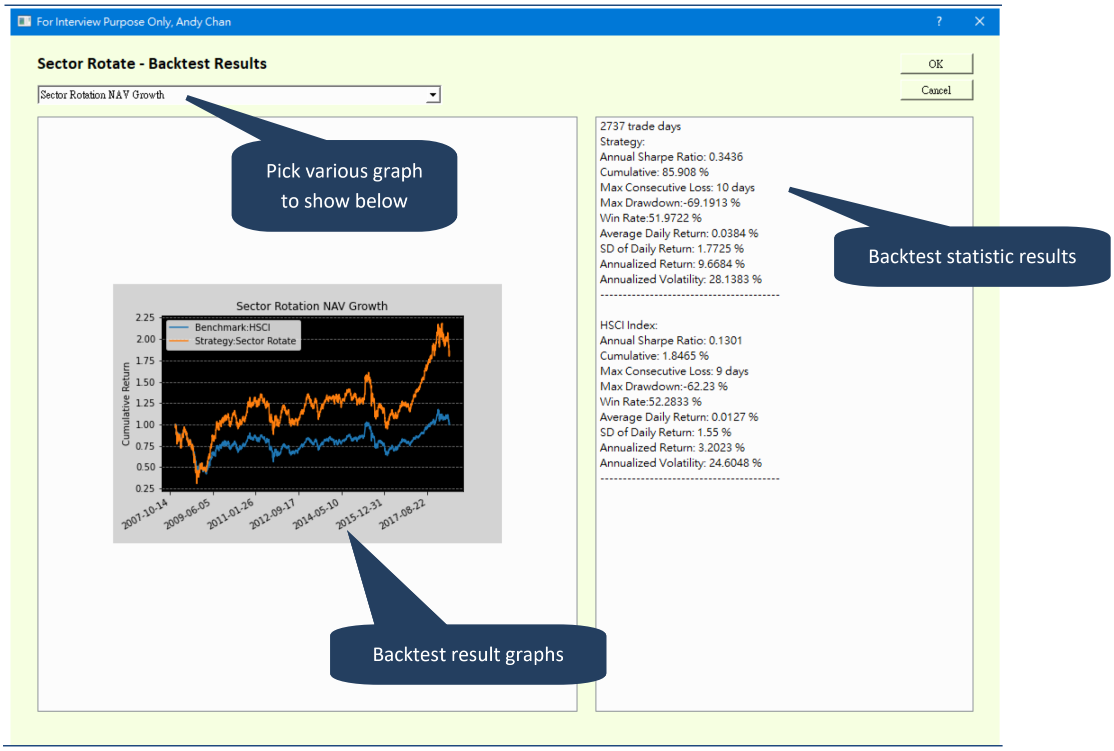
Main Window	To show input and result window forms
Update_Inputs	To cleanse Bloomberg data and update input CSV accordingly
Select_Sector	1. Calculate sector growth and rank them 2. Generate investment signals based on ranking
Backtest_Strategy	1. Calculate weighted price growth of each stock in each chosen sector, then calculate portfolio return performance accordingly 2. Perform backtesting, show backtest stats and relevant graphs afterwards
Config	Contains all global variable, most of independent variable, and utility methods for other modules to use during execution

Update_Inputs module can be executed individually to update inputs, while the updated inputs will be forwarded to Select_Sector module for further processing. Data from both Update_Inputs and Select_Sector module will then be forwarded to Backtest_Strategy module for final processing.

Most of input variables are in the config module, where user can interact with through window forms generated with Main Window module to perform various forms of backtesting. (fig. 2)

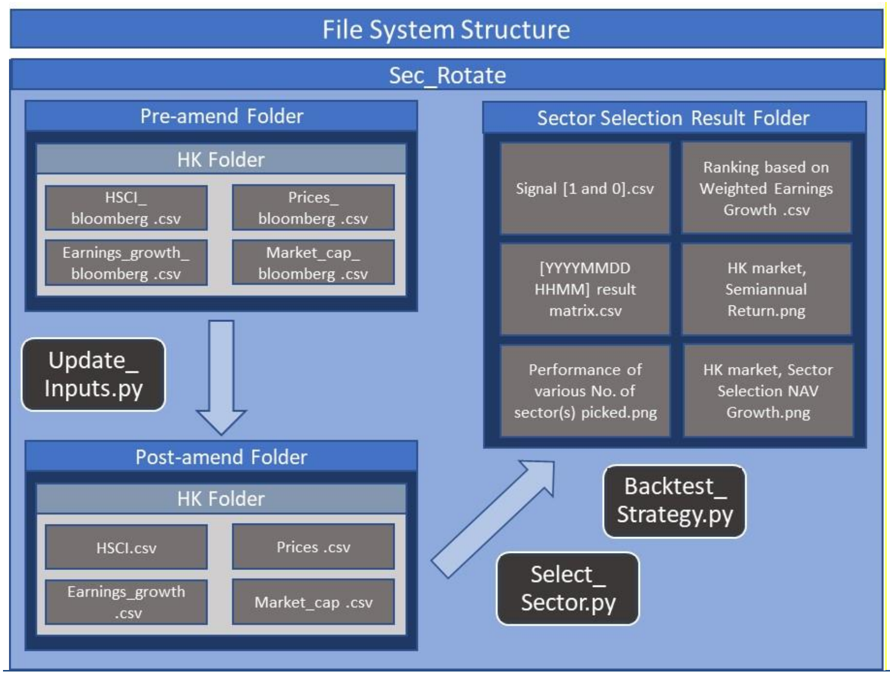
Figure 2 User Interface – Input Window & Result Window





