Evaluation of Machine Learning in Finance

Ze Yu Zhong

Supervisor: David Frazier

Monash University

Main Motivation

To evaluate the application of machine learning to predicting financial asset returns, with specific regard to how they deal with the unique challenges present in financial data.

Background

- Factors : a collection of regressors to be used in pricing returns that can be used to proxy for unknown underlying risk factors due to their correlation with cross sectional returns, (Harvey et al., 2016)
- Contrasts with the strict view that risk factors should be variables that have unpredictable variation through time, and be able to explain cross sectional returns independently

Background

- Early Example - Dividend Ratio
- Can be *persistent* (Goetzmann and Jorion (1993), Ang and Bekaert (2006))
- Dividend Ratios are correlated with lagged dependent variables on the right hand side of the regression equation.
- Violates the assumptions of independent regressors required for OLS: t stats which are biased upwards and increase with time horizon due to autocorrelated errors
- Corrections to t statistics using the GMM and NW errors are also shown to be biased, (Goetzmann and Jorion, 1993)

Background

- Traditionally, dividend ratios have been included because they have shown good in sample performance, particularly in the 1990s (Goyal and Welch, 2003)
- Proved to be not robust and have poor out of sample performance, especially since 2000s (Goyal and Welch (2003), Lettau and Ludvigson (2001), Schwert (2003))

Background

- Factors such as dividend ratios, earnings price ratio, interest and inflation etc. were “widely accepted” able to predict excess returns, (Lettau and Ludvigson, 2001)
- Welch and Goyal (2008) conclude that not a single variable had any statistical forecasting power, and the significance values of some factors change with the choice of sample periods.

Background

- More factors produced by literature: currently over 600 documented (Harvey and Liu, 2019)
- False discovery problem, (Harvey et al., 2016)
- Factors are cross sectionally correlated - inefficient covariances, factors may be subsumed within others, (Feng et al., 2019)

What is Machine Learning?

Hastie et al. (2009) define in *An Introduction to Statistical Learning* as a vast set of tools for understanding data.

We will define it as a diverse collection of:

- high dimensional models for statistical prediction,
- "regularization" methods for model selection and mitigation of overfitting in sample data
- efficient systematic methods for searching potential model specifications

Why apply Machine Learning in Finance?

- High dimensional - more flexible than traditional regression models, which make strong functional form assumptions and are sensitive to outliers, (Freyberger et al., 2017)
- Explicit methods for guarding against overfitting and generalizing poorly
- Methods to produce an optimal model from all possible at manageable computation cost

Applications in the Literature

- Kozak et al. (2017), Rapach and Zhou (2013), Freyberger et al. (2017), among others have applied shrinkage and selection methods to identify important factors
- Gu et al. (2018), Feng et al. (2018), among others have constructed machine learning portfolios that historically outperform traditional portfolios in terms of prediction error and predictive R^2
- Attribute their success to machine learning's ability to find non-linear interactions

Motivations

However, little work has been done on how machine learning actually recognises and deals with the challenges in financial data.

- Feng et al. (2018) cross validates their training set, destroying temporal aspect of data, and only explore a handful of factors
- Gu et al. (2018) only use data up until the 1970s to produce predictions in the last 30 years
- Gu et al. (2018)'s models do not have entirely consistent importance metrics - only their tree based methods recognise dividend yield as important

Motivations

- This paper will be the first to explore how machine learning performs in environments similar to financial data, with particular focus on how they deal with the challenges in financial data, acting as an extension to the simulation work of Gu et al. (2018).
- Models will also be evaluated again, but with more recent, representative financial data to explore robustness.

Model Overview

Returns are modelled as an additive error model

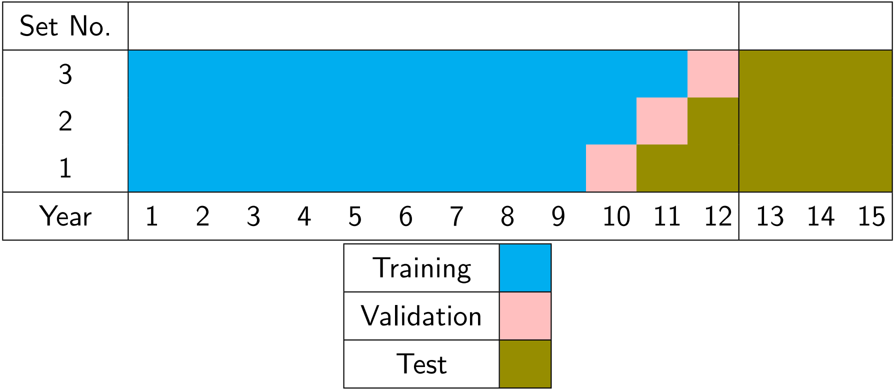
$$r_{i,t+1} = E(r_{i,t+1}|\mathcal{F}_t) + \epsilon_{i,t+1} \tag{1}$$

where

$$E(r_{i,t+1}|\mathcal{F}_t) = g^*(z_{i,t}) \tag{2}$$

Stocks are indexed as $i = 1, \ldots, N$ and months by $t = 1, \ldots, T$. $g^*(z_{i,t})$ represents the model approximation using the P dimensional predictor set $z_{i,t}$.

