Portfolio Optimization Lab

Christian Bongiorno | Séries Temporelle | 20-04-2021

Project Overview

Portfolio Selection.

- The Russell 1000 Index is a stock market index that tracks the highest-ranking 1,000 stocks in the Russell 3000 Index, which represent about 90% of the total market capitalisation of that index
- I will introduce you some performance metrics you should apply
- I will discuss some technical problem dealing with real data
- I will present the assignments
- We will work together till the end of the lesson. At the end of the lesson I’ll show you my implementation.

- Suggestions: Use numpy, vectorialize whenever you can. Use for-cycle only if it is strictly necessary

The Data-set

	A	AA	AAL	AAN	AAP	AAPL	ABBV	ABC	ABMD	ABT	...	YUMC	Z	ZAYO	Z
index															
2019-11-06	0.003172	-0.031021	-0.007081	-0.049157	0.009235	0.000428	0.002928	-0.016624	-0.013560	0.009428	...	-0.002343	0.008061	0.001751	0.020
2019-11-07	0.001582	0.029692	-0.002264	-0.037938	-0.002478	0.011475	-0.000853	-0.007465	0.020843	0.003243	...	0.020428	-0.005666	-0.000583	-0.009
2019-11-08	0.009571	-0.001331	-0.004217	-0.030116	-0.011168	0.002733	0.038278	0.023392	0.002234	0.004188	...	0.007554	0.115921	-0.001168	0.014
2019-11-11	0.002736	-0.013855	-0.005542	-0.014708	0.008447	0.007888	0.005267	-0.023868	0.002092	0.000239	...	-0.005030	0.028356	-0.000877	-0.004
2019-11-12	0.002599	-0.001802	-0.035605	0.016595	-0.078080	-0.000916	0.009065	0.017691	0.015599	0.006901	...	-0.018973	-0.020927	0.001169	0.008
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
2020-03-25	0.029414	-0.025071	0.100391	0.132570	0.045156	-0.005524	0.006204	-0.058006	0.022165	0.015096	...	0.035718	0.080084	NaN	0.036
2020-03-26	0.066168	-0.035898	0.017392	-0.002950	0.072035	0.051285	0.083582	0.110576	0.015403	0.069078	...	0.013183	0.022066	NaN	0.043
2020-03-27	-0.038863	-0.043323	-0.109199	-0.001690	-0.015775	-0.042284	-0.015837	-0.028687	-0.026699	-0.016626	...	-0.056858	-0.072186	NaN	-0.060
2020-03-30	0.027059	-0.076106	-0.136384	-0.047628	0.009370	0.028138	0.034754	0.074440	0.025492	0.062138	...	-0.005891	-0.023468	NaN	0.034
2020-03-31	-0.014554	0.014718	-0.004910	0.010148	-0.033509	-0.002043	0.012547	0.004530	-0.026043	-0.005434	...	0.007535	-0.028464	NaN	0.028

The data are daily adjusted close-to-close **log-returns.**

Some data could miss for different reasons. Of course the methods I show you cannot handle missing data.
Even if you should not use future information in-sample, I can accept to select the set of stocks that are without NaN both in-sample and out-of-sample

Inspecting Redundant Stocks

Typically, the first eigenvector is the largest, then they drop down

$$\boldsymbol{\Sigma}^{-1} = \frac{1}{\lambda_1}\boldsymbol{v}_1\boldsymbol{v}_1' + \frac{1}{\lambda_2}\boldsymbol{v}_2\boldsymbol{v}_2' + \frac{1}{\lambda_3}\boldsymbol{v}_3\boldsymbol{v}_3' + \ldots + \frac{1}{\lambda_n}\boldsymbol{v}_n\boldsymbol{v}_n'$$

Let us assume that $\lambda_n << \lambda_{n-1}$, then the composition of the eigenvector $\boldsymbol{v}_n$ will dominates the composition of the portfolio.

If some correlations are equal to +/- 1 (or close), the smallest eigenvalues converge to zero.