File System Structure

Figure 3 File system structure overview



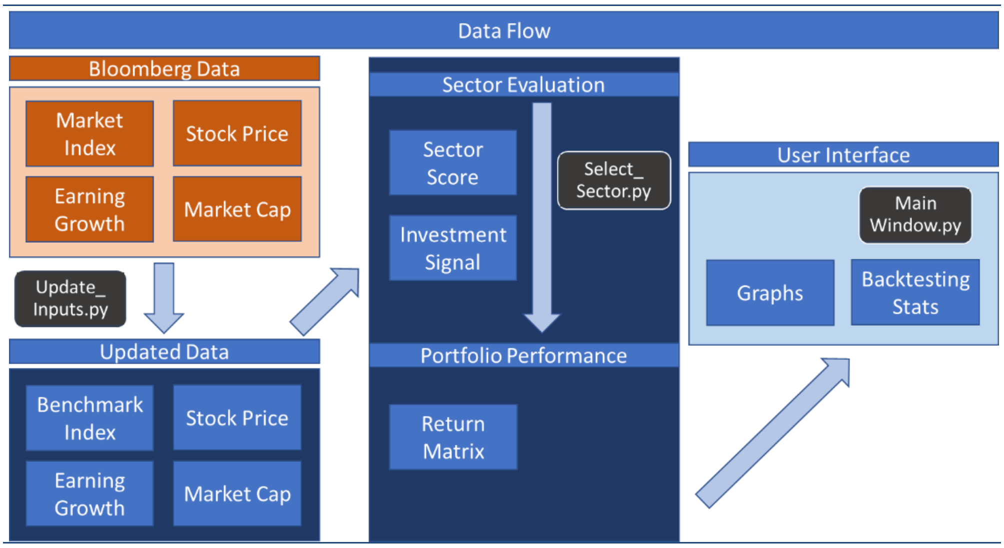
There are 3 folders containing all system input and output, namely 1) “pre-amend”, 2) “post-amend”, and 3) “sector selection”, which are placed under “sec_rotate” parent folder. Raw and cleansed input CSV are stored in “pre-amend” and “post-amend” folder respectively, while the signal data and final testing outcomes are stored in “sector selection” folder. (fig. 3)

For Interview Only, Not For Distribution

Explanation of System Execution Flow

Execution flow

Figure 4 Execution flow overview



Bloomberg Data

Four sets of data will be extracted from Bloomberg to csv, including: 1) market index, 2) earnings growth, 3) stock price, and 4) market cap. The first set (fig. 5) is for calculating benchmark performance while the latter three (fig. 6, 7, 8), which contain data of chosen stocks, are for calculating performance of sector rotation strategy.

Figure 5 Example of bloomberg raw data (HSCI_bloomberg.csv)

	A	B	C	D
1	Start Date	1/1/2018		Daily
2	End Date	11/7/2018		
3				
4	Securities	HSCI INDEX		
5				
6	PX_LAST			
7	2/1/2018	4224.21		
8	3/1/2018	4239.86		
9	4/1/2018	4264.77		
10	5/1/2018	4278.16		
11	8/1/2018	4303.51		

Figure 6 Example of bloomberg raw data (earnings_growth_bloomberg.csv)

	A	B	C	D	E	F	G	H	I	J
1	Start Date	1/1/2017								
2	End Date	28/6/2018								
3										
4	Securities	1728 HK Equity		1999 HK Equity		1929 HK Equity		2331 HK Equity		1114 HK E
5										
6	EPS_GROWTH									
7	30/6/2017	102.6087	31/3/2017	23.6407	31/3/2017	32.4783	30/6/2017	348.0226	30/6/2017	2
8	31/12/2017	180.7108	30/9/2017	-9.5861	30/9/2017	45.9016	31/12/2017	109.2066	31/12/2017	
9			31/3/2018	-14.2265	31/3/2018	26.3049				

Figure 7 Example of bloomberg raw data (prices_bloomberg.csv)

	A	B	C	D	E	F	G	H	I	J
1	Start Date	1/1/2018								
2	End Date	11/7/2018								
3										
4	Securities	1728 HK Equity		1999 HK Equity		1929 HK Equity		2331 HK Equity		1114 HK
5										
6	PX_LAST									
7	2/1/2018	8.25	2/1/2018	7.35	2/1/2018	8.21	2/1/2018	6.46	2/1/2018	20
8	3/1/2018	8.32	3/1/2018	7.4	3/1/2018	8.65	3/1/2018	6.6	3/1/2018	21.5
9	4/1/2018	8.29	4/1/2018	7.41	4/1/2018	8.68	4/1/2018	6.68	4/1/2018	21.4

Figure 8 Example of bloomberg raw data (market cap_bloomberg.csv)

	A	B	C	D	E	F	G	H	I	J
1	Start Date	1/1/2018		Quarterly						
2	End Date	28/6/2018								
3										
4	Securities	1728 HK Equity		1999 HK Equity		1929 HK Equity		2331 HK Equity		1114 HK Ec
5										
6	HISTORICAL_MARKET_CAP									
7	30/3/2018	14201.77	30/3/2018	23891.58	30/3/2018	89400	30/3/2018	17522.92	30/3/2018	82641.51
8										

Updated Data - Update_Inputs.py

Inputs	• HSCI_bloomberg.csv (fig. 5) – *raw market index data* • Earnings_growth_bloomberg.csv (fig. 6) – *raw earnings growth data of stocks* • market cap_bloomberg.csv (fig. 7) – *raw market cap data of stocks* • prices_bloomberg.csv (fig. 8) – *raw price data of stocks*
Outcomes	• HSCI.csv – *contains date index, and daily closed price data of specific index from 2000 to 2018.* • Earnings growth.csv (fig. 9) – *contains date index, stock code and sector header, and quarterly weighted earnings growth data* • *(earnings growth / total earnings growth of stock's sector) of chosen stocks from around 1985 to 2018.* • Prices.csv (fig. 10) – *contains date index, stock code header, and daily closed trading price data of chosen stocks from 2000 to 2018.* • Market cap.csv – *contains date index, stock code and sector header, and quarterly market cap data of chosen stocks from around 1985 to 2018.*

This module will check if there is any new data point in the raw Bloomberg data. New data will be cleansed, standardized, and updated to its corresponding input csv file accordingly (fig. 9 & 10) for further processing of other modules.

Mainly, the historical data of market index, stock prices, market cap, and earnings growth will be used to calculate benchmark KPI, price return of stock candidates, investment weights of stock candidates, and sector growth respectively.

