Measuring the Efficacy of Financial Ratios in Predicting Industry Weighted Returns

Kinbert Chou, Caroline Zhang, Eva Zhang
{klchou, czhang21, evazhang}@stanford.edu

1 Introduction

With recent advancements in financial technologies, algorithmic and high-frequency trading have grown in relevance. Many current methods in literature center on short-term capital investment strategies — this project instead explores the role of financial ratios in fundamental financial research and investigates their comparative efficacy in capturing macro-industry trends, in hopes of adding to the understanding of long-term investment analysis and limits of financial ratios in capturing industry trends for fundamental analysts.

1.1 Motivation

Accuracies in capturing industry trends over time were compared across the 3 different models that were used to predict on feature ratios. Specifically, we hoped to provide insight into success of 74 ratio features which fell within the 7 categories in the WRDS dataset: Capitalization, Efficiency, Liquidity, Solvency, Profitability, and Valuation.

1.2 Problem Formulation

The input to our algorithm is a continuous set of date, industry category, and 74 financial ratios falling into the aforementioned categories, with additional features indicating time and integer-encoded Fama-French 49-industry classified labels. For our final regression approaches (Elastic Net, Neural network regression, Ridge regression), models outputted predicted industry returns (% YOY change). We also experimented with Gaussian processes, Lasso regression, and a baseline linear regression model with the same input and output format. Next, for our classification approaches, the same financial ratios were used to output predicted industry returns binned uniformly in 8 categories from values in -1 to 1.

2 Related Work

There exists a large body of work analyzing financial ratios as a predictor of stock prices, mostly with focus on dividend yield, book-to-market ratio, earning yield or earnings price, and capital gain. Much of the work has been done in economics with linear regression and statistical confidence intervals, and recently with neural networks. However, there exist conflicting results on ratio efficacy —dividend yield [4], book-to-market[5], and capital gain [7] have all been cited as the most effective predictor. Our work takes a more comprehensive look at 70 different ratios, concentrating on contributing to fundamental analysis and industry valuation, which have been relatively overlooked, rather than pure technical analysis of stock patterns.

Existing techniques include ordinary least squares(OLS) regression, multiple regression predictive model [5], and stepwise multiple regression [7], but there is relatively limited previous machine learning work done with financial ratios: one recent study forecasts returns with historical return data, and obtains more optimal results on OLS with LASSO[6]. Neural networks have been explored to a small limited extent[9]; in one study, they performed worse in predicting industry rankings. From stock price prediction problems, backpropagation neural networks are proven to robustly outperform other models [10]. In another study, clustering was found to help determine ratio relationships[8]. Noticeably, there has been a lack of implementation classification techniques; we plan to explore these further as well as build upon the existing regression methodology for comparison.

3 Dataset and Features

We have 20577 valid examples occurring in monthly increments from 1970 January to 2017 February with date, industry code, and 74 financial ratio in addition to date, firm categories, and number of firms. We performed 80:20 split for training and testing, and 80:20 for training and dev within the 80% of data (split by time, sequential order). We obtained our data set from Wharton Research Database Services, utilizing the Graduate School of Business authorization portal. [1]

3.1 Data Preprocessing

We have preprocessed the data to remove any examples which include the absence of certain ratios (NAs), integer label-encoded industry category feature FFI49desc (industry categories). With regression models, output values were chosen to be Value-Weighted Industry Returns (by market capitalization) instead of Equally-Weighted Industry Returns in their original forms. With classification models, Value-Weighted Industry Returns were integer value-encoded into 8 uniform bins ranging from -1 to 1. The majority of industry returns fell between the range of less than 0.5 and greater than -0.5, as expected from the lack of drastic monthly changes, with a median value of 0.00109. Below are a few examples of our data set, with a subset of features displayed.

public_date	FFI49_desc	NFIRM	dpr_Median	PEG_trailing_Median	bm_Median	CAPEI_Median	divyield_Median
19931130	25	5	0.479	3.865	0.401	19.861	0.0273
19931231	25	5	0.479	4.122	0.401	21.179	0.0255
19940131	25	5	0.479	4.190	0.401	21.483	0.0212
19940228	25	6	0.394	11.991	0.406	22.148	0.0234
19940331	25	6	0.394	11.971	0.406	20.340	0.0233

Figure 1: A subset of features shown in our training data set.

