ESSAYS IN MACROECONOMIC FORECASTING AND MODEL EVALUATION

by

SUNGKYUNG LEE

A Dissertation Submitted to the

School of Graduate Studies

Rutgers, The State University of New Jersey

In partial fulfillment of the requirements

For the degree of

Doctor of Philosophy

Graduate Program in Economics

Written under the direction of

Norman R. Swanson

And approved by

_____

_____

_____

_____

New Brunswick, New Jersey

October, 2018

ABSTRACT OF THE DISSERTATION

ESSAYS IN MACROECONOMIC FORECASTING AND MODEL EVALUATION

By SUNGKYUNG LEE

Dissertation Director:
Professor Norman R. Swanson

This dissertation studies forecasting model specification, estimation, prediction, and evaluation in big data environments. In an effort to contribute to the discussions of macroeconomic forecasting, I examine the studies of forecasting model specification and forecast accuracy testings and introduce new methodologies in empirical frameworks. The whole set-up of forecasting model specification and forecasting evaluation framework is a continuum of decisions, which can lead to different forecasting results. In closely-connected two papers, I attempt to empirically evaluate the implications of using different methodologies throughout all stages of macro forecasting and provide insightful conclusions for future researches in the literature.

Chapter 2 revisits the question of predictive accuracy testing and model selection, and asks the question: does the loss function really matter, and if so, what can be gained when utilizing loss function-free model comparison and selection tests? So far in forecasting literature, forecasting results have been compared based on moment-based approaches which mostly concern about only first and second moment of forecasting errors and require to choose a loss function to begin with, which is an additional decisional problem. In Chapter 2, I compare forecasting results based on a distributional comparison approach suggested by Jin et al. (2016), which is technically based on the stochastic dom-

inance principles and loss-function robust. A series of empirical experiments are carried out using macroeconomic time series data modeled using big data methods, including a large number of dimension reduction, shrinkage, and machine learning methods. Analysis and ranking of these methods is found to depend crucially on whether loss function dependent evaluation of their accuracy is carried out, or not.

Chapter 3 builds on my first chapter by focusing on the usefulness of so-called "supervised" approaches to forecast model selection in big-data environments. When constructing forecasting models using latent factor variables that are designed to condense information from large datasets into a small set of useful explanatory variables, standard approaches involve extracting information relevant to the entire dataset, and not targeted to a particular variable being forecasted. Supervised approaches to model specification do not do this, but instead penalize model specifications according to metrics designed to focus on the particular target variable(s) of interest. In order to evaluate the efficacy of supervised approaches, I carry out Monte Carlo simulations and empirical exercises and empirical results suggest that supervised approaches that are geared for the purpose of forecasting do serve its own purpose.

Acknowledgements

I am greatly indebted to my advisor, professor Norman R. Swanson, for his guidance throughout the journey of my doctoral studies. Professor Swanson broadened my intellectual horizons and perspectives, guiding me to study in depth about time series analysis and macroeconomic forecasting.

I also sincerely appreciate professor John Landon-Lane and professor Yuan Liao. Professor John Landon-Lane introduced me Bayesian econometrics and gave me sincere advices whenever I needed. His knowledge on macroeconomics and macroeconometrics is quite insightful and opened to me alternative ways of thinking towards problems. Professor Yuan Liao's advice always has been sharp and gave me good directions based on his deep knowledge in theoretical statistics.

I would like to thank Ling Li, Mingmian Cheng, Yuliyan Miktov, Yixiao Jiang, Geoffrey Clarke, and Weiqi Xiong for their heartfelt helps over the last 6 years and I will cherish the time and friendship that we shared. I wish them all the luck and success in their careers and personal happiness. I owe special thanks to professor Hyun-Hak Kim. He has been always willing to provide generous help on my research and genuine advices on important decisions.

Also, I would like to express my gratitude to Linda Zullinger, Janet Budge, Paula Seltzer, Debra Holman and Donna Ghilino. They have been always there whenever I am in need of help and treated my problems as if those were their own. I wouldn't be able to finish my doctoral study successfully without their helps.