Figure 9 Example of cleansed data (earnings growth.csv)

	A	B	C	D	E	F	G	H	I	J	K
1		1728 HK E	1999 HK E	1929 HK E	2331 HK E	1114 HK E	3813 HK E	136 HK Equ	1211 HK E	3818 HK E	1828 I
2		Consumer I	Consumer I	Consumer I	Consumer I	Consumer I	Consumer I	Consumer I	Consumer I	Consumer I	Consu
3	1985Q1	0	0	0	0	0	0	0	0	0	
4	1985Q2	0	0	0	0	0	0	0	0	0	
75	2003Q1	0	0	0	0	3.6100676	0	0.0160868	0	0	
76	2003Q2	0	0	0	0	6.4450821	0	0.012363	5.6687918	0	

Figure 10 Example of cleansed data (prices.csv)

	A	B	C	D	E	F	G	H	
1		1728 HK E	1999 HK E	1929 HK E	2331 HK E	1114 HK E	3813 HK E	136 HK Eq	121
2	3/1/2000	0	0	0	0	0	0	0	
3	4/1/2000	0	0	0	0	1.35	0	16.206	
4	5/1/2000	0	0	0	0	1.345	0	15.144	
5	6/1/2000	0	0	0	0	1.32	0	20.192	

Sector Evaluation – Select_Sector.py

Inputs	• Earnings growth.csv (fig. 9) – *cleansed stock earnings growth data* • Market cap.csv – *cleansed stock market cap data*
Outcomes	• Ranking based on weighted earnings growth.csv (fig. 11) – *contains date index, sector header, and ranking data of user specified period.* • Signal [1 and 0].csv (fig. 12) – *contains date index, sector header, and investment signal data of user specified period.*

This module will generate investment signals for backtesting simulation.

Since not every HK listed stock will disclose its quarterly data, time unit of data record is semi-annual base. Semiannually, earnings growth of stocks under each sector will be accumulated and become sector growth. Sectors will then be ranked and scored based on their growth, while a certain number of outperforming sectors, which can be specified by user manually, will be picked and generate investment signals accordingly.

Cleansed earnings growth data is the main input to be used in this module. Sector score and investment signal results will be exported into csv called “Ranking based on Weighted Earnings Growth.csv” and “Signal [1 and 0].csv” respectively (fig. 11 & 12).

It should be noted that “1” and “0” shown in investment signal csv (fig. 12) respectively means “good to invest” and “skip the sector”.

Figure 11 Sector Ranking (Ranking based on weighted earnings growth.csv)

	A	B	C	D	E	F	G	H	I	J	K	L
1		Consumer	Consumer	Energy	Financials	Health Car	Industrials	Informatio	Materials	Real Estate	Telecommu	Utilities
2	1998Q4	8	3	0	2	10	6	1	7	9	5	4
3	1999Q2	5	0	6	9	10	7	3	4	2	1	8
4	1999Q4	7	0	8	9	10	6	2	5	1	3	4
5	2000Q2	8	7	9	5	10	4	0	2	3	1	6
6	2000Q4	5	2	6	4	8	10	1	3	9	0	7

Figure 12 Investment Signals (Signal [1 and 0].csv)

	A	B	C	D	E	F	G	H	I	J	K	L
1		Consumer	Consumer	Energy	Financials	Health Car	Industrials	Informatior	Materials	Real Estate	Telecommu	Utilities
2	1998Q4	0	1	1	1	0	0	1	0	0	0	1
3	1999Q2	0	1	0	0	0	0	1	1	1	1	0
4	1999Q4	0	1	0	0	0	0	1	0	1	1	1
5	2000Q2	0	0	0	0	0	1	1	1	1	1	0
6	2000Q4	0	1	0	1	0	0	1	1	0	1	0

Portfolio Performance & User Interface – Backtest_Strategy.py & Main Window.py

Inputs	• Signal [1 and 0].csv – *contains semiannually investment signal presented as "1" and "0"* • Market cap.csv –*cleansed quarterly stock market cap data* • Stock prices.csv – *cleansed daily stock price data* • Market Index.csv – *cleansed benchmark market index data*
Outcomes	• [YYYYMMDD HHMM] result matrix.csv – *contains weighted price return of each invested stock and returns of virtual portfolio* • Backtesting stats – *show backtest KPI, such as drawdown* • Performance graphs – *show benchmark comparison and semi-annual return of investment strategy*

This module will be responsible for generating virtual portfolio performance and conducting backtest.

Part 1 – Virtual portfolio performance

Semiannually, based on sector investment signals, market cap of stocks under chosen sectors will be picked to calculate their investment weights, which will be multiplied by each daily stock price return in next period to get weighted investment return. For example, daily price return on any trading date in 1H22 will multiply its investment weight in 2H21, and so on.

By accumulating all weighted investment returns on each trading date, daily return of virtual portfolio will be calculated, which will be used to calculate other KPI such as cumulative return and portfolio NAV over time.

Investment signals, stock price data and market cap data are the main inputs for portfolio return calculation. Portfolio return data generated will be exported into csv called "[YYYYMMDD HHMM] result matrix.csv". (fig. 13)

Figure 13 Return Matrix

	A	B	C	D	E	F	G	H	I	J	K
1		1728 HK E	1999 HK E	1929 HK E	2331 HK E	1114 HK E	3813 HK E	136 HK Eq	1211 HK E	3818 HK E	1828 H
2	1/1/2008										
3	2/1/2008										
4	3/1/2008										
264	1/1/2009										
265	2/1/2009				0.018934	0.003352		-3.16E-06	0.009363	0.015526	0.005
266	5/1/2009				0.027467	0.009583		1.93E-05	0.022296	0.032599	0.006
267	6/1/2009				-0.00438	-0.0007		-9.05E-06	0.032828	0.00705	-0.00

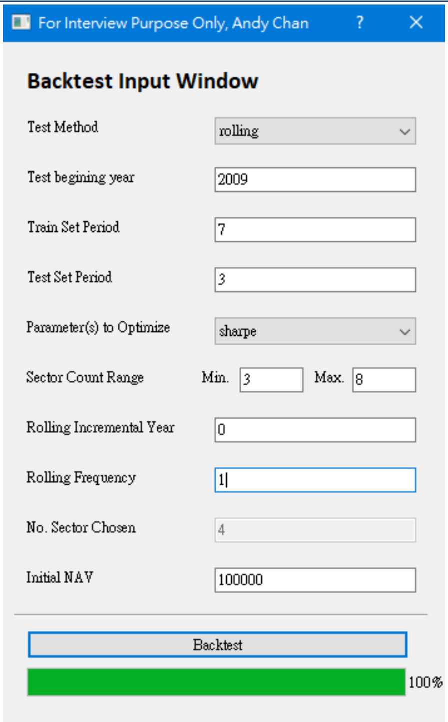
RP	RQ	RR	RS	RT
36 HK E	Daily Retu	Cumulative Ret	Stock Count	Portfolio NAV
	0	1	0	1000000
	-0.51361	0.994863917	127	994863.9171
	-3.53001	0.959745138	129	959745.138
	0	0.502095992	0	502095.9921
	3.789101	0.521120917	128	521120.9175
	3.923255	0.541565819	139	541565.8193
	1.427162	0.549294839	130	549294.8388

Part 2 – Backtesting

The second part of this module is to conduct backtesting on the sector selection strategy. There are two types of testing methods can be chosen from: 1) single holdout cross validation, and 2) rolling window testing. User can change testing parameters in input window (fig. 14) to alter the test.

