OUT-OF-SAMPLE COMPARISON OF THE PREDICTIVE POWER OF VARIOUS NONLINEAR METHODS

Contents

List of Figures

List of Tables

1 Introduction

There are a large number of papers studying the relationship between numerous economic variables such as dividend yield, book-to-market ratio,... and the movement of equity premium. Among these studies, Welch and Goyal (2008) conduct a comprehensive analysis of the predictive ability of 15 prominent variables from the literature to equity premium using simple linear regression. In their paper, they show that most of these variables have poor out-of-sample performance, many linear predictive regressions fail to consistently outperform a simple forecast based on the historical average out-of-sample. Many variables have negative performance both IS (in-sample) and OOS (out-of-sample) over the most recent 30 years. They claim that the poor out-of-sample performance of regression models could indicate structural instability. They highly recommend using OOS diagnostic to judge the stability of the models.

The goal of my paper is to examine the predictive relationship between a collection of potential variables and equity premium, and to compare the predictive performance of various non-linear methods. My research is in a similar spirit to Welch and Goyal (2008) study. However, instead of using traditional regression, I extend the linear model to accommodate non-linear predictive relationships by using various machine learning methods. Using machine learning to examine the behavior of equity premium allows us to explore a richer specifications of functional form. It is expected that, due to its flexibility, non-linear machine learning methods can discover the underlying relationship between equity premium and the potential predictor variables better than a traditional linear regression model can do, because for our particular time series data, some assumptions in linear model may be violated. However, flexibility always comes along with overfitting, especially for small dataset like this problem, in this paper, I also try to reduce overfitting problem.

In this paper, I examine the out-of-sample performance of monthly equity premium forecasts for the S&P 500. I conduct an analysis of non-linear methods for forecasting equity premium, including neural network, principal component regression (PCR), the combination of neural network and principal component analysis (PCANN), Mallows model averaging (MMA), and iterated combination method (IC) and the results show that these methods can actually improve predictions. I start with the univariate predictive models using single hidden layer feed-forward neural network where each model is based on one of the 5 selected variables, **b/m**, **ntis**, **d/y**, **tbl** and **e/p**. Next, I consider the multivariate models which incorporate information from all these 5 variables, including PCR with maximum 3 principal components, neural network using first principal component as its input, MMA with 2^5 individual models, and iterated combination of all aforementioned models with historical average.

The remainder of this paper is organized as follows. Section 2 describes the the research methodology, including framework used to obtain out-of-sample forecasts and the measures to evaluate these predictive models. Section 3 shows the empirical results. Section 4 discusses the conclusions of the paper.

2 Methodology

2.1 The Framework

Assume that there is some relationship between equity premium and a set of potential predictor variables, which can be defined in the general form

$$\mathrm{y}_{t+1} = f(x_t) + \epsilon_{t+1} \tag{1}$$

where y_{t+1} is a scalar variable denoting the excess return on the S&P 500, $f(\cdot)$ is some predictor functions which we will discuss below, x_t is a vector of length k denoting predictor variables available up to time t whose predictive ability is of interest, and ϵ_{t+1} is is a mean-zero random error term.

The target variable of interest, y_t, is equity premium, defined as

$$\text{Equity premium}_t = \log(\frac{P_t + D_t}{P_{t-1}}) - \log(1 + R_t) \tag{2}$$

where D_t is 12-month movings sums of dividend paid on the S&P 500 index, P_{t-1} and P_t is the month-end value of the S&P 500 index at time $t-1$ and time t, respectively, and R_t is the short-term risk-free rate.

The collection of potential predictor variables for equity premium forecast studied in the literature is large. In my research, I select variables, x_t, based on three criteria. The first one is that they have been considered the most prominent variables for forecasting equity premium explored in the literature. The second one is that they provide statistically significant positive monthly (at least) in-sample performance, $\mathrm{R}^2_{\mathrm{IS}}$, in Welch and Goyal (2008). And the last one is that they should contain information from different important aspects that potentially affect equity premium. All in all, I have a set of only five candidate variables, as shown below.

1. Stock characteristics:

   - **Dividend yield** (*logDY*): $\mathrm{logDY} = \log(\frac{D_t}{P_{t-1}})$
   - **Earnings-price ratio** (*logEP*): $\mathrm{logEP} = \log(\frac{\mathrm{Earnings}_t}{P_t})$, where $\mathrm{Earnings}_t$ is 12-month moving sums of earnings on the S&P 500 index.

2. Corporate operation activity: **Book-to-Market ratio** (bm) is the ratio of book value to market value for the Dow Jones Industrial Average.

3. Corporate issuing activity: **Net Equity Expansion** ($ntis$) is the ratio of 12-month moving sums of net issues by NYSE listed stocks to total end-of-year market capitalization of NYSE stocks.

4. Interest rate: **Treasury bill rates** (tbl) are the 3-month Treasury Bill.

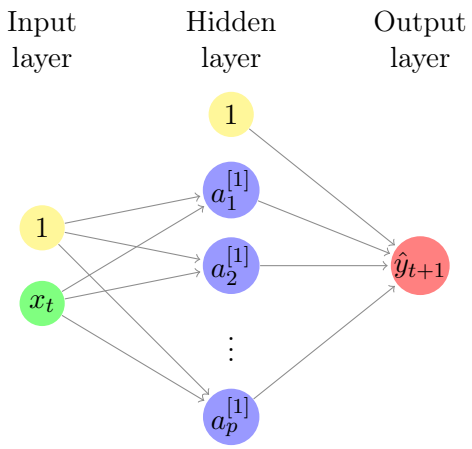
2.2 Models

Below I briefly describe various models considered for $f(\cdot)$ function in Equation (1).

2.2.1 Neural network

In this study, I will explore the performance of the most basic configurations for the feed-forward neural networks. In stock returns forecasting problem, due to limited data, a single hidden layer is often employed in research for two reasons. The first reason is that a single hidden layer network is shown to be sufficient to detect any complex nonlinear pattern in data as long as a suitable number of hidden units is used. The second one is that Gu, Kelly and Xiu (2018) finds that shallow network outperforms deeper network in forecasting stock return. The more complex network is, the more data is needed to achieve stable performance, but we have only finite data available so using deeper network more likely to suffer from the well-known problem of overfitting. Therefore, I will focus on the standard single hidden layer architecture which is characterized by: a single input unit (x_t), p nodes in the hidden layer ($a_1^{[1]}, \ldots, a_p^{[1]}$), and a single output unit ($\hat{y}_{t+1}$). The network is illustrated below:

Figure 1: Neural network



Before going on, let's define some notation first:

$$\mathbf{X}_{t,1\times n} = \begin{bmatrix} x_{t,1} & \dots & x_{t,n} \end{bmatrix}, \quad \mathbf{y}_{t+1,1\times n} = \begin{bmatrix} y_{t+1,1} & \dots & y_{t+1,n} \end{bmatrix}$$

$$\mathbf{W}^{[1]}_{p\times 1} = \begin{bmatrix} w^{[1]}_1 \\ \vdots \\ w^{[1]}_p \end{bmatrix}, \quad \mathbf{b}^{[1]}_{p\times 1} = \begin{bmatrix} b^{[1]}_1 \\ \vdots \\ b^{[1]}_p \end{bmatrix}$$

$$\mathbf{Z}^{[1]}_{p\times n} = \begin{bmatrix} z^{[1]}_{11} & \dots & z^{[1]}_{1n} \\ \vdots & \ddots & \vdots \\ z^{[1]}_{p1} & \dots & z^{[1]}_{pn} \end{bmatrix}, \quad \mathbf{A}^{[1]}_{p\times n} = \begin{bmatrix} a^{[1]}_{11} & \dots & a^{[1]}_{1n} \\ \vdots & \ddots & \vdots \\ a^{[1]}_{p1} & \dots & a^{[1]}_{pn} \end{bmatrix}$$

$$\mathbf{W}^{[2]}_{1\times p} = \begin{bmatrix} w^{[2]}_1 & \dots & w^{[2]}_p \end{bmatrix}, \quad \mathbf{b}^{[2]}_{1\times 1} = \begin{bmatrix} b^{[2]}_1 \end{bmatrix}$$

$$\mathbf{Z}^{[2]}_{1\times n} = \begin{bmatrix} z^{[2]}_1 & \dots & z^{[2]}_n \end{bmatrix}, \quad \mathbf{A}^{[2]}_{1\times n} = \begin{bmatrix} a^{[2]}_1 & \dots & a^{[2]}_n \end{bmatrix} = \hat{\mathbf{y}}_{t+1,1\times n}$$

where p is the number of nodes in hidden layer, n is the sample size, the input layer is represented by a row vector $\mathbf{X}_t$, $\mathbf{y}_{t+1}$ refers to the output vector at time $t+1$, $\mathbf{W}^{[1]}, \mathbf{W}^{[2]}$ are weight matrices, and $\mathbf{b}^{[1]}, \mathbf{b}^{[2]}$ denote bias vectors.

