Momentum-Based NIKKEI225 Portfolio Optimizer Using XGBoost Regressor

LI Chung Yat, 20422979

1 Introduction

This paper aim to build a trading algorithm that trade on all components in NEKKEI225 index. Using a variety of momentum factors and derive a model that can make accurate forecast in terms of both magnitude and direction on t+1 return. Then, to use the predicted return to perform portfolio optimization in which can yield a promising performance and provide proper explanations in stock price movement.

2 Methodology

2.1 Data Source

NIKKEI225 index components are extracted by yahoofiance python library. Daily close price and trading volume are crawled and further processed to produce factors used in model fitting. A note on the ticker 9343.T, Softbank Corp., that yahoofinance on python is unable to pull the desired data after 2019, thus, I saved the data as csv in a data dump and plug the missing data manually. The Repo rate is downloaded from Japan Securities Dealers Association website.

2.2 Factors

2.2.1 Return Lags

Return lags are included to capture stock price momentum. It is calculated directly through historical price.
Return lags to catch short term momentum: 1-day, 2-day, 3-day, 4-day, 5-day
Return lags to catch long term momentum: 14-day, 20-day
Frankly speaking, the 14-day and 20-day return lags are not the most ideal way to capture the long-term momentum. Return lag against 14-day VWAP and 20-day VWAP will be more desirable.

2.2.2 Average Daily Trading Volume

ADTV is an indicator that reflects the stocks' popularity over a period. In often, stocks that are frequently traded will experience a dramatic price shock or rally. This also implies that magnitude of the momentum is higher in the period. 3-day ADTV, 5-day ADTV, 14-day ADTV and 20-day ADTV are included.

2.2.3 Annualized Realized Volatility

Realized volatility tends to be larger in a bearish market. It is expected to see stock returns being negatively correlated to the realized volatility. 3-day, 5-day, 14-day and 20-day realized volatility are included.

2.2.4 Overnight Repo Rate

This rate is an implied rate from overnight repurchasing agreement (repo). Due to the use of collateral, a repo is considered free of credit risk, and thus the interest rate implied by the repo price is essentially a riskless interest rate.

2.3 Model

2.3.1 Parametric model vs non-Parametric model

In the past, parametric model dominated this kind of portfolio optimization problem as it can provide a numeric forecast results that help portfolio managers to determine and adjust the portfolio weights. It can also provide interpretable results and being properly justified with economy and market behaviors. However, in this paper, a non-parametric model XGBoost regressor is used instead of traditional parametric models. The motivation behind is to utilize the edge of non-parametric model making a more accurate forecast and ability to autofit the regressor.

2.3.2 XGBoost

XGBoost is a gradient boosting machine learning technique in the form of an ensemble of decision tree models. The algorithm uses the ensemble that has low prediction power to teach the main model by minimizing the mean square error in the model. Ensemble learning is powerful as it trains multiple learners and combines results to figure out the ultimate model that has the best predictive power. Variance and bias of a model can be effectively reduced by ensemble learning as more perspectives are considered when multiple models are built. Bagging and boosting are the two ensemble techniques widely used. Bagging, an abbreviation of Bootstrap Aggregation, is the technique to repeatedly build classifiers based on a subset of features selected from the full scope (Shubham, 2018). The prediction is then made by aggregating all predictions from all classifiers. Bias will be low when multiple advisors give suggestions from various perspectives. Still, the variance could be high as the decision trees embedded in bagging usually grow deep and overfit for the sake to build up significant characteristics in each learner. Boosting, on the other hand, brings significant improvement in lowering the model’s variance. Different learners will corporate and figure out the most significant features by assigning weights to them after trials and trials. All learners will eventually converge, and the optimal model is generated. For boosting, reduction of bias might not be as efficient as bagging while the variance of results can be significantly decreased.

2.3.2.1 XGBoost Regressor

It is the regressor module of the machine learning model. Knowing that traditional non-parametric model is only capable of performing categorical classification, it specifically addresses the needs of numerical output using the significant factors selected and automatically fit a best-fit regression model. This feature will play a crucial role in the portfolio optimization part of the project.

