Erasmus University Rotterdam
Erasmus School of Economics

Bachelor Thesis Quantitative Finance
Double Bachelor Econometrics and Economics

Forecasting the Equity Premium Under Parameter Instability: Simple Combinations Versus Nonlinear Machine Learning

Author:
Ewout Noort

Student Number:
503539en

Supervisor:
Dr. O. Kleen

Second Assessor:
B. van Os, MSc

Abstract

When forecasting the equity premium, models often suffer from model uncertainty and parameter instability. To tackle those issues, H. Zhang et al. (2020) use forecast combination methods and average window (AveW) estimation, respectively. By additionally shrinking the forecast to the historical average, their method delivers significantly positive out-of-sample R-squared. In this thesis, I show that simple forecast combinations perform well compared to advanced ones used by H. Zhang et al. (2020). However, they are unable to outperform the kitchen sink benchmark or the advanced weighted-average least squares. Nonlinear machine learning methods, on the other hand, perform poorly when parameter instability is accounted for. While almost all combination methods generate significantly positive out-of-sample R-squared, the nonlinear machine learning models struggle to do so. I show that the nonlinear machine learning models hardly improve with AveW or shrinking. At the same time, AveW greatly benefits simple combination methods, and shrinking stabilizes performance over time. Regarding shrinking, the results show that, although optimizing the shrinking factor remains difficult, there is potential in using regularized outlier-robust regression methods. I show my empirical findings to be consistent with the literature on finance and macroeconomics.

July 3, 2022

The views stated in this thesis are those of the author and not necessarily those of the supervisor, second assessor, Erasmus School of Economics or Erasmus University Rotterdam.

Contents

1 Introduction

Since many predictors of the equity premium fail to deliver out-of-sample forecasts that consistently outperform the simple historical average (Rapach et al., 2010; Welch & Goyal, 2008), stock return predictability remains controversial to this day. This may in part be attributed to model uncertainty: the *true* model is unknown (Pesaran & Timmermann, 1995; Welch & Goyal, 2008). To account for model uncertainty, forecast combination methods can be applied. However, they frequently rely on the assumption of parameter stability (H. Zhang et al., 2020), which is often violated in practice (Stock & Watson, 1996). H. Zhang et al. (2020) combine combination methods with the average window (AveW) estimation procedure developed by Pesaran and Timmermann (2007), which deals with parameter instability. Additionally, to reduce forecast variance, H. Zhang et al. (2020) shrink the obtained forecasts to the historical average. Their procedure provides forecasting gains that are statistically significant and economically meaningful.

In this thesis, I replicate the results of H. Zhang et al. (2020) and extend them in four ways. H. Zhang et al. (2020) use a selection of advanced combination methods including Bayesian model averaging (BMA), Mallows model averaging (MMA) introduced by Hansen (2008), jackknife model averaging (JMA) introduced by Hansen and Racine (2012) and X. Zhang et al. (2013), and weighted-average least squares (WALS) proposed by Magnus et al. (2010). These methods provide 'optimal' combination weights, in that they asymptotically achieve the lowest expected squared forecast error or minimize an unbiased estimator of the in- and out-of-sample mean squared forecast error (MSFE). However, according to Rapach et al. (2010) and Timmermann (2006), simple or performance-based weights often perform well compared to analytically optimal weights. This leads to the first research question:

> **RQ1** Do the advanced weighting schemes in H. Zhang et al. (2020) outperform simple or performance-based combination weights?

This question is evaluated when the models are estimated traditionally, but also when the AveW method and shrinking are applied. The simple and performance-based combination methods are the inverse residual variance (IV) method from Bates and Granger (1969), the trimmed mean (TM), and the discounted mean squared forecast error (DMSFE) method, both from Stock and Watson (2004). My second extension builds on the fact that H. Zhang et al. (2020) show that their combination methods outperform the linear machine learning models LASSO (Tibshirani, 1996) and elastic net (Zou & Hastie, 2005). Since Coulombe et al. (2019) conclude that nonlinearities in machine learning algorithms are useful in macroeconomic forecasting, I ask the following question:

> **RQ2** Can nonlinear machine learning models achieve higher forecasting performance than forecast combination methods?

The selected models are the Random Forest developed by Breiman (2001) and the Support Vector Regression from Vapnik (1995). Again, an expanding window, AveW, and shrinking are considered. Now that I apply two new classes of methods, a natural third extension is to generalize the results of H. Zhang et al. (2020) to simple combination methods and machine learning models. Thus, my next research question is:

RQ3 Do simple combination methods and machine learning models improve when estimated with the average window technique and shrinking is applied?

My final extension of H. Zhang et al. (2020) is inspired by the authors themselves. They shrink forecasts towards the historical average but admit that they do not manage to optimize the shrinking factor with a regression-based approach. Instead, they take the simple average of the two, called naive shrinking. Therefore, I use the suggestion from Liu et al. (2022) to use regularized regression, which has not been applied to combination forecasts yet. Thus, my final research question is:

RQ4 Can a regularized regression approach outperform naive shrinking when applied on forecasts that account for model uncertainty and parameter instability?

To answer the questions posed above, I evaluate forecasts of the equity premium over a period from 1957:01 to 2016:12. I use the updated database from Welch and Goyal (2008) over the period 1926:12 to 2016:12 to construct predictors and the equity premium, where the latter is defined as the log-returns on the value-weighted CRSP index including dividend minus the log-risk-free rate. Following Campbell and Thompson (2008), forecasts are evaluated statically via the out-of-sample R-squared, and economically by considering the annualized utility gain for a risk-averse mean-variance investor. Additionally, I compare models via the forecast encompassing test from Harvey et al. (1998).

My results show that the performance of the simple and advanced combination methods is similar, with the exception of WALS, which performs better and is the only model that outperforms all benchmarks. This indicates that while simple combination methods indeed perform well (Rapach et al., 2010; Timmermann, 2006), they can be beaten. My results contribute to the literature on financial forecasting with machine learning by showing that the nonlinear models outperform the combination methods when estimated traditionally. Nonetheless, when parameter instability is accounted for – which improves almost all forecasts – the nonlinear machine learning models are dominated by the forecast combination methods.

Comparing AveW to expanding window estimation, AveW has a particularly large effect on the simple combination methods. In fact, using AveW, they are able to generate significantly positive out-of-sample R-squared; most advanced methods cannot. The nonlinear machine learning methods, on the other hand, only moderately improve from AveW. This indicates that they either suffer less from parameter instability or AveW estimation is unsuited for nonlinear machine learning models. Contrary to what H. Zhang et al. (2020) find, shrinking only slightly improves performance of the simple combination and nonlinear machine learning models. Further analysis shows that shrinking mainly stabilizes performance over time. I show that my results are robust against using different predictors, but fail to hold up in a new out-of-sample period that includes the 2020 stock market crash caused by increasing instability over COVID-19.

Finally, using a regularized regression to determine the shrinking factor shows the potential to beat naive shrinking, but only if estimated with an outlier-robust estimation method. Using RANSAC estimation from Fischler and Bolles (1981) some combination methods improved over naive shrinking. For now, the main limitation of this method seems to be the selection of adequate hyperparameters.

My results add to the literature on macroeconomic finance in two ways. First, I show that the models closely track and predict business conditions. This provides evidence for the theories of Campbell and Cochrane (1999), Fama and French (1989), and Lettau and Ludvigson (2001b) that returns are predictable because expected returns vary with business conditions. Consistent with Neely et al. (2014) and Rapach et al. (2010), I find that macroeconomic predictors perform particularly well in periods of recession. Second, I show that predictability is higher in periods of extreme movement. This adds to the idea that predictability arises because the models are able to track risk (Fama & French, 1989). Variations in risk, in turn, relate to variations in expected returns (Merton, 1980).

The remainder of this paper is structured as follows. In Section 2 I present an overview of the relevant literature. The data and methodology are described in Sections 3 and 4. Section 5 presents the empirical results and in Section 6 I consider macroeconomic explanations for predictability in returns. Section 7 presents a robustness exercise and Section 8 concludes.

2 Literature

Many recent forecasting methods improved out-of-sample performance by taking into account model uncertainty and parameter instability (Rapach & Zhou, 2013). In this section, I provide an overview of this literature. Additionally, I present some literature on nonlinear machine learning in financial forecasting, and forecast shrinking.

2.1 Model Uncertainty

Timmermann (2006) explains that forecast combination methods, or model averaging methods, are a way to deal with model uncertainty. Rather than picking one forecast model, all available models are used and their forecasts are combined via, for example, a weighted average. This method not only addresses the fact that it is hard to select ex ante what model is superior; it acknowledges the fact that even the best model produces forecast errors. Therefore, combining it with another forecast may cancel out some of those errors. Successful applications of forecast combination methods in macroeconomic and financial forecasting include Makridakis and Hibon (2000) and Stock and Watson (2004).

Rapach et al. (2010) and Timmermann (2006) note that simple or performance-based combination schemes often perform well compared to analytically optimal schemes because the weights in the latter are hard to estimate accurately. Examples include using equal weights or weights based on the inverse of the residual variance of each model. The latter was introduced by Bates and Granger (1969). A performance-based example is the discounted mean squared forecast error (DMSFE) method from Stock and Watson (2004), where the DMSFE of each model is computed over a hold-out sample in the training data. The weights of the models are inversely related to the DMSFE. Timmermann (2006) shows that the performance of these methods may be enhanced by trimming off some forecasts.

Hoeting et al. (1999) provide a detailed tutorial on how to use Bayesian model averaging (BMA), which is more complex compared to the weighting schemes mentioned earlier. BMA assumes that each model under consideration has a prior probability to be the *true model* and then updates those probabilities using information from the estimated models. An alternative weighting scheme is introduced by Hansen (2008) and is called Mallows model averaging (MMA). This method works by minimizing Mallows's criterion. Jackknife model averaging (JMA) also works by minimizing an information criterion, only this one is based on leave-one-out cross-validation. JMA was developed by Hansen and Racine (2012) and X. Zhang et al. (2013). MMA and JMA are both frequentist techniques, whereas BMA is a bayesian technique. In frequentist methods, the weights are fully determined by the data and require no priors. As opposed to MMA, JMA allows for heteroskedasticity in the models. As pointed out by H. Zhang et al. (2020), Hansen (2008) and X. Zhang et al. (2013) show that the Mallows criterion is an unbiased estimator of the in- and out-of-sample MSFE. Moreover, X. Zhang et al. (2013) and Hansen and Racine (2012) demonstrate that JMA asymptotically achieves the lowest expected squared error. Finally, a method that combines Bayesian and frequentist techniques is weighted-average least squares (WALS), introduced by Magnus et al. (2010). According to H. Zhang et al. (2020), this method has both computational and theoretical advantages over BMA, MMA, and JMA. Moreover, Magnus and De Luca (2016) show it to be an effective approach to dealing with uncertainty.

2.2 Machine Learning

Contrasting with forecast combinations are methods to select a single 'optimal' model to create forecasts, also called pretesting (Leeb & Pötscher, 2003). One way to do this is by using variable selection techniques such as least absolute shrinkage and selection operator (LASSO) introduced by Tibshirani (1996). By using a subset of the available variables rather than all, prediction accuracy may be improved on (H. Zhang et al., 2020). The Elastic Net method, introduced by Zou and Hastie (2005), builds on this idea but overcomes some of the limitations of LASSO when, for example, predictor variables are correlated. Considering that LASSO and elastic net are linear, Coulombe et al. (2019) show that nonlinearities in machine learning models improve forecasting. Popular models that incorporate nonlinearity are the Support Vector Regression (SVR) from Vapnik (1995), which uses the kernel trick; the non-parametric random forest regression (RFR) from Breiman (2001); or an artificial neural network (ANN). For successful applications of SVR, RFR, and ANN on financial and macroeconomic time series, see among others, Patel et al. (2015), Sermpinis et al. (2014), Xiang-rong et al. (2010), and Gu et al. (2020). In this thesis, RFR and SVR are applied because (i) Gu et al. (2020) concluded that RFR and neural networks have similar performance for monthly asset price forecasting, and (ii) Coulombe et al. (2019) argue that SVR and RFR are generally sufficient to consider nonlinearity while being computationally less demanding than ANN, which is important when models are re-estimated every step.

2.3 Parameter Instability

Forecast combination and variable selection methods are often derived under the assumption of parameter stability (H. Zhang et al., 2020), while this is often not satisfied in practice (Stock & Watson, 1996). Earlier methods to deal with parameter instability, like those of Bai and Perron (1998, 2003), estimate the timing of structural breaks and then only use post-break data. More recently, Pesaran and Timmermann (2007) show that it can be optimal to use pre-break data in out-of-sample forecasting exercises. However, the optimal amount of pre-break data depends on the timing and size of the break, which are unknown. To overcome these practical limitations, Pesaran and Timmermann (2007) develop a method called average window (AveW) estimation, which is robust to structural breaks. Instead of picking one optimal period of data (or window) to estimate the model on, multiple windows are taken and the results are averaged. Pesaran and Pick (2011) show promising theoretical results in terms of MSFE, while at the same time the method has been shown to work well in a number of empirical applications (among others, Assenmacher-Wesche and Pesaran, 2008; Pesaran et al., 2009; Schrimpf and Wang, 2010).

2.4 Shrinking Forecasts

H. Zhang et al. (2020) create forecasts with AveW estimation and then shrink those forecasts, denoted as $\hat{r}_{T+1}$, to the historical average, denoted $\bar{r}_{T+1}$. This yields a final forecast $\hat{r}^{f}_{T+1} = (1-\delta)\bar{r}_{T+1} + \delta\hat{r}_{T+1}$, where δ is the shrinking factor. The idea behind shrinking is to reduce forecast variance while increasing its bias only slightly, resulting in a lower MSFE. According to Timmermann (2006), determining the optimal δ is difficult. This is demonstrated by the results of H. Zhang et al. (2020), who show that naive shrinking, with $\delta = 0.5$, outperforms a regression based approach from Granger and Ramanathan (1984) and Lin et al. (2017) to determine δ. Recently, Liu et al. (2022) extended this method to use a regularized regression to estimate δ. They show that their shrinkage method is able to generate significant positive out of sample R-squared. However, they only apply it to forecasts obtained from simple models, not those produced by forecast combination methods, machine learning models, and AveW estimation.

3 Data

To create forecasts, I use the same predictor variables as H. Zhang et al. (2020), who follow Elliot et al. (2013) and construct 12 economic variables based on the updated data from Welch and Goyal (2008)[1]. In particular, I take six variables directly from the data of Welch and Goyal (2008). These are the stock variance (SVAR), book-to-market ratio (BM), net equity expansion (NTIS), Treasury bill rate (TBL), long-term rate of returns (LTR), and inflation (INFL). Additionally, I construct six new variables using the data from Welch and Goyal (2008). These are the dividend price ratio (DP), the log of the ratio between the 12-month moving sum of dividends and the S&P500 index; dividend yield (DY), the log of the ratio between the 12-month moving sum of dividends and the S&P500 index of 12 months ago; earnings price ratio (EP), the log of the ratio between the 12-month moving sum of earnings and the S&P500 index; term spread (TMS), the long term yield on government bonds minus the treasury bill rate; default yield spread (DFY), the yield on BAA- minus AAA-rated rated bonds; default return spread (DFR), long-term corporate bond return minus long-term government bond returns. The dependent variable, equity premium, is defined as log-returns on stocks including dividend, minus the log-returns on the risk-free rate. Contrary to H. Zhang et al. (2020), I proxy stock

[1] An update database of the data from Welch and Goyal (2008) is accessible at http://www.hec.unil.ch/agoyal/. For a full description of this data I refer you to Welch and Goyal (2008)

returns with the value weighted CRSP index, rather than the S&P500. These values are highly correlated, but the CRSP includes all stocks and is, thus, arguably more representative of stock returns. As a result, my results can in part be interpreted as a robustness exercise of the results of H. Zhang et al. (2020) for the proxy used as equity premium. My sample period is from 1926:12 to 2016:12, where 1957:01 to 2016:12 is the out-of-sample period for which to create forecasts.

4 Methodology

In this section, I first explain the three components of my forecasting procedure: combination method or model, estimation procedure, and shrinking. Thereafter, I show the statistical and economic evaluation metrics.

4.1 Models and Forecast Combination Methods

I classify each model or combination method into one of three categories: simple model averaging methods, advanced model averaging methods, and machine learning models. Each method or model produces forecasts of returns at time $T + 1$, denoted $\hat{r}_{T+1}$, using an $1 \times N$ vector of predictors data observed at time T, denoted X_T.

Besides the three categories of models, I employ two simple benchmark models. These benchmarks should not be confused with the benchmark historical average, which is used in my evaluation metrics. The first benchmark is the simple combination method of Rapach et al. (2010), which uses a simple linear regression for each of the predictor variables individually, and then averages those forecasts. This method is also referred to as Pool-AVG. Second, I apply a multivariate linear regression for stock returns at time $t + 1$, r_{t+1}, which is written as

$$r_{t+1} = X_t\beta + \varepsilon_{t+1}, \quad t = 0, 1, ..., T - 1, \tag{1}$$

where β is an $N \times 1$ vector of unknown coefficients, ε_{t+1} is an error term, and T is the sample size. This model is referred to as the kitchen sink model and can be estimated with Ordinary Least Squares (OLS). Note that the linear regression serves as a starting point for many of the combination methods applied in this paper.