Main test parameters a user can set are successively testing method, beginning year of testing period, in-sample period, out-of-sample period, minimum and maximum sector count to be tested, time interval between each rolling window, rolling frequency, number of sectors chosen, and initial NAV.

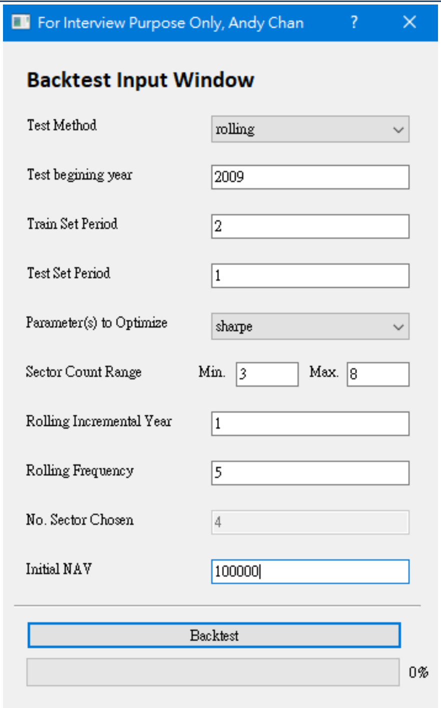
Figure 14 Test parameters for single holdout cross validation



For example, the setting shown in fig. 14 can be read as "10 years testing period since **2009**, of which **7** years in-sample and **3** years out-of-sample. Perform backtest **1** time, with **0** year time interval between each run. Try sector count from **3** to **8** with in-sample data and test the performance of the one that can maximize **sharpe** ratio with out-of-sample data."

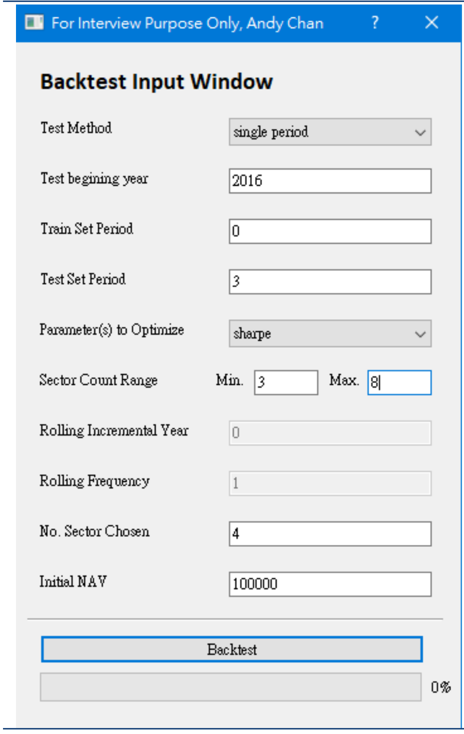
Under "rolling" test method, setting Rolling Incremental Year to 0 and Rolling Frequency to 1 for single holdout cross validation test (execute 1 time, no incremental year needed), while setting Rolling Incremental Year to > 0 and Rolling Frequency > 1 for rolling window testing (fig. 15).

Figure 15 Test parameters in config module for rolling window



On the other hand, "single period" method does not separate data into in-sample and out-of-sample. Instead of finding optimal trading parameters from the training data set, it directly uses user manual input of trading parameters in the whole testing period. Under this method, Rolling Incremental Year and Rolling Frequency will be set to 0 and 1 respectively (fig. 16).

Figure 16 Test parameters in config module for single period test method



For example, after running rolling window test, user may find that several versions of trading parameters are acceptable for the strategy. User can then conduct his/her own evaluation and judgement, manually input final decision, and test it with another set of data under single period test method. Further details will be elaborated in the next section.

Portfolio return matrix and market index data are the main inputs for backtesting. Backtesting results such as annual sharpe ratio will be shown in result window while related graphs will also be exported to corresponding folder. (fig. 17 & 18)

Figure 17 Outputs of single holdout cross validation: in-sample test

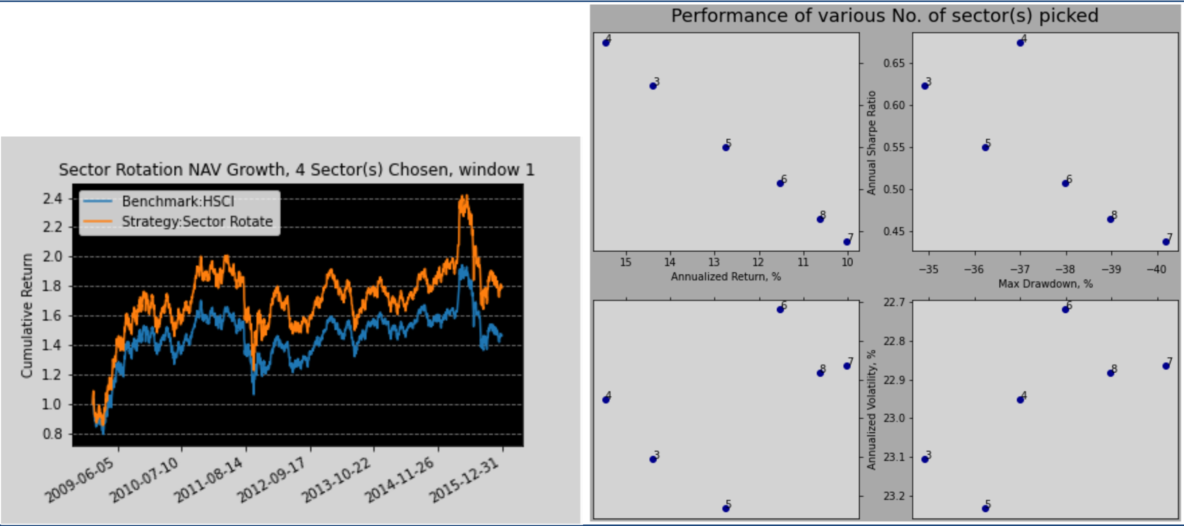
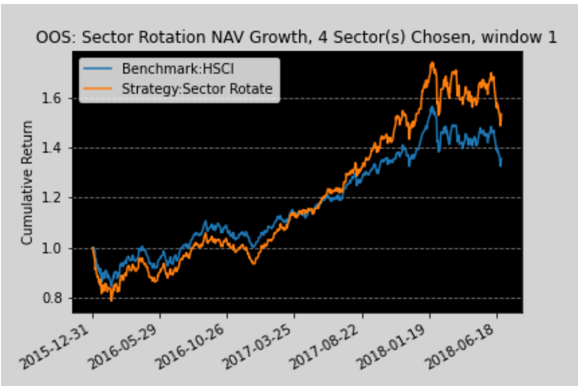