2.3.2.2 Rationale of XGBoost

XGBoost is a gradient boosting library designed to be more efficient, flexible and portable compared to traditional libraries. XGBoost provides different boosters: Tree Booster, Dart Booster and Linear Booster. The XGBoost model can be initially trained by the objective with the objective function:

$$\text{obj} = \sum_{i=1}^{n} l\,(y_i, \hat{y}_i^{(t)}) + \sum_{i=1}^{t} \Omega\,(f_i)$$

f_i represents the functions that contain the structure of the tree and the leaf scores. After that, by writing the prediction value at step t as $\hat{y}_i^{(t)}$ in

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k\,(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i)$$

The objective function will become

$$\text{obj}^{(t)} = \sum_{i=1}^{n} [\,l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t) + \text{constant}$$

where g_i and h_i are defined as

$$\begin{aligned} g_i &= \partial_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)}) \\ h_i &= \partial^2_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)}) \end{aligned}$$

After removing all the constants, the objective function at a specific step t becomes

$$\sum_{i=1}^{n} [\,g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t)$$

2.3.2.2 Advantages of XGBoost

2.3.2.2.1 Regularization

XGBoost prevents the model from overfitting by using in-built L1 and L2 regularization.

2.3.2.2.2 Parallel Processing

XGBoost allows users to use multiple CPU cores or GPU to execute the model, which makes the run much faster.

2.3.2.2.3 Handling Missing Values

In handling missing values, XGBoost tries both the left- and right-hand split and learns the way leading to higher loss for each node.

2.3.2.2.4 Cross Validation

XGBoost allows users to run cross-validation at each iteration of the boosting process. Hence, users can access the optimized parameters for the model easier, even with a single run.

2.3.2.2.5 Effective Tree Pruning

Normal Gradient Boosting Machine would stop splitting a node when a negative loss occurs in the split. However, in XGBoost, it would split up to the max_depth specified and then start pruning the tree backwards and remove splits beyond with no positive gain.

2.3.3 Trading Strategy

To keep the strategy as simple as possible, we will rebalance the long-only portfolio on a daily basis. Assuming the stock price is only continuous over the actual trading hours of exchanges, the close price at time t is the same as the open price at the time (t + 1). The model will take long positions of the corresponding stocks' opening price and unwind all the positions at market close.

2.3.4 Portfolio Optimization

2.3.4.1 Arithmetic average of non-negative Predicted Return

The portfolio weights of the indexes are assigned based on the predicted return of the corresponding index. The model will assign zero weight on the index if the predicted return is smaller than or equal to zero. Else, the weights of each index will be calculated as the weighted average of the predicted return. Thus, this asset allocation methodology will only resulted in long-only portfolio and control portfolios will be set up to evaluate long-only bias.

2.3.4.2 Arithmetic average of non-negative Predicted Risk-Adjusted Return

The weight calculation methodology is the same as the previous section, but instead of simple predicted return, predicted risk-adjusted return is used. The predicted risk-adjusted return is calculated as follow:

$$S_i = \frac{E\left[\widehat{R_\iota} - R_f\right]}{\sigma_i}$$

where $\widehat{R_\iota}$ is the predicted return from XGBoost Regressor, R_f is the Repo rate and σ_i is the 1-day realized volatility of excess return for i = 1, …, 225.

2.3.4.3 Mean-Variance Optimization Portfolio

MVO is added to evaluate the portfolio performance if short position is available in the trading strategy. MVO determines the optimal asset allocation weight with either of the following equivalent problems:

- Given the portfolio variance, solving for the highest expected return
- Given the portfolio expected return, solving for lowest portfolio variance

By solving the following quadratic programming problem:

$$Min_w \frac{1}{2} w^T V w$$

subject to the following constraints:

$$r^T w = \mu$$
$$1^T w = 1$$

As this paper will perform portfolio rebalance every day and trade on short-term momentum, a rolling window of 20 days, in other words, 19 days of actual return plus 1-day XGBoost forecast return will be fed to the MVO optimization. This ensured that only recent data are captured in the asset allocation process. MVO weights in which yield the highest expected risk-adjusted return will be selected.