4.1.1 Simple Forecast Combinations

In general, forecast combination methods work as follows. There are a number m candidate models. The i-th model is denoted as $\mathcal{M}_i$ and has weight ω_i. The weights are collected in $\omega = (\omega_1, ..., \omega_m)'$. If model $\mathcal{M}_i$ produces forecast $\hat{r}^{(i)}_{T+1}$, we can express the combination forecast as

$$\hat{r}^{\text{combi}}_{T+1} = \sum_{i=1}^{m} \omega_i \hat{r}^{(i)}_{T+1}, \tag{2}$$

where ω is restricted to be an element of $\mathcal{H} = \{\omega \in \mathbb{R}^m : \omega_i \geq 0, \sum_{i=1}^{m} \omega_i = 1\}$. Different forecast combination methods have different ways to determine the weights ω_i.

Trimmed Mean. The first simple combination scheme is based on the trimmed mean (TM) from Stock and Watson (2004). I estimate all $m = 2^N - 1$ linear regression models that include a constant and at least one predictor. The forecasts are then combined using the arithmetic average. However, I symmetrically trim off $100\alpha\%$ of the most extreme forecasts, giving weights

$$\omega_i^{\text{TM}} = \frac{1}{(1-\alpha)N} \tag{3}$$

to the remaining forecasts. Following Stock and Watson (2004), I trim off 5% of forecasts on both sides.

Discounted Mean Squared Forecast Error. Second, I apply the performance-based discounted mean squared forecast error (DMSFE) combination method from Stock and Watson (2004). With DMSFE, the weights are determined by pseudo-out-of-sample performance. For this purpose, I make an 80/20 split in the estimation data. The first part is the training data over which the model is estimated and it ends at time T_0. The second part is a hold-out period for which forecasts are created and the discounted MSFE is computed as as

$$\phi_i = \sum_{t=T_0}^{T-1} \theta^{T-t-1}(r_{t+1} - \hat{r}_{t+1}^{(i)})^2, \tag{4}$$

where θ is the discount factor. I set θ=0.99. Finally, using the entire estimation sample, forecasts are created and combined with weights

$$\omega_i^{\text{DMSFE}} = \frac{1/\phi_i}{\sum_{j=1}^{m} 1/\phi_j}. \tag{5}$$

Since forecast combination methods may be improved by trimming off forecasts based on some measure of quality or performance (Timmermann, 2006), I trim off a total of 10% of forecasts that have the highest DMSFE.

Inverse Residual Variance. The final simple combination scheme is based on Bates and Granger (1969). In every model, the residual variance is estimated. The weights of the forecasts are inversely proportional to the estimated residual variance, giving weights

$$\omega_i^{\text{IV}} = \frac{1/\hat{\sigma}_i^2}{\sum_{i=1}^{m} 1/\hat{\sigma}_i^2}, \tag{6}$$

where $\hat{\sigma}_i^2$ denotes the unbiased estimated residual variance in model $\mathcal{M}_i$. I trim off the 10% of forecasts that have the highest residual variance.

4.1.2 Advanced Forecast Combinations

In this section, I explain the basics of the more advanced combination methods, which can additionally be classified as Bayesian, frequentist, or a mix of both. For the interested reader, Appendix A, along with the references given therein, provide more detail on how exactly the weights of each method are derived.

Bayesian Model Averaging. The first forecast combination method from the class of advanced methods is the Bayesian Model Averaging (BMA) method. With BMA, the weights are determined by the relative posterior probabilities of the models. Following H. Zhang et al. (2020), I assume equal model priors, which means that ex ante all models have an equal probability to be the true model. Additionally, like H. Zhang et al. (2020), I assume diffuse model priors on parameters. This can be interpreted as having very little information on the parameters of each model, and therefore assuming them to be equal across models. Then, having observed (r_{t+1}, X_t) for $t = 1, ..., T-1$, the probabilities of each model are updated using Bayes rule, which results in the posterior probabilities. I approximate the weights with an expression derived in Buckland et al. (1997) and explained in Appendix A. For the implementation of BMA in python, I adapted a piece of sample code from Basener (2020).

Mallows Model Averaging. Mallows model averaging (MMA) is a forecast combination method that uses frequentist-, rather than Bayesian techniques. This implies that the weights are completely determined by the data, and require no priors (H. Zhang et al., 2020). MMA is developed by Hansen (2008) and sets the combination weights such that a Mallows criterion is minimized. This approach is appealing since Hansen (2008) shows that the Mallows criterion is an asymptotically unbiased estimate of the in-sample mean squared error and the out-of-sample mean squared forecast error. Rather than considering all possible $2^N - 1$ models like with BMA and the simple combining methods, I follow H. Zhang et al. (2020) and only consider N nested models, where the nesting order is determined by the coefficients on LASSO regression. The reasons for this choice are (i) because the theory regarding MMA developed in Hansen (2008) is only for nested models, and (ii) determining the weights ω^{MMA} requires solving a quadratic programming problem, thus, using $N << 2^N - 1$ models is computationally more feasible. The Mallows criterion and the minimization problem can be found in Appendix A. For the implementation of MMA I used the MATLAB code provided by Hansen and Racine (2012)[2].

Jackknife Model Averaging. The second frequentist method I employ is jackknife model averaging (JMA), developed by Hansen and Racine (2012) and X. Zhang et al. (2013). JMA selects weights by minimizing a leave-one-out, or jackknife, cross-validation criterion. This entails that every observation (r_T, X_{t-1}) is deleted once and forecasted with the remaining observations. Do this with each model and save the error that each model makes. Then, linearly combine the errors on a single observation with the combinations weights ω^{JMA}. Finally, set the weights such that the mean squared error of these combinations is minimal. For the mathematical details of this procedure, I refer you to Appendix A.

In contrast to MMA, JMA can accommodate heteroskedastic data. Moreover, X. Zhang et al. (2013) and Hansen and Racine (2012) demonstrate that JMA asymptotically achieves the lowest expected squared error. For similar reasons as before, only nested models based on a LASSO regression are considered. My implementation of JMA again uses the MATLAB code from Hansen and Racine (2012)[3].

[2] Code can be accessed via https://www.ssc.wisc.edu/ bhansen/progs/joe_12.html

[3] See footnote 2

Weighted Average Least Squares. The final forecast combination method I use is weighted-average least squares (WALS), developed by Magnus et al. (2010). WALS offers an attractive combination of Bayesian and frequentist techniques. For instance, the computational complexity of WALS is linear in the number of predictors, rather than exponential like BMA and the simple combination methods (Magnus et al., 2010). Moreover, WALS treats ignorance regarding the priors in a different manner than BMA does, which results in bounded prediction variance (Magnus et al., 2010). Specifically, whereas the priors in BMA are based on a Normal distribution, I can use the reflected Weibull distribution with WALS. My choice of using Weibull priors, rather than e.g., Laplace or Subbotin priors is based on H. Zhang et al. (2020), who motivate their choice based on Magnus and De Luca (2016). In my application of WALS, I put all predictor variables in the group of focus variables, which is the group of variables that may be included. The constant is in the group of variables that are required to be included. The expression for the averaging coefficients of the model is provided in Appendix A. For the implementation of WALS, I use the MATLAB code made available by Magnus et al. (2010), Magnus and De Luca (2016), De Luca and Magnus (2011), and Kumar and Magnus (2013)[4].

4.1.3 Machine Learning

Finally, I introduce the machine learning models. The first two models, LASSO and Elastic Net, are linear and use build-in variable selection techniques. The final two models, Support Vector Regression and Random Forest Regression, are nonlinear. Additional details can be found in Appendix A.

LASSO. The Least Absolute Shrinkage and Selection Operator (LASSO) model, introduced by Tibshirani (1996), is a linear regression model that performs variable selection. In essence, LASSO estimates the the coefficients β in Equation (1), but applies L_1-regularization. This means LASSO penalizes the parameter β by adding the L_1 norm, denoted as $\left\|\beta\right\|_1^1$, to the regular squared loss function that OLS uses. How much the parameter β is penalized is determined by λ, the regularization parameter. Tibshirani (1996) explain that, for sufficiently high λ, some coefficients are set equal to zero, resulting in a more stable and parsimonious model. In each forecasting step, the optimal λ is determined via 5-fold cross-validation using chronological splits. This means that the validation data always come after the estimation data, thus respecting the time series component of the data. This is contrary to H. Zhang et al. (2020) who use regular 5-fold cross-validation. Appendix A contains more details on the loss function and cross-validation.

Elastic Net. In addition to L_1-regularization, the Elastic Net method, introduced by Zou and Hastie (2005), applies L_2-regularization. With L_2-regularization, the squared norm of β is added to the loss function. Zou and Hastie (2005) show that the Elastic Net mimics the variable selection property of the LASSO while resolving some of its limitations. For instance, empirical applications show that L_2-regularization tends to perform better than L_1 in the case of correlated variables, which is relevant when using macroeconomic predictors. Moreover, LASSO

[4] Code can be accessed via https://www.janmagnus.nl/items/WALS.pdf. I used version 2.0, created 3 April 2010 and last revised on 18 December 2013

tends to select only one variable from a group of correlated variables and does not care which one. Similar to LASSO, the elastic net has regularization parameter λ, which I select via 5-fold cross-validation with a chronological split of the data. Additionally, it has a hyperparameter α that controls how the weight from λ is divided between the two penalties. I follow H. Zhang et al. (2020) and set $\alpha = 0.5$, thus, dividing the weight equally across penalties.

Support Vector Regression. The first nonlinear machine learning model I introduce is the ε-Support Vector Regression (SVR) introduced by Vapnik (1995). Appendix A provides the necessary details to use SVR for my application. Smola and Schölkopf (2004) contains additional derivations and background. Similar to the regression models we encountered so far, the SVR estimates a function of the explanatory variables $f(x)$, such that $f(x_t)$ is close to r_{t+1}. However, SVR differs in two main ways. First, SVR uses an ε-intensive loss function, as introduced in Bennett and Mangasarian (1992), rather than the squared loss function used in OLS, LASSO and elastic net. This means that if r_{t+1} is within a margin ε of $f(x_t)$, no penalty is given; however, for r_{t+1} outside the margin, a penalty the size of the distance to the margin is given. Second, SVR can incorporate nonlinearities via the kernel trick. Following Colombo and Pelagatti (2020) I use the radial basis function kernel (RBF). SVR uses L_2-regularization and thus has a regularization parameter, which is selected together with ε via 5-fold cross-validation with a chronological split.

Random Forest Regression. The final model I apply is the non-parametric Random Forest Regression (RFR). The random forest was developed by Breiman (2001) and is an ensemble of regression trees. Hastie et al. (2009) explain that in a regression tree, you iteratively partition the predictor space into rectangles. Starting at the root node of the tree, it determines the feature and specific value to split on. For example, the data can be split based on whether inflation was above or below 2%. The two new subsets are represented by a node and separately split on. This continues until the data at one node are sufficiently homogeneous. The splitting stops and this becomes a terminal node. The returns corresponding to this terminal node are modeled by a constant, which is set equal to the mean of returns in said node. For a new observation x_T, you follow the branches of the tree and forecast r_{T+1} with the value at the corresponding terminal node. *When* to stop splitting is controlled by the maximum allowed depth of a tree, which is a hyperparameter. If the tree is too deep it overfits the data, but if it is too shallow it does not capture some structure in the data.

As explained by Hastie et al. (2009), a random forest takes bootstrap samples of the original data and grows a regression tree on each of them. To create a forecast, you use each tree and aggregate the results. This procedure is called Bootstrap Aggregation, or Bagging. RFR reduces the variance of the forecast by decorrelating trees via random feature selection: instead of considering all predictors at each node, only a random subset is considered. The RFR thus has two hyperparameters that I will determine via 5-fold cross-validation with chronological split: the maximum depth of a tree, and the number of features considered at each node. The number of trees is essentially also a hyperparameter. However, Breiman (2001) shows that adding trees does not cause overfitting of the data. Therefore, it must be determined by balancing the extra benefit with the additional computation time. For details on how the splits inside the tree are determined and some intuition on bagging and random feature selection, see Appendix A; for examples of RFR see Hastie et al. (2009). I used the python machine learning package, sklearn (Pedregosa et al., 2011), for the implementation of LASSO, elastic net, SVR, and RFR.

4.2 Model Estimation

My first estimation method is to use an expanding window, meaning every step I re-estimate the model using all available historical data. Second, following H. Zhang et al. (2020), I use average window (AveW) estimation, developed by Pesaran and Timmermann (2007), to account for parameter instability.

Explained by Pesaran and Timmermann (2007), AveW estimation is useful when structural breaks are present, but their exact location and maginitude are unknown. In essence, the method estimates the model over different parts of the data and averages the result. Mathematically, let $\mathbb{W} = \{(r_{t+1}, X_t)\}_{t=0}^{T-1}$ be the set of all available observations. Then, define a number m of overlapping estimation windows as $\mathbb{W}_i = \{(r_{t+1}, X_t)\}_{t=T-w_i}^{T-1}$ for $i = 1, ..., m$, where $w_i = w_{\min} + (\frac{i-1}{m-1})(T - w_{\min})$ denotes window length and $w_{\min}$ is the minimum window length. Finally, letting $\hat{r}_{t+1}(\mathbb{W}_i)$ denote the forecast of r_{t+1} using the model estimated on window $\mathbb{W}_i$, we get AveW forecast

$$\hat{r}_{T+1}^{\text{AveW}} = \sum_{i=1}^{k} \hat{r}_{T+1}(\mathbb{W}_i). \tag{7}$$

4.3 Shrinking

As a final step, I apply the combining method from H. Zhang et al. (2020) where the forecasted return, denoted as $\hat{r}_{T+1}^{\text{FC}}$, is shrunk towards the histrocial average, defined as $\bar{r}_{T+1} = \frac{1}{T}\sum_{t=1}^{T} r_t$. This results in the combination forecast

$$\hat{r}_{T+1}^{\text{C}} = (1-\delta)\bar{r}_{T+1} + \delta \hat{r}_{T+1}^{\text{FC}}, \tag{8}$$

where δ is the shrinking factor. Following H. Zhang et al. (2020), I set $\delta = 0.5$ (naive shrinking). Additionally, I use a regression based approach from Lin et al. (2017) to determine the 'optimal' value of lambda. The procedure is based on Granger and Ramanathan (1984), who explain that the weights of a forecast combination can be determined by regressing the variable to be forecasted on the individual forecasts. In this regression, I restrict the coefficients to sum up to one and I set the intercept equal to zero[5]. As shown in Lin et al. (2017), the regression then becomes becomes

$$r_{t+1} = (1-\delta)\bar{r}_{t+1} + \delta \hat{r}_{t1}^{\text{FC}} + u_{t+1}, \tag{9}$$

where u_{t+1} is an error term. After subtracting $\bar{r}_{t+1}$ from both sides in Equation (9), it can be seen that δ can be estimated by regressing $r_{t+1} - \bar{r}_{t+1}$ on $\hat{r}_{t+1}^{\text{FC}} - \bar{r}_{t+1}$ with OLS. Based on the results from H. Zhang et al. (2020), I expect this to perform poorly. Therefore, I also use the method from Liu et al. (2022), who suggest that instead of OLS, a regularized regression is used to determine δ in Equation (9). Specifically, I use Ridge regression, which applies L_2-regularization. This

[5] Allowing for a constant or weights that do not sum to one affected forecasting performance extremely negatively, so results are not displayed

means the squared norm of δ is added to the quadratic loss function. Like we saw before, this add regularization parameter α, which is determined via 5-fold cross-validation with chronological split. Following Liu et al. (2022), I consider values $\alpha = k \times \text{var}_t(\hat{r}_t^{\text{FC}} - \bar{r}_t)$ for $k = 0, 1, ..., 100$[6].

Of course, realized values of r_{t+1}, $\bar{r}_{t+1}$, and $\hat{r}_{t+1}^{\text{FC}}$ are needed for this regression. I do this as follows. I produce forecasts for the entire out-of-sample period. Then, I split this period into two parts: 1957:01 to 1979:12 and 1980:01 to 2016:12; combination forecasts are only made for the latter. This way, the forecasts from the first period can be used for the regression. The regression is done with an expanding window, meaning that every period one observation is added to the regression. I further extend this regression approach by exponentially weighting the observations. To be precise, observation (y_t, X_t) gets weight $w_{T-t} = \theta^{T-t}$, where θ can be interpreted as a monthly discount factor.

4.4 Forecast Evaluation

In this section, I present the metrics used to evaluate forecasts. Throughout this section, $\hat{r}_t^{\mathcal{M}}$ denotes the forecast made by model $\mathcal{M}$, e.g., BMA with AveW and shrinking, or SVR with expanding window. Whenever a metric requires the use of a benchmark, the historical average is used.

4.4.1 Statistical Evaluation

I statistically evaluate the forecasts by means of the out-of-sample R-squared from Campbell and Thompson (2008), denoted as R^2_{OS}. If a total of p forecasts over the period $[T+1, T+p]$ are available, then the R^2_{OS} for model $\mathcal{M}$ is defined as

$$R^2_{\text{OS}} = 1 - \frac{\text{MSFE}^{\mathcal{M}}}{\text{MSFE}^{\text{BMK}}} = 1 - \frac{\sum_{t=T+1}^{T+p}(r_t - \hat{r}_t^{\mathcal{M}})^2}{\sum_{t=T+1}^{T+p}(r_t - \bar{r}_t)^2}, \tag{10}$$

where $\bar{r}_t$ denotes the benchmark historical average, and $\text{MSFE}^{\mathcal{M}}$ and MSFE^{BMK} are the mean squared forecast error of the model and the historical average, respectively.