Figure 18 Outputs of single holdout cross validation: out-of-sample test



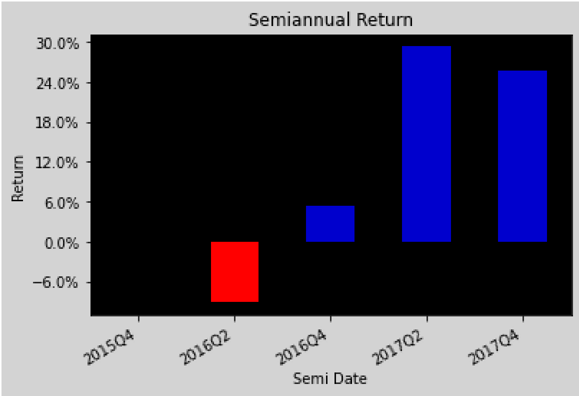
Picking 4 sectors is the best in terms of annualized sha

Out of sample test, rolling window: 1, 4 of sectors pick

649 trade days
Strategy:
Annual Sharpe Ratio: 1.01
Cumulative: 53.48 %
Max Consecutive Loss: 5 days
Max Drawdown:-21.4082 %
Win Rate:56.3725 %
Average Daily Return: 0.0726 %
SD of Daily Return: 1.1403 %
Annualized Return: 18.2833 %
Annualized Volatility: 18.1014 %

HSCI Index:
Annual Sharpe Ratio: 0.7735
Cumulative: 31.8276 %
Max Consecutive Loss: 6 days
Max Drawdown:-15.4676 %
Win Rate:56.6343 %
Average Daily Return: 0.05 %
SD of Daily Return: 1.0262 %
Annualized Return: 12.6 %
Annualized Volatility: 16.29 %

Figure 19 Outputs of single holdout cross validation: out-of-sample test



For single holdout cross validation and test method of "single period", an additional semiannual return graph will be shown and exported as well. (fig. 19)

Strategy Backtesting

Single holdout cross validation

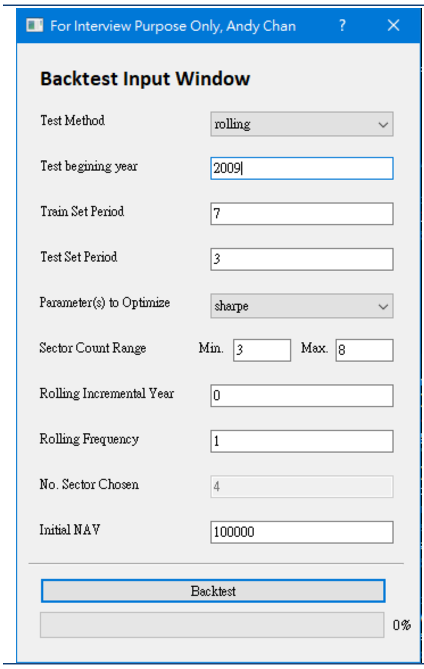
Over the 10 years after 2008, investment environment is characterized by record-low nominal interest rate and widespread usage of QE. A single holdout cross validation test will be conducted on these 10 years data.

For simplification purpose, we assume statistical stationarity during this period of 'new normal' and thus no regime analysis is needed to conduct accordingly. Transaction costs and dividend income are not considered during testing.

Backtest configuration:

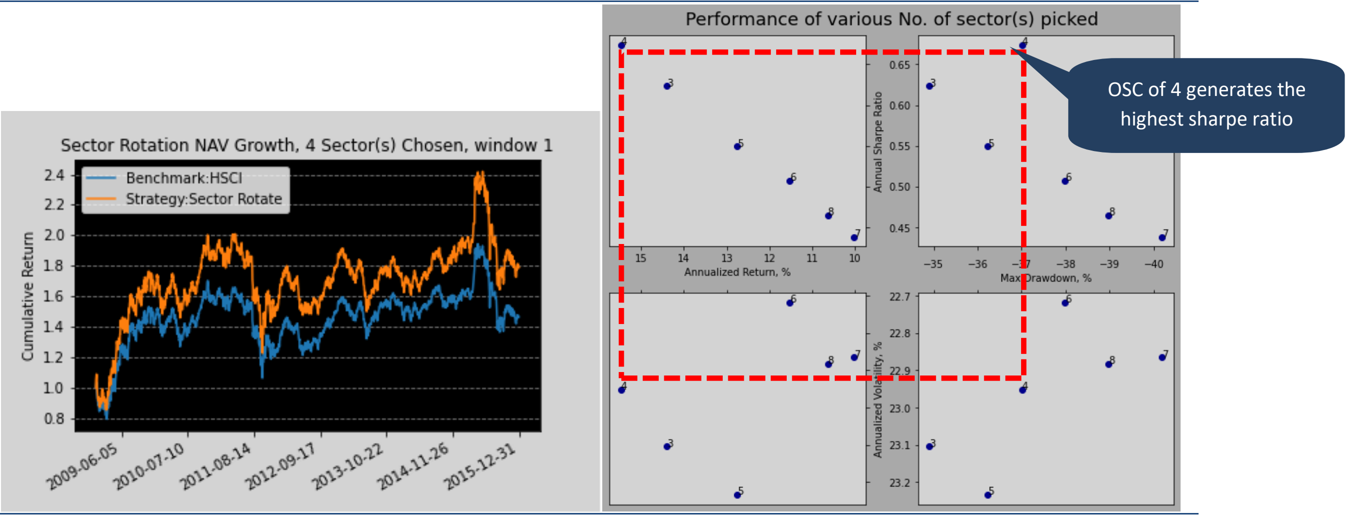
Testing market	Hong Kong
Benchmark	HSCI index
Evaluation metrics	Annual sharpe ratio (to be maximized)
Time period	10 years, from Jan 2009 to Jan 2019: ➢ Due to data accessibility, only contain data up to 2Q18 ➢ Using data after year 2008 to avoid distortion from 2008 financial crisis. ➢ Using data before year 2019 to avoid distortion from HK unrest and COVID-19
IS/OOS split ratio	2/3 data to be in-sample, remaining 1/3 to be out-of-sample
Testing sector count	3 to 8 sectors, out of total 11 sectors ➢ to diversify idiosyncratic risk, must pick at least 3 sectors