Our goal is to find weights and biases, $\mathbf{W}^{[1]}, \mathbf{W}^{[2]}, \mathbf{b}^{[1]}, \mathbf{b}^{[2]}$, for the network that minimize the loss function. For regression the loss function is given by

$$L(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{2}\sum_{t=1}^{n}(\hat{y}_t - y_t)^2 \tag{3}$$

The network construction comprises three steps in each epoch, one pass across the samples.[1] In the first step, forward propagation is performed in order to compute $\hat{\mathbf{y}}_{t+1}$ then obtain the loss function $L(\mathbf{y}, \hat{\mathbf{y}})$ as in Equation (3). To explore the non-linear relationship between the inputs and outputs, neural networks use non-linear activation functions. In the hidden layer of this network, I use the rectified linear unit activation function (ReLU), defined as $g(z) = \max(z, 0)$, a popular choice for regression problem. In the second step, backpropagation is performed to compute the derivative of the loss function with respect to any weight or bias in the network. In the last step, the weights and bias terms are updated using RMSprop method, an efficient version of the Stochastic gradient descent, in such a way that they minimize the loss function described above. This process is summarized in Algorithm 1.

[1] Assume processing the entire data set on each iteration (not use mini-batch in this section) to get an idea about how neural network model works in general. In practice, mini-batch approach is utilized.

Algorithm 1: Neural network

input : $\mathbf{X}_{t,1\times n}$, $\mathbf{y}_{t+1,1\times n}$

output: $\mathbf{W}^{[1]}_{p\times 1}$, $\mathbf{b}^{[1]}_{p\times 1}$, $\mathbf{W}^{[2]}_{1\times p}$, $\mathbf{b}^{[2]}_{1\times 1}$

initialize parameters $\mathbf{W}^{[1]}_{p\times 1}$, $\mathbf{b}^{[1]}_{p\times 1}$, $\mathbf{W}^{[2]}_{1\times p}$, $\mathbf{b}^{[2]}_{1\times 1}$ randomly

initialize $\mathbf{S}_{\mathrm{d}\mathbf{W}} = 0$, $\mathbf{S}_{\mathrm{d}\mathbf{b}} = 0$, $\beta = 0.999$, $\epsilon = 10^{-8}$

for *each iteration* **do**

//**Forward propagation**

$\mathbf{Z}^{[1]} = \mathbf{W}^{[1]}\mathbf{X}_t + \mathbf{b}^{[1]}$

$\mathbf{A}^{[1]} = g^{[1]}(\mathbf{Z}^{[1]}) = \max(0, \mathbf{Z}^{[1]})$

$\mathbf{Z}^{[2]} = \mathbf{W}^{[2]}\mathbf{A}^{[1]} + \mathbf{b}^{[2]}$

$\hat{\mathbf{y}}_{t+1} = \mathbf{A}^{[2]} = g^{[2]}(\mathbf{Z}^{[2]}) = \mathbf{Z}^{[2]}$

//**Backward propagation**

$\mathrm{d}\mathbf{Z}^{[2]} = \mathbf{A}^{[2]} - \mathbf{y}_{t+1}$

$\mathrm{d}\mathbf{W}^{[2]} = \frac{1}{n}\mathrm{d}\mathbf{Z}^{[2]}\mathbf{A}^{[1]\intercal}$

$\mathrm{d}\mathbf{b}^{[2]} = \frac{1}{n}\mathrm{d}\mathbf{Z}^{[2]}\mathbf{1}_n$ `// 1_n is a column vector of ones`

$\mathrm{d}\mathbf{Z}^{[1]} = \mathbf{W}^{[2]\intercal}\mathrm{d}\mathbf{Z}^{[2]} \cdot \mathrm{g}^{[1]\prime}(\mathbf{Z}^{[1]})$ `// · denotes an element-wise product`

$\mathrm{d}\mathbf{W}^{[1]} = \frac{1}{n}\mathrm{d}\mathbf{Z}^{[1]}\mathbf{X}_t^{\intercal}$

$\mathrm{d}\mathbf{b}^{[1]} = \frac{1}{n}\mathrm{d}\mathbf{Z}^{[1]}\mathbf{1}_n$

//**Update weights and biases using RMSprop**

Unroll $\mathrm{d}\mathbf{W}^{[1]}$, $\mathrm{d}\mathbf{W}^{[2]}$ to get vector $\mathrm{d}\mathbf{W}$

Unroll $\mathrm{d}\mathbf{b}^{[1]}$, $\mathrm{d}\mathbf{b}^{[2]}$ to get vector $\mathrm{d}\mathbf{b}$

$\mathbf{S}_{\mathrm{d}\mathbf{W}} := \beta\mathbf{S}_{\mathrm{d}\mathbf{W}} + (1-\beta)\mathrm{d}\mathbf{W}^2$

$\mathbf{S}_{\mathrm{d}\mathbf{b}} := \beta\mathbf{S}_{\mathrm{d}\mathbf{b}} + (1-\beta)\mathrm{d}\mathbf{b}^2$

$\mathbf{W} := \mathbf{W} - \alpha\frac{\mathrm{d}\mathbf{W}}{\sqrt{\mathbf{S}_{\mathrm{d}\mathbf{W}}}+\epsilon}$

$\mathbf{b} := \mathbf{b} - \alpha\frac{\mathrm{d}\mathbf{b}}{\sqrt{\mathbf{S}_{\mathrm{d}\mathbf{b}}}+\epsilon}$

end

As there is no reliable theory specifying the optimal number of hidden units for this equity premium forecast problem, defining structure of model is a difficult task. A much simpler network might not capture the non-linearity relationship in the predictors and the dependent variables, while a too complex network requires to estimate a large number of parameters which might cause high variance problem. Overfitting becomes more of a concern as the model complexity increases relative to the size of the data, especially for our limited data set, and therefore must be heavily regularized. In theory, neural network can simultaneously achieves low bias by building a large network and low variance by applying regularization methods. There are three popular approaches to deal with overfitting in neural network, including L1 and L2 regularization, dropout, and early stopping. In this specific problem, building a

larger neural network and applying dropout method in hidden units is a good choice. Dropout method will randomly eliminate hidden units from the neural network during training, end up with a sub-network from a larger network, potentially reduce the risk of overfitting.

2.2.2 Principal Component Analysis

When dealing with a limited data set in a high dimensionality space, there is a serious challenge we should consider when design a predictive model, *the curse of dimensionality phenomenon*, which occurs when a small number of observations are spread over high dimensional space. The curse of dimensionality phenomenon is very likely to cause overfitting, which means the model attempts to follow the noise too closely, thus produces a remarkably accurate prediction for training data, however it fails to generalize to new data. Therefore, a small number of observations per predictor will degrade the prediction accuracy. This problem becomes more severe for out-of-sample (OOS) prediction performance, a relatively poor OOS predictions early is expected when the number of observations in training set is relatively small. To deal with this problem, dimension reduction is an useful approach. In this paper, I use a well known dimensionality reduction technique, called Principal Component Analysis (PCA). The key concept behind PCA method is that in some case, a small number of principal components are sufficient to explain most of the variability in the data. Thus, information in k predictors can be summarized in much smaller M factors then can be used in forecasting target variable.

Dimension reduction methods work in three steps. In the first step, original predictors are standardized to have sample mean zero and sample variance one, as follow

$$\tilde{X}_{ij} = \frac{X_{ij} - \bar{X}_j}{\frac{1}{n-1}\sum_{i=1}^{n}(X_{ij} - \bar{X}_j)^2}, \quad j = 1, \ldots, k; \quad i = 1, \ldots, n \tag{4}$$

where $\bar{X}_j = \frac{1}{n}\sum_{i=1}^{n} X_{ij}$

In the second step, PCA condenses the set of predictors from a k-dimensional space to a smaller number of M uncorrelated, linear combinations of the original predictors that explain most of the variability in the original data.

$$Z_m = \sum_{j=1}^{k} \theta_{jm} X_j \quad m = 1, \ldots, M \tag{5}$$

$\theta_{1m}, \ldots, \theta_{km}$ are chosen wisely such that the first principal component Z_1 captures most of the information contained in the original predictors then the second principal component Z_2 is constructed in such a way that it has the largest variance

subject to the constraint of being uncorrelated with Z_1. Proceeding in this manner, we generate a series of M principal components.

The final step is fitting the model using these M components. Now, instead of using all k original variables, we use only the first M predictors, $Z_1, \ldots, Z_M$, to fit predictive model. Equation (1) becomes

$$\mathrm{y}_{t+1} = f(z_t) + \epsilon_{t+1} \tag{6}$$

In this paper, I focus on two forms of function $f(\cdot)$. The first one is a linear regression model, then Equation(6) becomes

$$\mathrm{y}_{t+1} = \beta_0 + \sum_{m=1}^{M} \beta_m z_{m,t} + \epsilon_{t+1} \tag{7}$$

We then fit the model in Equation (7) using least squares. This approach called *The principal components regression* (PCR).

The second form of $f(\cdot)$ we examine is the neural network, in which we will estimate the predictive model in Equation(6) using single hidden layer feed-forward neural network described above.

2.2.3 Mallows model averaging

Several studies show that model uncertainty and instability is one of the reasons that cause the poor forecasting ability of predictive models, thus combining individual forecasts is recommended. Model averaging is a frequentist combination approach, in which the estimator is a weighted average of predictions from different models which differ in their constituent variables. The idea is first to run the predictive regression on each set of predictors then combine the individual forecasts to obtain final estimator. Mallows model averaging (MMA), proposed by Hansen (2007), is a popular model averaging technique, in which the model weights are estimated by minimizing a Mallows criterion.