The R^2_{OS} represents the % reduction in MSFE that a model achieves compared to predicting the historical average (H. Zhang et al., 2020). Clearly, for $R^2_{\text{OS}} > 0$, the model performs better than the historical average. To evaluate statistical significance, I tests the null hypothesis of $R^2_{\text{OS}} \leq 0$ using the test developed by Clark and West (2007). This test adjusts for the fact that an increased number of parameters in a model increases the variance, and is thus appropriate when comparing MSFEs of models with different sizes. For this test, let $\hat{e}_{\text{HA},t+1}$ and $\hat{e}_{\mathcal{M},t+1}$ denote the one-step ahead forecast errors of the historical average and model $\mathcal{M}$ forecast, respectively. In addition, let $\hat{f}_{t+1} = \hat{e}^2_{\text{HA},t+1} - [\hat{e}^2_{\mathcal{M},t+1} - (\bar{r}_{\text{HA},t+1} - \hat{r}_{\mathcal{M},t+1})^2]$ and let $\bar{f} = \frac{1}{p}\sum_{t=T+1}^{T+p}\hat{f}_t$, the sample average. Finally, let $\hat{\sigma}_f^2$ denote the sample variance of $\hat{f}_{t+1}$. This yields test statistic

$$CW = \sqrt{p}\,\frac{\bar{f}}{\hat{\sigma}_f}. \tag{11}$$

The usual critical values for a t-statistic can be used, although it should be kept in mind that

[6] Increasing k beyond 100 did not impact the results significantly

this will lead to a slightly undersized test (Clark & West, 2007).

I also compare the forecasting methods to each other. Specifically, I employ the forecast encompassing test developed by Harvey et al. (1998), which tests the null hypothesis that the forecasts of model 1 encompass those of model 2 against the one-sided alternative hypothesis that the model 1 forecasts do not encompass model 2 forecasts. Intuitively, the forecasts of model 2 are *encompassed* by those of model 1 if they contain no information that is not contained in the forecasts of model 1. Using the same notation as before, I define $d_{t+1} = (\hat{e}_{1,t+1} - \hat{e}_{2,t+1})\hat{e}_{1,t+1}$. Then, denoting $\bar{d} = (1/p)\sum_{t=T+1}^{T+p} d_t$, we get test statistic

$$MHLN = \frac{p-1}{p}\frac{\bar{d}}{\sqrt{\hat{V}(\bar{d})}}, \tag{12}$$

where $\hat{V}(\bar{d}) = p^{-2}\sum_{t=T+1}^{T+p}(d_t - \bar{d})^2$. Harvey et al. (1998) recommend using critical values from the t_{p-1}-distribution.

4.4.2 Economic Evaluation

Following, among others Campbell and Thompson (2008) and H. Zhang et al. (2020), I evaluate economic performance via utility gains for a risk-averse mean-variance investor who allocates his or her wealth between risky stock returns and a risk-free asset. Here, I explain and argue which choices I made, for a complete derivation of the equations below, see Appendix B. I assume an investor with a one-period investment horizon and relative risk aversion parameter γ. Assuming the investor maximizes expected utility, it can be shown that at the end of period T, he or she decides to allocate a share

$$w_{T+1}^{\mathcal{M}} = \frac{1}{\gamma}\left(\frac{\hat{r}_{T+1}^{\mathcal{M}}}{\hat{\sigma}_{T+1}^2}\right) \tag{13}$$

to equities in period $T+1$, where $\hat{\sigma}_{T+1}^2$ is his or her forecast of the variance of equity returns. Following, Campbell and Thompson (2008), the variance is estimated with the historical variance using a five-year rolling window; γ is set equal to 3; and I restrict $0 \leq w_{T+1}^{\mathcal{M}} \leq 1.5$. This imposes the realistic constraint of no short-selling or taking over 50% leverage.

For each portfolio, constructed with forecasts from model $\mathcal{M}$, I compute the certainty equivalent return as

$$\text{CER}_{\mathcal{M}} = \hat{u}_{\mathcal{M}} - \frac{\gamma}{2}\hat{\sigma}_{\mathcal{M}}^2, \tag{14}$$

where $\hat{u}_{\mathcal{M}}$ and $\hat{\sigma}_{\mathcal{M}}^2$ are the sample mean and variance of portfolio returns, respectively. This is the value of returns such that the investor is indifferent between acquiring this return *for sure* and acquiring the risky returns of the portfolio. Comparing the CER of the portfolio using model $\mathcal{M}$ with the benchmark (bmk) is done by computing the difference in CER and scaling it to an annualized percentage. I compute this as $\Delta(\text{ann}\%) = 1200 \times (\text{CER}_{\mathcal{M}} - \text{CER}_{\text{bmk}})$. In addition, I compute the Sharpe ratio of each portfolio, defined as the returns minus the risk-free rate divided by the standard deviation of returns.

The reason I economically evaluate performance is that the literature shows that R_{OS}^2 values tend to be small (Campbell & Thompson, 2008). Consequently, evaluating forecasts via an asset allocation exercise helps to determine whether improvements in performance are economically meaningful. In fact, the annualized percentage difference can be interpreted as the fee the investor would be willing to pay to get the information of model $\mathcal{M}$ (Campbell & Thompson, 2008). The reason I choose the mean-variance framework with a risk-averse investor is because of risk. As pointed out by Marquering and Verbeek (2004), implementing a trading strategy and evaluating the returns does not necessarily account for the risk the investor faces. The framework presented above explicitly incorporates risk aversion in both forming and evaluating the portfolio. Furthermore, the framework is attractive because of its simplicity. I do not argue that mean-variance is the best representation of an investor, nor do I claim it yields the highest return or utility. In fact, using neural networks or GARCH type specifications to forecast the variance may increase returns and utility (Marquering & Verbeek, 2004). But this is outside the scope of my thesis. I simply want an indication of whether a model provides an economically meaningful increase in performance.

5 Empirical Results

In this section, I present and compare the out-of-sample performance of the combination methods and machine learning models when estimated with expanding window, with average window (AveW) (Pesaran & Timmermann, 2007), and shrinking. Finally, I present the results on the optimization of the shrinkage factor.

5.1 Expanding Versus Average Window Estimation

Table 1 presents the out-of-sample evaluation of forecasts made with an expanding and average window. The RFR used 200 trees[7]. In panel A, we observe that, when estimated traditionally, the machine learning models outperformed the combination methods. However, only SVR produced significantly positive out-of-sample R-squared and was not able to beat the Pool-AVG benchmark. The advanced combination schemes performed worst.

Panel B of Table 1 presents the out-of-sample performance of the models when estimated with the AveW method from Pesaran and Timmermann (2007) with $m = 10$ windows and a minimum window size of 240 observations. In bold, we see that TM performs best, and the shaded cells indicate that only the simple combination methods outperformed the benchmark models. Next, in order of declining performance, we have the linear machine learning models, the nonlinear machine learning models, and then the advanced combination methods, with the exception of WALS.

Indicated by a dagger, all forecast methods except SVR improved when estimated with AveW instead of expanding window. The combination methods have improved more than the machine learning methods. This may indicate that (i) the violation of the stationary environment assumption has larger consequences for the combination methods, or (ii) the AveW technique to deal with structural breaks simply works best for the combination methods. Based on the relatively high performance of machine learning models with expanding window, option (i) seems most plausible.

[7] Increasing the number of trees beyond 200 did not improve performance a lot but would make the computational burden for AveW too high given the limited time to generate results

Table 1. *Forecast Evaluation With Expanding and Average Window*

Model	R^2_{os}(%)	*CW*	$R^2_{\text{os, rec}}$(%)	$R^2_{\text{os, exp}}$(%)
Panel A: Expanding window				
Pool-AVG	**0.41**	1.86**	1.10	0.11
kitchen sink	-7.92	0.31	-6.37	-8.60
TM	-0.49	0.65	0.38	-0.88
DMSFE	-0.44	0.64	0.51	-0.85
IV	-0.88	0.42	-0.38	-1.10
BMA	-2.07	-1.35	-3.68	-1.35
MMA	-4.28	-0.57	-4.14	-4.34
JMA	-2.06	-1.38	-1.77	-2.18
WALS	-4.21	0.15	-2.79	-4.83
LASSO	0.11	0.89	-0.16	0.22
Elastic Net	0.00	0.20	-0.08	0.04
SVR	0.12	1.70**	1.74	-0.60
RFR	-0.11	0.16	0.07	-0.19
Panel B: Average window				
Pool-AVG	0.46†	1.80**	1.41	0.03
kitchen sink	-1.88†	1.95	-2.62	-1.55
TM	**0.87**†	2.02**	2.98	-0.07
DMSFE	0.85†	1.95**	2.95	-0.07
IV	0.68†	1.93**	2.70	-0.21
BMA	-0.97†	0.89	0.34	-1.56
MMA	-0.56†	1.72	0.20	-0.90
JMA	-0.18†	1.62	0.84	-0.63
WALS	0.14†	1.92**	-0.20	0.29
LASSO	0.25†	1.14	1.84	-0.45
Elastic Net	0.26†	1.18	1.99	-0.50
SVR	-0.19	0.78	1.29	-0.83
RFR	-0.06†	0.50	1.31	-0.67

Note. This table reports out-of-sample R-squared values of forecasts for the equity premium over the period 1957:12-2016:12. *CW* is the MSFE-adjusted statistic from Clark and West (2007). $R^2_{\text{os,exp}}$ and $R^2_{\text{os,rec}}$ denote the out-of-sample R-squared during NBER dated expansions and recessions, respectively. Average window estimation from Pesaran and Timmermann (2007) uses $m = 10$ windows and minimum window size of 240 months. The simple combination methods trim off a total of 10% of forecasts. DMSFE uses a monthly discount factor θ=0.99. *p <0.1, **p <0.05, ***p <0.01. Numbers are rounded to two decimal places. Bold entries indicate that the respective model performs best for the given estimation technique. Shaded cells indicate that the respective model outperforms the Pool-AVG and kitchen sink benchmarks and has positive R^2_{OS}. † Indicates AveW improved performance over expanding window. Abbreviations: TM, trimmed mean; DMSFE, discounted mean squared forecast error; IV, inverse variance; BMA, Bayesian model averaging; MMA, Mallows model averaging; JMA, jackknife model averaging; WALS, weighted-average least squares; SVR, support vector regression; RFR, random forest regression.

The final two columns of Table 1 show the out-of-sample R-squared values during NBER dated recessions and expansions. There is a clear pattern of high and low predictability during recessions and expansions, respectively. A notable exception is WALS, which performed better during expansions and significantly outperforms the historical average.

Finally, comparing these results to H. Zhang et al. (2020), who use a different proxy for the equity premium, I find two differences. First, the difference in predictability between recessions and expansions is more pronounced in my results. Section 6 elaborates on predictability during good and bad times. Second, under the AveW estimation scheme, LASSO and elastic net performed better in H. Zhang et al. (2020). This may be the result of using chronological- rather than regular 5-fold cross-validation. It does not necessarily indicate that regular cross-validation should be used in this application. If no theoretical justification for ignoring the time series component of the data can be given, the results might not generalize to new data. However, an in-depth discussion on the use of cross-validation in time series is outside the scope of this paper, and I refer the interested reader to Bergmeir and Benítez (2012).

5.2 Average Window Estimation Plus Shrinking

Table 2 presents the out-of-sample R-squared and economic performance of forecasts estimated with AveW and shrunk towards the historical average. For AveW, $m = 10$ windows and a minimum window size of 240 observations are used. Shrinking is done by taking the arithmetic average of the forecast and the historical average, also called naive shrinking. RFR is estimated with 200 trees.

The results show that all combination methods except BMA significantly outperformed the benchmark historical average. Highlighted in bold, WALS performed best and was the only model that could outperform the benchmark kitchen sink and Pool-AVG models (indicated by the shaded gray cell). Comparing the forecasting methods further, we see that the machine learning models are dominated by the forecast combination methods. In addition, we see that the simple combination methods perform similar to the advanced ones; except for WALS, which performs better, and BMA which performs worse.

As indicated by the daggers, all methods except LASSO improved when shrinking was applied. The improvement is not the same across all models. The simple combination methods and machine learning models improved only moderately, while the advanced combination schemes experienced the largest improvement.

To further illustrate the effect of AveW estimation and shrinking, Figure 1 plots the DCSFE of the kitchen sink and one model from each category: IV, WALS, and LASSO. The DCSFE is the difference in cumulative squared forecast error of the historical average and the model under consideration. Higher DCSFE means better performance. The figure shows that AveW always improves upon expanding window estimation. In contrast, it shows that over specific horizons, AveW performed better without shrinking for IV, WALS, and LASSO. However, this period is followed by a decline in DCSFE for AveW, unmatched by the decline in DCSFE for shrinking. H. Zhang et al. (2020) explain that shrinking may improve performance by reducing variance in forecasts. My results additionally show that shrinkage stabilizes performance; AveW sometimes performs better, but over the entire evaluation sample shrinking improves forecasting.

Three final observations regarding Figure 1 are in place. First, the bad performance of kitchen sink without shrinking is largely explained by poor performance from 1995 onward. Second, from 1975 onward, the DCSFE of LASSO with expanding window stays quite constant. This indicates that many coefficients were set to zero and the model forecasted a value close to the mean, which equals the historical average. Third, as addressed by H. Zhang et al. (2020), there is a jump in performance during the October 1987 stock market crash. This shows that the models performed relatively well during periods of extreme movement in the market, a

Table 2. *Forecast Evaluation With Average Window and Shrinking*

Model	$R^2_{\text{os}}(\%)$	*CW*	$R^2_{\text{os, rec}}(\%)$	$R^2_{\text{os, exp}}(\%)$	$\Delta(\text{ann}\%)$	SR
Pool-AVG	0.25	1.80**	0.74	0.04	1.33	0.11
kitchen sink	1.13†	1.95**	0.99	1.19	2.64	0.13
TM	0.98†	2.02**	2.19	0.45	2.02	0.13
DMSFE	0.96†	1.95**	2.20	0.41	2.34	0.13
IV	0.96†	2.36***	2.16	0.42	2.36	0.13
BMA	0.10†	0.89	0.93	-0.27	1.21	0.10
MMA	0.90†	1.72**	1.54	0.62	2.29	0.13
JMA	0.78†	1.62**	1.53	0.45	1.90	0.12
WALS	**1.23**†	1.92**	1.11	1.28	2.47	0.13
LASSO	0.25	1.14	1.05	-0.10	1.00	0.10
Elastic Net	0.28†	1.18	1.15	-0.11	1.17	0.10
SVR	0.09†	0.78	0.70	-0.18	0.76	0.10
RFR	0.03†	0.50	0.70	-0.27	0.20	0.09

Note. This table reports out-of-sample R-squared values of forecasts for the equity premium over the period 1957:12-2016:12. CW is the MSFE-adjusted statistic from Clark and West (2007). $R^2_{\text{os,exp}}$ and $R^2_{\text{os,rec}}$ denote the out-of-sample R-squared during NBER dated expansions and recessions, respectively. Δ(ann%) is the annualized increase in utility for a mean-variance investor with relative-risk parameter $\gamma = 3$. SR denotes the Sharpe ratio. Average window estimation from Pesaran and Timmermann (2007) uses $m = 10$ windows and minimum window size of 240 months. Naive shrinking with $\delta = 0.5$ is used. **p* <0.1, ***p* <0.05, ****p* <0.01. Numbers are rounded to two decimal places. Bold entries indicate that the respective model performs best. Shaded cells indicate that the respective model outperforms the Pool-AVG and kitchen sink benchmarks. † Indicates shrinking improved performance over AveW. Abbreviations: TM, trimmed mean; DMSFE, discounted mean squared forecast error; IV, inverse variance; BMA, Bayesian model averaging; MMA, Mallows model averaging; JMA, jackknife model averaging; WALS, weighted-average least squares; SVR, support vector regression; RFR, random forest regression.