We chose not to normalize ratios with non-percentage based formulas to preserve their original fundamental financial significance. In our milestone, we performed a naive f-test on our features, and found that median dividend yield was most indicative of the final industry equally-weighted value returns. After experimenting with feature selection and ablative analysis, features were kept in their original format for the final prediction. To capture potentially information-rich historical patterns, features and labels of the previous 1-year-rolling window were included for experiment in the regression models, but were not used in the ultimate result. For instance, one row with 70 financial ratios would instead have 12×70 worth of ratio features for prediction in the same month. In addition, the impact of significant financial world events on model performance was also considered. Before experimentation, we hypothesized that 2008 and subsequent years may have a fundamental difference in the comparative efficacy of financial ratios with industry output than that found in prior years. After experimentation, no long-lasting change in the efficacy of financial ratios in prediction industry returns were found.

[1] A link to our features and their corresponding significance can be found here. https://wrds-www-wharton-upenn-edu.stanford.idm.oclc.org/documents/793/WRDS_Industry_Financial_Ratio_Manual.pdf

4 Methods

Using methods from lecture in addition to the milestone feedback, our team experimented with both regression and classification approaches. Below are 3 of our final models that were used in evaluating financial ratio efficacy in capturing industry trends. A baseline model made with binary classification and linear regression was explored in the milestone, and will largely be discussed in performance assessment in the Results section.

4.1 Classification: Support Vector Machines

The SVM algorithm we used was derived from class, from the dual optimization objective as shown

$$\begin{aligned} \max_\alpha \quad & W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} y^{(i)} y^{(j)} \alpha_i \alpha_j \langle x^{(i)}, x^{(j)} \rangle. \\ \text{s.t.} \quad & \alpha_i \geq 0, \quad i = 1, \ldots, m \\ & \sum_{i=1}^{m} \alpha_i y^{(i)} = 0, \end{aligned}$$

in class:

We explored both OvR (one-vs-rest) and OvO (one-vs-one) methods for prediction on our 8-binned industry returns. Although OvR is computationally more efficient and does not produce $\frac{K(K1)}{2}$ for K classes, it ignores interclass dependence, and adds to an imbalance in data classes by pooling negative examples when making predictions. However, for our purposes, OvR suffices and had no significant performance gaps in comparison with OvO. Since our data was mostly distributed between the -0.5 to 0.5 range which meant the imbalance in negative example would not be accentuated by OvR, OvO made no significant improvements over OvR in accuracy during performance. The OvR algorithm can be seen as defined below, where each prediction is made from $\hat{y} = \underset{k \in \{1 \ldots K\}}{\text{argmax}} \; f_k(x)$

for $k \in \{1, ...K\}$ classes **do**
  z = new label vector
  if $y_i = k$ **then**
    $z_i \leftarrow 1$
  else
    $i \leftarrow 0$
  end if
  apply binary classifier algorithm to X, z, obtain f_k

Next, we explored Representer Function spaces as presented in the representer function lecture notes from class. We used a baseline linear kernel, explored polynomial kernels of different degrees, and finally trained with a Radial basis function kernel. The RBF, linear, and polynomial kernels are defined respectively by: $K(x,z) = Exp(\frac{-1}{2\tau^2} \|x - z\|_2^2)$,$K(x,z) = x \cdot z$, $K(x,z) = (x \cdot z + 1)^d$.

Kernel tricks, as shown in class, are used to work with non-linearly separable data, and provide computationally cheaper results. Further discussion of kernel choices and impact on performance can be found in the Results section.

4.2 Regression & Classification: Neural Network

Both regression on original value-weighted industry returns and classification on the binned returns were performed. Our model architecture for regression can be seen in the below diagram. 5 hidden layers were used in the classifier network, where a sigmoid input layer was followed by softmax, linear, and sigmoid hidden layers, finally yielding output from a softmax layer. Sparse categorical cross-entropy loss was used for the classifier network, as according to the integer-encoded labels of the uniform 8 bins. The regression network had 3 hidden layers, with Relu for the first 2 hidden layers, and finally a linear layer. The Adaptive Moment Estimation (Adam) optimizer and Cross-Entropy loss $-\Sigma_{j=1}^{k} y_j log(\hat{y}_j)$ as discussed in lecture were used in the regression network.

