Stock prediction in the S&P 500 Index based on a selection of Nikkei Index returns

Amine Laghaout

(Dated: July 1, 2016)

Executive summary: This paper presents a simple regression method for predicting the performance of the S1 stock traded in S&P 500 index based on the daily returns of the stocks S2, S3, $\cdots$, S10 traded in the Nikkei index. The relevance of the Japanese stocks in predicting S1 is quantified, showing that the two stocks that have the most impact on S1 are S5 and S6.

Keywords: S&P 500 Index, Nikkei Index, return, prediction, cross-sectional regression, toy model

Contents

I. THE REGRESSION MODEL

A. Techniques used

The present analysis was implemented in Python 3 and the `sklearn` and `pandas` packages for machine learning. Several regression techniques were tested, including regular linear regression, ridge regression, stochastic gradient descent (SGD) regression, and support vector regression. These techniques were tuned manually to different parameters and it was concluded that the best estimator was the support vector regressor with a linear kernel, a penalty parameter of 0.02, and an epsilon-SVR parameter of 0.22. A more rigorous optimization of these parameters can be obtained by a grid search (which is also included in the Python source code).

Cross-validation was performed by a cyclical folding of the training set into three sections, one of which is used for the validation. This has the advantage of effectively obtaining an average figure of merit for the score/accuracy of the estimator over the *entire* training set.

B. Assumptions

The work herein rests on two major assumptions regarding the causality of the correlation between the American stock S1 and the Japanese stocks S2, S3, $\cdots$, S10. The first is based on the fact that because the trading at the Nikkei occurs *before* the trading at the S&P 500, then the latter are necessarily reacting to the former. Although this might hold on a daily basis, it may be a naive assumption in the long run. Indeed a cursory literature review indicates that the opposite is more likely on the long run so that the Nikkei index *follows* the movements of S&P 500, as discussed by Becker and Finnerty in Refs. [1, 2] as well as in Ref. [3]. For simplicity, however, we shall maintain the assumption that the S1 reacts to the Japanese stocks on a daily basis.

The second assumption is in the same vein in that we do not treat the data set as a time series problem but rather as a simple cross-sectional sample that overlooks the chronology of the data points. This assumption, may again be naive, since economic factors are eminently time-sensitive. However, as will be shown in the next sections, the inclusion of time-dependence in the regression analysis using moving averages and compound returns, does not seem to improve the predictions. We have therefore opted to consider the data as cross-sectional.

A final assumption is that the fluctuations in currency exchange rate between the US dollar and the Japanese yen can be factored in as negligible noise.

C. Preliminary synopsis

We shall here provide a synopsis of the raw data. Table 1 shows the mean and standard deviation of the returns. One plausible insight that can be extracted from this table is that any prediction on stock $S1$ which has an error less than its standard deviation of 0.93% over the sampling period can be deemed as satisfactory. A formal expression of how this error is quantified is provided in Sec. I D 1.

stock	**S1**	S2	S3	S4	S5
mean	**0.118352**	-0.250501	0.351924	0.296563	0.453432
std. dev.	**0.930020**	1.062886	1.879886	1.137385	2.801194

stock	S6	S7	S8	S9	S10
mean	0.160605	0.278186	-0.036940	0.493267	0.009384
std. dev.	0.902715	2.013317	0.641218	1.947678	0.645085

FIG. 1: Mean and standard deviation of the various stocks over the 50 trading days of the training set.

A raw plot of the stock returns versus time is provided in Fig. 2.

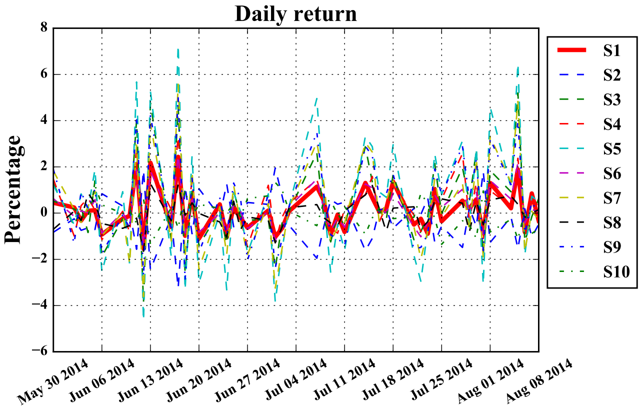

FIG. 2: Daily stock returns over the sampling period. The stock of interest, S1, is maked with a thicker red line.