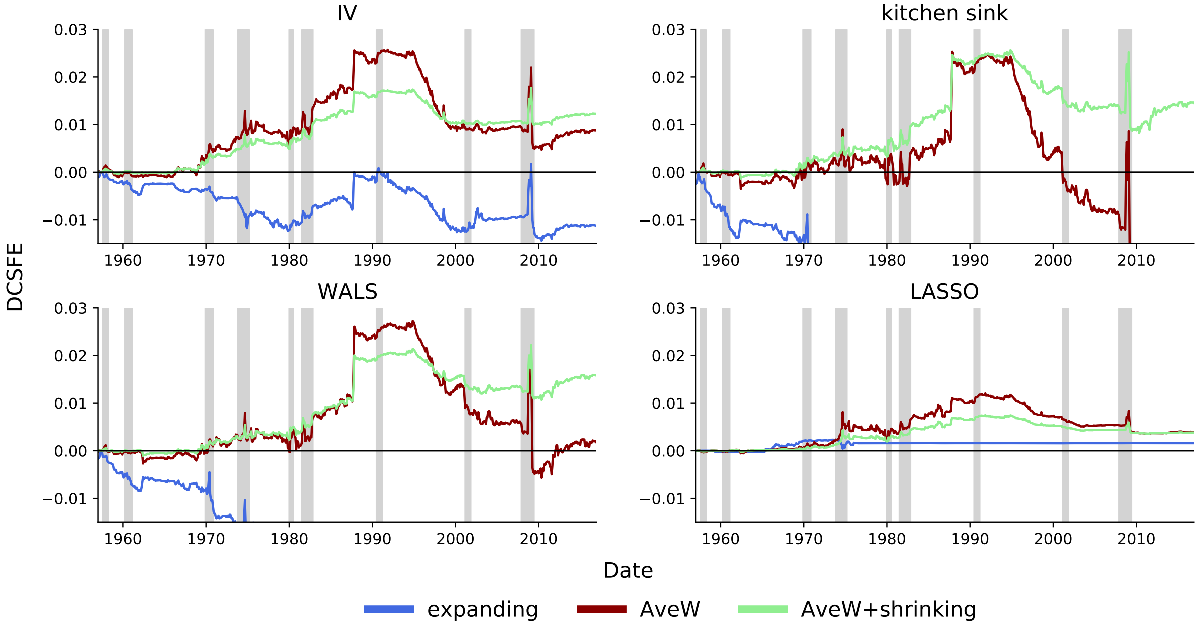
phenomenon elaborated on in Section 6.2.

Going back to Table 2, we see that most models were able to generate positive out-of-sample R-squared in both good and bad times, which was not the case without shrinking. Nevertheless, performance was higher during recessions than during expansions, a pattern consistent with the results from Rapach et al. (2010). Two notable exceptions are the kitchen sink model and WALS, which both perform well across business cycles.

The final two columns of Table 2 show the economic performance of the models. When considering a mean-variance investor with relative risk parameter γ=3 who allocates his or her wealth to stocks and risk-free bonds, all models produced an increase in annual utility. In fact, almost all combination methods produced utility gains of over 2%. Put differently, the investor would be willing to pay a portfolio management fee of over 2% if his or her funds were invested according to the strategy described in Section 4. This shows that the increase in performance is economically meaningful. Comparing the models to each other via utility increase or Sharpe ratio yields the same results as before. Advanced and simple combinations performed similarly, except for WALS (better) and BMA (worse). Machine learning models performed the worst, in particular the nonlinear ones. In terms of economic performance, none of the models outperformed the simple kitchen sink benchmark.

To put the economic performance in more perspective, I compare them to the 60-40 portfolio popularized by John Bogle which invests 60% in stocks and 40% in bonds. This method does not

Figure 1. *Cumulative Forecasting Performance Over Time*



Note. The figures plot the difference in cumulative squared forecast error (DCSFE) of the historical average benchmark and the given model. Higher DCSFE indicates better model performance. Evaluation period is 1957:01-2016:12. Average window (AveW) estimation from Pesaran and Timmermann (2007) uses $m = 10$ windows and minimum window size of 240 months. Shrinking uses $\delta = 0.5$. Shaded areas indicate a NBER dated recession. Expanding window line is cropped out to improve visibility. Abbreviations: IV, inverse variance; WALS, weighted-average least squares.

directly use information on stock returns or predictors and achieved an annualized utility gain of 1.11 and Sharpe ratio of 0.10. The annualized increase in utility of the forecast combination-based portfolios are roughly twice as high, and their Sharpe ratios are around 30% higher.

My results on AveW plus shrinking are qualitatively similar to H. Zhang et al. (2020). The only difference is that LASSO and elastic net were unable to significantly outperform the historical average in my results, while they did in H. Zhang et al. (2020). This may again be the result of the chronological cross-validation used in this paper.

5.3 Forecast Encompassing

In this section, I present the results of the forecast encompassing test from Harvey et al. (1998). Table 3 shows the p-values of the MHLN statistic using the forecasts over the period 1957:12 to 2016:12. I first compare the three classes to each other and thereafter to the benchmark Pool-AVG and kitchen sink.

Table 3. *Results Forecast Encompassing Test*

	Pool-AVG	kitchen sink	TM	DMSFE	IV	BMA	MMA	JMA	WALS	LASSO	elastic net	SVR	RFR
Pool-avg		0.27	0.71	0.71	0.65	0.20	0.38	0.43	0.50	0.44	0.42	0.16	0.03**
kitchen sink	0.04**		0.17	0.16	0.17	0.04**	0.18	0.14	0.45	0.05	0.05**	0.03**	0.03**
TM	0.04**	0.24		0.34	0.33	0.00***	0.25	0.18	0.42	0.04**	0.03**	0.03**	0.02**
DMSFE	0.05*	0.25	0.62		0.42	0.00***	0.26	0.19	0.43	0.05**	0.04**	0.03**	0.02**
IV	0.05*	0.27	0.63	0.51		0.00***	0.27	0.19	0.46	0.05**	0.04**	0.03**	0.03**
BMA	0.34	0.60	0.99	0.99	0.99		0.85	0.93	0.90	0.43	0.42	0.21	0.23
MMA	0.07*	0.49	0.36	0.35	0.36	0.04**		0.25	0.83	0.09*	0.08*	0.05**	0.05**
JMA	0.09*	0.42	0.55	0.53	0.55	0.03**	0.58		0.72	0.11	0.10*	0.06*	0.06*
WALS	0.04**	0.29	0.17	0.16	0.16	0.02**	0.05**	0.07*		0.05**	0.05**	0.04**	0.03**
LASSO	0.31	0.32	0.79	0.79	0.74	0.20	0.45	0.50	0.57		0.35	0.16	0.09
elastic net	0.34	0.33	0.82	0.82	0.76	0.20	0.46	0.52	0.59	0.61		0.17	0.10*
SVR	0.49	0.29	0.65	0.65	0.61	0.23	0.38	0.42	0.52	0.52	0.50		0.25
RFR	0.93	0.32	0.81	0.80	0.75	0.30	0.46	0.53	0.58	0.74	0.71	0.36	

Note. This table reports p-values of forecast encompassing test from Harvey et al. (1998). Tests the null hypothesis that the equity premium forecasts of the model in the first row encompasses the forecasts of the model in the first column, against the one sided alternative that it does not. The p-values are based on t_{p-1}-distribution. *p <0.1, **p <0.05, ***p <0.01. The out-of-sample evaluation period is 1957:12-2016:12. Average window estimation from Pesaran and Timmermann (2007) is used with $m = 10$ windows and minimum window size of 240 months. Naive shrinking with $\delta = 0.5$ is used. Numbers are rounded to two decimal places. Abbreviations: TM, trimmed mean; DMSFE, discounted mean squared forecast error; IV, inverse variance; BMA, Bayesian model averaging; MMA, Mallows model averaging; JMA, jackknife model averaging; WALS, weighted-average least squares; SVR, support vector regression; RFR, random forest regression.

Table 3 shows that we cannot reject the null that the forecasts from the simple combination methods (TM, DMSFE, and IV) encompassing those of the advanced weighting schemes and machine learning models. Similarly, we cannot reject the null that the forecasts from the advanced combination methods encompass the others. Unsurprisingly, BMA forms an exception and does not encompass any of the simple combination methods. Finally, the machine learning methods. For LASSO and elastic net, we reject the null that they encompass the simple averaging schemes and some of the advanced averaging methods. For the nonlinear SVR and RFR, we reject the null of forecast encompassing for almost averaging methods. These results confirm patterns we saw in Tables 1 and 2: with the exception of BMA, the simple and advanced combination methods performed similarly, linear machine learning models performed worse, and nonlinear models were the worst. With this, I obtain the answer to the first two research questions.

Following, I compare the models to the benchmark Pool-AVG and kitchen sink. Table 3 shows that the Pool-AVG does not encompass most of the averaging methods; conversely, for none of the models (except RFR) we reject the null that they encompass Pool-AVG. For the kitchen sink benchmark, we cannot reject the null of encompassing for any of the models. At the same time, for some of the models, we do reject the null that they encompass the kitchen sink. This is in line with the out-of-sample R-squared values we previously analyzed: most averaging methods can outperform the Pool-AVG but not the kitchen sink.

5.4 Optimal Shrinking

Table 4 compares the out-of-sample performance of different methods to select the shrinking factor. As explained in Section 4.3, a hold-out period was required and thus the evaluation period is 1980:01 to 2016:12, which differs from the period used in the main analysis. As expected, determining δ via an ordinary regression estimated with OLS lead to a decrease in R^2_{OS} for all models. Using ridge, a regularized regression, with 5-fold cross-validation for the regularization parameter α improved performance compared to OLS, but still underperformed compared to naive shrinking. When the observations were weighted exponentially to discount the past, performance stayed roughly the same for a monthly discount factor $\theta = 0.99$. However, Table 1 in Appendix C demonstrates that the performance is very sensitive to the choice of θ. Therefore, I would advise against using weighted Ridge.

Further analysis of weighted Ridge showed that the performance of some models improved when $\theta > 1$ was chosen, as displayed in column six of Table 4. These results are merely illustrative, as they lead to my final estimation method for δ, which is to use RANSAC estimation (Fischler & Bolles, 1981). I do not recommend using $\theta > 1$ unless an economic justification can be given for why distant observations should be counted more heavily than near ones.

A closer look at the estimated values of δ showed that $\theta > 1$ resulted in a lower variance in the series of estimates of δ. A possible explanation for this may be that the data contain some outlier-like values. When such a value enters the estimation window and is weighted heavily because $\theta < 1$, it affects the estimate of δ too much and creates instability. On the other hand, if $\theta > 1$ is used, observations that enter the estimation window affect δ less heavily, resulting in a more stable series of estimated δs, which positively affects the performance of shrinking. Therefore, I estimated the Ridge regression with the Random Sample Consensus

Table 4. *Forecast Evaluation for Different Shrinking Methods*

Model	δ=0.5	OLS	Ridge	Ridge (θ=0.99)	Ridge (θ=1.01)	Ridge + RANSAC
Pool-AVG	0.11	-0.67	-0.59	-0.36	-0.42	0.08
kitchen sink	1.29	0.56	0.84	0.82	1.42	1.37
TM	0.86	0.17	0.47	0.70	0.87	0.84
DMSFE	0.85	0.15	0.50	0.55	0.85	1.20
IV	0.84	0.11	0.42	0.60	0.82	0.85
BMA	-0.17	-0.60	-0.32	-0.48	-0.05	-0.81
MMA	0.98	0.07	0.47	0.60	0.97	0.78
JMA	0.69	-0.24	0.35	0.70	0.68	-0.16
WALS	1.56	0.78	0.99	0.93	1.23	1.84
LASSO	0.20	-0.05	-0.03	0.38	0.08	0.08
Elastic Net	0.19	-0.07	-0.07	0.44	-0.10	0.11
SVR	0.14	-0.19	-0.19	-1.35	-0.25	-0.57
RFR	-0.04	-0.26	-0.22	-0.61	-0.47	-0.31

Note. This table reports out-of-sample R-squared values of forecasts for the equity premium using Average window estimation from Pesaran and Timmermann (2007) ($m = 10$ windows, minimum window size is 240 months) and shrinking. Column header denotes method to select shrinking factor. 1957:12-1979:12 is used as first estimation sample to determine δ and expanded iteratively. The out-of-sample evaluation period is 1980:01-2016:12. θ Denotes the monthly discount factor in weighted Ridge. RANSAC is estimated using a proportion 0.8 of observations. Numbers are rounded to two decimal places. Shaded cells indicate improvement over naive shrinking. Abbreviations: TM, trimmed mean; DMSFE, discounted mean squared forecast error; IV, inverse variance; BMA, Bayesian model averaging; MMA, Mallows model averaging; JMA, jackknife model averaging; WALS, weighted-average least squares; SVR, support vector regression; RFR, random forest regression.

(RANSAC) algorithm, which is an outlier-robust estimation method often used in computer vision applications (Fischler & Bolles, 1981). This method randomly samples subsets of the data. Each subset contains a minimum of $p \times T$ observations, where p is a number between zero and one. The Ridge model is estimated over each subset. Then, for every model, you classify each observation from the *original* data as an inlier or outlier based on whether its residual is below or above some threshold value, respectively. I set this threshold equal to the median absolute deviation of returns. The model with the largest number of inliers is chosen as the best one, and the estimated δ of this model is used. The final column of Table 4 shows the performance of this method. As can be seen, the forecasting performance of some models improved. In fact, some performed better than naive shrinking, but there is no general pattern of improvement. Table 1 in Appendix C shows that for different values of p, different methods perform well. This may indicate that the method has potential, but a data-driven way to determine p should be used.

6 Sources of Equity Premium Predictability

My empirical results show that when model uncertainty and parameter stability are taken into account, and the variance of forecasts is decreased via shrinking, returns become predictable to some extent. At the intersection of finance and macroeconomics, predictability of returns is often argued to be the result of time-varying expected returns (Cochrane, 2005; Fama & French, 1989). Therefore, in this section, I further analyze my results and link predictability to economic theory.

6.1 Predictability in Good and Bad Times

One of the main characteristics of the equity premium is its counter-cyclical movement (Cochrane, 2005). therefore, I discuss two theories that relate business cycles to expected returns. The first is based on consumption. Fama and French (1989), argue that varying expected returns can be linked to business cycles via consumption smoothing or permanent income. According to permanent income models, consumption depends on wealth over a person's lifetime rather than his or her current income. During expansions, income may exceed wealth and leaves people with extra capital to invest. All else equal, this increases the supply of capital for (among others) investments in equity, which results in a lower return on equity investments, and vice-versa. In alternative link between consumption smoothing and expected returns is made in Lettau and Ludvigson (2001b). They argue that forward-looking investors will smooth out consumption-based on expected changes in asset wealth. Thus, if returns are expected to rise, consumption out of current assets and income goes up, and vice versa.

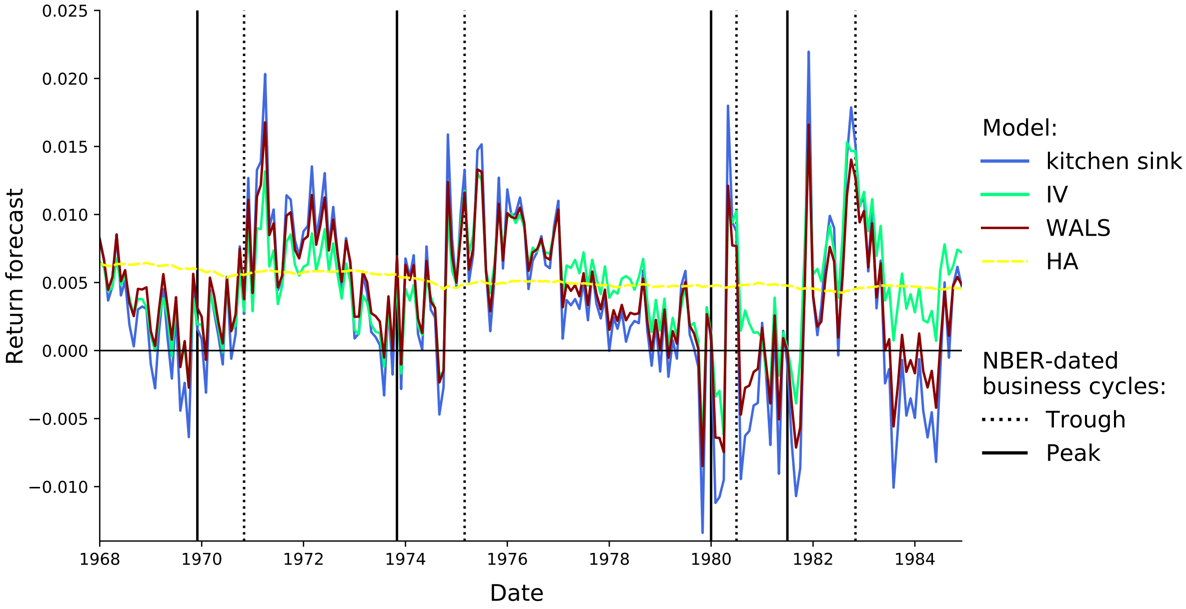
Second, time-varying expected returns can be linked to business cycles via changes in risk aversion. Campbell and Cochrane (1999) present an adjusted representative-agent consumption-based asset pricing model that includes slowly-moving consumer habits driven by past aggregate consumption. This model is able to explain many observed characteristics of stock prices, such as the countercyclical nature of returns. The model shows that risk aversion is counter-cyclical; hence, investors require a higher risk premium in bad times, and vice-versa. Consumption smoothing and time-varying risk aversion do not exclude each other, and may each explain a part of the variation in expected returns.

To illustrate the relationship between predictability and business cycles, Figure 2 displays the forecasts of the kitchen sink, IV, WALS, and historical average over the period 1968:01-1984:12. The models are estimated with AveW and shrinking. The figure also displays NBER-dated recessions and expansions. For the kitchen sink, IV, and WALS, a well-defined pattern is observed. From peak to trough, the forecasts had an increasing tendency and reaches a high peak at or just after the trough. Conversely, the equity premium forecasts had a strong declining tendency from trough to peak. This illustrates that the forecasts are closely related to business cycles. On the other hand, the historical average model is based on the idea of constant equity premium and does not incorporate any macroeconomic information. Using the theoretical relation between expected returns and business cycles, we can explain that the models gain an advantage over the historical average since they are able to track business cycles.