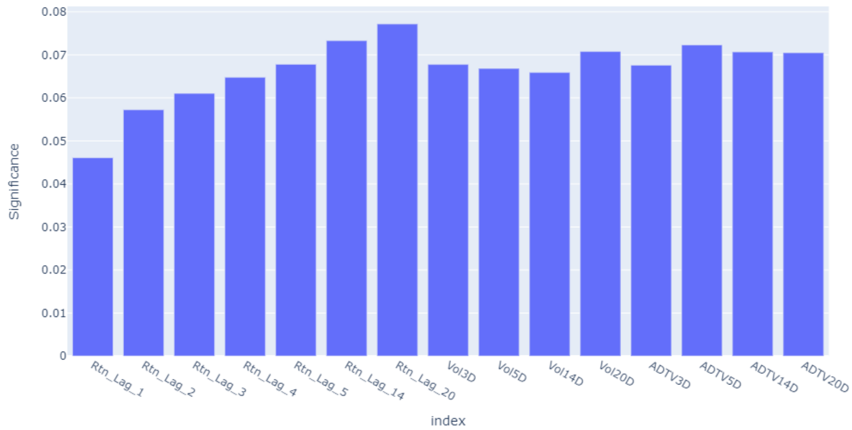
2.3.4.4 Control Portfolio: Long-bias Portfolio

A control portfolio is set up to estimate the biased return of daily long-only strategy. The portfolio will equally assign weights to all the instruments. This is motivated by the fact that the NIKKEI225 has been carrying a strong long-term momentum since 2020 and reached its historical high in 2021 March. It is necessary to carry out this control portfolio to determine whether the arithmetic average and the MVO portfolio can bring excess return in bullish market.

3 Results & Discussions

3.1 Model Performance: XGBoost Regressor

3.1.1 Factors significance



The XGBoost regressor gave a surprising result regarding the features significance in the model. The long-term return lag is deemed as higher importance features in the model compared to the short-term return lag. This reflects that trading on short-term momentum is less desirable than the long-term one. However, this phenomenon is justifiable by the fact that NIKKEI225 experience a steady growth since 2017, again, the long bias in the model. No conclusive statements can be made on realized volatility and ADTV, they need to be further examined with different tenors.

3.1.2 Statistics

Accuracy	Precision	Recall
0.495199	0.535234	0.494273

3.1.2.1 Accuracy

Accuracy examines the model's chance to "do the right thing", long when the price will increase, and pass/short when the price will drop or stay at the time (t + 1). This indicator can give us a general sense of how well the model is performing but cannot bring too much value to the model from a financial perspective. The average accuracy of the model is roughly 49.5%. This is also the main reason why short is forbidden in the portfolio optimization as the accuracy is worst than a blind guess.

3.1.2.2 Precision

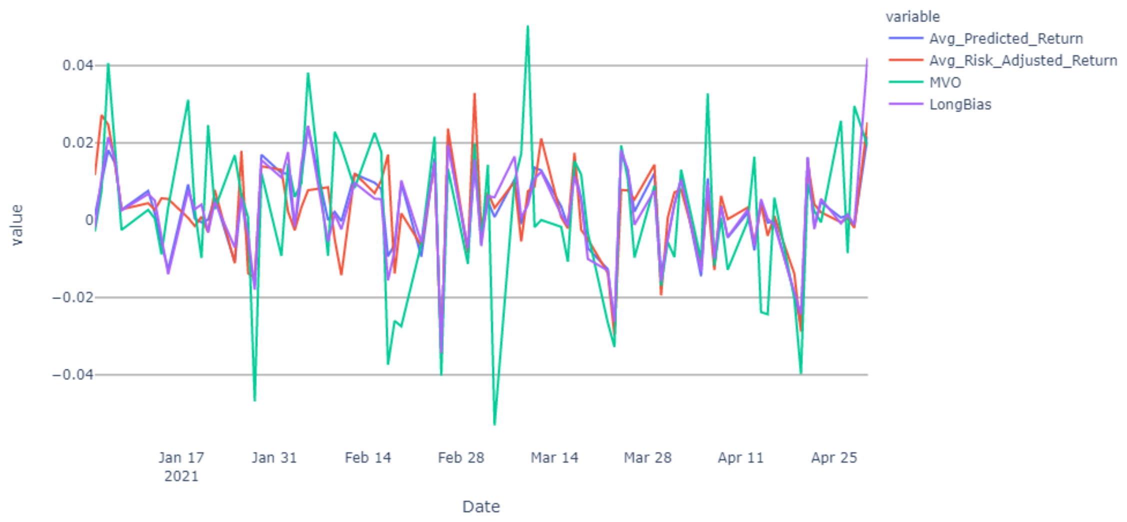
The precision is the indicator that reflects the model's ability to make a correct decision given that the model decided to engage in a trade. This statistic brings more financial interpretation to the model compared to the others. As there will be no profit and loss incurred if the model did not engage in a trade, precision is a more informative factor to be evaluated on when profitability is the final goal of the model. The model in general can make precise investment decisions. The average precision of the model is roughly 53.5%. The only concern is the long bias during the bullish market in which probability of resulting a positive actual return is higher than negative one. Therefore, a control portfolio to test long bias in portfolios is set up.