Figure 20 Test configuration



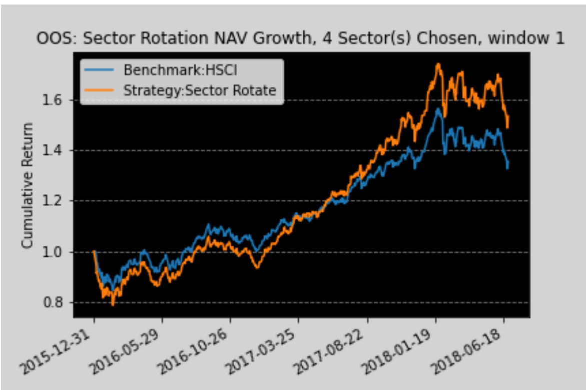
2:1 ratio is applied to determine in-sample and out-of-sample data. As a result, 10 years of data after year 2018 will be separated into two segments: first 7 years are in-sample while remaining ~3 years are out-of-sample (fig. 20). Based on system simulation on the in-sample data, the optimal sector count (hereafter 'OSC'), which generate the highest annual sharpe ratio, is 4. (fig. 21)

Figure 21 Outputs of single holdout cross validation: in-sample test



The system then refreshes investment signals based on OSC parameter and tests the strategy again on the second segment of data. Based on the backtest stats (fig. 22), the rotation strategy performance outperforms HSCI benchmark performance in terms of annual sharpe ratio (1.01 vs 0.774) and cumulative return (53.5% vs 31.83%).

Figure 22 Outputs of single holdout cross validation: out-of-sample test



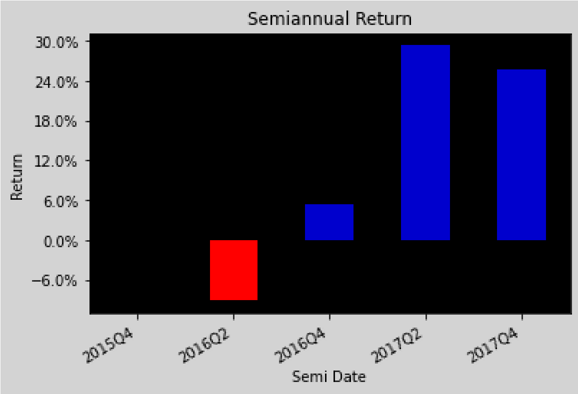
Picking 4 sectors is the best in terms of annualized sha

Out of sample test, rolling window: 1, 4 of sectors pick

649 trade days
Strategy:
Annual Sharpe Ratio: 1.01
Cumulative: 53.48 %
Max Consecutive Loss: 5 days
Max Drawdown:-21.4082 %
Win Rate:56.3725 %
Average Daily Return: 0.0726 %
SD of Daily Return: 1.1403 %
Annualized Return: 18.2833 %
Annualized Volatility: 18.1014 %

HSCI Index:
Annual Sharpe Ratio: 0.7735
Cumulative: 31.8276 %
Max Consecutive Loss: 6 days
Max Drawdown:-15.4676 %
Win Rate:56.6343 %
Average Daily Return: 0.05 %
SD of Daily Return: 1.0262 %
Annualized Return: 12.6 %
Annualized Volatility: 16.29 %

Figure 23 Outputs of single holdout cross validation: out-of-sample test



Walk Forward Optimization – Optimal Sector Count

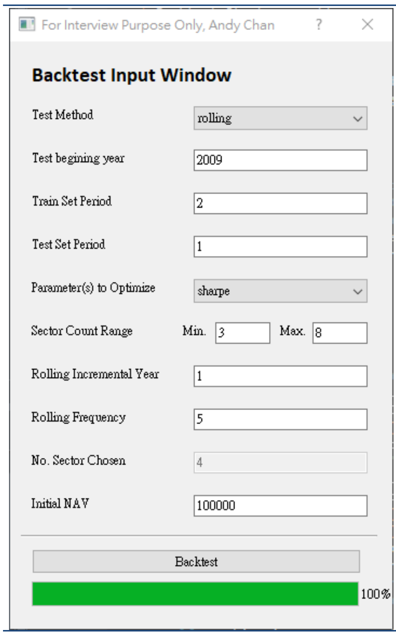
To reduce potential overfitting, rolling window test will be conducted on the first segment of testing data to further optimize OSC.