Lastly, I would like to say special thanks to my parents, who always have been my emotional prop even though we have been physically away for a long time. I am immensely indebted to their support, love, and prayer.

Dedication

To my parents,

Table of contents

List of tables

List of illustrations

Chapter 1
Introduction

My doctoral study has been focused on both theoretical and empirical studies of macroeconomic forecasting and my dissertations attempt to evaluate state-of-the-art forecasting methodologies empirically and present further discussion questions and relevant policy implications. The two closely-related papers cover all stages of macroeconomic forecasting, from model specification, estimation, and prediction, to evaluation and derive the empirical implications of different choices made in each stage.

Specifically, the first chapter examines the topic of forecasting accuracy testings. Forecast accuracy has long been measured by using loss function-specific measures of predictive accuracy. However, it is well known that ordinal rankings associated with such measures are affected under generic model misspecification. Given that virtually all models used in forecasting are approximations, and hence misspecified to some degree, it is of interest to consider using loss function robust measures when comparing alternative models. Thus, instead of carrying out model selection solely based on moment-based criteria and tests, such as those based on mean square forecast error (MSFE) and mean absolute forecast error (MAFE) loss (e.g., Diebold and Mariano (2002) and White (2000)), one might instead utilize robust stochastic dominance based model selection, such as the predictive accuracy tests due to Jin, Corradi and Swanson (Jin et al. (2016)). In my first chapter, the above tests are examined in a series of empirical experiments carried out on macroeconomic time series filtered using big data methods including dimension reduction, shrinkage and machine learning (e.g., ridge regression, the elastic net, bagging, boosting, factor augmented autoregression, and least angle regression), and diffusion index (factor) analysis (e.g. principal components).

Analysis and ranking of the models and methods analyzed in my experiments is found to depend crucially on whether loss function dependent pre-

dictive evaluation is carried out. Namely, model selection depends on whether the "best model" is chosen using distributional comparison based on Jin et al. (2016) tests or moment based comparison based on Diebold and Mariano (2002) or White (2000)'s tests. This is not surprising, given the above discussion, but it is noteworthy, as the implications for forecast model building are immediate. In addition, it is found that loss function specific test results are sensitive to the specification of loss function, and in particular to whether MSFE or MAFE loss is used. Again, while this is not surprising, it is a key element to understanding the importance of whether or not to utilize loss function robust predictive accuracy tests in forecasting contexts. With regard to the actual prediction models and methods utilized in my experiments, for a given variable, it is not possible to claim that a specific forecasting model "uniformly" dominates, across different forecast horizons and specification methods. However, it is noteworthy that hybrid models that combine shrinkage estimation with diffusion indexes perform better than benchmark models including linear models, purely factor-based models, and model averaging. Thus, there appears to be useful information in "big data", and the manner in which this information is extracted matters. In conclusion, I find that loss function-specific and loss function-robust tests are complementary, and that much can be learned by utilizing both types of tests for empirical forecast model selection.

My third chapter builds on my second chapter by focusing on the usefulness of so-called "supervised" approaches to forecast model selection in big-data environments. When constructing forecasting models using latent factor variables that are designed to condense information from large datasets into a small set of useful explanatory variables, standard approaches involve extracting information relevant to the entire dataset, and not targeted to a particular variable being forecasted. Supervised approaches to model specification do not do this, but instead penalize model specifications according to metrics de-

signed to focus on the particular target variable(s) of interest. For example, regardless of which variable is being forecasted (targeted), standard principal component methods involve constructing diffusion indexes based on eigenvectors corresponding to the r largest eigenvalues of the correlation matrix of the dataset. However, this practice is questionable, in the sense that it is not clear whether such latent factors (i.e., diffusion indexes) are effective in terms of predicting a specific variable. Supervised methods tackle this issue by "training" involving simple approaches such as taking into account the correlations between big data and the variable being predicted.

