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

Define "factor" the same way as Harvey et al. (2016): 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.

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

Individual factors are typically firm characteristics and show little variation over time, or are pre-known

Background

- Goetzmann and Jorion (1993) and Ang and Bekaert (2006) note the persistence present in dividend ratio factors
- Movements in dividend ratios are dominated by movements in price and therefore 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 to be unbiased: t stats which are biased upwards and increase with time horizon due to autocorrelated errors
- Goetzmann and Jorion (1993) show that corrections to t statistics using the GMM and NW errors also appear to be biased upwards

Background

- Goyal and Welch (2003) conclude that while models incorporating dividend related factors were able to achieve higher in sample performance prior to 1990 than the historical mean, they could not have outperformed the historical mean out of sample
- Attributed to the increasing persistence and non-stationarity of dividend ratios, noting that they have become like random walks as of 2001
- Mirrors the sentiment of (Lettau and Ludvigson (2001), Schwert (2003) and others) who conclude that models incorporating dividend ratios seemed to break down in the 2000s despite having performed well in the 1990s

Background

Despite the controversy, the prevailing tone within the literature was that various factors such as dividend ratios, earnings price ratio, interest and inflation etc. were able to predict excess returns, with Lettau and Ludvigson (2001) remarking that this was now "widely accepted."

Welch and Goyal (2008) extend upon the work of Goyal and Welch (2003) by including a more comprehensive set of variables and time horizons. Conclude that not a single variable had any statistical forecasting power. Show that the significance values of some factors change with the choice of sample periods.

Background

- Literature has continued to produce more factors: quantitative trading firms were using 81 factor models as the norm by 2014 (Hsu and Kalesnik, 2014), and Harvey and Liu (2019) currently document well over 600 different factors suggested in the literature.
- Harvey et al. (2016) detail the false discovery problem when the number of potential factors is extremely high
- Produce a multiple testing framework to mitigate this, and conclude that many of the historically discovered would have deemed significant by chance

Background

- Number of potential factors discovered in the literature has increased to the same scale as, if not greater, than the number of stocks considered in a typical portfolio, or the time horizon, (Feng et al., 2019)
- Produces highly inefficient covariances in a standard cross sectional regression
- Moreover, when the number of factors exceeds the sample size, traditional cross sectional regressions become infeasible

Background

Many factors are cross sectionally correlated

Factors may be significant because they are correlated with a true, underlying factor and do not provide independent information themselves, a concern which Cochrane (2011) calls the multidimensional challenge

Especially challenging for traditional regression models, which make strong functional form assumptions and are sensitive to outliers, (Freyberger et al., 2017)

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 and more likely to approximate underlying data generating processes
- 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

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) conclude that all models agree on same subset of factors for importance, but in terms of relative importance of factors there are differences, and 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

- Two main approaches to dealing with temporal data
  - Rolling window - training, validation, and test set lengths are fixed and move forwards in time
  - Growing window - training set grows in size, but validation and test set lengths are fixed and move forwards in time
- Hybrid approach was chosen for feasibility
- Define a training set, validate on the next year, forecast for the next year
- Increase training set by one more year and move the validation and test sets forward one year

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)

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

Mean Squared Error (MSE)

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

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 Limitations:

- Need to manually consider and specify non-linear interactions
- Struggles with high dimensionality

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 and 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
- The eventual model 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, with repsect to minimizing the loss function

Regression Trees and Random Forests

The prediction of a tree, $\mathcal{T}$, with K "leaves" (terminal nodes), and depth L is

$$g(z_{i,t};\theta, K, L) = \sum_{k=1}^{K} \theta_k \mathbf{1}_{z_{i,t} \in C_k(L)} \tag{8}$$

where $C_k(L)$ is one of the K partitions in the model.

Only recursive binary trees are considered.

Regression Trees and Random Forests

Trees can be grown with respect to a variety of loss functions, including mean absolute error, mean squared error and Huber Loss:

$$H(\theta, C) = \frac{1}{|C|} \sum_{z_{i,t} \in C} L(r_{i,t+1} - \theta) \tag{9}$$

where $|C|$ denotes the number of observations in set C (partition). Optimal prediction for each partition is the mean of the partition to minimize MSE, and the median of the partition to minimize MAE.