In addition to the daily fluctuations in returns, it would be interesting to cumulate them so as to visualize the overall evolution of the stocks over the sampling period of $N = 50$ days. For any given stock $s \in \{S1, \cdots, S10\}$, the compound return up to trading day $i \in [1, N]$, which we shall label as $CR_i^{(s)}$, is given by

$$CR_i^{(s)} = IR^{(s)} \times \prod_{j=1}^{i} \left(1 + DR_j^{(s)}/100\right), \tag{1}$$

where $IR^{(s)} = CR_0^{(s)}$ is the initial compound return of stock s just before the period of data sampling begins. Because it is unknown, we shall set it to $IR^{(s)} = 100, \forall s$. This is done without loss of generality since we are interested in the *relative* evolution of the compound returns and we may therefore set all stocks to the same initial value of 100 arbitrary units. $DR_j^{(s)}$ is the return for some day $j \in [1, N]$ as readily provided in our data set. Note that only the daily returns DR are expressed in percentages whereas the compound return CR and initial return IR are expressed in arbitrary units (i.e., either in USD or JPY). The compound return is plotted in Fig. 3.

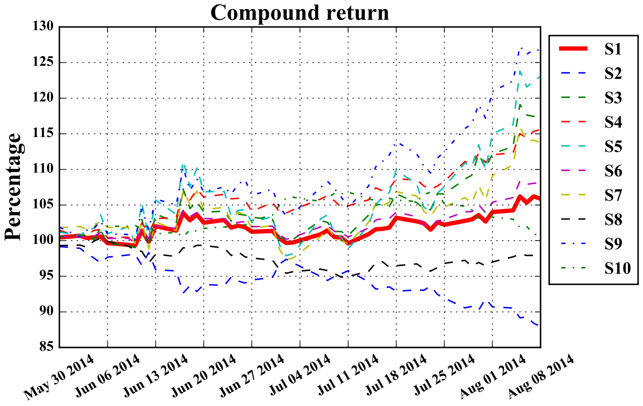

FIG. 3: Evolution of the compound returns $CR^{(s)}$ over the sampling period assuming that all stocks are initially fixed at an initial value of 100 arbitrary units. On can therefore easily read off the percentage evolution of the stocks.

D. Algorithm settings

1. *Figures of merit*

In order to assess the performance of our regression algorithm, we shall use two figures of merit. One is the standard score that is generated by the various `sklearn` methods of regression. The other, which we shall refer to as the error ϵ is obtained from the average absolute difference between the predicted and actual returns of S1. It is obtained by

$$\epsilon = \frac{1}{N} \sum_{i=1}^{N} \left| \vec{P}_i - \vec{A}_i \right|, \tag{2}$$

where $\vec{P}$ and $\vec{A}$ are the vectors of predicted and actual returns of S1, respectively. N is the number of data points. The indices i running from 1 to N label the different data points, i.e., the different vector elements in $\vec{A}$ and $\vec{P}$, and effectively enumerate the trading days.

E. Feature engineering

As discussed in Sec. I A, we have opted for a support vector regressor (SVR) with a linear kernel as it produced the optimal figures of merit (see Sec. I D 1).

1. *Feature selection*

The first step in the feature engineering is to quantify the readily available features by relevance. This is made possible by the `selectKBest` function of `sklearn`. The relevance of the different Japanese stocks in predicting S1 are plotted in Fig. 4, where we can clearly see that S4, S8, and S10 are the least relevant features and that their removal may be beneficial so as to minimize over-fitting.

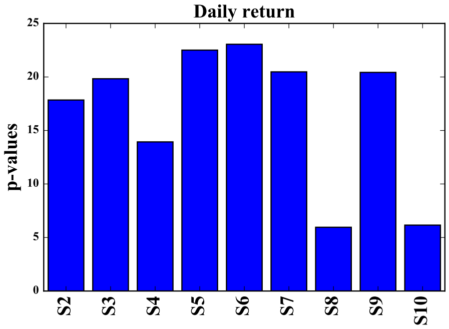

FIG. 4: Relevance of the Japanese stocks S2 $\cdots$ S10 in predicting S1 based on the p-value scoares.

In order to justify the previous statement as to how many of the low-relevance features to include in our analysis, we have plotted the regression score and the error ϵ of Eq. (2) as a function of the number k of highest-scoring features included in the regression. These plots are provided in Figs. 5 and 6, respectively.

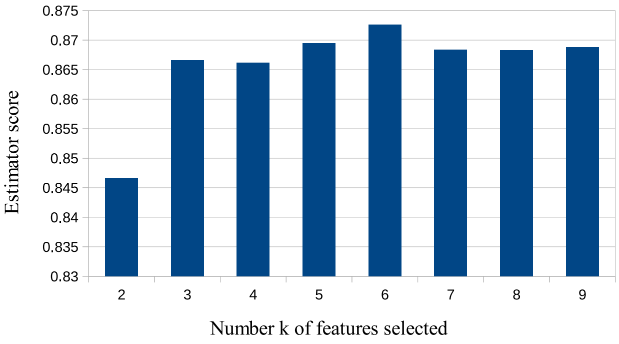

FIG. 5: Score of the regression algorithm as a function of the number k of best features included in the model.