Sample Splitting



Set No.															
3	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟥	🟫	🟫	🟫
2	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟥	🟫	🟫	🟫	🟫
1	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟥	🟫	🟫	🟫	🟫	🟫
Year	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Training	🟦
Validation	🟥
Test	🟫

Figure 1: Sample Splitting Procedure

Loss Functions

Mean Absolute Error (MAE)

$$\text{MAE} = \frac{1}{n}\sum_{j=i}^{n} |y_j - \hat{y}_j| \tag{3}$$

Mean Squared Error (MSE)

$$\text{MSE} = \frac{1}{n}\sum_{j=i}^{n} (y_j - \hat{y}_j)^2 \tag{4}$$

Models Considered

- Linear Models
- Penalized Linear Models (Elastic Net)
- Random Forests
- Neural Networks

Linear Models

Linear Models assume that the underlying conditional expectation $g^*(z_{i,t})$ can be modelled as a linear function of the predictors and the parameter vector θ:

$$g(z_{i,t};\theta) = z_{i,t}'\theta \tag{5}$$

Optimizing θ with respect to minimizing MSE yields the Pooled OLS estimator

Penalized Linear Models

Penalized linear models have the same underlying statistical model as simple linear models, add a new penalty term in the loss function:

$$\mathcal{L}(\theta; .) = \underbrace{\mathcal{L}(\theta)}_{\text{Loss Function}} + \underbrace{\phi(\theta; .)}_{\text{Penalty Term}} \tag{6}$$

Focus on the popular "elastic net" penalty (Zou and Hastie, 2005), which takes the form for the penalty function $\phi(\theta; .)$:

$$\phi(\theta; \lambda, \rho) = \lambda(1-\rho) \sum_{j=1}^{P} |\theta_j| + \frac{1}{2}\lambda\rho \sum_{j=1}^{P} \theta_j^2 \tag{7}$$

Regression Trees & Random Forests

- Fully non-parametric models that can capture complex multi-way interactions.
- A tree "grows" in a series of iterations:
    1. Make a split ("branch") along one predictor, such that it is the best split available at that stage with respect to minimizing the loss function
    2. Repeat until each observation is its own node, or until the stopping criterion is met
- Slices the predictor space into rectangular partitions, and predicts the unknown function $g^*(z_{i,t})$ with the "average" value of the outcome variable in each partition to minimize the loss function

Random Forests

Trees have very low bias and high variance

They are very prone to overfitting and non-robust

Random Forests were proposed by Breiman (2001) to address this

- Create *B* bootstrap samples
- Grow a highly overfit tree to each, but only using *m* random subset of all predictors for each
- Average the output from all trees as an ensemble model

Neural Networks

Feed forward neural networks consist of an “input layer” of scaled data inputs, “hidden layers” which interact and non-linearly transform the inputs, and finally an “output layer” that aggregates the hidden layers and transform them a final time for the final output.

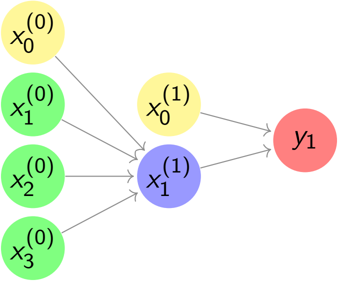


Figure 2: Sample Neural Network

Neural Network Specifications

- Neural networks with up to 5 hidden layers were considered.
- The number of neurons is each layer determined by geometric pyramid rule (Masters, 1993)
- All units are fully connected