3.1.2.3 Recall

The recall reflects the model's ability to catch the profitable scenario. It evaluates the probability of the model to engage in trades given that the actual return is positive. The average recall of the model is roughly 49.4%. Recall is not necessarily to be high, a model with precision 100% but low recall is also desirable.

<u>3.2 Portfolio Performance: Return</u>

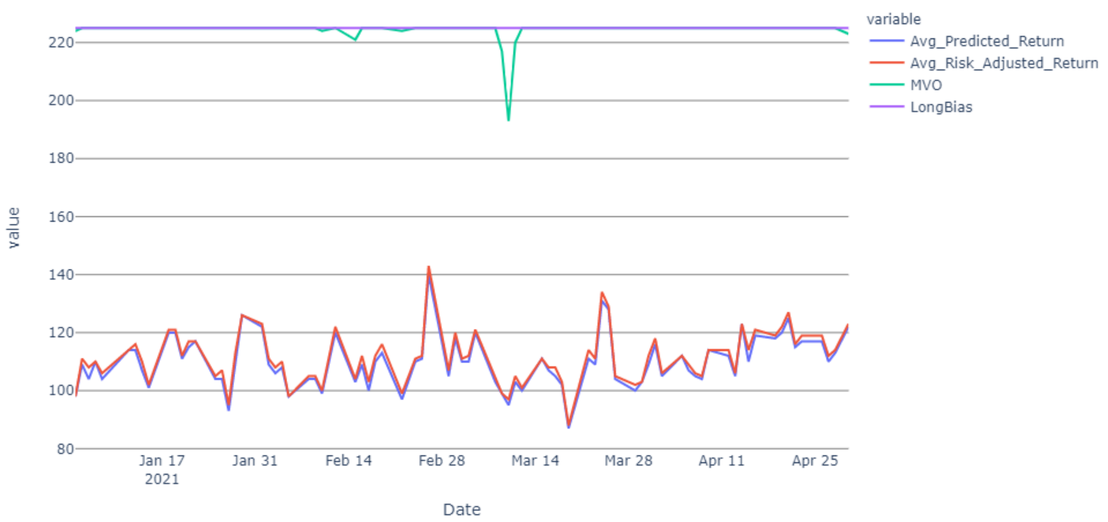
Portfolios Return:



Positively correlated returns:

	Avg_Predicted_Return	Avg_Risk_Adjusted_Return	MVO	LongBias
Avg_Predicted_Return	1.000000	0.838841	0.631666	0.963063
Avg_Risk_Adjusted_Return	0.838841	1.000000	0.476351	0.796183
MVO	0.631666	0.476351	1.000000	0.603792
LongBias	0.963063	0.796183	0.603792	1.000000

Number of long position in portfolio:



All the portfolios are positively correlated with the long bias portfolio, it is expected because all the 3 portfolios are long-only portfolios. Still, it is necessary to evaluate that whether the portfolio can produce excess return compared to the control portfolio, otherwise, the 3 portfolio returns are purely driven by positive momentum in Japan stock market. (The performance in generate excess return over the long bias portfolio will be further examined in the cumulative return section.)