Following Hansen (2007), assume the model is the homoskedastic linear regression as

$$y_i = \mu_i + e_i \quad i = 1, \ldots, n \tag{8}$$

where $\mu_i = \sum_{j=1}^{\infty} \theta_j x_{ji}$, $E(e_i|x_i) = 0$,and $E(e_i^2|x_i) = \sigma^2$. We consider a sequence of predictive models $m = 1, 2, \ldots, M$ where the m^{th} model uses the first k_m features of X_i, where $0 \leq k_1 < k_2 < \ldots$. The m^{th} predictive model is defined as

$$y_i = \sum_{j=1}^{k_m} \theta_{j(m)} x_{ji(m)} + b_{i(m)} + e_i, \quad i = 1, \ldots, n \quad j = 1, \ldots, k_m \tag{9}$$

where the approximation error is $b_{i(m)} = \sum_{j=k_m+1}^{\infty} \theta_{j(m)} x_{ji(m)}$.

In matrix notation, Equation (9) can be written as

$$\mathbf{Y} = \mu + \mathbf{e} = \mathbf{X_{(m)}}\Theta_{(m)} + \mathbf{b}_{(m)} + \mathbf{e} \tag{10}$$

where $\mathbf{Y} = (y_1, \ldots, y_n)'$, $\mu = (\mu_1, \ldots, \mu_n)'$, $\mathbf{X_{(m)}}$ is an $(n \times k_m)$ matrix, $\Theta_{(m)} = (\theta_{1(m)}, \ldots, \theta_{k_m(m)})'$, $\mathbf{b}_{(m)} = (b_{1(m)}, \ldots, b_{n(m)})'$ and $\mathbf{e} = (e_1, \ldots, e_n)'$

The least squares estimator of $\Theta_{(m)}$ in the m^{th} model is

$$\hat{\Theta}_{(m)} = (\mathbf{X_{(m)}}'\mathbf{X_{(m)}})^{-1}\mathbf{X_{(m)}}'\mathbf{Y} \tag{11}$$

The corresponding estimator of μ is

$$\hat{\mu}_{(m)} = X_{(m)}\hat{\Theta}_{(m)} = X_{(m)}(X'_{(m)}X_{(m)})^{-1}X'_{(m)}Y \equiv P_{(m)}Y \tag{12}$$

Let $W = (\omega_1, \ldots, \omega_M)'$ be a weight vector in the unit simplex in $\mathbb{R}^M$: $H = \{W \in [0,1]^M : \sum_{m=1}^{M} \omega_m = 1\}$. The model average estimator of μ is:

$$\hat{\mu}(W) = \sum_{m=1}^{M} \omega_m \mathbf{P}_{(m)}\mathbf{Y} \equiv \mathbf{P}(W)\mathbf{Y} \tag{13}$$

The Mallows criterion for the model average estimator is

$$C_n(W) = (\mathbf{Y} - \hat{\mu}(W))'(\mathbf{Y} - \hat{\mu}(W)) + 2\sigma^2 tr(P(W)) \tag{14}$$

The weight vector based on the Mallows criterion is:

$$\hat{W} = \underset{W \in H}{\operatorname{argmin}}\, C_n(W) \tag{15}$$

2.2.4 Iterated combination approach

Another combination method is shown to improve forecasting performance is *iterated combination approach*, introduced by Lin, Wu, and Zhou's (2017), which combines the historical average with forecast from predictive models, defined as

$$y^I_{t+1} = (1-\delta)\bar{y}_t + \delta\hat{y}_{t+1|t} + u_{t+1} \tag{16}$$

where $\bar{y}_t$ is the sample mean of y_t using samples until time t, $\hat{y}_{t+1|t}$ is the forecast from non-linear models conditional on information available at time t, u_{t+1} is the noise, and δ is the combination weight.

The combination weight, δ, solves the optimization problem:

$$\min_{\delta} E_t(y_{t+1} - \hat{y}^I_{t+1})^2 = \min_{\delta} E_t(y_{t+1} - (1-\delta)\bar{y}_t - \delta\hat{y}_{t+1|t})^2 \tag{17}$$

The solution is defined as

$$\hat{\delta} = \frac{\text{cov}_t(y_{t+1} - \bar{y}_t, \hat{y}_{t+1|t} - \bar{y}_t)}{\text{var}_t(\hat{y}_{t+1|t} - \bar{y}_t)} \tag{18}$$

In general, the population ratio is unknown, but can be estimated based on the data, by replacing with that of the sample covariance to the sample variance.

The intuition behind iterated combination model forecast is the same as the idea of portfolio diversification, in which a risk-free asset and a risky asset are combined to improves portfolio performance. In the same way, it is expected that a suitable portfolio of $\bar{y}_t$ and $\hat{y}_{t+1|t}$ will produce a better forecast than using either $\bar{y}_t$ or $\hat{y}_{t+1|t}$.

2.3 Forecast Evaluation

2.3.1 Forecast evaluation measures

To evaluate predictive performance of a forecasting model, we use three measures: R^2 statistic, root mean square error difference (ΔRMSE), and MSE-F statistics.

Before computing these statistics, we need to calculate MSE_{Null}, the mean squared error (MSE) of the NULL model which using historical average to forecast equity premium, and MSE_{Alt}, the MSE of a forecasting ALTERNATIVE model which includes predictor variables we are interested in.

$$\text{MSE}_{\text{Null}} = \frac{1}{n}\sum_{t=1}^{n}(y_t - \bar{y}_t)^2 \tag{19}$$

$$\text{MSE}_{\text{Alt}} = \frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2 \tag{20}$$

where n is the number of fitted values, $\hat{y}_t$ is the fitted value from a predictive model, and $\bar{y}_t$ is either the full-period equity premium mean for IS or the prevailing equity premium mean for the OOS. IS statistics are computed from full sample data, OOS statistics are computed from data available up to the time at which the forecast is made.

Both R^2 and ΔRMSE indicate the quality of the fit, and computed as follows:

$$\text{R}^2 = 1 - \frac{\text{MSE}_{\text{Alt}}}{\text{MSE}_{\text{Null}}} \tag{21}$$

$$\Delta\text{RMSE} = \sqrt{\text{MSE}_{\text{Null}}} - \sqrt{\text{MSE}_{\text{Alt}}} \tag{22}$$

R^2 compares the fit of an Alternative model with that of a Null model, which measures the reduction in MSE for an Alternative model relative to the Null model. R^2 is positive if the Alternative model fit the data better than the historical value

in terms of the MSE.

ΔRMSE has the same sign as R^2 and conveys the same message as R^2. But ΔRMSE is measured in the units of the dependent variables, it explicitly tells us the difference in the root mean squared error between the historical average forecasts and the forecasts provided by a predictive model. A large positive ΔRMSE means that the given forecast model beats the historical average forecast.

We further test whether the MSE of the predictive model forecast is significantly less than that of the historical average forecast. We are interested in testing

$$H_0 : \Delta\text{MSE} = 0$$

against the one-sided alternative

$$H_a : \Delta\text{MSE} > 0$$

If ΔMSE $= 0$ then the given model is not superior to historical average forecast.

To test the null hypothesis, I use MSE-F statistic, proposed by McCracken's (2004).

$$\text{MSE-F} = T \times \frac{\text{MSE}_{\text{Null}} - \text{MSE}_{\text{Alt}}}{\text{MSE}_{\text{Alt}}} \tag{23}$$

Clark and McCracken (2001) shows that MSE-F statistic follows a nonstandard distributions when testing nested models. Together with the fact that the data size is relative small, I will use MSE-F statistics' bootstrapped critical values to provide statistical significance levels for the Null hypothesis test.

Besides the forecast evaluation measures discussed above, we also use OOS diagnostics graph, which is highly recommended by Welch and Goyal (2008), to evaluate predictive performance of the forecasting models. Graph shows the relative performance of the predictive models compared to the historical average benchmark model, in terms of the cumulative sum of squared error difference, which is defined as the cumulative squared forecast errors of the NULL model minus the cumulative squared forecast errors of the ALTERNATIVE model. The units in the graphs are not intuitive, but the line pattern does matter.[2] An increase in a line indicates that the examined predictive model perform better than the NULL model; a decrease in a line indicates that the NULL model achieves better performance. This plot helps us to evaluate the stability of the models. Ideally, the plot should show an upward drift in general, not just in unusual sample periods. Welch and Goyal suggest that like other diagnostics such as heteroscedasticity, collinearity, etc., if a model does not pass the diagnostics, we should suspect any conclusion drawn from that model. They also highlight that OOS performance is only considered as a complement to IS

[2] The IS performance is vertically shifted, so that the line begins at zero on the date of our first OOS prediction.

performance meaning that if a model is not significant IS, its OOS performance is not interesting.

2.3.2 Bootstrap

Follow the bootstrap procedure described in Welch and Goyal (2005), we impose the NULL of no predictability for computing the critical values of MSE-F.

$$\begin{aligned} y_{t+1} &= \alpha + u_{1,t+1} \\ x_{t+1} &= \mu + \rho x_t + u_{2,t+1} \end{aligned} \tag{24}$$

For each predictor variable, we estimate the models in Equation (24) by ordinary least squares using the full data set. We then simulate 999 resampled data sets and obtain critical values of MSE-F from the bootstrap procedure as follows:

- Randomly draw n observations $u^*_{1,t+1}$, and $u^*_{2,t+1}$ with replacement from $\{u_{1,t+1}\}$, and $\{u_{2,t+1}\}$ with $t = 1, \ldots, n$
- Apply Equation (24) to generate (Y^*_{t+1}, X^*_{t+1})
- The initial observation-preceding the resampled data is selected by randomly draw a pairs (Y^*_{t+1}, X^*_{t+1}) from the original sample $\{Y_{t+1}, X_{t+1}\}$
- Perform IS and OOS fit then compute the bootstrapped statistic of MSE-F* based on resampled data
- Repeat above steps for $b = 1, \ldots, B$, with $B = 999$, to obtain the bootstrapped distribution $\{\text{MSE-F}^*\}_{b=1,\ldots,B}$
- Compute critical values based on the this bootstrapped distribution

With this bootstrap procedure, the autocorrelation structure of the predictor variable is preserved.[3]

3 Empirical Results

3.1 Empirical Procedure

3.1.1 Design of experiments

Data. I use the updated monthly data from Welch and Goyal (2008) over the period from 1926/12 to 2018/12. The sample contains a total of 1104 observations.