Cochrane (2005) argues that the hypothesis that predictability is caused by time-varying expected returns, which are linked to the real economy, is more plausible if the models are able to forecast macroeconomic conditions. Therefore, I follow H. Zhang et al. (2020) and forecast seven variables that represent the real economy using the original 12 economic variables.

The variables and their original sources are the CFNAI: the Chicago Fed National Activity Index collected from the Federal Reserve Bank of Chicago (1967:03 to 2017:12); the SRP: smoothed US recession probabilities, collected from the Federal Reserve Bank of St. Louis (1967:06 to 2017:12); IPG: the industrial production growth from the Federal Reserve Bank of St. Louis (1919:01 to 2017:12); the MU: Macroeconomic Uncertainty Index from Jurado et al. (2015) (1961:01 to 2017:12); GAP: the output gap, from the Federal Reserve Bank of St. Louis

Figure 2. *Equity Premium Forecasts With Average Window and Shrinking*



Note. The figure plots forecasts of the equity premium over the period 1968:01-1984:12. Average window estimation from Pesaran and Timmermann (2007) uses $m = 10$ windows and minimum window size of 240 months. Naive shrinking with $\delta = 0.5$ is used. Abbreviations: IV, inverse variance; WALS, weighted-average least squares; HA: historical average

(1949:01 to 2017:12); UNRATE: civilian unemployment rate collected from the Federal Reserve Bank of St. Louis (1948:01 to 2017:12); and finally, Cay: the Cointegrating residual between log-consumption, log-asset wealth, and log-labor income, introduced by Lettau and Ludvigson (2001a) and Lettau and Ludvigson (2001b) (quarterly data transformed to monthly data from 1952:01 to 2015:12). The data have been made available by H. Zhang et al. (2020)[8].

Following H. Zhang et al. (2020), I forecast four different forms of the macroeconomic variables. First, Y_{T+1}, the value of the variable in the next month. Second, Y_{T+12}, the value in a year from now. Third, $\Delta Y_{T+1} = Y_{T+1} - Y_{T-11}$, the change in the variable next month compared to a year ago. And fourth, $\Delta Y_{T+12} = Y_{T+12} - Y_T$, the change over the next year. I estimate the kitchen sink model with AveW, using $m = 10$ windows and $w_{min} = \lfloor (2/3)T \rfloor$, where T is the number of observations in the initial estimation sample. The forecasts are shrunk to the historical average with a shrinking factor $\delta = 0.5$.

Table 5 shows that our model, using AveW and shrinking, has significantly positive predictive power for all seven macroeconomic variables. This is not surprising in light of our earlier results. The predictive power provides evidence that predictability of returns stems from variations in macroeconomic conditions (Cochrane, 2005).

The difference in predictability across good and bad times observed in Table 2 is consistent with Neely et al. (2014). They conclude that macroeconomic predictors are particularly well at picking up increases in equity premium during a trough. Conversely, they say technical indicators are better at predicting equity premium declines during a business cycle peak. Since my models employ economic predictors, we can expect increased performance during business

[8] Data can be accessed via http://qed.econ.queensu.ca/jae/datasets/zhang002/

Table 5. *Forecast Evaluation of Macro Economic Conditions*

Variable	Evaluation period	Y_{T+1}	Y_{T+12}	ΔY_{T+1}	ΔY_{T+12}
CFNAI	2000:01–2016:12	48.47	12.68	15.56	22.11
SRP	2000:01–2016:12	41.56	11.85	15.38	22.93
IPG	1957:01–2016:12	63.16	63.50	49.78	23.36
MU	1995:01–2016:12	43.23	30.37	38.07	15.47
GAP	1980:01–2016:12	54.59	40.21	41.85	21.89
Cay	1985:01–2015:12	16.69	13.39	8.81	8.37
UNRATE	1980:01–2016:12	50.80	46.97	51.92	31.36

Note. This table reports the out-of-sample R-squared values for forecasting macroeconomic conditions with the kitchen sink model; the same model used for forecasting the equity premium. Average window estimation from Pesaran and Timmermann (2007) is used with $m = 10$ windows, minimum window size is two-thirds of the size of first estimation sample. Naive shrinking with $\delta = 0.5$ is used. All values are significant at the 1% level. Numbers are rounded to two decimal places. The macroeconomic conditions are: Chicago Fed National Activity Index, CFNAI; smoothed US recession probabilities, SRP; Macroeconomic Uncertainty Index (Jurado et al., 2015), MU; output gap, GAP; Cay from Lettau and Ludvigson (2001a) and Lettau and Ludvigson (2001b); and civilian unemployment rate, UNRATE

troughs. This implies that predictability, *in general*, need not be lower during peaks than during troughs. Instead, it may be the result of the chosen predictors.

6.2 Predicting Extreme Movement

In addition to consumption smoothing, Fama and French (1989) discuss the influence of risk in predictability of returns. In periods of high risk, investors demand a higher risk premium, thus expected returns increase. They argue that predictor variables like default spread, dividend yield, or term spread may form a proxy of risk, and thus track *that* part of expected returns that varies with business risk. Volatility in the equity premium may be a source of risk. Merton (1980) proposed a model with a positive relationship between the equity premium and volatility. Similar to Fama and French (1989), Merton (1980) argues that risk-averse investors require a positive compensation for positive levels of volatility. Note that this theory differs from Campbell and Cochrane (1999) because the explanation in Merton (1980) holds for a given level of risk aversion.

Recall that Figure 1 showed that model performance made a leap during a period of extreme movement. To further explore the effect of risk caused by extreme movements in the market, I evaluate forecasting gains during periods of extreme movement. Table 6 divides performance into periods where $|r_t| > 0.5, 1, 1.5, 2$, with r_t denoting the returns after standardization by subtracting the mean and scaling by the standard deviation. The models are estimated with AveW and shrinking. With the exception of BMA, SVR, and RFR, all models are able to generate significantly positive out-of-sample R-squared values for (almost) all periods. The table also shows a general pattern of higher R^2_{OS} statistics during more extreme periods. For example, the R^2_{OS} statistic for IV and MMA when $|r_t| > 2$ is more than double compared to when $|r_t| > 0.5$.

Table 6 also shows performance during periods of extreme negative returns: $r_t < -0.5, -1, -1.5, -2$. All models except SVR obtain highly significant R^2_{OS} statistics. The models perform even better in extreme negative periods than in regular extreme periods. However, is no clear pattern in performance across periods of extreme negative returns. That is, the R^2_{OS} statistics for $r_t < -0.5$ are not generally smaller or higher than for $r_t < -2$.

Table 6. *Forecast Evaluation During Periods of Extreme (Negative) Returns*

	Extreme returns				Negative returns			
Model	$\lvert r_t\rvert > 0.5$	$\lvert r_t\rvert > 1$	$\lvert r_t\rvert > 1.5$	$\lvert r_t\rvert > 2$	$r_t < -0.5$	$r_t < -1$	$r_t < -1.5$	$r_t < -2$
Pool-AVG	0.29**	0.38**	0.45**	0.55**	1.16***	1.12***	1.10***	0.96***
kitchen sink	1.87**	2.68**	3.01**	4.05*	5.35***	5.80***	6.48***	6.95***
TM	1.25**	1.78**	2.13**	2.71	4.29***	4.34***	4.43***	4.14***
DMSFE	1.23**	1.74**	2.10**	2.68**	4.26***	4.32***	4.45***	4.21***
IV	1.26**	1.79**	2.16**	2.79**	4.39***	4.47***	4.60***	4.32***
BMA	0.39	0.70	0.81	1.13	3.55***	3.41***	3.32***	2.78**
MMA	1.38**	2.04**	2.51**	3.16*	5.80***	5.81***	6.08***	5.91***
JMA	1.15**	1.71**	2.10**	2.62*	5.42***	5.23***	5.21***	4.80***
WALS	1.68**	2.16**	2.41*	3.25*	3.95***	4.29***	4.86**	5.37**
LASSO	0.44*	0.50*	0.60*	0.95*	1.24***	1.23***	1.39***	1.64***
elastic net	0.43*	0.50*	0.59	0.96*	1.21***	1.22***	1.40***	1.67***
SVR	0.20	0.05	-0.14	0.03	-1.43	-0.85	-0.46	-0.05
RFR	0.15	0.23	0.14	0.41*	0.77***	0.79***	0.70***	0.86***

Note. This table reports out-of-sample R-squared values for forecasting the equity premium using Average window estimation (Pesaran & Timmermann, 2007) ($m = 10$ windows, minimum window size is 240 months) and shrinking (naive with δ=0.5) for periods of extreme (negative) returns. Returns are normalized by subtracting the mean and scaling by standard deviation. *p <0.1, **p <0.05, ***p <0.01, based on *CW* statistic from Clark and West (2007). The out-of-sample period is 1957:12-2016:12. DMSFE uses monthly discount factor θ=0.99. Numbers are rounded to two decimal places. Abbreviations: TM, trimmed mean; DMSFE, discounted mean squared forecast error; IV, inverse variance; BMA, Bayesian model averaging; MMA, Mallows model averaging; JMA, jackknife model averaging; WALS, weighted-average least squares; SVR, support vector regression; RFR, random forest regression.

Based on the theory it is not surprising that the models perform better during more extreme periods. As argued previously, a component of the expected returns varies due to differences in risk. The models contain predictors or indicators of risk (Fama & French, 1989), allowing them to adapt their forecast based on changing risk conditions. The historical average, on the other hand, does not incorporate this information and assumes constant returns. As a result, the models are able to produce superior forecasts in these periods. When the returns become more extreme, the historical average ‘overlooks’ more information that *is* included in the models. Consequently, the difference in performance increases.

7 Robustness

In this section, I check whether my results are not dependent on the data or evaluation period used. In Appendix D I use the database from FRED-MD which contains monthly observations on 134 indicators of macroeconomic conditions in the US[9] to forecast the equity premium. Tables 1 and 2 in Appendix D display the same patterns we saw in Section 5. I also find that the difference in predictability across expansions and recessions is larger. Considering that the predictors are even more closely related to macroeconomic conditions than the Welch and Goyal (2008) data, it makes sense that most performance gains stem from recessions, as explained in Section 6.1.

In order to make my results comparable to the literature, I used data that span until 2016:12.

[9] The data is publicly available at https://research.stlouisfed.org/econ/mccracken/sel/

However, the data from Welch and Goyal (2008) have been updated and thus I check whether the results hold up in a new out-of-sample period 2017:01-2021:12. As it turns out, this is not the case. Therefore, I present and discuss the results here.

Table 7 contains the statistical and economical evaluation of forecasts of the equity premium. Since this period only contains 3 months of recessions, I do not evaluate performance during NBER dated recessions and expansions separately. The results show that the patterns observed in the original sample are not visible in the new period. In fact, most models seem to have no predictive power compared to the benchmark historical average. The nonlinear machine learning models form an exception and obtained positive out-of-sample R-squared values, although only significantly positive for SVR with expanding window. The Sharpe ratios are higher than in the original period. However, this can be attributed to the fact that in returns, in general, were higher over this period. For comparison, the 60-40 stocks-bonds portfolio attains a Sharpe ratio of 0.28.

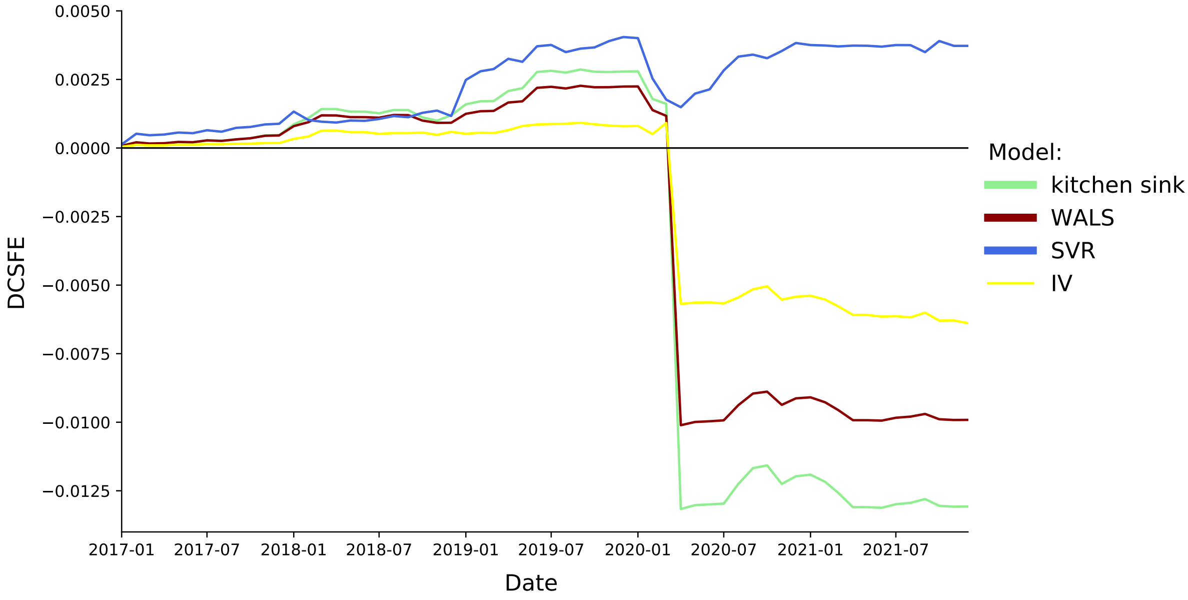
Table 7. *Forecast Evaluation Over New Period*

	Expanding		AveW		AveW+shrinking			
	R^2_{OS}	CW	R^2_{OS}	CW	R^2_{OS}	CW	$\Delta(\%ann)$	SR
Pool-AVG	-1.17	-1.20	-1.39	-1.03	-0.68†	-1.03	-1.38	0.24
kitchen sink	-10.20	-0.53	-27.00	-0.80	-10.76	-0.80	-2.89	0.21
TM	-6.72	-0.85	-10.67	-0.93	-4.78†	-0.93	-1.61	0.24
DMSFE	-6.97	-0.87	-11.28	-0.94	-5.04†	-0.94	-1.57	0.24
IV	-7.09	-0.84	-11.83	-0.93	-5.26†	-0.93	-1.6	0.24
BMA	-2.72	-1.21	-14.91	-0.91	-6.51	-0.91	-1.91	0.23
MMA	-6.22	-0.68	-20.23	-0.84	-8.43	-0.84	-1.93	0.23
JMA	-5.14	-0.95	-16.84	-0.85	-7.20	-0.85	-1.36	0.24
WALS	-7.35	-0.61	-19.78	-0.80	-8.16	-0.80	-2.36	0.22
LASSO	-0.01	-0.83	-4.32	-0.94	-2.05	-0.94	-2.20	0.22
elastic net	-0.01	-1.09	-5.46	-0.95	-2.57	-0.95	-2.26	0.22
SVR	**3.07**	1.77**	**1.08**	0.94	**0.64**	0.94	0.31	0.26
RFR	-0.81	-0.54	0.04	0.13	0.04	0.13	0.43	0.27

Note. This table reports out-of-sample R-squared values of forecasts for the equity premium over the updated period 2017:01-2021:12. CW is the MSFE-adjusted statistic from Clark and West (2007). Δ(ann%) is the annualized increase in utility for a mean-variance investor with relative-risk parameter $\gamma = 3$. SR denotes the Sharpe ratio. Average window estimation from Pesaran and Timmermann (2007) uses m=10 windows and minimum window size of 240 months. Naive shrinking with $\delta = 0.5$ is used. The simple combination methods trim off a total of 10% of forecasts. DMSFE uses a monthly discount factor θ=0.99. *p <0.1, **p <0.05, ***p <0.01. Numbers are rounded to two decimal places. Bold entries indicate that the respective model performs best for the given estimation technique. Shaded cells indicate that the respective model outperforms the Pool-AVG and kitchen sink benchmarks and attains positive R^2_{OS}. † Indicates that AveW+shrinking improves performance compared to expanding window. Abbreviations: TM, trimmed mean; DMSFE, discounted mean squared forecast error; IV, inverse variance; BMA, Bayesian model averaging; MMA, Mallows model averaging; JMA, jackknife model averaging; WALS, weighted-average least squares; SVR, support vector regression; RFR, random forest regression.

As the plot of the DCSFE in Figure 3 shows, most of the underperformance was realized in April 2020, during the stock market crash that resulted from the growing instability caused by the COVID-19 pandemic. This is in sharp contrast with the *leap* in performance of the models during the October 1987 stock market crash, as seen in Figure 1. This difference indicates that the performance of the methods applied in this paper does not respond equally to all not all crashes.