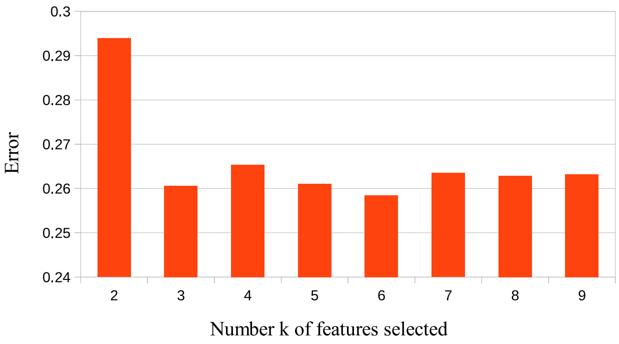

FIG. 6: Error ϵ in the predicted targets generated by the algorithm as a function of the number k of best features included in the model.

From these two figures, we conclude that a selection of $k = 6$ features from the initial nine S2 $\cdots$ S10 allows for the best performance of the regression as it leads simultaneoulsy to the highest score and smallest error. The optimal set of features consists therefore of S2, S3, S5, S6, S7, and S9.

2. Compound return

The second step in the feature engineering is the creation of new—possibly more relevant—features from the readily available ones, i.e. the daily returns. As discussed earlier, the evolution of the stocks is essentially a time series and time-dependence should therefore be factored in the features. One first way to do so is by considering the compound return already plotted in Fig. 3. A feature analysis on these compound return rates is given in Fig. 7.

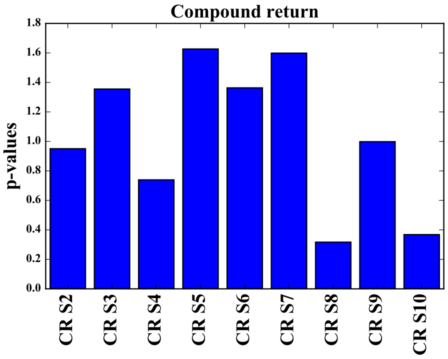

FIG. 7: Relevance of the compound return rates in predicting S1 quantified as p-values.

Although the relative ordering of these new features is similar to that of the raw return return rates, their relevance, as per their p-value, is much lower. We shall therefore discard the compound returns as viable features.

3. Exponential moving average

Another standard way to factor in time-dependence is by considering the moving average. Here, we used an exponentially-decaying moving average with a half-life of one trading day [4]. Again, the relevance of these new features, plotted in Fig. 8, though better than the compound return, is lower than that of the "raw" daily returns of Fig. 4.

We shall therefore conclude this preliminary feature engineering by keeping only the daily return rates as opposed to attempting to factor-in time-dependence.

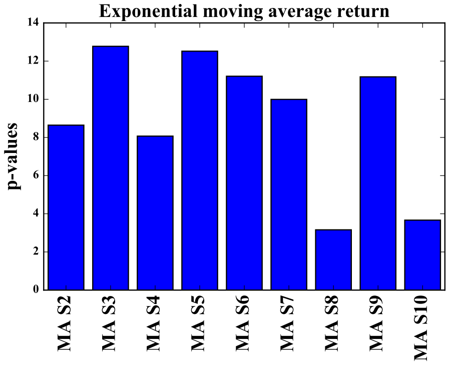

FIG. 8: Relevance of the exponentially-weighted moving average of the return rates in predicting S1 quantified as p-values.

F. Algorithm performance

To summarize, we have chosen a support vector regressor with a linear kernel, a penalty parameter of 0.02, and an epsilon-SVR parameter of 0.22. Without any further feature engineering, the six best features were identified to be S2, S3, S5, S6, S7, and S9. After implementing a cyclical cross-validation over three folds, the average figures of merit are as follows:

$$\begin{aligned} \text{Score} &: 0.8726 \\ \text{Error} &: 0.2585\% \end{aligned}$$

Note that the error of 0.2585% is less than a third of the standard deviation of the S1 stock over the entire sampling period. This is a promising indication that our model is not misbehaving.

G. Prediction

Finally, we shall visualize the predictions generated by our model by looking at the subsequent 50 trading days in the test set. The projected daily return and compound return on S1 are plotted in Figs. 9 and 10, respectively.

II. ANSWERS TO THE MACHINE LEARNING CHALLENGE

A. Which variables matter for predicting S1?

This was addressed quantitatively in Fig. 4. The two most relevant features are S6 and S5 followed with a tie of S3, S7, and S9, and then finally by S2. The three least relevant features—S4, S8, and S10—though, taken individually, exhibit some correlation with S1, are best kept out of the regression analysis, as argued in Sec. I E 1.