The equity premium, y, computed using the continuously compounded return on the S&P 500 index, including dividends and the Treasury bill rate. Our predictor

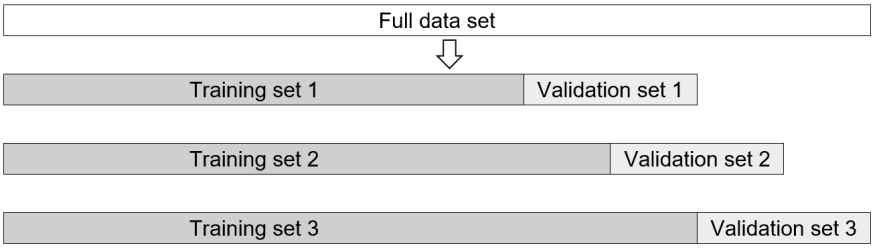
[3] If we randomly draw n pairs $(u^*_{1,t+1}, u^*_{2,t+1})$ in step 1, the cross-correlation structure of the two residuals is also preserved

variable set, x, contains 5 individual variables: dividend yield, earnings price ratio, book-to-market ratio, net equity expansion, and Treasury-bill rate.

We focus on one-step-ahead forecasts. The out-of-sample forecast evaluation period begins in 1965. We generate out-of-sample forecasts of the equity premium using recursively expanding windows, as in Welch and Goyal (2008).

Tuning hyperparameters. As non-linear models are often highly complicated thus may suffer from overfitting, we need to control the degree of model complexity via hyperparameters tuning. We adopt the following rolling cross-validation procedure in training.

Figure 2: Recursive scheme



We optimize the tuning hyperparameters using the "recursive" performance evaluation scheme with three training sets and three validation sets, as illustrated in Figure 2. For in-sample computation, the *full data set* is all observations from 1926/12 to 2018/12. For out-of-sample computation, the *full data set* is all observations available up to the time at which the forecast is made. Starting at the end of the full data set then going back 8 years to form the first training set and validation set. The training and validation samples gradually includes the most recent 2 years. But the training set still retain the historical data from beginning, thus the window size increases gradually while the validation set maintain the fixed size of 4 years.

The training process is performed as follows: The predictive model is initially trained on the subset of the in-sample data, *the training sets* in Figure 2, using a specific set of tuning hyper-parameter values. Then the fitted model is evaluated with the *validation sets* based on MSE. The final forecasting model, which yields the smallest 3-fold cross-validation MSE, is used to generate outputs.

Campbell and Thompson restriction. Welch and Goyal (2008) and Campbell and Thompson (2008) recommend that when examining the equity premium forecast, we should take into account the perspective of a real-world investor. Campbell and Thompson (2008) suggest to set both the historical mean forecast and the predictive model forecast to zero whenever they are negative because it does not make sense for an investor to forecast a negative equity premium. They also pointed out that this restriction improve the out-of-sample performance. Therefore, in my study, I impose the restriction that the equity premium forecast be positive, other-

wise it is set to zero on all out-of-sample forecasts. For IS, I use unrestricted forecasts because if restriction is imposed on IS forecast, we should incorporate it in objective function meaning that model is estimated such that the loss function is minimized subject to the constraint that the fitted values are non-negative, then the solutions are very complicated and are outside of the scope of this paper.

3.1.2 Models

The first non-linear method we examine is the feed-forward neural networks for each feature. Beside activation functions, the number of hidden nodes is also a factor to enhance the nonlinearity of a neural network model. As already mentioned, defining the number of hidden units in neural network is a difficult and important task, thus we must tune hyperparameters.

I employ the single hidden layer network model, illustrated in Figure 1 and Algorithm 1, in which the number of learning iterations (epochs) is 80,[4] the batch size is 128, the learning rate used in RMSprop method is set to 0.002.[5] Three levels of the number of hidden units, 10, 15, 20,[6] are experimented in this study. The idea is that I try to build a relatively large neural network compared to the size of data set in an attempt to detect nonlinear relationship between inputs and outputs, therefore must use dropout as a regularization method with a large dropout rate of 80% which means 80% of the hidden nodes will be eliminated from the network,[7]. For each predictor, three neural network models with different hidden units are examined and the recursive scheme described above is adopted to select the optimum neural network architecture.

It is worth noting that we should be careful when make a conclusion from a neural network forecast because neural networks are a stochastic algorithm which can yield different models when repeatedly running the same algorithm on the same data, due to randomness in parameters initialization and dropout regularization process in this case. When faced with this problem, it is recommended to not only report a single number but also provide statistical number as an evidence support the result. Here, due to limited resources, I will run the model several times then choose the model result which has the best predictive performance in terms of the statistical significance to report here.

Next, I will explore the multiple variables models that incorporate all five predictors, including Principal Components Regression (PCR), neural network using the first principal component as the input, and Mallow's Model Averaging models.

[4] The number of epochs is set to 80 because convergence was achieved after about 80 epochs.

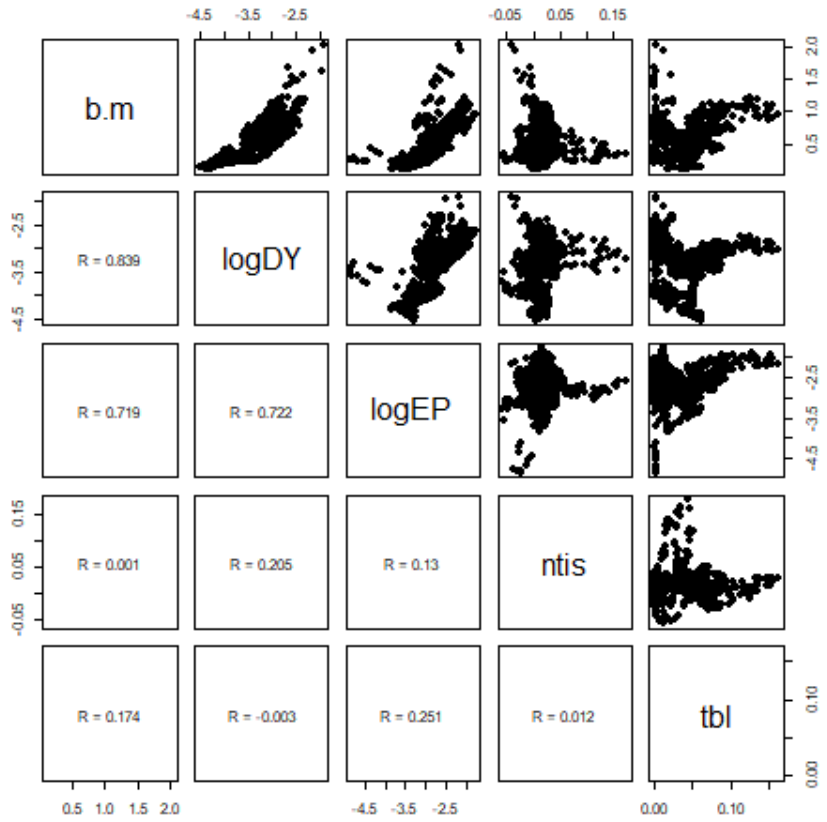
[5] The number of epochs, batch size and learning rater are selected through experimentation.

[6] For bootstrap in neural network, the number of hidden units is set to 15, without tuning (due to time limit)

[7] I also try to use these figuration in combination with early stopping, but it has poor forecasting ability so the results are not reported here.

Among all possible methods dealing with a relatively small data compared to the number of features situation, PCA approach is the most suitable for our problem for 2 reasons. First, it is a method to reduce overfitting in a problem associated with high dimension. Second, it is a method to deal with collinearity problems, in which two or more predictor variables are closely related to one another. The correlation matrix of the predictors displayed in Figure 3 shows that these five predictors appear to have some relationship, especially, **b.m**, **logDY** and **logEP** have high correlation with each other. The presence of collinearity in data can lead to a great deal of uncertainty in the model coefficient estimates meaning that a small change in the data can results in a large change in the model coefficient estimates.

Figure 3: Scatterplot matrix and correlation of the predictors



There are two popular solutions to deal with the problem of collinearity, dropping one of the problematic variables from models and combining them together into a single predictor. In my study, I use PCA as a method to combine information contained in the original features to obtain a smaller set of new uncorrelated predictors.

For PCR model, the number of principal components is determined using recursive scheme described early. Three PCR models with the number of principal

components ranging from 1 to 3 are tuned.[8] For neural network using the first principal component as the input, the first principal component is fed to the input node of the neural network described above, then the number of hidden units is selected by recursive scheme.