Figure 3. *Cumulative Forecast Performance in New Period*



Note. The figures plot the difference in cumulative squared forecast error (DCSFE) of the historical average benchmark and the given model. Higher DCSFE indicates better model performance. Evaluation period is 2017:01-2021:12. SVR is estimated with expanding window. Other models are estimated with Average window (AveW) from Pesaran and Timmermann (2007) using $m = 10$ windows and minimum window size of 240 months and shrinking with $\delta = 0.5$. Abbreviations: IV, inverse variance; WALS, weighted-average least squares; SVR, support vector regression

8 Conclusion

This thesis is an extension of the work of H. Zhang et al. (2020), who forecast the equity premium under model uncertainty and parameter instability. First, I compare the performance of simple forecast combination methods to the advanced methods used in H. Zhang et al. (2020) and find that without shrinking, but accounting for model uncertainty and parameter instability, they are superior. However, when shrinking is applied, WALS performs best and the other advanced methods perform similar to the simple combination methods. Second, I investigate how nonlinear machine learning models perform compared to combination methods. Using an expanding window, the nonlinear models perform a bit better. When parameter instability is accounted for, the forecast combination methods dominate the nonlinear machine learning models. Since these results are robust against using different predictor data, I recommend using WALS from Magnus et al. (2010) with AveW and shrinking when forecasting the equity premium.

Third, I look at the effect of AveW estimation and shrinking on the simple combination methods and nonlinear machine learning methods. The simple combination methods improve a lot under AveW estimation, indicating they suffered from parameter instability. This is not the case for the nonlinear models, which hardly improve or even decrease in performance. Both classes of models seem to only moderately benefit from shrinking, which contrasts with the large improvements we see for advanced combination methods. However, I show that shrinking stabilizes performance over time. As a result, I still recommend using shrinking, even though the benefits may not be as high as expected from H. Zhang et al. (2020). Regarding the first three research questions, I acknowledge that my research has limitations since the results do not hold in a sample that contains the stock market crash from 2020.

In my final extension, I try to optimize the shrinking factor. The results show that a regularized regression, as suggested by Liu et al. (2022), has potential when estimated with an outlier-robust method. Using RANSAC estimation, some models improved over naive shrinking, but no general pattern is present.

I also provide two economic reasons why my models do well. First, the results show that the models are able to track business conditions. This provides evidence for the hypothesis that the equity premium is predictable because a part of expected returns is linked to business cycles via consumption smoothing (Fama & French, 1989; Lettau & Ludvigson, 2001b) and time-varying risk aversion (Campbell & Cochrane, 1999). Second, the models perform well in extreme periods. Among others, Fama and French (1989) and Merton (1980) argue that part of expected returns varies with risk. Our results, therefore, add to the idea of Fama and French (1989) that the equity premium may be predictable because the models are able to track risk.

Finally, I provide two interesting avenues for future research. First, I suggest a data-driven approach be employed to select the optimal parameters of RANSAC to estimate a Ridge regression to select the shrinking factor. Alternatively, other outlier-robust estimation methods could be used. Second, consistent Neely et al. (2014) I find that macroeconomic predictor variables are particularly well at predicting increased levels of returns during troughs. Neely et al. (2014) also finds that technical indicators tend to perform well in periods of expansion. Consequently, I suggest that the AveW method and shrinking are used with both macroeconomic variables and technical indicators as this might result in improved performance across business cycles.

References

Assenmacher-Wesche, K., & Pesaran, M. H. (2008). Forecasting the Swiss economy using VECX models: An exercise in forecast combination across models and observation windows. *National Institute Economic Review*, *203*, 91–108.

Bai, J., & Perron, P. (1998). Estimating and testing linear models with multiple structural changes. *Econometrica*, 47–78.

Bai, J., & Perron, P. (2003). Computation and analysis of multiple structural change models. *Journal of applied econometrics*, *18*(1), 1–22.

Basener, B. (2020). *Bayesian Model Averaging Regression Tutorial.* Retrieved May 11, 2022, from https://www.kaggle.com/code/billbasener/bayesian-model-averaging-regression-tutorial

Bates, J. M., & Granger, C. W. J. (1969). The combination of forecasts. *Journal of the Operational Research Society*, *20*(4), 451–468. https://doi.org/10.1057/jors.1969.103

Bennett, K., & Mangasarian, O. (1992). Robust linear programming discrimination of two linearly inseparable sets. *Optimization Methods and Software*, *1*(1), 23–34.

Bergmeir, C., & Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, *191*, 192–213.

Breiman, L. (2001). Random forests. *Machine learning*, *45*(1), 5–32.

Buckland, S. T., Burnham, K. P., & Augustin, N. H. (1997). Model selection: An integral part of inference. *Biometrics*, 603–618.

Campbell, J. Y., & Cochrane, J. H. (1999). By force of habit: A consumption-based explanation of aggregate stock market behavior. *Journal of political Economy*, *107*(2), 205–251.

Campbell, J. Y., & Thompson, S. B. (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., & West, K. D. (2007). Approximately normal tests for equal predictive accuracy in nested models. *Journal of econometrics*, *138*(1), 291–311.

Cochrane, J. H. (2005). Financial markets and the real economy. *Foundations and Trends in Finance*, *1*(1), 1–101.

Colombo, E., & Pelagatti, M. (2020). Statistical learning and exchange rate forecasting. *International Journal of Forecasting*, *36*(4), 1260–1289.

Coulombe, P. G., Leroux, M., Stevanovic, D., & Surprenant, S. (2019). How is machine learning useful for macroeconomic forecasting? *Journal of Applied Econometrics.*

De Luca, G., & Magnus, J. R. (2011). Bayesian model averaging and weighted-average least squares: Equivariance, stability, and numerical issues. *The Stata Journal*, *11*(4), 518–544.

Elliot, G., Gargano, A., & Timmermann, A. (2013). Complete subset regressions. *Journal of Econometrics*, *177*(2), 357–373.

Fama, E. F., & French, K. R. (1989). Business conditions and expected returns on stocks and bonds. *Journal of financial economics*, *25*(1), 23–49.

Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, *24*(6), 381–395.

Granger, C., & Ramanathan, R. (1984). Improved methods of combining forecasts. *Journal of Forecasting*, *3*(2), 197–204.

Gu, S., Kelly, B., & Xiu, D. (2020). Empirical asset pricing via machine learning. *The Review of Financial Studies*, *33*(5), 2223–2273.

Hansen, B. E. (2008). Least-squares forecast averaging. *Journal of Econometrics*, *146*(2), 342–350.

Hansen, B. E., & Racine, J. S. (2012). Jackknife model averaging. *Journal of Econometrics*, *167*(1), 38–46.

Harvey, D. I., Leybourne, S. J., & Newbold, P. (1998). Tests for forecast encompassing. *Journal of Business & Economic Statistics*, *16*(2), 254–259.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (Vol. 2). Springer.

Hoeting, J. A., Madigan, D., Raftery, A. E., & Volinsky, C. T. (1999). Bayesian model averaging: A tutorial (with comments by M. Clyde, David Draper and E. I. George, and a rejoinder by the authors. *Statistical Science*, *14*(4), 382–417.

Jurado, K., Ludvigson, S. C., & Ng, S. (2015). Measuring uncertainty. *American Economic Review*, *105*(3), 1177–1216.

Kumar, K., & Magnus, J. R. (2013). A characterization of Bayesian robustness for a normal location parameter. *The Indian Journal of Statistics*, *75*(2), 216–237.

Leeb, H., & Pötscher, B. M. (2003). The finite-sample distribution of post-model-selection estimators and uniform versus nonuniform approximations. *Econometric Theory*, *19*(1), 100–142.

Lettau, M., & Ludvigson, S. (2001a). Resurrecting the (C)CAPM: A cross-sectional test when risk premia are time-varying. *Journal of Political Economy*, *109*(6), 1238–1287.

Lettau, M., & Ludvigson, S. (2001b). Consumption, aggregate wealth, and expected stock returns. *the Journal of Finance*, *56*(3), 815–849.

Lin, H., Wu, C., & Zhou, G. (2017). Forecasting corporate bond returns with a large set of predictors: An iterated combination approach. *Management Science*, *64*(9), 4218–4238.

Liu, L., Pan, Z., & Wang, Y. (2022). Shrinking return forecasts. *The Financial Review*, forthcoming.

Magnus, J. R., & De Luca, G. (2016). Weighted-average least squares (WALS): A survey. *Journal of Economic Surveys*, *30*(1), 117–148.

Magnus, J. R., Powell, O., & Prüfer, P. (2010). A comparison of two model averaging techniques with an application to growth empirics. *Journal of Econometrics*, *154*(2), 139–153.

Magnus, J. R., Wang, W., & Zhang, X. (2016). Weighted-average least squares prediction. *Econometric Reviews*, *35*(6), 1040–1074.

Makridakis, S., & Hibon, M. (2000). The M3-competition: Results, conclusions and implications. *International Journal of Forecasting*, *16*(4), 451–476.

Marquering, W., & Verbeek, M. (2004). The economic value of predicting stock index returns and volatility. *Journal of Financial and Quantitative Analysis*, *39*(2), 407–429.

McCracken, M., & Ng, S. (2016). FRED-MD: A monthly database for macroeconomic research. *Journal of Business & Economic Statistics*, *34*(4), 574–589.

Merton, R. C. (1980). On estimating the expected return on the market: An exploratory investigation. *Journal of financial economics*, *8*(4), 323–361.

Neely, C. J., Rapach, D. E., Tu, J., & Zhou, G. (2014). Forecasting the equity risk premium: The role of technical indicators. *Management science*, *60*(7), 1772–1791.

Noort, E., Geuzinge, T., Smilde, A., & Telgenkamp, E. (2022). *The Effects of Relaxed Assumptions on Forecasting the Gold Price* [Unpublished bachelor's seminar]. Erasmus University Rotterdam.

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. *Expert Systems with Applications*, *42*(4), 2162–2172.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Pesaran, M. H., & Pick, A. (2011). Forecast combination across estimation windows. *Journal of Business & Economic Statistics*, *29*(2), 307–318.

Pesaran, M. H., Schuermann, T., & Smith, L. V. (2009). Forecasting economic and financial variables with global VARs. *International journal of forecasting*, *25*(4), 642–675.

Pesaran, M. H., & Timmermann, A. (1995). Predictability of stock returns: Robustness and economic significance. *The Journal of Finance*, *50*(4), 1201–1228.

Pesaran, M. H., & Timmermann, A. (2007). Selection of estimation window in the presence of breaks. *Journal of Econometrics*, *137*(1), 134–161.

Rapach, D. E., Strauss, J., & Zhou, G. (2010). Ouf-of-sample equity premium prediction: Combination forecasts and links to the real economy. *The Review of Financial Studies*, *23*(2), 821–862.

Rapach, D. E., & Zhou, G. (2013). Forecasting stock returns. *Handbook of economic forecasting* (pp. 328–383). Elsevier.

Schrimpf, A., & Wang, Q. (2010). A reappraisal of the leading indicator properties of the yield curve under structural instability. *International Journal of Forecasting*, *26*(4), 836–857.

Schwarz, G. (1978). Estimating the dimension of a model. *The annals of statistics*, 461–464.

Sermpinis, G., Stasinakis, C., Theofilatos, K., & Karathanasopoulos, A. (2014). Inflation and unemployment forecasting with genetic support vector regression. *Journal of Forecasting*, *33*(6), 471–487.

Smola, A., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, *14*, 199–222.

Stock, J. H., & Watson, M. W. (1996). Evidence on structural instability in macroeconomic time series relations. *Journal of Business & Economic Statistics*, *14*(1), 11–30.

Stock, J. H., & Watson, M. W. (2004). Combination forecasts of output growth in a seven-country data set. *Journal of forecasting*, *23*(6), 405–430.

Tibshirani, R. (1996). Regression srinkage and selection via the LASSO. *Journal of the Royal Statistical Society*, *58*(1), 267–288.

Timmermann, A. (2006). Forecast Combinations. *Handbook of economic forecasting* (pp. 135–196). Elsevier.

Vapnik, V. (1995). *The nature of statistical learning theory*. Springer.

Welch, I., & Goyal, A. (2008). A comprehensive look at the emperical performance of equity premium prediction. *The Review of Financial Studies*, *21*(2), 1455–1508.

Xiang-rong, Z., Long-ying, H., & Zhi-sheng, W. (2010). Multiple kernel support vector regression for economic forecasting. *2010 International Conference on Management Science & Engineering 17th Annual Conference Proceedings*, 129–134.

Zhang, H., He, Q., Jacobsen, B., & Jiang, F. (2020). Forecasting stock returns with model uncertainty and parameter instability. *Journal of Applied Econometrics*, *35*(5), 629–644.

Zhang, X., Wan, A. T., & Zou, G. (2013). Model averaging by jackknife criterion in models with dependent data. *Journal of Econometrics*, *174*(2), 82–94.

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society*, *67*(2), 301–320.

Appendix

A Advanced Forecast Combination Methods and Machine Learning

In this appendix, I provide further details on the forecast combination methods and machine learning algorithms used in this thesis.

A.1 Bayesian Model Averaging

The first forecast combination method from the class of advanced methods is the Bayesian Model Averaging (BMA) combination method. For an in-depth discussion and tutorial on BMA, I refer you to Hoeting et al. (1999). Here, I only explain the basics necessary to obtain the BMA weights. With BMA, the weights are determined by the posterior probabilities of the models. Let $P(\mathcal{M}_i)$ denote the prior probability that model $\mathcal{M}_i$ is the true model, and let $P(R|\mathcal{M}_i)$ denote the probability of observing returns $R = (r_1, ..., r_T)'$ when $\mathcal{M}_i$ is the true model. The posterior probability of model $\mathcal{M}_i$ being the true model is thus $P(\mathcal{M}_i|R)$. H. Zhang et al. (2020) explain that by applying Bayes rule, we get the weights

$$\omega_i^{\text{BMA}} = P(\mathcal{M}_i|R) = \frac{P(\mathcal{M}_i)P(R|\mathcal{M}_i)}{\sum_{j=1}^{m} P(\mathcal{M}_j)P(R|\mathcal{M}_j)}, \qquad i = 1, ..., m. \tag{15}$$

Following H. Zhang et al. (2020), I then use the assumptions of equal model priors (i.e., $P(\mathcal{M}_i) = \frac{1}{m}$) and diffuse model priors on parameters. The latter can be interpreted as the model parameters being equal across models. Buckland et al. (1997) show that under these assumptions the BMA weights can be approximated by

$$\omega_i^{\text{BMA}} = \frac{\exp\{-\frac{1}{2}BIC_i\}}{\sum_{j=1}^{m} \exp\{-\frac{1}{2}BIC_j\}}, \qquad i = 1, ..., m, \tag{16}$$

where BIC_i is the Bayesian Information Criterion for model $\mathcal{M}_i$, introduced by Schwarz (1978). It is computed as $BIC_i = Tlog(\hat{\sigma}_i^2) + log(T)(N_i + 1)$, where T is the number of observations, N_i the number of predictors in model i (thus $N_i + 1$ is the number of free parameters), and $\hat{\sigma}_i^2$ the Maximum Likelihood estimate of the residual variance in model $\mathcal{M}_i$. Using forecast combination with BMA weights, I combine all possible kitchen sink models. Using N predictors, that gives $m = 2^N - 1$ possible models with constant and at least one predictor.

A.2 Mallows Model Averaging

The weights in frequentist model averaging (FMA) methods do not require any priors and are completely determined by the data H. Zhang et al. (2020). The first FMA method I use is Mallows Model Averaging. Hansen (2008) developed a forecast combination method where the weights of the forecasts are determined by minimizing a Mallows criterion. This criterion is appealing because Hansen (2008) shows that it is an asymptotically unbiased estimate of the in-sample mean squared error and the out-of-sample mean squared forecast error. Using the linear regression models like in Equation (1) for individual forecasts, we note that the combination forecast can be written as

$$\hat{r}^{\mathrm{c}}_{T+1} = \sum_{i=1}^{m} \omega_i \hat{r}^{(i)}_{T+1} = X_t \hat{\beta}(\omega), \tag{17}$$

where $\hat{\beta}(\omega)$ is the $((N+1) \times 1$ averaging estimator of β and is a function of the weights ω. Now let $K_i = N_i + 1$, the number of parameters in model $\mathcal{M}_i$, and $K = (K_1, ..., K_m)'$. Then Mallows criterion is defined as

$$C_p(\omega) = (R - X\hat{\beta}(\omega))'(R - X\hat{\beta}(\omega)) + 2\hat{\sigma}^2 \omega' K, \tag{18}$$

where R is a $T \times 1$ vector of returns, X is a $T \times (N+1)$ matrix of N observed variables over time and a constant, and $\hat{\sigma}^2$ is the unbiased estimate of the residual variance in the largest model. The weights of the MMA method are the values that minimize $C_p(\omega)$ and thus

$$\omega^{\mathrm{MMA}} = \underset{\omega \in \mathcal{H}}{\operatorname{argmin}} \{C_p(\omega)\}, \tag{19}$$

where $\mathcal{H} = \{\omega \in \mathbb{R}^m : \omega_i \geq 0, \sum_{i=1}^{m} \omega_i = 1\}$. Although MMA can be used with all $2^N - 1$ possible models, to my knowledge the theory behind the method has only been developed for nested models (see, Hansen (2008)). Moreover, since the weights of the MMA are the solution of a quadratic programming problem, a large number of models entails a large computational burden. Therefore, I use N nested models rather than all $2^N - 1$ available models. Similarly to H. Zhang et al. (2020), the nesting order is determined based on the coefficients of a LASSO regression.

