An analysis of heterogeneous ensembles at predicting stock prices of Brazilian power companies

Dennis Carnelossi Furlaneto
Federal University of Parana (UFPR)
Rua Cel. Francisco Herclito dos Santos, 100
Centro Politecnico - Jardim das Americas
81531-990 - Curitiba, PR, Brasil
dennis.furlaneto@gmail.com

Abstract

Financial time series analysis has long been a target of study for time series modeling and machine learning, in particular the application of different predictive models to classify and estimate the value for a given asset. In this work, the author analyses the performance of a heterogeneous ensemble using stacking, composed of a SVM, a Ridge regressor and a Random Forest model on the task of predicting the close value for two stocks from the Brazilian energy sector. Instead of achieving the lowest possible prediction error, the goal of this work is to compare the prediction performance of an ensemble of regressors and its base models with the objective of validating the use of ensembles on this problem.

1. Introduction

Financial time series analysis has long been a target of study for time series modeling and machine learning. These time series are non-stationary and pseudo-chaotic in nature, which means the standard deviation and mean of the series change over time, and the correlation between past and present is not necessarily high. The fact that forecasting such time series is challenging and that an accurate forecasting method would obviously bring financial gains in good amounts keep it as one of the top topics in research on time series.

More traditional models such as Autoregressive (AR), Moving Average (MA), Exponential Moving Average (EMA), Autoregressive Moving Average (ARMA), Autoregressive Integrated Moving Average (ARIMA) and Generalized Autoregressive Conditional Heteroscedasticity (GARCH) have been used and studied substantially[1], achieving the good results in a number of studies with financial time series. On the other hand, given the nature of these time series, the results of these tests vary drastically in respect to the the time period used and the source of the time series (most commonly stocks, indexes and currencies).

In the last decade, many new machine learning techniques have achieved state-of-the-art results on a number of problems on different fields, such as natural language, image analysis and power consumption prediction. Techniques such as Deep Neural Networks, SVMs, Recurrent Neural Networks and Ensembles have become popular not only for their prediction capacity, but also for the fact that they can now be actually trained in a stable, timely manner due to the increase in computation power and the advent of new training methods, such as Long-Short team memory for Recurrent Neural Networks.

Given the success they achieved in other problems, machine learning techniques have been used on financial time series prediction extensively, with a number of studies reporting results that exceed that of traditional techniques such as ARIMA and GARCH. Artificial Neural Networks have been the most popular technique to date in its many forms, generally Multi-layer Perceptron with 1 hidden layer, Recurrent Neural Networks and less often Deep Neural Networks. SVMs have also received much attention in the past years, outperforming Neural Networks in a number of studies [11]. On the other hand, other studies have shown that techniques such as Random Forests and Ridge regression have performed well on financial time series prediction, surpassing state-of-the-art techniques such as SVMs.

The lack of consistency in the results among models on different time periods and financial instruments also suggest that there might not be a single model that will perform better along a long enough time series. Ensemble methods, more specifically heterogeneous ensembles for this work, attempt to address inconsistency in accuracy by consolidating the regression or classification result of many classi-
The idea of heterogeneous ensembles is not new to the realm of computational finance; it has been used before, primarily as voting ensembles for trending prediction. This work is an analysis of a Heterogeneous Ensemble of regressors used to predict the closing prices of stocks in the Brazilian electrical sector. The author compares this ensemble with each of its parts in respect to their predictive accuracy and consistency in predictions in a time span of approximately 5 years.

The main contribution of this study is the performance analysis of a stacking heterogeneous ensemble, with a linear regressor meta-model, on the prediction of closing prices on stocks from the Brazilian stock market. To the best of this author’s knowledge, such an analysis hasn’t been done before.

2. Heterogeneous Ensembles

As stated by Reid in his review, ensembles combine the predictions of multiple base models, where these models can be trained using resampling, randomization of the training data and parameter selection methods in order to achieve heterogeneity among the models. A number of studies have shown the advantages of ensembles of classifiers when compared to their base models. Bagging, Boosting and AdaBoost are well known and popular ensemble methods, but these are techniques used for Homogeneous Ensembles; in other words, ensembles containing multiple base models of the same type (e.g: multiple Neural Networks). The key idea is that multiple models of the same kind, but with different hyper-parameters, initialized and trained in different ways can achieve better classifications and regression results when their results are combined.

Heterogeneous Ensembles, on the other hand, are ensembles created with a number of different models. The objective of these ensembles is still to combine the results of many models, but by training different types of models such as SVMs and Neural Networks on the training data and combining the results using other techniques. Many techniques for Heterogeneous Ensembles have been proposed, but in this study we will use Stacking to consolidate the results of SVMs, Random Forests and a Ridge linear model as an attempt to improve our prediction accuracy. This method was chosen for its simplicity and for the number of studies demonstrating its effectiveness.