Huang and Lee (2010) shows that with multiple explanatory variables available, besides combination of information (CI) method which using all features in a single forecasting model like PCA models above, combination of forecasts (CF) method could possibly improve the out-of-sample predictive performance. CF method generates an ultimate forecast by combining forecasts from many models each using a part of the whole information set available. And for out-of-sample equity premium forecast problem, they find that CF scheme typically outperforms CI scheme. One of the source of improvement is that the combination forecast substantially reduces the volatility of the individual forecasts. Therefore, here I also investigate the out-of-sample forecasting capability of MMA, a popular combination technique which is in a similar spirit to CF scheme. To build MMA model, I use all 2^5 individual models.

The final model we investigate is iterated combination model. As shown by Lin, Wu, and Zhou's (2017) in the context of corporate bond returns, "predictability generated by the iterated combination is both statistically and economically significant". Inspired by its remarkable forecasting performance, I also build out-of-sample iterated combination models by combining the prevailing historical average forecast with the unrestricted forecasts obtained from all the aforementioned models.[9]

[8] The choice of 3 principal components is based on the correlation matrix of the predictors in Figure 3.

[9] "unrestricted" means not imposing CT constraint on forecasts.

3.2 Empirical results

3.2.1 Predictive power of univariate predictive models

In this section, we examine the in-sample and out-of-sample forecasting performance of neural network models where each model is based on one of the five variables described above. Table 1 and Figure 5 below show predictive performance of these forecasting models, several observations can be made.

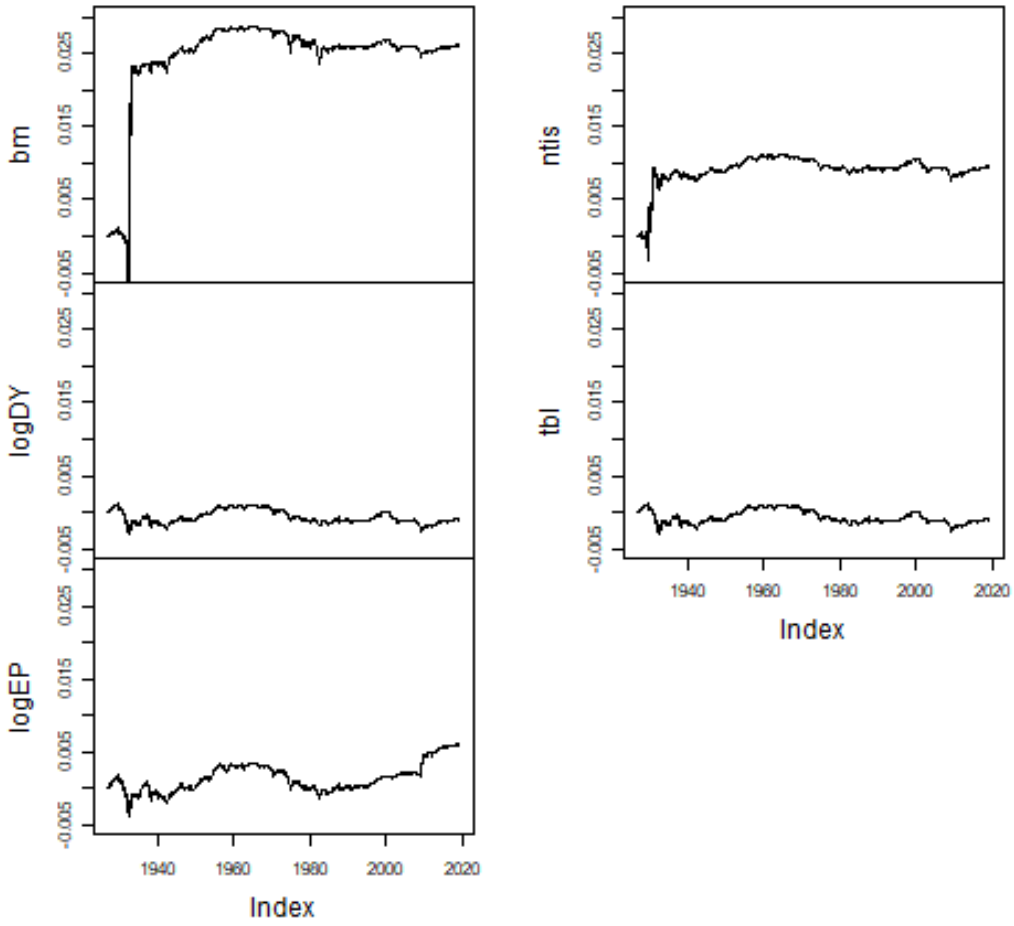


Figure 4: IS performance

For IS, the plots in Figure 5 show that IS performances over OOS period from all examined model have the similar patter, but difference in the magnitude of volatility, for example IS lines of all models drop in the recession periods but **b/m** tends to decrease more than **ntis**. This view indicates that neural network forecasts based on these predictor variables react in the same way to the events occuring in economic conditions.

Table 1 and Figure 4 shows that **logDY** and **tbl** have apparently same patter, both have negative IS R^2, indicating neural network based on these predictors is not useful to explain the variability of equity premium, thus can be ignored. Table 1 shows that three out of five models, **b/m**, **ntis** and **e/p**, are significant IS at least at the 5% level but two of them (**b/m**, **ntis**) have lost statistical significance when we exclude the period before 1965. The IS R^2 statistic over the entire in-sample period for **b/m**, **ntis** are 0.8% and 0.28%, respectively, but they become negative over the OOS period. From Figure 4, we can see that the relatively good IS performance of **b/m** and **ntis** mainly result from their remarkable performance in the period before 1965. An implausible performance IS in period before 1965 and an unremarkable performance thereafter make us question about the reliable of these model forecasts. One possible explanation for this highly unstable predictability is that a single predictive model is not sufficient to detect the relationship between equity premium and explanatory variables over this long sample period. This suggest that we should consider structural instability in building models.

Next, we examine **e/p** model which achieves R^2 of 1.18% with a 5% level of significance over entire in-sample period and 0.19% over OOS period. In contrast to other panels in Figure 4, the curve in **e/p** plot have slopes that are predominantly positive, even in OOS period, making it the only one that have positive R^2 statistics over both entire in-sample period and OOS period.

Now, we focus on OOS performance. Figure 5 shows a common pattern across five plots that there are roughly six peaks in OOS performance curves occuring at the oil crisis recession of 1973 to 1975, the Volcker disinflation in the early 1980s, the energy crisis recession of 1981 to 1982, the oil price shock in early 1990s, the dot-com buble and 9/11 recession of 2001 to 2002, and the global financial crisis recession of 2008 to 2009.[10] It is interesting to note that while we can see a clear upward trend in OOS curves in these recession periods, IS curves are going in the opposite direction, tending to decrease, in these periods. With this view, OOS neural network apparently perform better in bad times than a single IS neural network. It is also important to note that the OOS performance of these univariate predictive models are consistently better than the performance of their IS counterpart. This indicates that when applying non-linear model, particularly neural network model, it is bad idea to use one single IS model to predict equity premium. This once again support the idea that we should incorporate structural breaks in forecasting using non-linear model.

The plots in Figure 5 shows that except **e/p** model, most variables have good performance early then bad performance thereafter, affirming Welch and Goyal (2008). This means it is unreasonable to attribute the poor OOS performance of individual predictive models to parameter estimation uncertainty due to the lack of data, but

[10] `https://en.wikipedia.org/wiki/List_of_recessions_in_the_United_States`

it is more about the data structure.

Book-to-market ration: b/m has good OOS predictive performance from 1965 to 1981, poor performance until 1990, and neither good nor bad thereafter, except two unusual upward drifts in recession periods mentioned earlier. The early upward trend in OOS performance make it has significant positive OOS R^2, 0.35%, at 10% level of significance. Its IS performance is poor over OOS period (1965 to 2018), that is also consistent with statistics reported in table 1, while the IS R^2 is statistically significant positive, 0.8%, at 1% level of significance over the entire IS period, it is negative, -0.23%, over OOS period. Both IS and OOS shows the instability in neural network model based on **b/m**.

Net Equity Expansion: ntis model has excellent performance until 1990s, then underperformance thereafter. Neural network perform well out-of-sample to extract non-linear relationship between equity premium and **ntis** in period of 1965 to 1990s, but since then its predictive ability weaker. It has significant positive OOS R^2 of 0.47% at 5% level of significance, even exceeds its IS R^2. Its IS performance behave the same as that of **b/m**, IS R^2 is statistically significant positive, 0.28%, at 5% level of significance over the entire IS period, it is negative, -0.15%, over OOS period. The volatility in both IS and OOS indicates highly unstable predictability of **ntis**.

Dividend Yield: In general, **d/y** has superior performance until the early 1980s, has nondescript performance from 1980 to the early 2000s, then slightly decline thereafter. Along with this curve pattern, it has significant positive OOS R^2 of 0.26% at 10% level of significance, but its IS R^2 is negative implying that the historical mean beats the neural network model based on **d/y**.

Treasury Bills: tbl's performance generally declines in OOS period both IS and OOS. Therefore, both IS and OOS R^2 are negative. It is interesting to note that for other variables, OOS iterated combination performance is almost identical to OOS performance but for **tbl** OOS iterated combination model act like the insurance tool, when the OOS line go up, it goes down and vice versa. Together with the remarkable increase in 1975 makes it OOS iterated combination has significant positive R^2 1.24% at 5% level of significance. But, it is obvious to see that the **tbl**'s performance relies heavily on the superior performance in period from 1965 to 1975, thus according to Welch and Goyal (2008), one exceptional good performance in one particular unusual period is not meaningful.