ReLU activation function was chosen for all hidden layers for computational speed, and hence popularity in literature:

$$\text{ReLU}(x) = max(0, x) \tag{8}$$

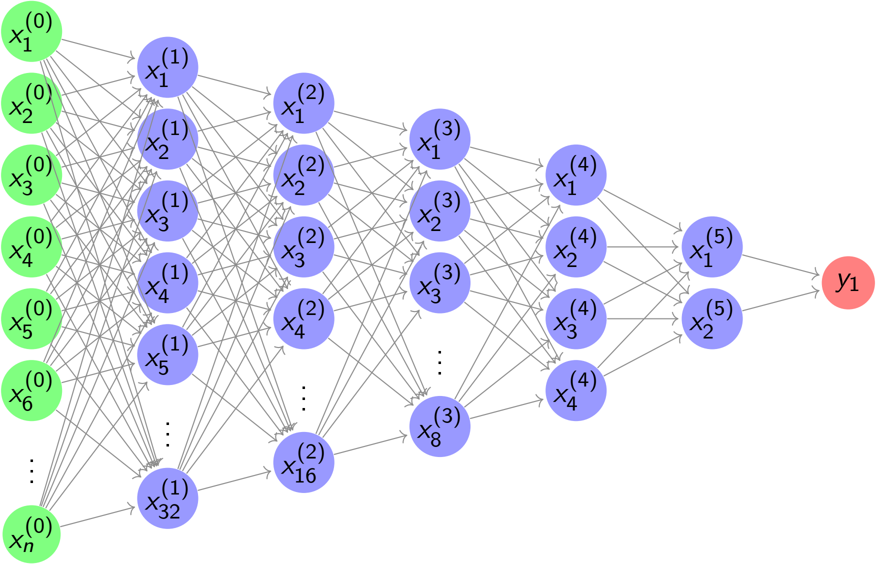


Figure 3: Neural Network 5 (most complex considered)

Overall Simulation Design

Simulate a latent factor model with stochastic volatility for excess return, r_{t+1}, for $t = 1, \ldots, T$:

$$r_{i,t+1} = g\left(z_{i,t}\right) + \beta_{i,t+1} v_{t+1} + e_{i,t+1}; \tag{9}$$

$$z_{i,t} = \left(1, x_t\right)' \otimes c_{i,t}; \quad \beta_{i,t} = \left(c_{i1,t}, c_{i2,t}, c_{i3,t}\right); \tag{10}$$

$$e_{i,t+1} = \exp\left(\sigma^2_{i,t+1}\right) \varepsilon_{i,t+1}; \tag{11}$$

$$\sigma^2_{i,t+1} = \omega + \gamma_i \sigma^2_{t,i} + w_{i,t+1} \tag{12}$$

v_{t+1} is a 3 × 1 vector of errors, $w_{i,t+1}, \varepsilon_{i,t+1}$ are scalar error terms.

Variances tuned such that the R squared for each individual return series was 50% and annualized volatility 30%.

Simulating Characteristics

The matrix C_t is an $N \times P_c$ vector of latent factors. The $P_x \times 1$ vector x_t is a 3×1 multivariate time series, and ε_{t+1} is a $N \times 1$ vector of idiosyncratic errors.

Simulation mechanism for C_t that gives some correlation across the factors time was used. First consider drawing normal random numbers for each $1 \leq i \leq N$ and $1 \leq j \leq P_c$, according to

$$\overline{c}_{ij,t} = \rho_j \overline{c}_{ij,t-1} + \epsilon_{ij,t}; \quad \rho_j \sim \mathcal{U}\left(\frac{1}{2}, 1\right) \tag{13}$$

Simulating Characteristics

Then, define the matrix

$$B := \Lambda\Lambda' + \frac{1}{10}\mathbb{I}_n, \quad \Lambda_i = (\lambda_{i1}, \ldots, \lambda_{i4}), \quad \lambda_{ik} \sim N(0,1), \; k = 1, \ldots, 4 \tag{14}$$

Transform this into a correlation matrix W via

$$W = (\text{diag}(B))^{\frac{-1}{2}} (B) (\text{diag}(B))^{\frac{-1}{2}} \tag{15}$$

To build in cross-sectional correlation, from the $N \times P_c$ matrix $\bar{C}_t$, we simulate characteristics according to

$$\widehat{C}_t = W\overline{C}_t \tag{16}$$

Simulating Characteristics

Finally, the "observed" characteristics for each $1 \leq i \leq N$ and for $j = 1, \ldots, P_c$ are constructed according to:

$$c_{ij,t} = \frac{2}{n+1} \operatorname{rank}\left(\hat{c}_{ij,t}\right) - 1. \tag{17}$$

with the rank transformation normalizing all predictors to be within $[-1, 1]$

Simulating Macroeconomic Time Series

For simulation of x_t, a 3×1 multivariate time series, we consider a VAR model, a generalization of the univariate autoregressive model to multiple time series:

$$x_t = Ax_{t-1} + u_t, \quad u_t \sim N\left(\mu = (0,0,0)', \Sigma = \mathbb{I}_3\right) \tag{18}$$

