Carnegie Mellon University, MSCF
Machine Learning II, Spring 2021

Beating Human Analysts at Earnings Predictions

Tai Pong Lie, Jared Greenberg, Michael Djaballah

March 2021

1 Introduction

1.1 Background

The quarterly release of a company's balance sheet can make or break its valuation. It is especially important since it is the only time that new information about a company's financial health is released. One of the most important indicators is the earnings of the company, i.e., how much money was made during the quarter.

Since earnings are highly telling of the investing future of a company, they are crucial to investors. Investors and investing institutions dedicate much effort to predicting the quarterly earnings of various companies. These analysts have access to private and proprietary data, models, and estimation methods.

1.2 The Original Paper

The paper that we are basing our work on is "Beating Human Analysts in Nowcasting Corporate Earnings by using Publicly Available Stock Price and Correlation Features" by Kamp, Boley, and Gärtner.

The original paper, in short, aims to beat human analyst estimations using publicly available data. The difficulty of this task, referenced in the abstract and introduction, is very high. The paper describes it: "...automatized methods based on classic econometric and statistical approaches fail to reach the quality of human experts by orders of magnitude."[1] This leads the authors to use a machine learning algorithm, ridge regression, within a nowcasting framework. Nowcasting is the prediction of current or very near-term data that is inaccessible at the time of prediction. They are usually subject to a delay in release due to audit reasons. Thus, being able to conduct nowcasting on earnings using publicly available data, namely just stock prices, would be an ideal method for the independent investor who does not have access to private or subscription information.

Using ridge, the feature space was:

- Daily prices of the target stock
- Daily prices of a set of other stocks
- Moving averages of the target stock
- Moving averages of the set of other stocks, weighted by correlations to the target stock

Kamp, Boley, and Gärtner used S&P 100 index components during the period from 2008 to 2012 for testing the model. 100 randomly chosen stocks from 2004 to 2006 were used to cross validate and tune two parameters.

The parameters were:

- Length of look back window in days: 11, 50, 125, 200, 250, 350, 500, 700
- Regularization penalty for ridge: 10, 100, 1000, 3000, 5000, 10000

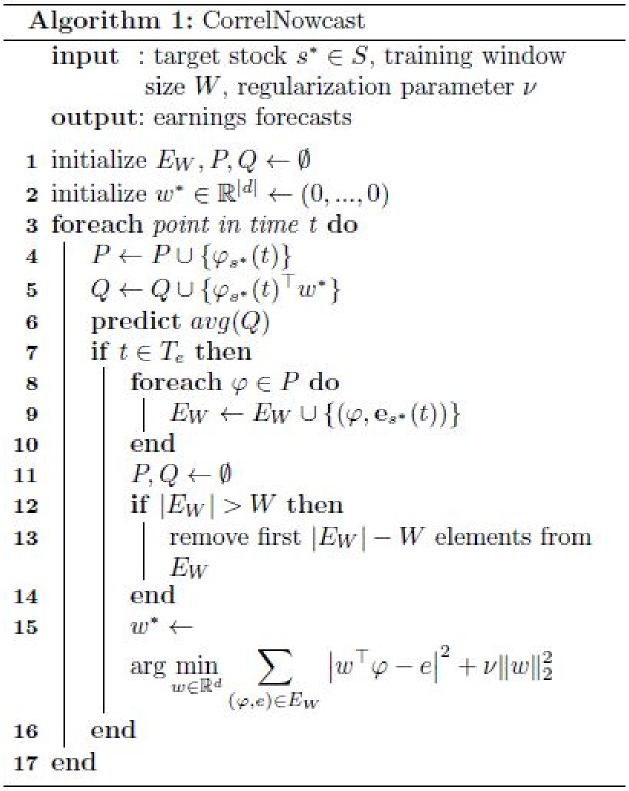
The goal of our investigation is to verify or refute the claims made in the original paper, as well as explore the limits of nowcasting within the predetermined feature space.

2 Methods

We will be using the same framework as the original paper to ensure that we are testing the original hypothesis. This means using the same parameters available for tuning, the same features, and the same base model, with the exception of the look back window lengths 500 and 700, which will not be used due to limited data.

The pseudo-code (from the original paper)[1] of the authors' CorrelNowcast algorithm is include on the next page with line references. A few key properties of their algorithm is as follows:

- The algorithm assumes cross validation is completed for the look back window length and the regularization parameter for the ridge, and takes these as input
- After an earnings result is announced, the rows of data starting from 1 day after the last announcement date to the current announcement date is added to the training data and the ridge is retrained (see lines 8-10)
- Size of the training data is always kept to be equal to the look back window length W, so when new rows are added, old rows are dropped (see lines 11-14)
- In between announcement dates, a prediction from the ridge is averaged with all the predictions after the last announcement date to give the final prediction for a row (see line 6)



```
Algorithm 1: CorrelNowcast
input  : target stock $s^* \in S$, training window
         size $W$, regularization parameter $\nu$
output: earnings forecasts
initialize $E_W, P, Q \leftarrow \emptyset$
initialize $w^* \in \mathbb{R}^{|d|} \leftarrow (0, ..., 0)$
foreach point in time $t$ do
    $P \leftarrow P \cup \{\varphi_{s^*}(t)\}$
    $Q \leftarrow Q \cup \{\varphi_{s^*}(t)^\top w^*\}$
    predict $avg(Q)$
    if $t \in T_e$ then
        foreach $\varphi \in P$ do
            $E_W \leftarrow E_W \cup \{(\varphi, \mathbf{e}_{s^*}(t))\}$
        end
        $P, Q \leftarrow \emptyset$
        if $|E_W| > W$ then
            remove first $|E_W| - W$ elements from
            $E_W$
        end
        $w^* \leftarrow$
        $\arg\min_{w \in \mathbb{R}^d} \sum_{(\varphi,e) \in E_W} \left|w^\top \varphi - e\right|^2 + \nu \|w\|_2^2$
    end
end
```