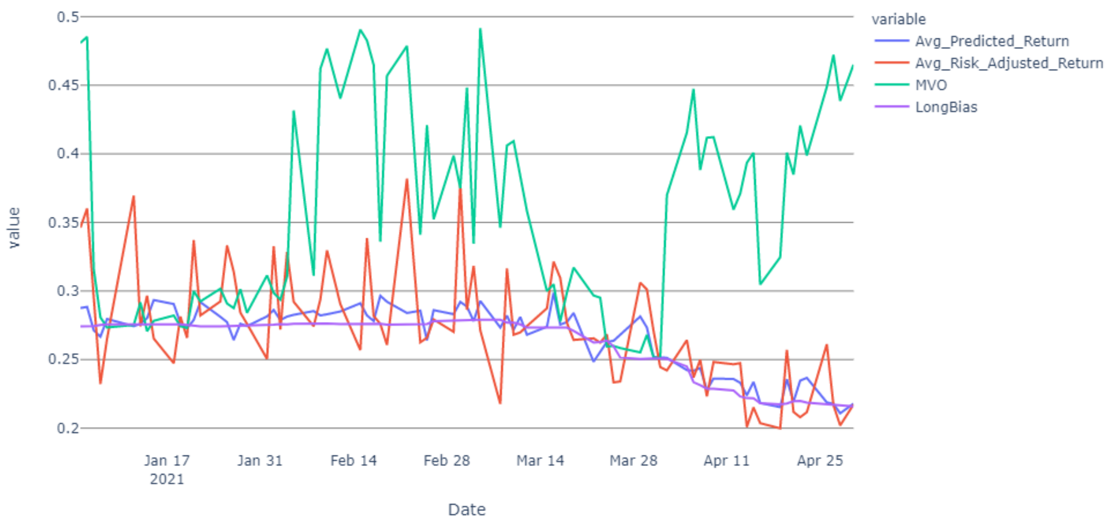
It is noticeable that average return portfolio is highly correlated (ρ=0.963) with the control portfolio. This implies that the portfolio in most of the time yields positive return only when general market is promising. Originally, high correlation is suspected to be caused by the XGBoost Regressor being frail that it always makes positive predictions and assign roughly equal weights to all the instruments. However, the number of long position plot proved the invalidity in the above claim in which the average return portfolio engaged in on average 110 positions throughout the period. Despite the invalid claim, the high correlation between the 2 portfolios will yet be a significant weakness of the average return portfolio if there is no excess return in this allocation strategy. Otherwise, there is no point to use the model as the fund manager can simply take evenly weighted long positions of NIKKEI225 components to yield the same return.

The only difference between average risk-adjusted return portfolio and average return portfolio is the weights assigned to the index components. Co-movement behavior with the average return portfolio is foreseeable but remarkably the returns' correlation falls to 0.796. Differences in portfolios' return between risk-adjusted return portfolio and long bias portfolio will be more significant. The return plot suggested that the risk-adjusted return portfolio generate higher return or loss compared to the average return portfolio.

Although the MVO portfolio basically assigns a portion of weight to all instruments, its return is relatively less correlated to the long bias portfolio. It is observable that the MVO portfolio return sometimes negatively correlated to the control portfolio. This is a desirable feature for long-only portfolio in which the ability to yield returns in bearish market.

3.3 Portfolio Performance: Volatility

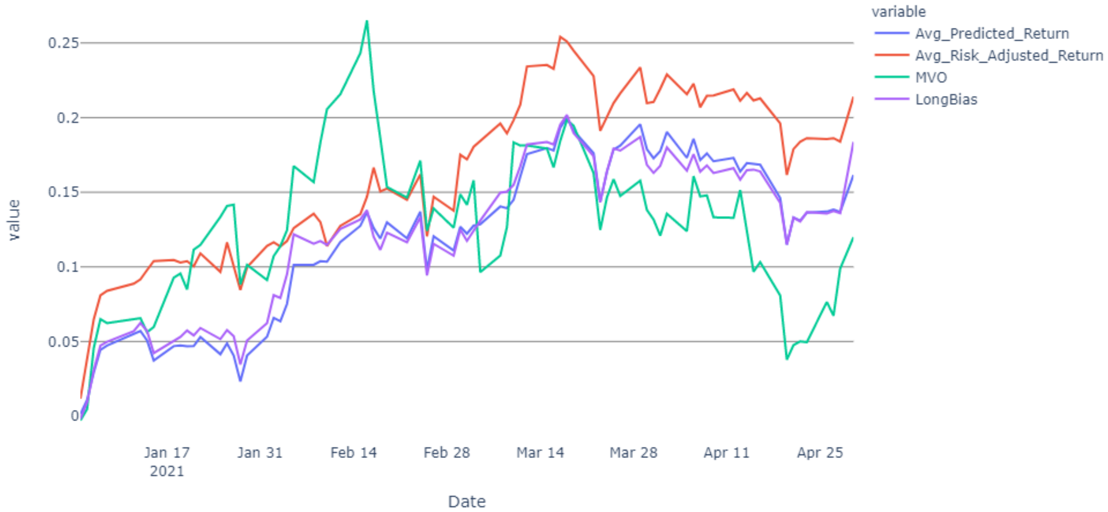
Portfolios Annualized Volatility:



On average, the MVO portfolio return is the most volatile among the 4, followed by the average risk-adjusted return portfolio. The remaining 2 portfolios have similar volatility. There is an apparent trend of decrease in volatility starting from March, implying the overall Japan stock market volatility is decreasing. The MVO portfolio, however, experienced a reversed trend in annualized realized volatility.

3.4 Portfolio Performance: Cumulative Return

Portfolios Cumulative Return:



To evaluate the actual performance of the asset allocation strategies, it is crucial to examine the cumulative return plot of the 4 portfolios. The similarity of cumulative return between the average return portfolio and long bias portfolio validated the hypothesis raised earlier on the high correlation. A conclusive result can be made on the average return strategy that it is heavy long biased and unable to generate excess return. The return it generated over the period is mainly driven by the bullish sentiment in the Japan stock market.

Although the average risk-adjusted return portfolio still exhibits strong co-movement feature with the long bias portfolio, there exists a surplus return buffer over the control portfolio. The strategy itself can bring value to investor regardless of the general market. This seems to be contradictory against the conclusion made on average return portfolio as the selected stocks in the 2 portfolios are the same. Recall the fact that number of long instruments of "averaging" strategies differs from the long bias portfolio, the strategies do undergo stocks selection process that is subjected to XGBoost Regressor's return forecast. This implies that the stocks selection based on regression result indeed contributes to the surplus return of the model which match the statistical summary of XGBoost Regressor (53% Precision).

MVO performance is the worse as the objective function set in the optimization process targets the highest risk-adjusted return weight. There is not constraint in the portfolio variance, thus, the returns generated bring larger impacts to the portfolio. The MVO portfolio starts crashing since mid-February when the NIKKEI225 index reached the historical high. This is weird but can be justified by the fact that realized volatility is high in a surplus environment which leads to a low risk-adjusted return, thus, underweight the blooming stocks.

4 Conclusions

4.1 Summary

The application of XGBoost Regressor for portfolio optimization in the composite of NIKKEI225 components is promising. The portfolio performs quite well in the surplus Japan stock market.

4.2 Future developments

Undoubtedly, a more sophisticated trading strategy is desired. All the associated cost incurred from market transaction is neglected in the calculation of portfolio return, for example, broker fees and exchange fees. Although stamp duties and funding cost are relatively low in Japan, the small cost embedded will accumulate and ends up to a big drawback of the model to be implemented in actual stock market. Especially, the trading strategy used rebalance of the portfolio daily, there will be a huge amount of transaction cost when market fictions are added into the return calculation. Other than that, the model misses the ability to identify buy-in and unwind opportunities, e.g. the model will unwind the position even if it predicts the index price will keep increasing over 2 days.

5 Reference

Japan Securities Dealers Association. Retrieved from https://www.jsda.or.jp/en/statistics/bonds/trr/index.html

Markowitz H., Portfolio Selection, The Journal of Finance, vol. 7, no. 1, pp. 77-91, Mar., 1952.

Shubham, J. (2018, July 6). Ensemble Learning - Bagging and Boosting. Retrieved from https://becominghuman.ai/ensemble-learning-bagging-and-boosting-d20f38be9b1e.

Vijayalakshmi Pai G. A., Metaheuristics for Portfolio Optimization- An Introduction using MATLAB, Wiley-ISTE, 2018. Retrieved from https://www.mathworks.com/academia/books/metaheuristics-for-portfolio-optimization-pai.html