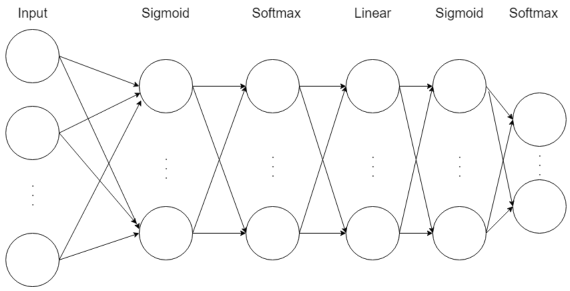


Figure 2: Neural Net Classifier Architecture

4.3 Regression: Ridge, Lasso, & Elastic Net

We had several variations of regression with different regularization parameters and implemented by minimizing the cost function. Lasso optimization problem can be viewed as $\min \|Y - X\theta\|_2{}^2 + \lambda\|\theta\|_1$. Ridge optimization problem can be viewed as L_2 regularization, and can be posed as $\min \|Y - X\theta\|_2{}^2 + \lambda\|\theta\|_2{}^2$. Finally, the Elastic net optimization problem can be posed as a mixture of L_1 and L_2 regularization:

$$\min_{\beta} \|\vec{y} - X\beta\|^2 + \lambda[\alpha\|\beta\|_2{}^2 + (1-\alpha)\|\beta\|_1]$$

5 Experiment Results & Discussion

In addition to the 3 main aforementioned models, experiments were also performed on baseline Ordinary Least Squares regression, and Gaussian Processes as suggested in the milestone.

5.1 Results

Respective accuracies for classification models and mean-squared errors for regression models can be found in the figures below. Accuracy was calculated by $\frac{TP + TN}{TP+TN+FP+FN}$, whereas mean-squared error follows the equation of $\frac{1}{n}\Sigma_{i=1}^{n}(Y_i - \hat{Y}_i)$, where Y_i, $\hat{Y}_i$ respectively represent the true and predicted industry return, and *TP*, *TN*, *FP*, *FN* stand for the respective probabilities in a confusion matrix. A sample Precision-Recall curve for the RBF SVM can be found in the figure in the next page. All the SVM results included apply for both OvO and OvR experiments.

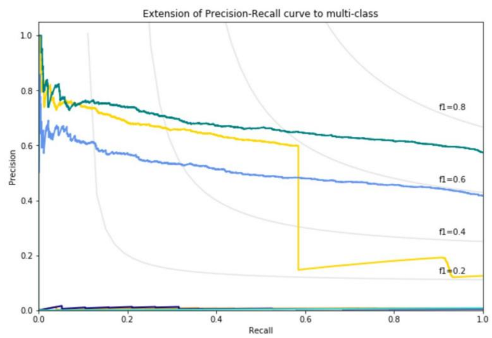
Table 1: Classification Accuracy and Regression Mean-Squared Error

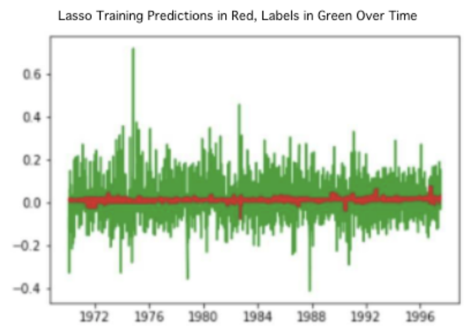
Classification Accuracy					Regression MSE				
Model	Size	Training	Size	Test	Model	Size	Training	Size	Test
SVM RBF	13168	0.6520	4116	0.5958					
SVM Poly	13168	0.6174	4116	0.5677	OLS	13168	0.00454	4116	0.00738
SVM Linear	13168	0.5805	4116	0.5694	Ridge	13168	0.00454	4116	0.007403
SVM Sigmoid	13168	0.5234	4116	0.5222	Elastic Net	13168	0.00456	4116	0.00736
NN	13168	0.5642	4115	0.614	NN	13168	0.0047	4116	0.0692

5.2 Discussion

Among the classification methods, neural network performed the best, achieving 0.614 classification accuracy on test data, with the use of sparse categorical entropy loss for integer-encoded labels. We performed hyper-parameter grid searches with learning rates and batch-sizes, and utilized a learning rate of 0.02 in our final model. Activation neurons were also tuned and selected based on preliminary experiment results and function: more specifically, our final model incorporated a sigmoid, softmax, and linear layer, which structure intuitively reasonably captures the sign of the industry return change, Around the same margin of accuracy was achieved in training and testing, and our concerns on overfitting and imbalanced class issues did not arise to the same extent as expected. We suspect that the test data performed marginally better the training data since some of the financial trends and macro-industry trends over time in the training data set were harder to capture and more noisy than the smaller test data set.