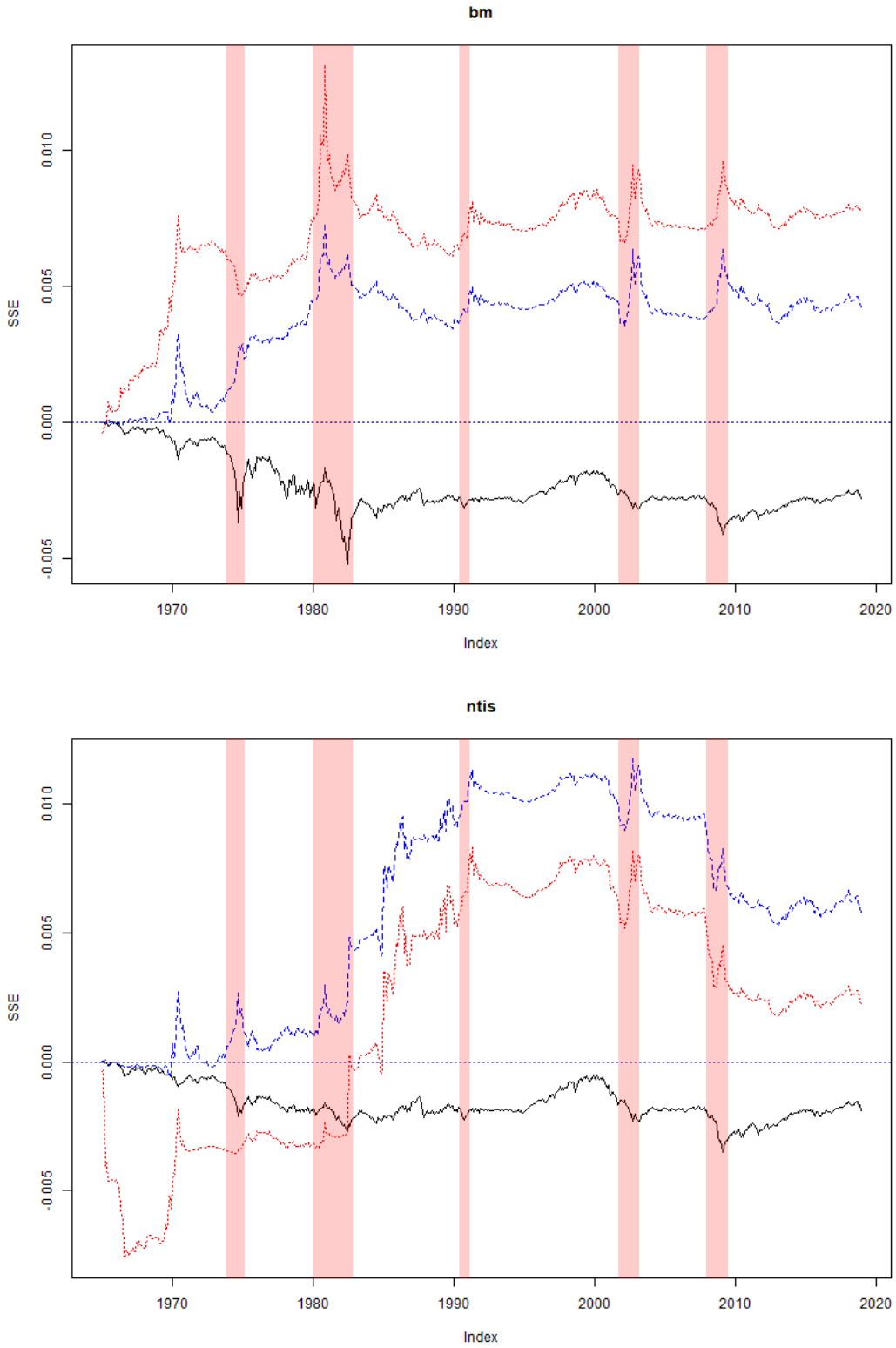


Figure 5: **Diagnostic Plot for univariate predictive models.** The IS prediction relative performance is plotted in black, the OOS prediction relative perfomance is plotted in blue, the OOS iterated combination prediction relative perfomance is plotted in red. The recession periods are marked by a red vertical line.

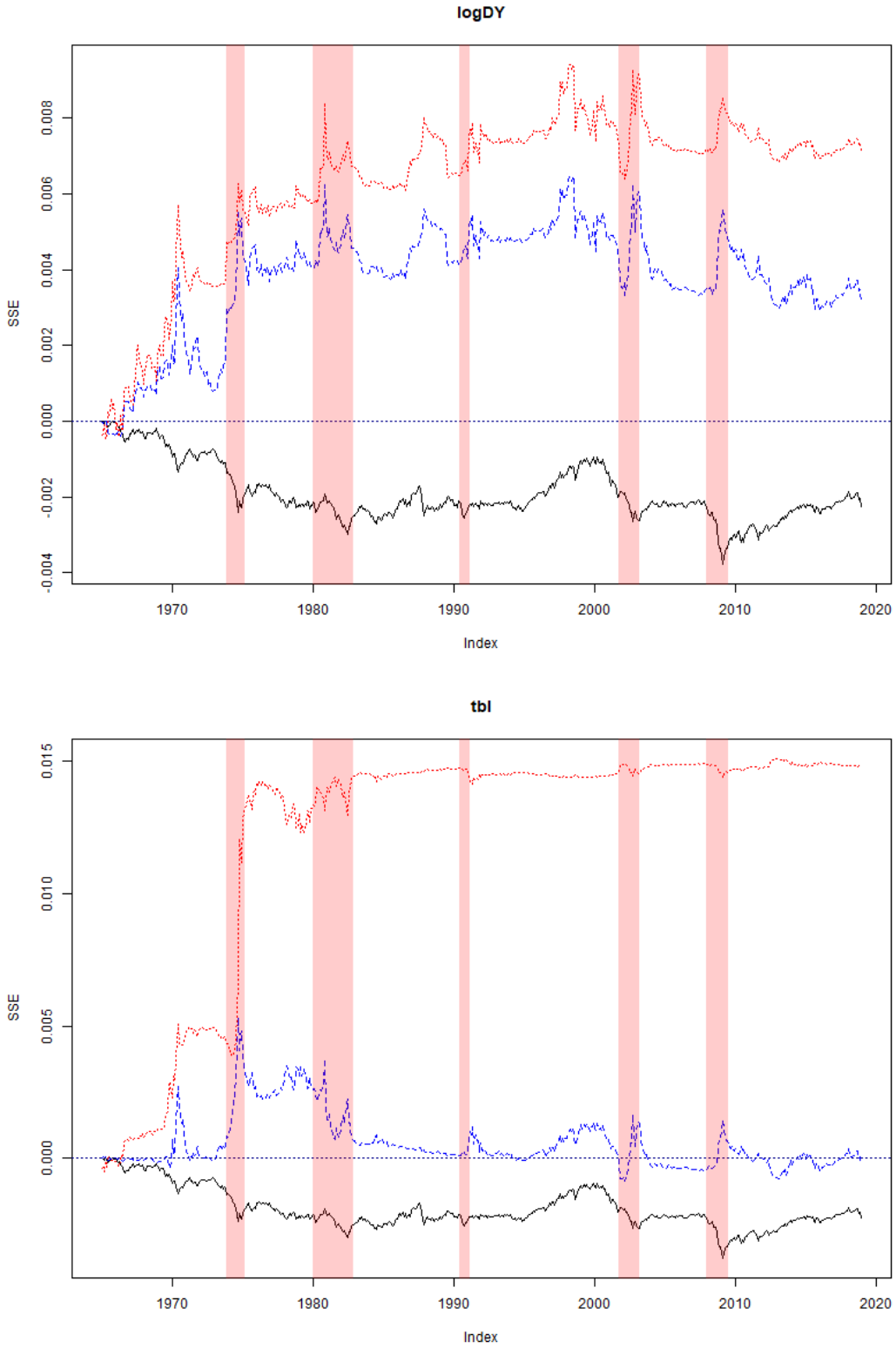


Figure 5: Diagnostic Plot for univariate predictive models (cont.)