A.3 Jackknife Model Averaging

The second FMA method I employ is the jackknife model averaging (JMA) method developed by Hansen and Racine (2012) and X. Zhang et al. (2013). The idea is as follows: the observation (r_T, X_{t-1}) is omitted and all models are estimated on the remaining data. The estimated models are used to forecast r_t using X_{t-1}. The weights deletedwe estimate the i-th model. Let us consider model i first. Let $\tilde{R}^{(i)} = (\tilde{r}^{(i)}_1, ..., \tilde{r}^{(i)}_T$ denote the Jackknife estimator of returns in model $\mathcal{M}_i$.

First, we need to obtain the jackknife residuals of each model. For this, observation (r_t, X_{t-1}) is omitted and model $\mathcal{M}_i$ is estimated on the remaining observations. The estimated model is used to forecast r_t based on X_{t-1}. This gives the jackknife estimator of r_t in model $\mathcal{M}_i$

$$\hat{r}^{(i)}_t = X_t \hat{\beta}_{(-t)}, \tag{20}$$

where $\hat{\beta}_{(-t)}$ denotes the estimate of β when observation (r_t, X_{t-1}) is left out. The jackknife estimator is obtained for $t = 1, ..., T$. We get the jackknife estimator of R in model $\mathcal{M}_i$ as $R^{(i)} = (\hat{r}^{(i)}_1, ..., \hat{r}^{(i)}_T)$. Finally, this gives $\tilde{e}^{(i)} = R - R^{(i)}$, the jackknife residual for model i. The residuals for all models are collected in $\tilde{e} = (\tilde{e}^{(1)}, ..., \tilde{e}^{(m)})$. The jackknife residuals are combined using the forecast combination weights ω, which gives the averaging residuals

$$\tilde{e}(\omega) = \sum_{i=1}^{m} \omega_i \tilde{e}^{(i)} = \tilde{e}\omega. \tag{21}$$

Intuitively, you want to set the weights ω such that the averaging residuals are small. For this purpose, the leave-one-out (or jackknife) cross-validation criterion is used, which is defined as

$$CV_t(\omega) = \frac{1}{T}\tilde{e}(\omega)'\tilde{e}(\omega). \tag{22}$$

The JMA weights are obtained by minimizing the jackknife cross-validation criterion and can thus be expressed as

$$\omega^{\text{JMA}} = \underset{\omega\in\mathcal{H}}{\text{argmin}}\big\{CV_t(\omega)\big\}, \tag{23}$$

where $\mathcal{H}$ is defined the same as in Equation (19). Similar to MMA, I only use nested models for JMA.

A.4 Weighted-Average Least Squares

Weighted-average least squares was developed by Magnus et al. (2010) and offers a combination of Bayesian and frequentist averaging techniques. The computational complexity of estimating WALS is linear in the number of predictors, rather than exponential, like BMA and the simple combination methods (Magnus et al., 2010). Compared to BMA, WALS offers an attractive way to deal with ignorance regarding the model priors, leading to bounded risk (Magnus et al., 2010). I follow H. Zhang et al. (2020) and use the reflected Weibull distribution as prior; they motivate this choice based on Magnus and De Luca (2016). To explain the WALS method, I first rewrite Equation (1) into matrix notation and split the predictor matrices. This gives

$$R = X_1\beta_1 + X_2\beta_2 + e \tag{24}$$

where R is the $T \times 1$ vector of returns, X_1 is a $T \times (N_1 + 1)$ matrix of a constant and N_1 predictors the model should always include, X_1 is a $T \times N_2$ matrix of predictors that may be included in the model. The second step is to orthogonalize the columns in X_2. Define $M_1 := I_T - X_1(X_1'X_1)^{-1}X_i'$ where I_T is a $T \times T$ identity matrix. Then, let P be an orthogonal $N_2 \times N_2$ matrix such that $P'X_2'M_1X_2P = \Lambda$, for Λ a diagonal matrix with positive diagonal elements. Then define

$$X_2^* := X_2P\Lambda^{-1/2}, \text{and} \tag{25}$$

$$\beta_2^* := \Lambda^{-1/2}P'\beta_2. \tag{26}$$

Then $R = X_1\beta_1 + X_2^*\beta_2^* + e$, since $X_2^*\beta_2^* = X_2\beta_2$. Moreover, the transformed variables are orthogonal in the sense that $X_2^{*\prime}M_1X_2^* = I_{N_2}$. Denoting $\beta_2^* = (\beta_{2,1}^*, ..., \beta_{2,N_2}^*)$, Magnus et al.

(2010) show that the WALS estimators of β_1 and β_2 in Equation (24) can be obtained as

$$\hat{\beta}_1 = \hat{\beta}_{1r} - Q^* \hat{\beta}_2^* \tag{27}$$

$$\hat{\beta}_{2,h}^* = \hat{\beta}_{2u,h}^* - \hat{\sigma}_h \frac{A_1\left(\frac{\hat{\beta}_{2u,h}^*}{\hat{\sigma}_h}\right)}{A_0\left(\frac{\hat{\beta}_{2u,h}^*}{\hat{\sigma}_h}\right)}, \qquad h = 1, ..., N_2, \tag{28}$$

where $\hat{\beta}_{1r} = (X_1'X_1)^{-1}X_1'R$, $Q^* = (X_1'X_1)^{-1}X_1'X_2^*$, $\hat{\beta}_{2u,h}^* = X_2^{*\prime}M_1R$, $\hat{\sigma}_h$ the standard error of $\beta_{2,h}^*$, and

$$A_j(x) = \int_{-\infty}^{+\infty} (x-\gamma)^j \phi(x-\gamma)\pi(\gamma)d\gamma, \qquad j = 0, 1, \tag{29}$$

where $\phi(\cdot)$ denotes the standard normal pdf, and $\pi(\cdot)$ is the pdf of the reflected Weibull distribution. The reflected Weibull distribution is defined as

$$\pi(\gamma) = \frac{qc}{2}|\gamma|^{-(1-q)} exp\{-c|\gamma|^q\}, \tag{30}$$

where $q = 0.8876$ and $c = log(2)$. The expressions in Equations (27) and (28) are based on the equivalence theorem from Magnus et al. (2010) and Magnus et al. (2016).

A.5 LASSO

The Least Absolute Shrinkage and Selection Operator (LASSO) model, introduced by Tibshirani (1996), is a linear model that. LASSO estimation estimates the coefficients β in Equation (1), but applies L_1-regularization. This means LASSO penalizes the parameter β by adding the L_1 norm, denoted as $\|\beta\|_1^1$, to the loss function. As such, the estimated coefficients are defined as

$$\hat{\beta}_{\text{LASSO}} = \underset{\beta}{\text{argmin}} \Big\{ \frac{1}{T} \sum_t (r_{t+1} - X_t\beta)^2 + \lambda \|\beta\|_1^1 \Big\}, \tag{31}$$

where λ is the regularization parameter. A high value of λ means a high regularization, i.e., the parameter values are shrunk to zero more heavily. Moreover, for high enough λ, LASSO sets some coefficients to zero and thus works as a variable selection technique (Tibshirani, 1996). The optimal regularization parameter is determined via five-fold cross validation.

As opposed to H. Zhang et al. (2020), I perform cross-validation with a rolling window since we are dealing with time series. In regular k-fold cross-validation, the training data is split in k roughly equal parts. For a given value of the hyperparameter, the model is estimated on $k - 1$ parts of the data and evaluated on the data that is left out. This is done for all k combinations of estimation and evaluation data and the evaluation scores are averaged. The procedure is repeated for all indicated values of the hyperparameter. The hyperparameter for which the highest average score is attained is then said to be the optimal one. In a time-series setting, the data come with a chronological component. Therefore, cross-validation is performed with a chronological split in the data. This means the data are split in k equal parts, ordered by time. For given hyperparameters, the model is first estimated on the first part and evaluated on the second, then estimated on the first two parts and evaluated on the third, and so on. This way, the evaluation data always comes after the estimation data, replicating reality.

A.6 Elastic Net

In addition to L_1-regularization, the Elastic Net method, introduced by Zou and Hastie (2005), applies L_2-regularization. With L_2-regularization, the squared norm of β is added to the loss function. Denoting the squared norm by $\left\|\beta\right\|_2^2$, the Elastic Net estimate of β is given by

$$\hat{\beta}_{\text{EN}} = \underset{\beta}{\operatorname{argmin}}\Big\{\frac{1}{T}\sum_t (r_{t+1} - X_t\beta)^2 + \lambda\big(\frac{1-\alpha}{2}\|\beta\|_1^1 + \alpha\|\beta\|_2^2\big)\Big\}, \tag{32}$$

where λ is again the regularization parameter and α controls how weight is divided between the two penalty terms. I follow H. Zhang et al. (2020) by setting $\alpha = 0.5$ and determining λ by five-fold cross validation, again using a chronological split. Zou and Hastie (2005) show that the Elastic Net mimics the variable selection property of the LASSO, while resolving some of its limitations.

A.7 Support Vector Regression

The first nonlinear machine learning model that I will employ is the Support Vector Regression (SVR). In this section, I describe the ε-SVR, introduced by Vapnik (1995). The explanation may contain overlap with the one I give in Noort et al. (2022), but I largely modified it for the current application.

Similar to regression, an SVR model estimates a function $f(X_t)$ such that $f(x_t) \approx r_{t+1}$. However, SVR uses an ε-intensive loss function, rather than squared loss. This penalty function was introduced by Bennett and Mangasarian (1992) and is denoted by ξ_t. If r_{t+1} is within a margin ε of $f(x_t)$, no penalty is given, and after that a linear penalty the size or the distance to the margin is given. The hyperparameter ε is set via 5-fold cross-validation with chronological splits of the data. SVR also uses L_2-regularization to prevent overfitting and increase the generalizability of the model. This yields a loss function:

$$L(w) = \|w\|^2 + C\sum_{t=1}^{T}\xi_t, \tag{33}$$

where C is the regularization parameter and is determined via 5-fold cross-validation with chronological split.

What makes SVR work well, is its ability to incorporate nonlinearity via a kernel function (Coulombe et al., 2019; Sermpinis et al., 2014), denoted $k(x_t, x)$. Smola and Schölkopf (2004) provides an overview of kernel functions and the so called kernel trick. For this application, it is enough to understand that the kernel trick provides an efficient way to map you input data to a higher dimensional space. The result is that the estimated relation is nonlinear in the original inputs. The chosen kernel thus determines the possible shapes . Following Colombo and Pelagatti (2020), I use the radial basis function kernel (RBF): $k(x_t, x) = \exp(-\gamma\|x_t - x\|)$. The RBF introduces a hyperparameter, γ. Intuitively, γ determines the influence a single observation x_s has on the value $f(x_t)$ when $s \neq t$. I use the default value of γ in the python module sklearn, which is $\gamma = 1/(\text{n\_observations} \times \text{n\_predictors})$.

A.8 Random Forest Regression

The second nonlinear machine learning model that I employ is the non-parametric Random Forest, developed by Breiman (2001). A random forest regression (RFR) is an ensemble consisting of multiple regression trees. I thus first explain the regression tree. My explanation is based on Hastie et al. (2009) and uses the same notation as previous sections. For further details and examples, I refer to them.

A regression tree iteratively partitions the feature space into rectangles. Starting at the root node, it decides the feature that is split on, and the specific split value, also named threshold. For example, you can split based on whether the inflation is above or below 2%; the feature space is then partitioned into two regions. The tree is then grown iteratively: the two subsets of the feature space, both represented by a node in the tree, are separately split into new regions, and so on. Suppose we have partitioned the feature space into M non-overlapping regions $R_1, \ldots, R_M$. In region R_m, we model the dependent variable by a constant c_m, which is equal to the mean value of all observed dependent variables r_{t+1} whose predictor x_t is in region R_m. Mathematically, the return r_{t+1}, with observed variable x_T is forecasted as

$$f(x_T) = \sum_{m=1}^{M} c_m I(x_T \in R_m). \tag{34}$$

The splits at each node are determined via the following greedy algorithm: for every predictor variable j, we consider all possible splitting values s. The feature space is then split into $R_1(j,s) = \{x_t | x_t, j \leq s\}$ and $R_2(j,s) = \{x_t | x_t, j > s\}$. Set prediction constants $c_i = mean(r_{t+r} | x_t \in R_i)$. Subsequently, choose j^* and s^* such that the sum of squares in the resulting nodes is minimized. That is,

$$(j^*, s^*) = \underset{j,s}{\operatorname{argmin}} \Big\{ \sum_{x_t \in R_1(j,s)} (r_{t+1} - c_i)^2 + \sum_{x_t \in R_2(j,s)} (r_{t+1} - c_2)^2 \Big\}. \tag{35}$$

An important issue is how deep to grow the tree. A shallow tree may not capture the structure in the data well, while a very deep tree is likely to overfit. I control the size of the tree by limiting how deep the tree can grow. Gu et al. (2020) show that for financial time series, on average the optimal number of layers is one to five. This means rather shallow trees, which they explain by the high noise-to-signal ratio in financial time series. I determine maximum depth via 5-fold cross-validation with chronological split.

Hastie et al. (2009) also explain how a random forest from Breiman (2001) is created. In a random forest k trees are created. For each tree, a bootstrap sample of size T is taken from the original data. On each sample, a tree is grown. However, when growing the tree, at each node only $p \leq N$ of the predictor variables are available to split on. The available predictors are randomly chosen at each node. For a new observation X_T, each tree in the random forest is used to forecast and the outputs are averaged into one final forecast. The idea behind bootstrapping the data and then averaging (Bagging) the results is to reduce the variance of the final estimator. Hastie et al. (2009) show that the random feature selection aims at decreasing the variance by decorrelating the trees while increasing the variance of the individual trees by only a minimal amount.

Compared to regression trees, the random forest add two additional hyperparameters: k and q. First, Breiman (2001) show that adding more trees does not cause overfitting, but is computationally more heavy. Therefore, k is set such that increasing it only gives minor improvement. Second, q can be determined via a five-fold grid search.t

B Mean-Variance Investor and Economic Evaluation

Following H. Zhang et al. (2020), I evaluate economic performance via utility gains for a risk-averse mean-variance investor who allocates his wealth between risky stock returns and a risk-free asset. This section provides a more thorough derivation of the asset allocation exercise presented in Section 4.4. Assume an investor with a one-period investment horizon who chooses at the end of period T what share $w_{T+1}^{\mathcal{M}}$ of his portfolio is allocated to equities in period $T+1$. The $\mathcal{M}$ indicates that the investor makes forecasts of the equity premium with model $\mathcal{M}$. The remaining share $1-w_{T+1}^{\mathcal{M}}$ is allocated to risk-free bills. Since r_{T+1} denotes stock returns in excess of the risk-free rate, this gives a total realized return of

$$R_{p,T+1}^{\mathrm{M}} = w_{T+1}^{\mathrm{M}}(r_{T+1} + r_{T+1}^{f}) + (1 - w_{T+1}^{M})r_{T+1}^{f} = w_{T+1}^{\mathrm{M}} r_{T+1} + r_{T+1}^{f}, \tag{36}$$

where r_{T+1}^{f} is the risk-free rate. I assume a mean-variance investor with relative risk parameter γ. Assume that the investors utility is given by the return minus the risk. The source of risk is the variance of equity returns, denoted as σ_{T+1}^2. Since the investor is only exposes a fraction w_{T+1}^{M} of his portfolio to equity, the variance in his portfolio amounts to $(w_{T+1}^{\mathrm{M}})^2\sigma_{T+1}^2$. Scaling the variance by a half times the risk parameter gives risk

$$\frac{\gamma}{2}(w_{T+1}^{\mathrm{M}})^2\sigma_{T+1}^2. \tag{37}$$

Subtracting the risk from the returns gives total utility at time $T+1$ of

$$U_{T+1} = R_{p,T+1}^{\mathrm{M}} - \frac{\gamma}{2}(w_{T+1}^{\mathrm{M}})^2\sigma_{T+1}^2. \tag{38}$$

The investor maximizes the expected utility. Thus, forecasts of equity premium, r_{T+1}, and volatility, σ_{T+1}^2, are needed. Returns are forecasted by model $\mathcal{M}$, denoted as $\hat{r}_{T+1}^{\mathcal{M}}$. As suggested by Campbell and Thompson (2008), the variance is estimated using a five-year rolling window of returns and denoted $\hat{\sigma}_{T+1}^2$. This gives an expected utility

$$U_{T+1} = w_{T+1}^{\mathrm{M}}\hat{r}_{T+1}^{\mathcal{M}} + r_{T+1}^{f} - \frac{\gamma}{2}(w_{T+1}^{\mathcal{M}})^2\hat{\sigma}_{T+1}^2, \tag{39}$$

which we maximize by taking the derivative with respect to w_{T+1}^{M} and setting it equal to zero. This gives first order condition:

$$\frac{\partial U_{T+1}}{\partial w_{T+1}^{\mathcal{M}}} = r_{T+1}^{\mathcal{M}} - \gamma w_{T+1}^{\mathcal{M}}\hat{\sigma}_{T+1}^2 = 0, \tag{40}$$

which is solved for $w_{T+1}^{\mathcal{M}}$ by

$$w_{T+1}^{\mathrm{M}} = \frac{1}{\gamma}\left(\frac{\hat{r}_{T+1}^{\mathcal{M}}}{\hat{\sigma}_{T+1}^2}\right). \tag{41}$$

Finally, the second-order condition

$$\frac{\partial^2 U_{T+1}}{\partial (w_{T+1}^{\mathcal{M}})^2} = -\gamma\hat{\sigma}_{T+1}^2 < 0, \tag{42}$$

is satisfied for all $w_{T+1}^{\mathcal{M}}$ if γ >0. Similar to H. Zhang et al. (2020), Rapach et al. (2010), and Campbell and Thompson (2008) I set $\gamma = 3$ and thus the weights in equation 41 provide the maximum expected utility. Following H. Zhang et al. (2020), Rapach et al. (2010), and Campbell and Thompson (2008) I restrict $0 \leq w_{T+1}^{\mathrm{M}} \leq 1.5$. A share $1 - w_{T+1}^{\mathrm{M}}$ is allocated to risk-free bills in period $T + 1$. The realized return of the portfolio in period $T + 1$ is given by Equation 36 after plugging in the weight w_{T+1}^{M}, and the returns r_{T+1}, and r_{T+1}^{f}.