Simulating Macroeconomic Time Series

Consider 3 different specifications for matrix A:

$$A_1 = \begin{pmatrix} .95 & 0 & 0 \\ 0 & .95 & 0 \\ 0 & 0 & .95 \end{pmatrix} \tag{19}$$

$$A_2 = \begin{pmatrix} 1 & 0 & .25 \\ 0 & .95 & 0 \\ .25 & 0 & .95 \end{pmatrix} \tag{20}$$

$$A_3 = \begin{pmatrix} .99 & .20 & .10 \\ .20 & .90 & -.30 \\ .10 & -.30 & -.99 \end{pmatrix} \tag{21}$$

Simulating Return Series

We will consider four different functions $g(\cdot)$:

(1) $g_1\left(z_{i,t}\right) = \left(c_{i1,t}, c_{i2,t}, c_{i3,t} \times x_t'\right)\theta_0$

(2) $g_2\left(z_{i,t}\right) = \left(c_{i1,t}^2, c_{i1,t} \times c_{i2,t}, \operatorname{sgn}\left(c_{i3,t} \times x_t'\right)\right)\theta_0$

(3) $g_3\left(z_{i,t}\right) = \left(1[c_{i3,t} > 0], c_{i2,t}^3, c_{i1,t} \times c_{i2,t} \times 1[c_{i3,t} > 0], \operatorname{logit}\left(c_{i3,t}\right)\right)\theta_0$

(4) $g_4\left(z_{i,t}\right) = \left(\hat{c}_{i1,t}, \hat{c}_{i2,t}, \hat{c}_{i3,t} \times x_t'\right)\theta_0$

$g_1\left(z_{i,t}\right)$ is a linear specification

$g_2\left(z_{i,t}\right)$ and $g_3\left(z_{i,t}\right)$ is a non-linear specification with interactions

$g_4\left(z_{i,t}\right)$ builds returns using $\hat{c}$, which are the unobserved characteristics without cross sectional correlation built in

θ^0 was tuned so that the cross sectional R^2 was around 25%, and the predictive R^2 5%.

The simulation design results in $3 \times 4 = 12$ different simulated datasets,

Data Source

CRSP/Compustat database for stock returns with stock level characteristics such as accounting ratios and macroeconomic factors will be queried.

Only more recent data will be used, such as the period before and after 2008 GFC

Out of Sample R Squared

Overall predictive performance for individual excess stock returns were assessed using the out of sample R^2:

$$R^2_{OOS} = 1 - \frac{\sum_{(i,t) \in \mathcal{T}_3} (r_{i,t+1} - \widehat{r}_{i,t+1})}{\sum_{(i,t) \in \mathcal{T}_3} \left(r_{i,t+1} - \bar{r}_{i,t+1}\right)^2} \tag{22}$$

where $\mathcal{T}_3$ indicates that the fits are only assessed on the test subsample, which is never used for training or tuning.

Diebold Mariano Tests for Predictive Accuracy

- Compares the forecast accuracy of two forecast methods, (Diebold and Mariano (2002) and Harvey et al. (1997))
- Tests whether or not the difference series ($d_t = e_{1t} - e_{2t}$) between two forecast methods' errors is different from zero
- e_{1t} and e_{2t} will instead represent the average forecast errors for each model

Variable Importance

The importance of each predictor j is denoted as VI_j, and is defined as the reduction in predictive R-Squared from setting all values of predictor j to 0, while holding the remaining model estimates fixed.

Despite obvious limitations, this allows us to visualize which factors machine learning algorithms have determined to be important.

Results

Work is currently being done on trying to tune the R-Squared values of the simulated datasets

References

Ang, A., Bekaert, G., 2006. Stock return predictability: Is it there? The Review of Financial Studies 20, 651–707.

Breiman, L., 2001. Random forests. Machine learning 45, 5–32.

Diebold, F. X., Mariano, R. S., 2002. Comparing predictive accuracy. Journal of Business & economic statistics 20, 134–144.

Feng, G., Giglio, S., Xiu, D., 2019. Taming the factor zoo: A test of new factors. Tech. rep., National Bureau of Economic Research.

Feng, G., He, J., Polson, N. G., 2018. Deep Learning for Predicting Asset Returns. arXiv:1804.09314 [cs, econ, stat] ArXiv: 1804.09314.

Freyberger, J., Neuhierl, A., Weber, M., 2017. Dissecting characteristics nonparametrically. Tech. rep., National Bureau of Economic Research.

Goetzmann, W. N., Jorion, P., 1993. Testing the predictive power of dividend yields. The Journal of Finance 48, 663–679.

Questions and Answers