Stacking consists of training a meta-classifier or regressor that takes as an input the predictions of the base models composing the ensemble. The meta-model learns the relationship between the models output and target value instead of initial features and target value; an example of such an ensemble would be to use another SVM as a final consolidation step for the results of base models.

In this study, we will use a simple linear regression model to consolidate the results, which effectively attribute weights to each model’s prediction in order to best fit the training data. This linear model is then transferred along with the models to the testing phasing, where this model is responsible for fitting the results from the models to the testing target.

3. Dataset

The idea for the dataset used in this study came from the work of Laboissiere et. al. whom used stocks from companies in the Brazilian energy sector. The data is composed of daily stock prices from 2 large Brazilian electrical companies: CMIG4 (Companhia Energética Minas Gerais), CESP6 (Companhia Energética de São Paulo). These stocks are part of the electrical sector component of the Bovespa index, which makes their movement correlate to the entire market movement. Furthermore, because these stocks are components of an important index, daily rates are readily available on a variety of providers on the web for an extended historical period.

The Bovespa index (BSVP), the IEE index (IEEX) and the currency conversion rate USDBRL were also used. IEEX is the index that tracks stocks of the energy sector in Brazil, and BVSP is the Brazilian stock market index. As these stocks are theoretically strongly correlated to the market and subject to international influence as they are part of market indexes and are part of an emerging market, the currency rate between the US Dollar and Reais is also used.

Daily data for each trading day from Jan-2008 to Dec-2013 composes this dataset, and for each type of financial instrument, we have a different number of features available:

- Stocks (CMIG4, CESP6): High, Low, Open, Close, Volume and Adjusted Close
- Indexes (BSVP, IEEX): High, Low, Open, Close, Volume
- Currency (USDBRL): Close

The data for these stocks and indexes was downloaded from Yahoo! Finance, and the currency rate between the US dollar and Reais was downloaded from the website of the Federal Reserve Bank of St. Louis (FRED). Both websites provide a web API that allow downloads of structured files given the name of the stock, index or currency rate.

These time series, as mentioned before, have a moving mean and standard deviation; they also contain quite a few non-linearities, which makes the prediction problem particularly challenging. The following images show the series for the stocks aforementioned just so the reader has a good picture of the dataset we will work with:
4. Estimating closing prices

Aside from the Data Acquisition phase which was previously described, the workflow built for the tests consists of another 6 distinct phases: Feature Extraction, Feature Selection, Normalization, Model Training and Stacking. (as shown in Figure 3)

The author used Python as the programming language to develop each of these phases, along three of its most popular packages: numpy, pandas and scikit-learn. Scikit-learn is definitely one of the most intuitive and complete machine learning package to date; pandas was specially useful at downloading and extracting the features from the financial time series and numpy was used for general data manipulation and normalization of the features.

The features extracted from the data were Exponential Moving Averages (EMA) with 3 different time spans: 5, 10 and 15 days

- Stocks (CMIG4, CESP6): EMA of High, Low, Open and Close
- Indexes (BSVP, IEEX): EMA of High, Low, Open, Close, Volume

Exponential Moving Averages are very popular in the financial industry, and normally used in the technical analysis of stock prices as trend indicators [3]. Such is the popularity of this indicator that studies were dedicated to analyse it as the sole predictor model of prices and indexes [12]. Each of the Exponential Moving Averages are then lagged in 1 day, which means that the features dataset and their respective target values are related as follow:

\[
\text{Close}_{t+1} = F(\text{EMA}_t(\text{Stock, Indexes, Currency}))
\]

Where \(F\) represents the model we are using to predict the closing value; \(\text{Stock, Indexes} and USDBRL\) represent all the features in each of these sets.

After the feature extraction and lagging, a correlation analysis between each of the lagged features and the target value is done. If the correlation between a given EMA and the Close price is larger than 0.5, the feature is selected. Selecting features with a high correlation coefficient is a technique that is used in a number of studies and was highlighted [7] as being effective in increasing the accuracy prediction in the prediction task on financial time series.

The features and the Close prices are then normalized using z-score, which normalizes all the values accounting not only for their distance from the mean, but also their standard deviation. The final array of values have zero mean and standard deviation of one, which has been shown to improve the training and classification/regression.