I compare the performance of the portfolio using model $\mathcal{M}$ with a portfolio that estimates r_{T+1} with the historical average, our benchmark model. To do so, I first compute the certainty equivalent return (CER) of each portfolio. This is the value of returns such that the investor is indifferent between acquiring this return *for sure*, and acquiring the risky returns of the portfolio. Assuming the investor compares investing opportunities by comparing their expected utility, we get that the CER of the portfolio using model $\mathcal{M}$ equals its expected utility, where the expectations can now be computed ex-post. This gives

$$\mathrm{CER}_{\mathrm{M}} = \hat{u}_{\mathrm{M}} - \frac{\gamma}{2}\hat{\sigma}_{\mathrm{M}}^2, \tag{43}$$

where $\hat{u}_{\mathrm{M}}$ and $\hat{\sigma}_{\mathrm{M}}^2$ are the sample mean and variance of portfolio returns, respectively. Comparing the CER of the porfolio using model $\mathcal{M}$ with the benchmark (bmk) is done by computing the utility difference in annualized percentage return, given by

$$\Delta(\mathrm{ann}\%) = 1200 \times (\mathrm{CER}_{\mathrm{M}} - \mathrm{CER}_{\mathrm{bmk}}), \tag{44}$$

where returns are assumed to have a monthly frequency. This value can be interpreted as the fee the investor would be willing to pay to get the information of model $\mathcal{M}$ to form his portfolio, rather than using the benchmark.

C Weighted Ridge and RANSAC

This appendix presents results on weighted ridge and ridge with RANSAC for different parameter values. Table 1 shows the out-of-sample performance of a weighted ridge regression for different values of θ, the monthly discount factor. The results indicate that the performance is very sensitive to the chosen value of θ; comparing the third to the fourth column, we see that a change of 0.005 in θ induced a change in R^2_{OS} of up to 0.75.

Table 1 also shows the performance of forecasts when the shrinking factor is determined with ridge using RANSAC estimation for different proportions of observations required in each model. The results show there is potential in this method; as some models perform better for specific values of p. The results seem a bit sensitive to the chosen p. Therefore, as argued in the main text, I suggest finding a data-driven way to determine the optimal value of p.

Table C 1. *Shrinking Performance With Weighted Ridge or RANSAC*

		Weighted Ridge					Ridge+RANSAC		
Model	δ=0.5	θ=0.98	θ=0.99	θ=0.995	θ=0.999	θ=1	p=0.7	p=0.8	p=0.9
Pool-AVG	0.11	-1.29	-0.36	-0.27	-0.43	-0.59	-0.32	0.08	0.11
kitchen sink	1.29	0.10	0.82	0.07	0.73	0.84	1.29	**1.37**	1.10
TM	0.86	-0.21	0.70	0.51	0.51	0.47	**0.96**	0.85	**1.52**
DMSFE	0.85	-0.25	0.55	0.43	0.52	0.50	0.78	**1.20**	**1.31**
IV	0.84	-0.35	0.60	0.46	0.48	0.42	0.57	0.58	**1.32**
BMA	-0.17	-1.53	-0.48	-0.23	-0.19	-0.32	-0.37	-0.81	-0.44
MMA	0.98	-0.47	0.60	0.43	0.43	0.47	-0.76	0.78	0.36
JMA	0.69	-0.32	0.70	0.56	0.44	0.35	-0.55	-0.16	-0.08
WALS	1.56	-1.43	0.93	0.23	0.87	0.99	**1.97**	**1.84**	1.55
LASSO	0.20	-0.25	0.38	0.25	0.03	-0.03	0.12	0.08	**0.35**
Elastic Net	0.19	-0.24	0.44	0.28	0.00	-0.07	**0.34**	0.11	**0.62**
SVR	0.14	-2.48	-1.35	-0.67	-0.30	-0.18	-0.42	-0.57	-0.31
RFR	-0.04	-1.54	-0.61	-0.31	-0.18	-0.22	-0.32	-0.31	-0.17

Note. This table reports out-of-sample R-squared values of forecasts for the equity premium using Average window estimation from Pesaran and Timmermann (2007) (m=10 windows, minimum window size is 240 months) and shrinking. Column header denotes method to select shrinking factor. 1957:12-1979:12 is used as first estimation sample to determine δ and expanded iteratively. The out-of-sample evaluation period is 1980:01-2016:12. θ Denotes the monthly discount factor in weighted Ridge. The simple combination methods trim off a total of 10% of forecasts. DMSFE uses a monthly discount factor θ=0.99. Abbreviations: TM, trimmed mean; DMSFE, discounted mean squared forecast error; IV, inverse variance; BMA, Bayesian model averaging; MMA, Mallows model averaging; JMA, jackknife model averaging; WALS, weighted-average least squares; SVR, support vector regression; RFR, random forest regression.

D Alternative Predictor Variables

As mentioned in the main text, I perform two robustness exercises. The first one is a test for robustness against the predictor variables, the second checks the results in a different period and is shown in the main text. For the first exercise, I forecast the equity premium with a different set of predictor variables. Following H. Zhang et al. (2020), I use the database from FRED-MD which contains monthly observations on 134 indicators of macroeconomic conditions in the US[10]. For a detailed explanation of the data I refer you to McCracken and Ng (2016). The data span from 1959:01 to 2016:12 where 1980:01 to 2016:12 is the out-of-sample evaluation period. Rather than using all 134 variables directly, I follow H. Zhang et al. (2020) who extract seven factors from the data using principal component analysis. The factor values from H. Zhang et al. (2020) are publicly available[11]. I use the models described in section 4.1 with an expanding window, average window (AveW) from Pesaran and Timmermann (2007), and AveW with shrinking towards the historical average.

Table 1 displays the out-of-sample R-squared for forecasts using expanding and average window estimation. The results are similar to those using Welch and Goyal (2008) data. For expanding window, the machine learning models perform relatively well. With average window estimation, the simple combination methods perform best, followed by the machine learning methods, while the advanced combination methods are unable to outperform the historical average. As the daggers in the table show, almost all methods improve with AveW. The combination methods seem to benefit from it most.

When shrinking is applied, all models were able to attain positive out-of-sample R-squared and positive economic value, as shown in Table 2. The daggers in the table show that shrinking improves performance for almost all models. Again, the simple combination methods and WALS perform relatively well, although only IV outperforms the kitchen sink benchmark. The performance of linear machine learning models, RFR, and advanced combination methods is comparable.

The similar patterns observed in these results add to the robustness of the results. There is, however, one notable difference with the main results. When using Welch and Goyal (2008) data, all models that generated significantly positive out-of-sample R-squared had positive R-squared statistics in both recessions and expansions, with the exception of BMA. Using FRED-MD data, this was not the case. Even when estimated with AveW and shrinking, all models got negative R^2_{OS} during expansions. In fact, across all models and estimation techniques, we see that the difference in predictability between expansions and recessions is larger when using the macroeconomic predictors from FRED-MD.

[10] The data is publicly available at https://research.stlouisfed.org/econ/mccracken/sel/.

[11] Available at http://qed.econ.queensu.ca/jae/datasets/zhang002/

Table D 1. *Forecast Evaluation With Expanding and Average Window Using FRED MD Data*

Model	R^2_{os}(%)	*CW*	$R^2_{\text{os, rec}}$(%)	$R^2_{\text{os, exp}}$(%)
Panel A: Expanding window				
Pool-AVG	-0.03	0.44	1.71	-0.67
kitchen sink	-3.15	0.94	-0.24	-4.23
TM	-0.19	1.08	2.57	-1.21
DMSFE	-0.22	1.11	2.88	-1.35
IV	-0.33	1.11	2.46	-1.36
BMA	-0.57	1.47	4.09	-2.29
MMA	-0.88	1.29	3.40	-2.48
JMA	-0.75	1.20	3.72	-2.40
WALS	-1.65	0.77	1.09	-2.66
LASSO	0.29	1.71**	4.39	-1.22
Elastic Net	0.36	1.77**	4.55	-1.19
SVR	-0.44	-0.25	0.09	-0.64
RFR	**0.80**	1.85	7.78	-1.76
Panel B: Average window				
Pool-AVG	0.00†	0.50	1.96	-0.72
kitchen sink	-1.74†	1.46	5.39	-4.36
TM	0.49†	1.40*	5.14	-1.21
DMSFE	**0.74**†	1.40*	5.39	-1.34
IV	0.45†	1.44*	5.36	-1.36
BMA	-0.02†	1.18	4.55	-1.71
MMA	-0.38†	1.18	5.48	-2.44
JMA	-0.44†	0.97	4.43	-2.23
WALS	-0.31†	1.35	5.48	-2.44
LASSO	0.56†	1.47*	4.07	-0.73
Elastic Net	0.66†	1.54*	4.57	-0.78
SVR	-0.47	0.28	1.86	-1.33
RFR	0.48	1.31*	5.43	-1.34

Note. This table reports out-of-sample R-squared values of forecasts for the equity premium over the period 1980:01-2016:12, using FRED MD data. CW is the MSFE-adjusted statistic from Clark and West (2007). $R^2_{\text{os,exp}}$ and $R^2_{\text{os,rec}}$ denote the out-of-sample R-squared during NBER dated expansions and recessions, respectively. Average window estimation from Pesaran and Timmermann (2007) is used with m=10 windows, minimum window size is two-thirds of the size of first estimation sample. Naive shrinking with δ=0.5 is used. The simple combination methods trim off a total of 10% of forecasts. DMSFE uses a monthly discount factor θ=0.99. *p <0.1, **p <0.05, ***p <0.01. Numbers are rounded to two decimal places. Bold entries indicate that the respective model performs best for the given estimation technique. Shaded cells indicate that the respective model outperforms the Pool-AVG and kitchen sink benchmarks. † Indicates AveW improved performance over expanding window. Abbreviations: TM, trimmed mean; DMSFE, discounted mean squared forecast error; IV, inverse variance; BMA, Bayesian model averaging; MMA, Mallows model averaging; JMA, jackknife model averaging; WALS, weighted-average least squares; SVR, support vector regression; RFR, random forest regression.

Table D 2. *Forecast Evaluation With Average Window and Shrinking Using FRED MD Data*

Model	R^2_{os}(%)	*CW*	$R^2_{os,rec}$(%)	$R^2_{os,exp}$(%)	Δ(ann%)	SR
Pool-AVG	0.09†	0.50	1.12	-0.30	0.31	0.15
kitchen sink	0.75†	1.46*	5.31	-0.93	0.61	0.15
TM	0.73†	1.40*	3.36	-0.24	0.67	0.15
DMSFE	0.74	1.40*	3.56	-0.30	0.91	0.16
IV	**0.79**†	1.45*	3.62	-0.25	0.82	0.16
BMA	0.58†	1.18	3.35	-0.44	1.09	0.16
MMA	0.55†	1.18	3.86	-0.67	0.78	0.16
JMA	0.37†	0.97	3.20	-0.67	0.57	0.15
WALS	0.74†	1.35*	4.22	-0.54	0.63	0.15
LASSO	0.63†	1.47*	2.69	-0.12	1.11	0.16
Elastic Net	0.70†	1.54*	2.95	-0.14	1.17	0.16
SVR	-0.06†	0.28	1.02	-0.46	0.04	0.14
RFR	0.49†	1.31*	3.10	-0.48	0.63	0.16

Note. This table reports out-of-sample R-squared values of forecasts for the equity premium over the period 1980:01-2016:12, using FRED MD data. CW is the MSFE-adjusted statistic from Clark and West (2007). $R^2_{\text{os,exp}}$ and $R^2_{\text{os,rec}}$ denote the out-of-sample R-squared during NBER dated expansions and recessions, respectively. Δ(ann%) is the annualized increase in utility for a mean-variance investor with relative-risk parameter $\gamma = 3$. SR denotes the Sharpe ratio. Average window estimation from Pesaran and Timmermann (2007) is used with $m = 10$ windows, minimum window size is two-thirds of the size of first estimation sample. The simple combination methods trim off a total of 10% of forecasts. DMSFE uses a monthly discount factor θ=0.99. *p <0.1, **p <0.05, ***p <0.01. Numbers are rounded to two decimal places. Bold entries indicate that the respective model performs best for the given estimation technique. Shaded cells indicate that the respective model outperforms the Pool-AVG and kitchen sink benchmarks. † Indicates shrinking improved performance over AveW. Abbreviations: TM, trimmed mean; DMSFE, discounted mean squared forecast error; IV, inverse variance; BMA, Bayesian model averaging; MMA, Mallows model averaging; JMA, jackknife model averaging; WALS, weighted-average least squares; SVR, support vector regression; RFR, random forest regression.

E Python and Matlab Code

In this section, I provide a brief description of the code used to generate my results. Since I used many files to structure the code, I describe them in the table below.

Table E 1. *Description Python and MATLAB Files*

File	Explanation
DataPrep.py	Loads Welch and Goyal (2008) data and creates explanatory variables according to Elliot et al. (2013).
DataPrepFREDMD.py	Loads and prepares the seven factors from FRED MD database
DataPrepMacro.py	Loads and prepares the seven macroeconomic conditions. Also makes and evaluates the forecasts, described in section D.
AVG_Pool.py	Makes forecasts with Pool AVG benchmark method from Rapach et al. (2010).
KitchenSink.py	Makes forecasts with the benchmark kitchen sink method.
SimpleCombination.py	Implements the class forecastCombi which can be supplied data. Class can be called to make forecasts for BMA, IV, TM, and DMSFE. Implements two functions exp_forecast and AveW_forecast which create and object of forecastCombi class and forecast one value with expanding and average window, respectively.
TrimmedMean.py	Makes forecasts for Trimmed Mean method from Stock and Watson (2004). Imports functions exp_forecast and AveW_forecast from SimpleCombination.py
InverseVariance.py	Makes forecasts for Inverse Variance method from Bates and Granger (1969). Imports functions exp_forecast and AveW_forecast from SimpleCombination.py. Also contains code used to evaluate all forecasts over specific periods of extreme movement
DMSFE.py	Makes forecasts for Discounted Mean Squared Forecast Error method from Stock and Watson (2004). Imports functions exp_forecast and AveW_forecast from SimpleCombination.py
BMA.py	Makes forecasts for Bayesian Model Averaging method. Although not a simple combination method it imports functions exp_forecast and AveW_forecast from SimpleCombination.py because programming-wise it fit in the same framework. Implementation of BMA in SimpleCombination.py uses some code from Basener (2020)
WALS_via_matlab.py	Imports WALS forecasts made in MATLAB and evaluates them.
JMA_via_matlab.py	Imports JMA forecasts made in MATLAB and evaluates them.
MMA_via_matlab.py	Imports MMA forecasts made in MATLAB and evaluates them.

Table E 1. *Description Python and MATLAB Files*

File	Explanation
Lasso.py	Makes and evaluates forecasts for LASSO method
Elastic_net.py	Makes and evaluates forecasts for elastic net method
RFR.py	Makes and evaluates forecasts for random forest regression
SVR.py	Makes and evaluates forecasts for support vector regression
Shrinking.py	Shrinks forecasts using the different methods described in Secion 4.3 and applied RANSAC
explanation_results.py	Used to create all figures in this thesis.
Evaluation.py	Implements the evaluation metrics used in this thesis: out-of-sample R-squared, MSFE, increased utility, Sharpe ratio, CW, MHLN, etc. Also implements a function to print all metrics
MHLN.py	Imports all forecasts and computes all p-values for MHLN test.
gma.m	Implements gma function which allows you to do JMA and MMA. Implemented by Hansen and Racine (2012).
JMA_forecasts.m	Creates JMA forecasts. Uses gma function from gma.m
MMA_forecasts.m	Creates MMA forecasts. Uses gma function from gma.m
WALS_forecasts.m	Creates WALS forecasts. Uses wals function from wals.m
wals.m	Implements wals function, can do WALS estimation with. Implemented by De Luca and Magnus (2011), Kumar and Magnus (2013), Magnus and De Luca (2016), and Magnus et al. (2010). Uses the distribution values tabulated by Table_postmoments.m
Table_postmoments.m	Creates two text documents with distributional values used by WALS. Uses the function postmoments from postmoments.m to get each value in the table. Code published by De Luca and Magnus (2011), Kumar and Magnus (2013), Magnus and De Luca (2016), and Magnus et al. (2010).
postmoments.m	Implements a function postmoments that generates distributional values. Code published by De Luca and Magnus (2011), Kumar and Magnus (2013), Magnus and De Luca (2016), and Magnus et al. (2010).