Given 7 years of data availability (from Jan 2009 to Dec 2015), we set rolling window period as 3 years and testing frequency as 5 times. 2:1 ratio is again applied, splitting each 3 years testing window into 2 years in-sample and 1 year out-of-sample. (fig. 24)

Additional configuration:

Rolling increment	1 year *(next window slides 1 year)*
Rolling frequency	5 times *(5 windows)*

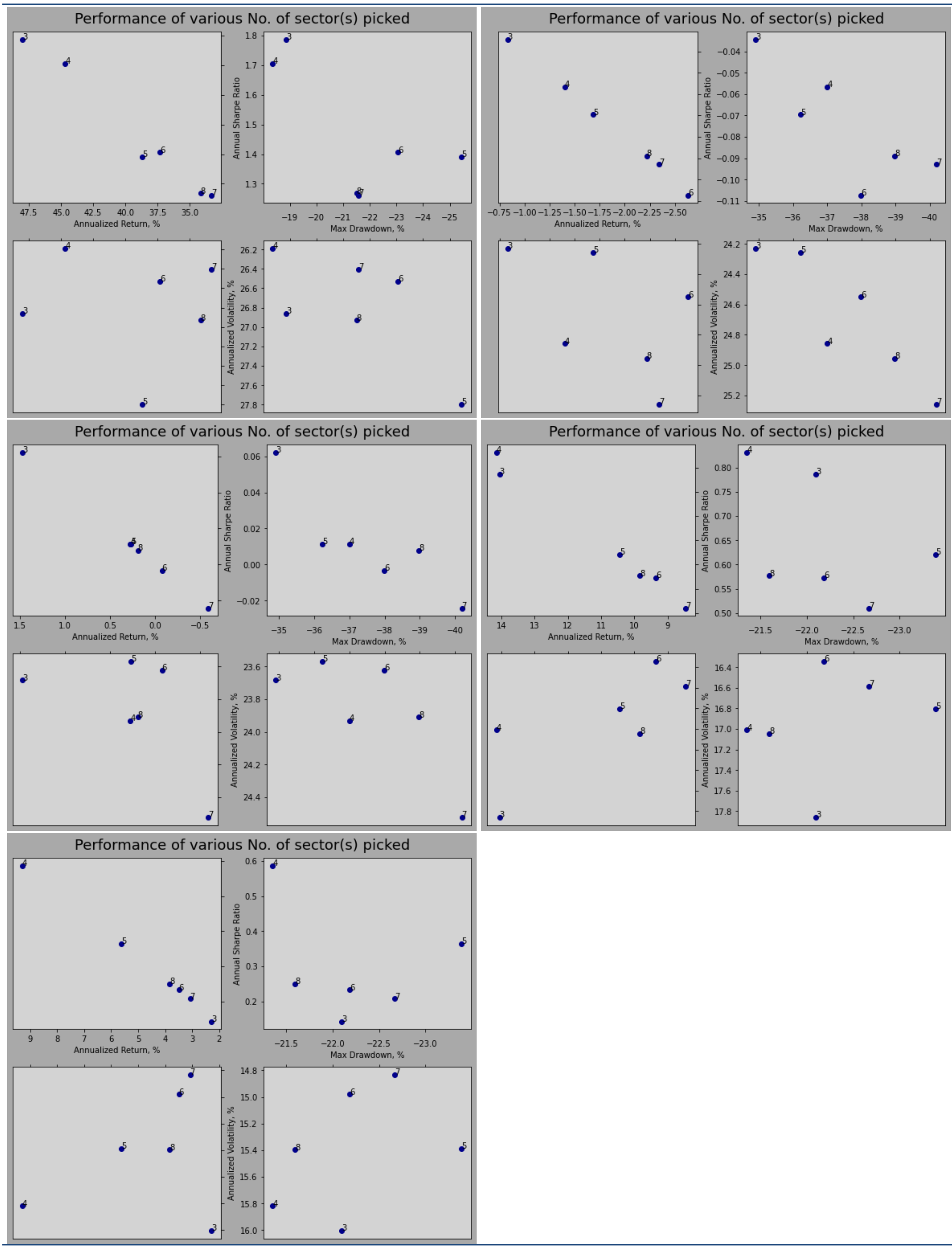
Figure 24 Test configuration



In-sample tests – optimal sector count filtering

In 5 in-sample test runs, OSC of 3 and 4 perform well in terms of annual sharpe ratio (fig. 25). As a result, both OSC are further evaluated in their respective 1 year out-of-sample test.

Figure 25 In-sample test KPI comparison (time window 1 to 5, from left to right)



Out-of-sample tests – optimal sector count testing

Based on out-of-sample results of 5 test runs (3 tests with OSC of 3; 2 tests with OSC of 4), sector rotation strategy with OSC of 4 beats benchmark in terms of annual sharpe ratio in both tests while OSC of 3 loses to benchmark in one of the three tests (fig. 26). Given that other KPI such as win rate do not offer strong evidence to differentiate one OSC's performance from the other, OSC of 4 will be picked considering that it offers higher diversification against idiosyncratic risk of sector.

Figure 26 Out-of-sample test KPI comparison

Picking 3 sectors is the best in terms of annualized sharpe, with 1.7875

Out of sample test, rolling window: 1, 3 of sectors picked

260 trade days
Strategy:
Annual Sharpe Ratio: -0.6715
Cumulative: -20.6664 %
Max Consecutive Loss: 7 days
Max Drawdown:-34.9069 %
Win Rate:47.7551 %
Average Daily Return: -0.0738 %
SD of Daily Return: 1.7451 %
Annualized Return: -18.6007 %
Annualized Volatility: 27.702 %

HSCI Index:
Annual Sharpe Ratio: -0.8591
Cumulative: -22.9207 %
Max Consecutive Loss: 8 days
Max Drawdown:-36.0094 %
Win Rate:48.1633 %
Average Daily Return: -0.0918 %
SD of Daily Return: 1.6968 %
Annualized Return: -23.1417 %
Annualized Volatility: 26.9363 %

Picking 3 sectors is the best in terms of annualized sharpe, with -0.0343

Out of sample test, rolling window: 2, 3 of sectors picked

261 trade days
Strategy:
Annual Sharpe Ratio: 1.141
Cumulative: 22.6391 %
Max Consecutive Loss: 8 days
Max Drawdown:-20.3443 %
Win Rate:51.8219 %
Average Daily Return: 0.0852 %
SD of Daily Return: 1.1855 %
Annualized Return: 21.4722 %
Annualized Volatility: 18.8185 %

HSCI Index:
Annual Sharpe Ratio: 1.1849
Cumulative: 19.6805 %
Max Consecutive Loss: 8 days
Max Drawdown:-17.1014 %
Win Rate:53.8776 %
Average Daily Return: 0.0786 %
SD of Daily Return: 1.0527 %
Annualized Return: 19.8007 %
Annualized Volatility: 16.7111 %