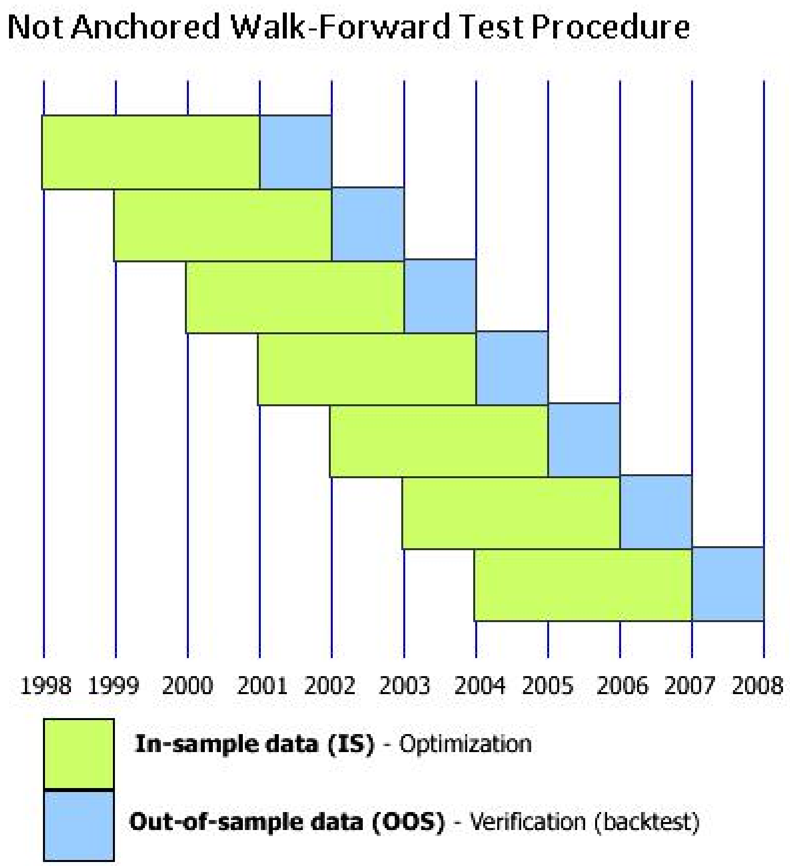
Null eigenvalues could come also for $T \leq N$

Cross-Validation in Finance

We must use historical observations (in-sample), to predict to predict the next future (out-of-sample).

In-sample: Find the weight of the portfolio

out-of-sample: Use the weight obtained in-sample to measure the performances



Risk Measure

- **Expected Risk**: $\sigma_P^{in} = \sqrt{252\ \boldsymbol{w}\boldsymbol{\Sigma}^{\boldsymbol{in}}\boldsymbol{w}'}$ where $\boldsymbol{\Sigma}^{\boldsymbol{in}}$ is the in-sample covariance matrix.
- **Realized Risk**: $\sigma_P^{out} = \sqrt{252\ \boldsymbol{w}\boldsymbol{\Sigma}^{\boldsymbol{out}}\boldsymbol{w}'}$ where $\boldsymbol{\Sigma}^{\boldsymbol{out}}$ is the out-of-sample covariance.

Note that the covariance could be defined as

$$\frac{1}{N-1}\sum_{i=1}^{N}\left(X_{ij}-\bar{X}_j\right)\left(X_{ik}-\bar{X}_k\right)$$

np.cov(X)

$$\frac{1}{N}\sum_{i=1}^{N}\left(X_{ij}-\mathrm{E}(X_j)\right)\left(X_{ik}-\mathrm{E}(X_k)\right).$$

np.cov(X, bias=True) [**USE THIS**]

Sharpe-Ratio

- $SR_P^{in} \;=\; = \frac{\langle r_P^{in} \rangle}{\sigma_P^{in}} \sqrt{252}$
- $SR_P^{out} \;=\; = \frac{\langle r_P^{out} \rangle}{\sigma_P^{out}} \sqrt{252}$
- To be correct the SR must be computed on the log-returns rather than on vanilla-returns. However, on short time-horizon they are approximately equivalents. Alternatively, you can computed the time-series of the daily log-returns of your portfolio, then $\sigma_P^{\blacksquare}$ will be the standard-deviation

Portfolio Metrics

- **The effective portfolio diversification is defined as**

$$N_{eff} = \frac{1}{\sum_{i=1}^{N} w_i^2}$$

which represent the effective number of stocks with a significant amount of money invested.

- **Gross Leverage**

$$G = \sum_{i=1}^{1} |w_i|$$

When no short selling is allowed G=1. Portfolios with $G > 1$ have an additional intrinsic risk due to high level of short-selling

Pantaleo, E., Tumminello, M., Lillo, F., & Mantegna, R. N. (2011). When do improved covariance matrix estimators enhance portfolio optimization? An empirical comparative study of nine estimators. Quantitative Finance, 11(7), 1067-1080.

Assignments

1. Use the first 2000 days to obtain an efficient frontier that include the minVar and the maxSR portfolio.

   1a. Check and remove possibly redundant stocks.

   1b. Compare the out-of-sample (next 252 days) efficient frontier with the in-sample one

2. Study for different in-sample window size the minVar portfolio performances (in-sample vs out-of-sample)

   2a. Plot out-of-sample Risk, Gross-leverage, and lowest eigenvalue of the covariance matrix for different value of the in-sample windows [N+1, .., 2000] days

3. If you have time, implement the long-only constrains (w_i >0)