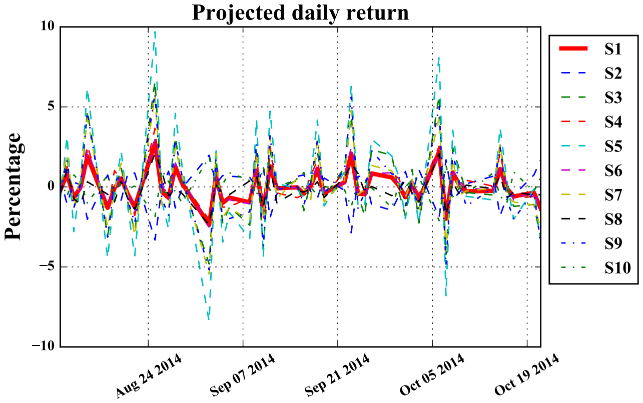

FIG. 9: Projected daily return on S1 over the period of the testing set.

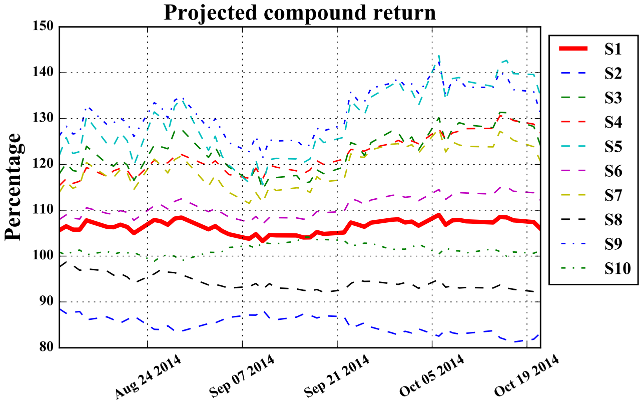

FIG. 10: Projected compound return on S1 over the period of the testing set. The initial values $IR_{50}^{(s)}$ are arranged so as to coincide with the ending values of the training period.

B. Does S1 go up or down cumulatively (on an open-to-close basis) over this period?

Over the training period, S2 goes up cumulatively by about 6% as shown in Fig. 3.

C. How much confidence do you have in your model? Why and when would it fail?

As mentioned in Sec. I B, we have treated the data as cross-sectional rather than as a time series. It may break down if the time-dependence of S1 on the *histories* of the Japanese stocks introduces a non-linearity in their relationship. Furthermore, some of the literature on the correlation between the S&P and Nikkei markets indicates that the causality operates in the direction opposite that we have adopted, making the prediction of S1 based on the Nikkei much less straightforward, if not impossible.

As for our confidence on the model, it can be summarized by the estimator score and error presented in Sec. I F.

D. What techniques did you use? Why?

Please refer to the discussion in Sec. I A.

III. RECOMMENDATIONS FOR FUTURE STUDIES

The model presented here is extremely simple and based on assumptions that may turn out to be too naive to allow for generalizations. The current work can therefore be refined by following the following recommendations:

- The history of the returns, as expressed by either the moving average or the cumulative products, was of no help in the analysis. A more elaborate *signal analysis* of the data as times series is needed since the treatment of the training data as cross-sectional is likely to be an over-simplification.

- The training set, with only 50 data points, is relatively small. A longer sampling period will be useful in reducing any over-fitting in our current model. If there does not exist a longer sampling period, synthetic data can be generated by introducing Gaussian noise with a standard deviation much smaller than the standard deviation of the returns.

- We have selected our estimator, the SVR with linear kernel, using manual methods as well as with an automated grid search of its optimal parameters. For the sake of robustness, however, multiple linear estimators could be combined using ensemble learning. This will prevent any single regressor to break down due to an idiosyncratic lack of generalization.

[1] K. G. Becker and J. E. Finnerty, *Do the Nikkei Stock Index Futures Follow the S&P 500? A Weak Form Efficiency Test* (University of Illinois at Urbana-Champaign, 1989), Faculty Working Paper No. 89-1611, URL `https://www.ideals.illinois.edu/bitstream/handle/2142/28936/donikkeistockind1611beck.pdf`.

[2] K. G. Becker, J. E. Finnerty, and M. Gupta, The Journal of Finance **45**, 12971306 (1990).

[3] B. McCormick, *The Nikkei-S&P 500 correlation* (2009), Blog Post, URL `http://www.optionmonster.com/news/article.php?page=the_nikkeisp_500_correlation_30654.html`.

[4] This was implemented with the `pandas.ewma` function of Python.