Picking 3 sectors is the best in terms of annualized sharpe, with 0.0622

Out of sample test, rolling window: 3, 3 of sectors picked

261 trade days
Strategy:
Annual Sharpe Ratio: 0.3934
Cumulative: 5.5543 %
Max Consecutive Loss: 10 days
Max Drawdown:-22.0978 %
Win Rate:49.5902 %
Average Daily Return: 0.0263 %
SD of Daily Return: 1.0629 %
Annualized Return: 6.6374 %
Annualized Volatility: 16.8734 %

HSCI Index:
Annual Sharpe Ratio: 0.1937
Cumulative: 1.7475 %
Max Consecutive Loss: 8 days
Max Drawdown:-17.2208 %
Win Rate:48.1481 %
Average Daily Return: 0.0117 %
SD of Daily Return: 0.9628 %
Annualized Return: 2.9608 %
Annualized Volatility: 15.2845 %

Picking 4 sectors is the best in terms of annualized sharpe, with 0.8316

Out of sample test, rolling window: 4, 4 of sectors picked

261 trade days
Strategy:
Annual Sharpe Ratio: 0.7755
Cumulative: 11.6665 %
Max Consecutive Loss: 5 days
Max Drawdown:-10.2435 %
Win Rate:53.4413 %
Average Daily Return: 0.0469 %
SD of Daily Return: 0.9596 %
Annualized Return: 11.8131 %
Annualized Volatility: 15.2336 %

HSCI Index:
Annual Sharpe Ratio: 0.0742
Cumulative: 0.0861 %
Max Consecutive Loss: 7 days
Max Drawdown:-9.3662 %
Win Rate:52.8455 %
Average Daily Return: 0.004 %
SD of Daily Return: 0.8545 %
Annualized Return: 1.0061 %
Annualized Volatility: 13.564 %

Picking 4 sectors is the best in terms of annualized sharpe, with 0.5869

Out of sample test, rolling window: 5, 4 of sectors picked

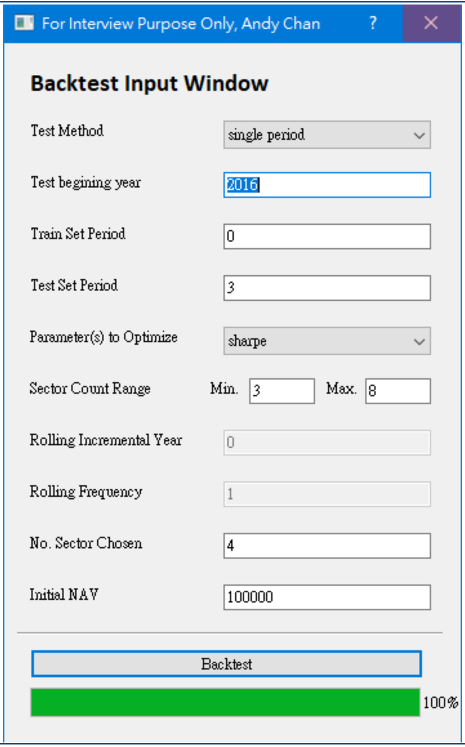
261 trade days
Strategy:
Annual Sharpe Ratio: -0.0773
Cumulative: -5.3346 %
Max Consecutive Loss: 8 days
Max Drawdown:-33.7907 %
Win Rate:48.583 %
Average Daily Return: -0.0079 %
SD of Daily Return: 1.622 %
Annualized Return: -1.9916 %
Annualized Volatility: 25.7483 %

HSCI Index:
Annual Sharpe Ratio: -0.3118
Cumulative: -8.6612 %
Max Consecutive Loss: 9 days
Max Drawdown:-29.8616 %
Win Rate:47.9675 %
Average Daily Return: -0.0272 %
SD of Daily Return: 1.3866 %
Annualized Return: -6.8637 %
Annualized Volatility: 22.011 %

Test on second segment of data

Once the optimal OSC is determined, user can then input it into No. Sector Chosen field and start testing with the second segment of data from 2016 to 2019.

Figure 27 Test configuration



With the same testing period and OSC, the same result with single holdout cross validation is generated (fig. 28). The performance sector rotation strategy successfully outperforms the performance of HSCI index benchmark, with annual sharpe ratio of 1.053 vs benchmark's 0.855, and cumulative return of 58.22% vs benchmark's 35.62%, representing an alpha of 22.6%.

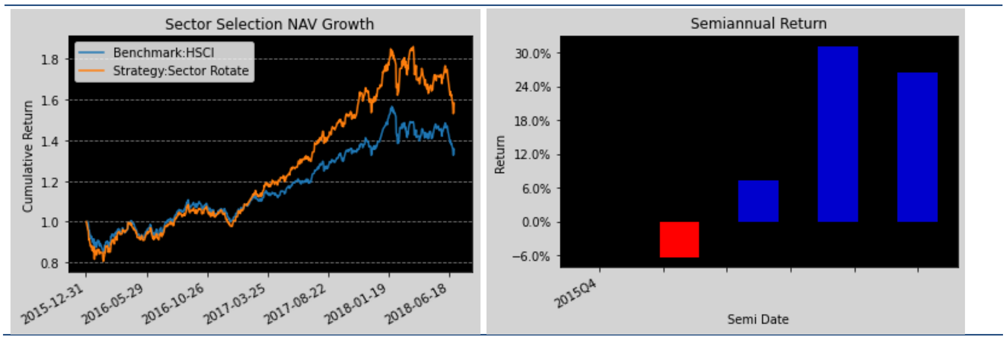
Figure 28 Test results

649 trade days
Strategy:
Annual Sharpe Ratio: 1.0534
Cumulative: 58.2207 %
Max Consecutive Loss: 6 days
Max Drawdown:-19.4795 %
Win Rate:56.0458 %
Average Daily Return: 0.0776 %
SD of Daily Return: 1.1691 %
Annualized Return: 19.5493 %
Annualized Volatility: 18.5584 %

HSCI Index:
Annual Sharpe Ratio: 0.8553
Cumulative: 35.6247 %
Max Consecutive Loss: 6 days
Max Drawdown:-15.4676 %
Win Rate:56.9558 %
Average Daily Return: 0.0551 %
SD of Daily Return: 1.0233 %
Annualized Return: 13.8935 %
Annualized Volatility: 16.2445 %

For Interview Only, Not For Distribution

Figure 29 Semi return and cumulative return of the strategy



In conclusion, investors can benefit more from investing using sector rotation strategy than passive index tracking strategy.