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

Most complex type of model available

Able to capture several non-linear interactions through their many layers, hence its other name “deep learning”

Highly flexible and therefore often the most parameterized and least interpretable models

The scope of this paper is limited to traditional “feed-forward” networks.

Neural Networks

The feed forward network consists of an “input layer” of scaled data inputs, one or more “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.

A neural network with no hidden layers reduces to already familiar regression models, such as OLS and Logit (depending on activation function and choice form of output).

Neural Network Specifications

Neural networks with up to 5 hidden layers were considered. The number of neurons is each layer was chosen according to the geometric pyramid rule (Masters, 1993)

All units are fully connected: each neurons receives input from all neurons the layer before it

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

$$\text{ReLU}(x) = max(0, x) \tag{10}$$

Computation

- Stochastic Gradient Descent using ADAM
- Batch Normalization
- Randomize initial starting weights and biases, then average these into an ensemble model
- See references for specific details

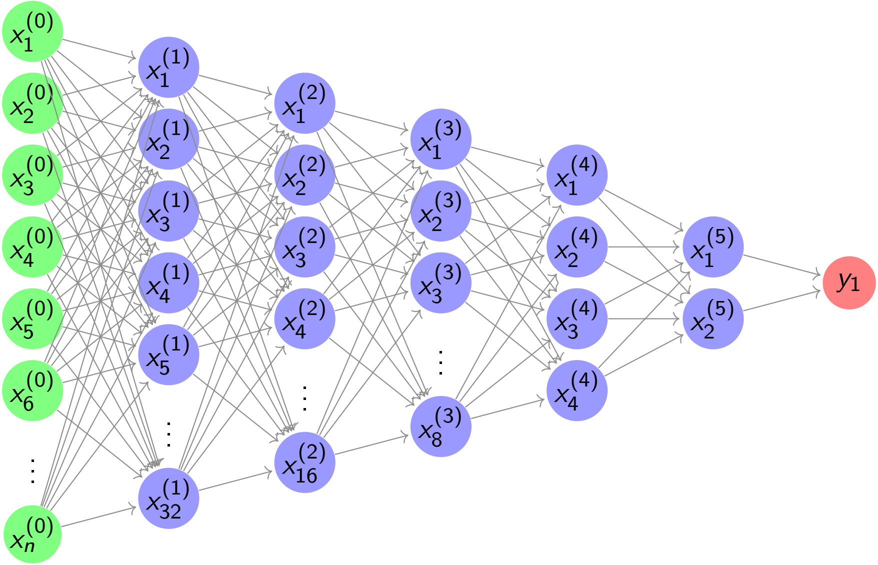


Figure 2: Neural Network 5 (most complex considered)

Overall Simulation Design

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

$$
\begin{aligned}
r_{i,t+1} &= g\left(z_{i,t}\right) + \beta_{i,t+1} v_{t+1} + e_{i,t+1}; && (11)\\
z_{i,t} &= (1, x_t)' \otimes c_{i,t}; \quad \beta_{i,t} = \left(c_{i1,t}, c_{i2,t}, c_{i3,t}\right); && (12)\\
e_{i,t+1} &= \exp\left(\sigma^2_{i,t+1}\right) \varepsilon_{i,t+1}; && (13)\\
\sigma^2_{i,t+1} &= \omega + \gamma_i \sigma^2_{t,i} + w_{i,t+1} && (14)
\end{aligned}
$$

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

$$\overline{c}_{ij,t} = \rho_j \overline{c}_{ij,t-1} + \epsilon_{ij,t}; \quad \rho_j \sim \mathcal{U}\left(\frac{1}{2}, 1\right) \tag{15}$$

Simulating Characteristics

Then, define the matrix

$$B := \Lambda\Lambda' + \frac{1}{10}\mathbb{I}_n, \quad \Lambda_i = (\lambda_{i1}, \dots, \lambda_{i4}), \quad \lambda_{ik} \sim N(0,1), \; k = 1, \dots, 4 \tag{16}$$

Transform this into a correlation matrix W via

$$W = (\text{diag}(B))^{\frac{-1}{2}} (B) (\text{diag}(B))^{\frac{-1}{2}} \tag{17}$$

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

$$\widehat{C}_t = W\overline{C}_t \tag{18}$$

Simulating Characteristics

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

$$c_{ij,t} = \frac{2}{n+1} \operatorname{rank}\left(\hat{c}_{ij,t}\right) - 1. \tag{19}$$