To test the limits of the feature space, several other models were also tested on the data. First, instead of using the cross validation method in the paper to choose a penalization parameter value, a ridge with built-in cross validation for the penalization parameter was used. Since the largest penalization parameter was discovered to be optimal, a LASSO model with built in cross validation for its penalization parameter was used to see the effect of heavier regularization. To check more flexible models, a random forest and gradient boosted trees were checked as well. These models were checked with the look back window resulting from original cross validation, as a single run of the CV models took almost 2 hours.

3 Data

All data was sourced from Bloomberg, namely closing prices for the stocks via the PX_LAST field, and earnings data via EARN_ANN_DT_TIME_HIST_WITH_EPS, which contains multiple fields of data. In particular, the EPS data is GAAP adjusted and the analysts' consensus estimate that came along was adjusted by Bloomberg to be comparable with that particular EPS measure. Same as the paper, we sought for data on all the stocks in the S&P 100 index. The data pull actually yielded 101 stocks. After dropping stocks with NaN values in our interested period, we ended up with 88 stocks. This set was used for cross validation and testing.

In order to test whether the authors' claim holds in a more recent period, we used we used data from February 2013 to February 2015 for tuning, and data from March 2015 to March 2019 for testing.

Another limit of our data was that the date of the publishing of the analyst estimates were not known, so we do not get as good of a cross section of the estimate timeline as the original paper. Analysts also do not publish all of their estimates on the same day, so our way is more correct by using the analyst consensus and comparing to the day of estimate from our algorithm.

4 Results

Cross validation results (mean relative error):

Sequence length		**11**	50	125	200	250	350
Penalization	10	0.2188	0.2291	0.3120	0.3826	0.3305	0.3706
	100	0.2185	0.2256	0.2666	0.2641	0.2704	0.2978
	1,000	0.2176	0.2207	0.2374	0.2257	0.2344	0.2540
	3,000	0.2171	0.2195	0.2287	0.2226	0.2264	0.2412
	5,000	0.2169	0.2190	0.2257	0.2224	0.2249	0.2370
	10,000	**0.2167**	0.2185	0.2232	0.2227	0.2246	0.2330

Test results (mean relative error):

Model prediction: 0.2369

Analyst consensus prediction: 0.1155

Additional Model Test Performance:

Model	Mean Relative Error
RidgeCV	0.2368
LassoCV	0.2369
Random Forest	0.2368
Gradient Booster	0.2367

5 Discussion

The MRE from model predictions was actually higher than the MRE from analyst consensus predictions. This does not support the paper's claim.

Even without including the cases W = 500 and W = 750 in our cross validation, the cross validation still took around 6 hours to run. We found that the code's greatest bottle neck was fitting Sci-Kit Learn's Ridge regression (and other models), so making the code faster would be difficult without switching to a faster language or having a faster implementation of ridge. In any case, the cross validation process was computationally intensive since we had 36 parameter combinations. Each run involves 88 stocks over a two year period, with refitting of the ridge regressor each time there was an earnings announcement, for each stock. The run on the test set took much less time since it was only one run that involved 88 stocks over a four year period.

It was interesting to find that cross validation yielded the combination of shortest look back window (W = 11) and the largest regularization (ν = 10000) to be the best. The fact that the best look back window was the shortest suggested that the model had the most predicting power when predictions were based only on moving averages over a short period of recent data. The fact that the best regularization parameter was the largest was likely to address the fact that we have 352 dimensions in the feature space but only 541 rows of data in the cross validation (for the two years). So naturally, high regularization was needed to cut down the variance.

This was the reasoning in adding self-tuning models from Sci-Kit Learn, perhaps the papers penalization tuning did not go high enough. The self-tuning Ridge model barely outperformed the hand tuned version, and the LASSO version performed about on par. The best test performance was found in the Gradient Booster, but this outcome may not have been significant. Interesting enough, all final models had nearly the same mean relative error, meaning that there may be limits to the predictive power of price movements for earnings.

6 Conclusion

Our work failed to replicate the results of the initial paper. However, the fact that the method was not as far-reaching as the authors hoped is not the same as saying it was useless. The method, while poorer than the analyst consensus, can still be used as a tool to aid analysts themselves. The method can also be used as a (higher variance) predictor of earning reports available outside of when analysts release their estimates. However, as noted above, the method appears to work best when using the most shortest back look window, so predicting earning reports from further out will add more error to predictions.

Future research may include more searching around parameter space to allow for more accurate predictions. Using this method as a part of a larger predictor, such as through stacking, may allow it to be better used.

We also note that as the cross-validation found only the shortest sequence length to be most accurate, a large amount of information is contained within the last 11 days of trading. This may imply a certain amount of "data leakage" from right before earning reports are released, either in the form of traders having more confidence in analyst predictions, or through insider trading. Future research could shed more light on the reasons for this.

7 Appendix

Link to original paper:

`http://www.ferari-project.eu/wp-content/uploads/2014/12/earningsPrediction.pdf`

References

[1] Michael Kamp, Mario Boley, Thomas Gärtner. *Beating Human Analysts in Nowcasting Corporate Earnings by using Publicly Available Stock Price and Correlation Features.* Siam International Conference on Data Mining (SDM'14), 2014.