The pre-processed data is then used to train and test a SVM, a Ridge linear model and Random Forest ensemble in a walk-forward sliding window routine, described by Kaasstra and Boyd on [6]:

- Currency (USDBRL): EMA of Close
This type of test is prevalent in the financial industry and is best know as back-testing. The time series is treated as a data stream, where the testing data comes right after the training data on the time series. This routine was chosen in favour of the traditional train/test/validation routine because:

- Conventional train/test/validation division of the data points may scramble the data. The training set must only contain points in the past when compared to the test and validation sets.
- Future are generally conditioned to events in the recent past if they are not completely random.

As the last step, the predictions are stacked with a linear regression model, responsible for giving the actual regression value.

5. Testing and Results

Accounting for the 1 day lagging, which reduces the dataset by 1 data point, the number of points contained in the time series is equal to 1564, which is equivalent to 1564 days. The size of the windows for the training and the test set were set, arbitrarily, as follow:

<table>
<thead>
<tr>
<th>Table 1. Train/Test split</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Span (%) / days</strong></td>
</tr>
<tr>
<td>Training           25% (391)</td>
</tr>
<tr>
<td>Test               1.2% (20)</td>
</tr>
</tbody>
</table>

Before the full backtesting starts, the correlation analysis and feature selection take place using the first training segment as input, starting on Jan-2008 and spanning through 391 training days. The features selected at this phase are used on all the subsequent train/test steps. The features that were used for the tests, for both stocks, where:

- EMA(5 days): BSVP, Stock(High), Stock(Low), Stock(Open), Stock(Close)
- EMA(10 days): BSVP, Stock(High), Stock(Low), Stock(Open), Stock(Close)
- EMA(15 days): BSVP, Stock(High), Stock(Low), Stock(Open), Stock(Close)

For each base model in the ensemble (SVM, Random Forest and Ridge), Grid Search and 3-fold cross validation were used to optimize the parameters on each of the training and testing iterations. The predictions are then used as input to the Linear Regressor, which outputs the true regression. The following results show how well this stacking ensemble can track the true close time series for the stocks we’ve chosen:
Table 2. Train/Test results

<table>
<thead>
<tr>
<th></th>
<th>RMSE</th>
<th>SVM</th>
<th>Ridge</th>
<th>RF</th>
<th>Stacking</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMIG4</td>
<td>Mean</td>
<td>0.139</td>
<td>0.1302</td>
<td>0.2374</td>
<td>0.1263</td>
</tr>
<tr>
<td></td>
<td>Std</td>
<td>0.1013</td>
<td>0.0981</td>
<td>0.1743</td>
<td>0.0904</td>
</tr>
<tr>
<td></td>
<td>Median</td>
<td>0.1065</td>
<td>0.1027</td>
<td>0.1732</td>
<td>0.0903</td>
</tr>
<tr>
<td>CESP6</td>
<td>Mean</td>
<td>0.0881</td>
<td>0.0826</td>
<td>0.1512</td>
<td>0.0732</td>
</tr>
<tr>
<td></td>
<td>Std</td>
<td>0.0526</td>
<td>0.0505</td>
<td>0.0973</td>
<td>0.0394</td>
</tr>
<tr>
<td></td>
<td>Median</td>
<td>0.0793</td>
<td>0.0725</td>
<td>0.127</td>
<td>0.0673</td>
</tr>
</tbody>
</table>

The results on table 2 are worthy of attention. The stacking ensemble performed better than the best model, even considering that one of the models (Random Forest) performed considerably worse than SVM and Ridge regression. Furthermore, for the stock CESP6, stacking improved the mean RMSE in 11.4% and the standard deviation in 22%.

These values indicate that, aside from computational limitations such as training time and memory for storing the models (which shouldn’t be a concern for this specific ensemble), the heterogeneous ensemble performed statistically better from an accuracy standpoint. Respecting limitations aforementioned, there would be no apparent reason to use a single model in favour of the ensemble for regression.

6. Conclusion

In this work, the author proposes a framework containing a heterogeneous ensemble of a SVM, a Ridge regressor and a Random Forest to estimate the closing prices of 2 electrical sector stocks from an emerging market (Brazil). The main objective of this study was to compare the prediction performance of the heterogeneous ensemble with its base models instead of achieving state of the art prediction results.

The mean, standard deviation and median of the RMSEs computed on each of the training/testing iterations shown that the ensemble performed better than the best of its base models in respect to these indicators for both stocks; in one of the cases, the ensemble performed significantly better in respect to the mean and the standard deviation, and on both of them the median RMSE was about 10% smaller than the median RMSE of the best regressor.

These results provide encouragement for the author to improve on this work. Additions to this ongoing research would be:

- A comparison between the results of heterogeneous ensembles with different composing meta-models.
- Running the same tests on a larger body of stocks
- Test ensembles with other base models, such as Neural Networks.

References