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:

$$x_t = Ax_{t-1} + u_t, \quad u_t \sim N\left(\mu = (0,0,0)', \Sigma = \mathbb{I}_3\right) \tag{20}$$

Simulating Macroeconomic Time Series

Consider 3 different specifications for matrix A:

$$A_1 = \begin{pmatrix} .95 & 0 & 0 \\ 0 & .95 & 0 \\ 0 & 0 & .95 \end{pmatrix} \tag{21}$$

$$A_2 = \begin{pmatrix} 1 & 0 & .25 \\ 0 & .95 & 0 \\ .25 & 0 & .95 \end{pmatrix} \tag{22}$$

$$A_3 = \begin{pmatrix} .99 & .20 & .10 \\ .20 & .90 & -.30 \\ .10 & -.30 & -.99 \end{pmatrix} \tag{23}$$

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

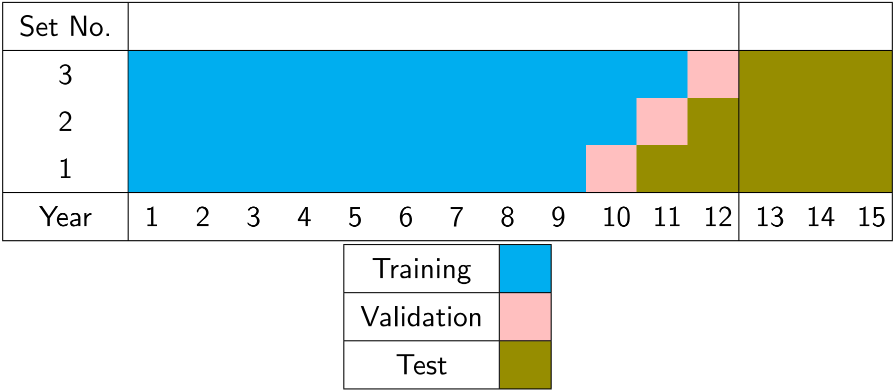
$g_4\left(z_{i,t}\right)$ builds returns using $\hat{c}$, which are the unobserved characteristics without cross sectional correlation built in

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

Sample Splitting

$T = 180$ monthly periods corresponds to 15 years. The training sample was set to start from $T = 108$ or 9 years, a validation set 1 year in length. The last 3 years were reserved as a test set never to be used for validation or training.

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 3: Sample Splitting Procedure

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:

$$R^2_{OOS} = 1 - \frac{\sum_{(i,t)\in\mathcal{T}_3}(r_{i,t+1} - \widehat{r}_{i,t+1})}{\sum_{(i,t)\in\mathcal{T}_3}\left(r_{i,t+1} - \bar{r}_{i,t+1}\right)^2} \tag{24}$$

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

- The Diebold-Mariano test (Diebold and Mariano (2002) and Harvey et al. (1997)) compares the forecast accuracy of two forecast methods
- Different to the overall R squared metric because it tests whether or not the models' forecast accuracy is significantly different
- Tests whether or not the difference series ($d_t = e_{1t} - e_{2t}$) between two forecast methods' errors is different from zero
- As all models in this paper will be producing forecasts for an entire cross section of stocks, e_{1t} and e_{2t} will instead represent the average forecast errors for each model

Diebold Mariano Tests for Predictive Accuracy

Under the null hypothesis (forecast errors from compared models are the same):

$$S_1^* = \left[\frac{n+1-2h+n^{-1}h(h-1)}{n}\right]^{1/2} S_1; \quad S_1^* \sim N(0,1) \tag{25}$$

$$S_1 = \left[\hat{V}(\bar{d})\right]^{-1/2} \bar{d} \tag{26}$$

$$\hat{\gamma}_k = n^{-1} \sum_{t=k+1}^{n} (d_t - \bar{d})(d_{t-k} - \bar{d}) \tag{27}$$

$$V(\bar{d}) \approx n^{-1} \left[\gamma_0 + 2\sum_{k=1}^{h-1} \gamma_k\right] \tag{28}$$

where d_t represents the difference series between the forecast errors of the

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

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.

Cochrane, J. H., 2011. Presidential Address: Discount Rates. The Journal of Finance 66, 1047–1108.

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.

Questions and Answers