SVMs for OvR and OvO achieved comparative around the same accuracy, which did not come as a surprise as mentioned before: most of the data fall between the 0.5 to -0.5 range, which means that the benefit OvO has on correcting negative class imbalance had less of an impact on more equally distributed label classes.





Among the regression methods, Elastic Net did the best: from experimentation, we found that ElasticNet performed better than Lasso and Ridge. Since some financial ratios, in preliminary analysis in the milestone, were found to not have significant contributions to performance in modelling industry trends, it is reasonable that ElasticNet, a compromise between Ridge and Lasso, would perform better than the two models. According to lecture and Elements of Statistical Learning, elastic-net selects variables like the lasso, and "shrinks together the coefficients of correlated predictors like ridge", in addition to holding performance advantages over L_q penalties. [2] Comparing lasso and ridge performance, while lasso serves both a regularization and selection method, we observe that Lasso's overt penalization of high coefficient β lead to predictions of 0 without any usage of industry trends, as per the figure above.

6 Conclusion

Overall, neural network classification achieved the highest performance accuracy: neural networks were able to produce a marginally better result than SVMs in classifying industry return change between months. Due to the minor change between monthly industry trends, regression methods which had heavy penalization and no movement between month to month performed the best, yet it is not necessarily true these methods utilized insight to make predictions from past trends and ratios.

Given more resources and time, we would explore more with data normalization and neural network construction. With more time, advanced data preprocessing and selection can be performed: currently, ratios exist in their original form to preserve their fundamental financial significance, without removal of possibly noisy features. Identifying noisy and unnecessary variables, in addition to adding advanced forms of historical and cumulative insight could increase accuracy in both regression and classification. Finally, more in-depth investigation of neural network architecture fit for financial ratios could be performed and experimented with, both for regression and classification, to gain more insight into ratio efficacy in feature selection. Our codebase for this project can be accessed at: `https://github.com/kl-chou/CS-229-Project`.

[2] Hastie, T., Tibshirani, R.,, Friedman, J. (2001). The Elements of Statistical Learning. New York, NY, USA: Springer New York Inc.

7 Contributions

As a group, we contributed equally to the entire project. Eva worked additionally on preprocessing data, coding the classification algorithms, and writing milestone and final reports. Kinbert preprocessed data, coded the regression algorithms, improved data processes, and generated diagrams. Caroline worked on coding neural networks, reviewing background literature, and making the poster and reports.

8 References

[1] Alexander, J.A. & Mozer, M.C. (1995) Template-based algorithms for connectionist rule extraction. In G. Tesauro, D.S. Touretzky and T.K. Leen (eds.), *Advances in Neural Information Processing Systems 7*, pp. 609–616. Cambridge, MA: MIT Press.

[2] Bower, J.M. & Beeman, D. (1995) *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System.* New York: TELOS/Springer–Verlag.

[3] Hasselmo, M.E., Schnell, E. & Barkai, E. (1995) Dynamics of learning and recall at excitatory recurrent synapses and cholinergic modulation in rat hippocampal region CA3. *Journal of Neuroscience* **15**(7):5249-5262.

[4] Lewellen J. (2004), “Predicting Returns with Financial Ratios”, *Journal of Financial Economics*, 74, 209-235.

[5] Kheradyar, S., Ibrahim, I., & Nor, F.M. (2011) Stock Return Predictability with Financial Ratios. *International Journal of Trade, Economics and Finance,* 2(5): 1-6.

[6] Rapach, David and Strauss, Jack and Tu, Jun and Zhou, Guofu, Industry Return Predictability: A Machine Learning Approach (October 8, 2018). Available at SSRN: https://ssrn.com/abstract=3120110 or http://dx.doi.org/10.2139/ssrn.3120110.

[7] Karami, G. R. and Talaeei, L. (2013). Predictability of stock returns using financial ratios in the companies listed in Tehran Stock Exchange, *International Research Journal of Applied and Basic Sciences* 5(3): 360–372.

[8] Tupe, Tom, "Financial Ratio Analysis For Stock Price Movement Prediction Using Hybrid Clustering" (2014). Master’s Projects. 377.

[9] Vertes, Druce (2018). "Machine Learning for Financial Market Prediction – Time Series Prediction with Sklearn and Keras." Retrieved from https://alphaarchitect.com

[10] Song, Yue-Gang & Zhou, Yu-Long Han, Ren-Jie. (2018). Neural networks for stock price prediction.

[11] Hastie, T., Tibshirani, R.,, Friedman, J. (2001). The Elements of Statistical Learning. New York, NY, USA:Springer New York Inc