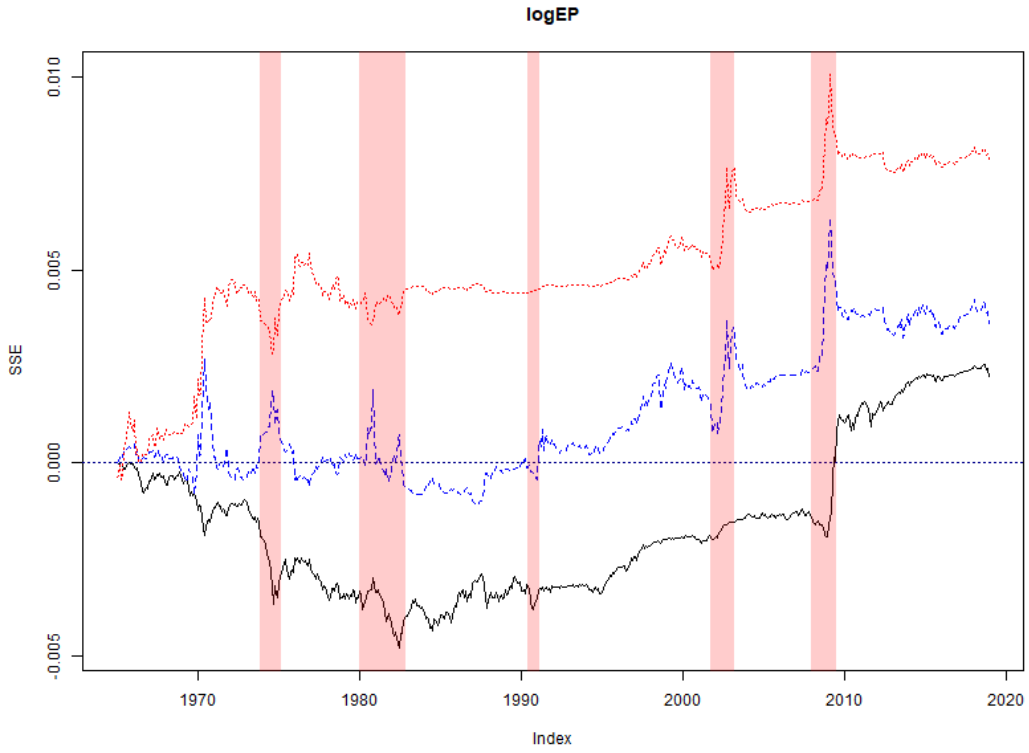


Figure 5: Diagnostic Plot for univariate predictive models (cont.)

Earnings Price Ratio: e/p has neither good nor bad performance (except for recession times) until 1980s, then superior performance thereafter so undoubtedly it has significant positive OOS R^2 of 0.3% at 10% level of significance, exceeding its IS R^2 of 0.18%. **e/p** is the only case that has promising IS and OOS pattern which underperform early but then perform well later. Campell and Thompson (2005) explain that the good predictability of **e/p** is due to "the changes in payout policy as firms have shifted from paying dividends to repurchasing shares".

The poor performance of a single IS model in tracking the movements in equity premium over the entire in-sample period together with the volatility in OOS performance indicate that structural breaks exist in the data thus affect the quality of the model and the forecast accuracy. We should take the structural instability into account when forecasting equity premium.

Finally, OOS iterated combination performance patterns look roughly like OOS performance, the difference only occurs in early periods until 1974-1975 recession, when δ in Equation (18)is estimated using limited data. Therefore, except increasing R^2, it does not really improve the forecasting performance of individual model. As expected, except for **tbl** having significant positive R^2 but not meaningful, these models generally have positive OOS R^2 but not statistically significant, thus can be ignored.

After investigating five predictors, we see none of these variables outperform the historical average IS and/or OOS. Thus, with non-linear model like neural network, finding individual predictors which consistently outperform the historical average is still difficult task.

3.2.2 Predictive power of multiple predictive models

Next we explore multiple variables prediction models which use all five features to forecast equity premium.

All the multiple predictive models achieve a statistically significant positive IS R^2 all at 1% level of significance over entire in-sample period, demonstrating a strong predictive ability. It is obvious that all multivariate models generate higher IS R^2 than the univariate model due to the fact that as number of predictors used in model increases, R^2 increases. Among our three multivariate models, **MMA** has the highest IS R^2 1.23% compared to 0.88% and 1.15% for neural network with first PCA (denoting **PCANN**) and **PCR** model, respectively. The reason for this is because there is loss of information in **PCR** and **PCANN** models whereas **MMA** incorporates information from all five variables in building forecast. Like individual neural newtwork in previous section, **PCANN**'s IS performance becomes negative over OOS period. **PCR** and **MMA**, have almost identical IS performance which performed well only in the first half of the sample period. The similarity of their curve form is due to the fact that even though they are complicated, they are just built on linear scheme basis. Although their IS R^2 over OOS period are pretty lower then the IS R^2 for entire period, they are still positive, 0.49% for PCR and 0.5% for MMA. Compared with univariate model in previous section, this indicate that it is unreasonable to just use one single variable in predicting equity premium because a single variable alone capture different aspect of the economic conditions thus can generate false signal leading to poor forecasting performance. However, using multivariate models may cause overfitting and here it is shown in OOS performance in Panel B of Table 1.

Table 1: **Forecasting performance.** This table presents statistics on IS and OOS performance of equity premium forecasts at monthly frequency, forecasts begin in 1965 January. Campbell and Thompson (2008) restriction that the forecasted equity premium be positive is imposed on all OOS forecasts. Panel 1 reports the performance of univariate neural network models, providing statistics for the forecast model that incorporate individual predictive variables given in each row. Panel 2 shows the performance of multivariate predictive models, including PCANN, PCR and MMA in which PCANN indicates the neural network using the first principal component as the predictor, PCR indicates Principal Components Regression, and Mallows model averaging labeled as MMA. We use the ΔRMSE and R^2 as measures of the predictive capacity of a given forecast model relative to the historical average benchmark forecast. MSEF statistic is used for testing whether MSE of the given model is less than that of the historical average forecast based on its bootstrapped distributions. Asterisks indicate significance at the *10%, **5%, and ***1% levels.[11]The column 'IS for OOS period' gives the IS R^2 and IS ΔRMSE for the OOS period, computed based on the errors from the overall full in-sample model over the OOS period. All numbers are percentages.

	IS					**OOS**					
	IS			IS for OOS period		OOS			Iterated combination		
	ΔRMSE	R^2	MSEF	ΔRMSE	R^2	ΔRMSE	R^2	MSEF	ΔRMSE	R^2	MSEF
Panel A: Univariate prediction models											
b.m	0.0217	0.7995	**8.8976*****	-0.0049	-0.2302	0.0076	0.3526	**2.2928***	0.0137	0.6364	4.1502
ntis	0.0077	0.2849	**3.1542****	-0.0033	-0.1556	0.0101	0.4700	**3.0599****	0.0038	0.1780	1.1555
logDY	-0.0010	-0.0361	-0.3983	-0.0039	-0.1825	0.0057	0.2624	**1.7047***	0.0128	0.5920	3.8593
tbl	-0.0010	-0.0361	-0.3983	-0.0039	-0.1825	-0.0005	-0.0210	-0.1361	0.0268	1.2395	**8.1328****
logEP	0.0048	0.1766	**1.9535****	0.0041	0.1925	0.0065	0.3018	**1.9619***	0.0141	0.6530	4.2594
Panel B: Multiple predictive models											
PCANN	0.0239	0.8806	**9.8085*****	-0.0024	-0.1103	-0.0382	-1.7808	-11.3380	0.0246	1.1407	**7.4771***
PCR	0.0313	1.1533	**12.8810*****	0.0106	0.4921	-0.0052	-0.2404	-1.5538	0.0171	0.7946	**5.1904***
MMA	0.0335	1.2324	**13.7754*****	0.0109	0.5059	-0.0342	-1.5966	-10.1831	0.0211	0.9763	**6.3890***

Iteration combination of MMA and HA gives MSEF statitics with p-value of 10.3%

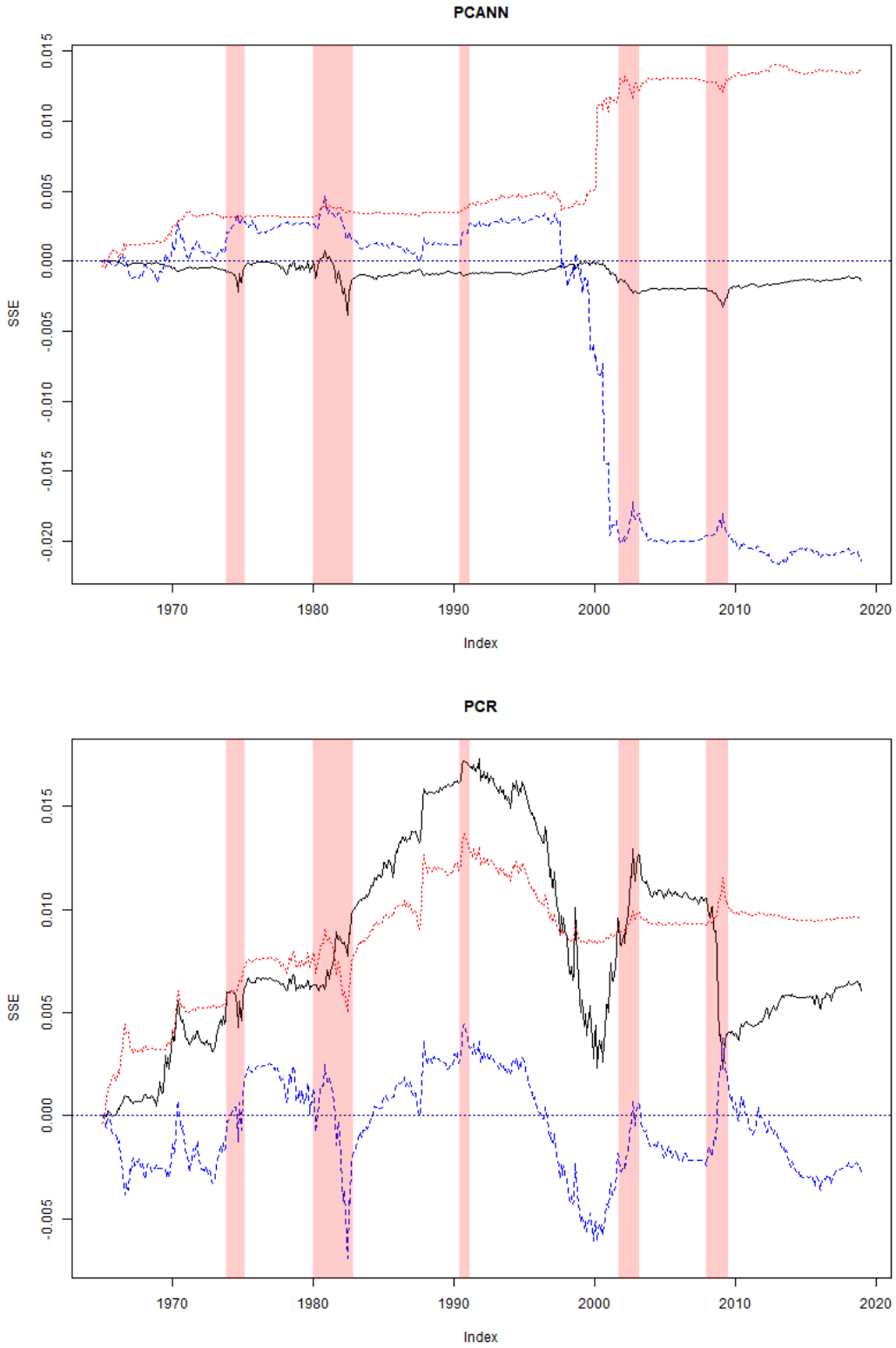


Figure 6: Diagnostic Plot for multiple predictive models

Figure 6: Diagnostic Plot for multiple predictive models (cont.)

PCANN: The plot in Figure 6 shows that **PCANN** perform well until the mid-1990s then drop dramatically from from 1995 to 2000, then has unremarkable performance since then. [12] This substantial drop make it has negative OOS R^2. **PCANN** only use the first principal component as its input so it is unreasonable to claim that this poor OOS performance is due to overfitting. The possible reason is maybe the neural network based on only one factor is not sufficient to extract the non-linear pattern in data in period after 1995. Next looking at the iterated combination forecast performance, here we see more clearly that the curve of OOS iterated combination goes in the opposite direction of OOS curve which make its OOS R^2 significantly postitive 1.14% at the 10% level.

PCR: Table 2 presents the **PCR** performance with varying the number of principal components, in which the first five rows consider **PCR** model built with the fixed number of principal component ranging from 1 to 5 and the next two rows consider **PCR** model built with tuning procedure. For the first five models, we can see the presence of overfitting, as the number of factors increases, the IS R^2 increases monotonically but the OOS R^2 first increase then decreases. In Table 1, the **PCR** with number of factors ranging from 1 to 3 tuning is presented. We see that for IS, the tuning procedure choose the **PCR** with 3 factors, but for OOS, tuning pro-

[12] this pattern shown in every repeat I take.

Table 2: Principal Components Regression

PCs	IS R^2	OOS R^2	OOS IC R^2
1	0.29	0.04	-148.55
2	0.3	0.05	-616.95
3	1.15	-1.62	0.57
4	1.28	-1.32	0.50
5	1.31	-2.16	0.63
Tuning 1:3	1.15	-0.24	0.79
Tuning 1:5	1.31	-0.12	0.74

cedure does not choose a single model. Because we use the recursive procedure to obtain out-of-sample forecasts, each data point in OOS period is predicted by different model with optimal number of factors after tuning process, leading to little improvement in OOS R^2 compared to using fixed number of factors which exceeds 3. This once again shows that data structure play an important role in forecasting performance, different form of model will suitable for different period.

MMA: In contrast to other models, in **MMA** we do not tune the number of individual model used in the final combination model, but use full 2^5 models constituted from 5 variables. Forecasts generated from 32 individual models are highly correlated because each model incorporates a part of the whole information set of 5 variables. Therefore, a combination of these high correlated forecasts is more likely to suffer from overfitting. Plot of **MMA** in Figure 6 seems to support this claim by showing an excellent IS performance together with a consistently poor OOS performance. Moreover, its IS is only good for half of period, meaning one single complicated model is still not enough to fit data. Once again, like **PCANN**, we see OOS iterated combination take advantage of its diversity property to improve the OOS performance. When OOS curve decrease, OOS iterated combination increase and vice verse, leading to significant positive R^2 of 0.97% at 10% level.

We find that the iterated combination method dramatically improves the forecasting performance of all the multiple predictive models and all of them is statistically significant positive at 10% level. Moreover, its patterns tend to be less volatile and more stable than the plain OOS performance. As mentioned earlier, the iterated combination scheme improves forecasting performance in the same manner that portfolio diversification does to improve portfolio performance in which stocks with weakly correlated returns are combined together to achieve less volatile portfolio return. For our particular issue, in the Figure 6, we can see **MMA** and **PCANN** forecasts consistently underperform the historical average forecast, thus a combination of unconditional forecast and forecasts generated from these models will be likely achieve better performance. This suggests that iterated combination method is effective for equity premium prediction in case of using multiple variables.

It is noteworthy that, for this particular issue, although R^2 statistics are very

small in magnitude, they are still economically meaningful. Campell and Thompson (2005) prove that even a small R^2 can significantly increase portfolio returns.

4 Conclusion

Following Welch and Goyal (2005), we have examined forecasting models according to four criteria: IS significance, reasonable OOS performance, effect of unusual periods and recent performance. The results show that most of univariate predictive models are uncertain, in consistent with Welch and Goyal (2008) findings. The good OOS performance early and bad OOS performance late pattern of these models shown in Welch and Goyal (2008) is still discovered here even though more data are available now indicating that we should not attribute the poor OOS performance to the parameter estimate uncertainty. The significant positive IS R^2 for full in-sample period becomes negative once data before OOS period excluded raising doubt about the stability of the predictive models, data before OOS period heavily affect the conclusion about predictive capability of the variables. $\mathbf{e/p}$ is the only variable that meets all four requirements over this examined period and has statistically significant both IS and OOS performance. One possible reason is that there is a change in policy. A large drop in **PCANN** and a tuning benefit shown in **PCR** suggest that different model configurations should be considered for different periods. All in all, these results indicate that single model over a long period is not sufficient and structural changes in the data play important role in analyzing predictability of a model. However, contrary to the findings of Welch and Goyal (2008) that OOS test destroys the predictability discovered in IS, in this paper we see that for univariate predictive models, IS and OOS deliver the same results, when a variable passes IS test, it also passes OOS test. Finally, the iterated combination models actually improve the OOS performance of multivariate predictive models due to its diversification property.

To improve OOS performance, I have some suggestions. First, we should consider model structure and data structure in building forecasters. About model structure, we can try to use different neural network configurations and other machine learning methods. To incorporate structural instability when building model, Pesaran and Timmermann (2007) and Zhang, He, Jacobsen and Jiang (2020) proposed the average windows forecasting method which first estimate individual sophisticated models over different estimation windows then average the their predictions to obtain final forecast. Next, we should consider multiple variables in building models because multivariate models are shown perform better than univariate models once we carefully choose the set of potential economic variables and integrate them together. The performance of the models depends on the choice of estimation period, thus varying the estimation period is also a way to view the relationship between the variables and the equity premium. The forecast horizon is another factor which affect

the predictive performance of the models, we should examine the effect of forecast horizon.

Bibliography

Campbell, J. Y.; Thompson (2008): Predicting excess stock returns out of sample: Can anything beat the historical average?, *The Review of Financial Studies* 21 (4): 1509–1531

Clark, T. E.; McCracken, M. W. (2001): Tests of equal forecast accuracy and encompassing for nested models, *Journal of econometrics* 105 (1): 85–110.

Gu, S.; Kelly, B.; Xiu, D. (2018): Empirical asset pricing via machine learning, *National Bureau of Economic Research*

Hansen, B. E. (2007): Least squares model averaging, *Econometrica* 75 (4): 1175-1189.

Huang, H.; Lee, T. H. (2010): To combine forecasts or to combine information?, *Econometric Reviews* 29 (5-6): 534–570.

Lin, H.; Wu, C.; Zhou, G.; Smits, T. (2018): Forecasting corporate bond returns with a large set of predictors: An iterated combination approach, *Management Science* 64 (9): 4218–4238.

McCracken, M. W. (2000): Asymptotics for out–of–sample tests of causality, *manuscript, University of Missouri*

Pesaran, M. H.; Timmermann, A.; Zhou, G.; Smits, T. (2007): Selection of estimation window in the presence of breaks, *Journal of Econometrics* 137 (1): 134–161.

Schomaker, M.; Heumann, C. (2014): Model selection and model averaging after multiple imputation, *Computational Statistics and Data Analysis* 71: 758-770.

Schomaker, M. (2017): MAMI: Model averaging (and model selection) after multiple Imputation, `http://mami.r-forge.r-project.org/`

Zhang, H.; He, Q.; Jacobsen, B.; Jiang, F. (2020): Forecasting stock returns with model uncertainty and parameter instability, *Journal of Applied Econometrics*

Affirmation

I hereby declare that I have composed my Master's thesis *"OUT-OF-SAMPLE COMPARISON OF THE PREDICTIVE POWER OF VARIOUS NONLINEAR METHODS"* independently using only those resources mentioned, and that I have as such identified all passages which I have taken from publications verbatim or in substance. I agree that the work will be reviewed using plagiarism testing software. Neither this paper, nor any extract of it, has been previously submitted to an examining authority, in this or a similar form.

I have ensured that the written version of this thesis is identical to the version saved on the enclosed storage medium.

_____

(Date, Signature)