In order to analyze the efficacy of such approaches, I carry out a Monte-Carlo simulation study in which I simulate different factor structures associated with a variety of data generating processes (with and without heterogeneity, for example), including cases where: (i) there are very few latent factors; (ii) there are many (up to 50) underlying factors; (iii) there are very few factors, and only 1 or 2 are relevant for forecasting the target variable; and (iv) no factors are relevant for forecasting. As might be expected, supervised methods are least useful under (i) and (ii); and are most useful under case (iii) and (iv). Empirical exercise results that horse-race compare 16 forecasting models also indicate that supervised forecasting schemes outperform other un-supervised forecasting models, proving the efficacy of supervision. Especially, the predictive gains when using PLS (partial least squares) are quite substantial and I observe that CFPC (combining forecast principal component) outperforms the simple average model most of the time. Therefore, I conclude that supervised approaches that are geared for the purpose of forecasting do serve its own purpose.

Chapter 2
Stochastic Dominance Based Evaluation of Factor and Shrinkage Based Prediction Models

1 Introduction

In the study of economic time series, forecast accuracy has long been measured by using loss function specific measures of predictive accuracy, such as the mean square forecast error (MSFE). However, it is well known that ordinal rankings associated with such measures are affected by generic model misspecification. Given that virtually all models used in forecasting are approximations, and hence misspecified to some degree, it is of interest to consider the use of loss function robust measures when comparing alternative models. The notion of a loss function is also closely related to the decision problem associated with how one chooses to weight and summarize forecast errors. [1] Thus, another way of viewing the choice between using loss function dependent evaluations and loss function free forecast evaluation involves assessing whether the relevant decision problem involves directly addressing loss function dependence, or eschewing the use of loss functions altogether. In this paper, I address this issue in a series of empirical forecasting experiments. In these experiments, a large number of state of the art dimension reduction, shrinkage, and machine learning methods are used to forecast various macroeconomic series, and both loss function specific and loss function free methods are used to "select" among the different models and methods.

[1] Machina and Granger (2005) and Granger and Machina (2006) proved that a loss function has a close, but not unique, relationship with an objection function and choice variable and Tideman and Timmermann (2008) stated that "a decision theory provides a framework for both the construction and evaluation of forecasts."Detailed discussion on the linkage between the two topics, decision theory and economic forecasting, will not be covered here.

When assessing forecast accuracy, the so-called Diebold-Mariano (DM) tests due to Diebold and Mariano (2002) comes most frequently to mind, although a whole host of alternative moment-based forecast evaluation tests has also been developed. Suppose that $e_{1,t}$ is a sequence of forecast errors from a given benchmark model, and $e_{k,t}$ are errors from an alternative model. Let g refers to a chosen loss function. Under the null hypothesis of equal forecast accuracy, the DM tests involve testing the following hypothesis: $H_0 : E(d_{1k,t}) = 0$, where $d_{1k,t} = g(e_{1,t}) - g(e_{k,t})$ for $k = 2, \ldots, l$. White (2000) develops a related "data-snooping" test of the null hypothesis: $H_0 : max_{k=2,\ldots,l} E(g(e_{1,t}) - g(e_{k,t})) \leq 0$, where l denotes a fixed number of alternative models against which the benchmark model is being compared. [2]

However, as discussed above, moment-based tests are potentially quite restrictive, and can be non-informative. For this reason, Jin et al. (2016) introduce loss function free forecast superiority tests that are based on the application of first- and second-order stochastic dominance principles, as discussed in Linton et al. (2005). In particular, Jin et al. (2016) introduce two concepts: general loss (GL) forecast superiority and convex loss (CL) forecast superiority. Simply put, a forecast error sequence GL outperforms other sequences if an economic agent with a GL loss function prefers the former to the latter. Similarly, a forecast error sequence CL outperforms other sequences if an economic agent with a CL loss function prefers the former to the latter. These authors have established links between tests for GL (CL) forecast superiority and tests for first-order (second-order) stochastic dominance. This in turn allows for the introduction of forecast evaluation procedures that are based on an out-of-sample generalizations of the tests introduced by Linton et al. (2005). As the tests are based on the empirical distributions of raw forecast errors, they are robust not only to the choice of the loss function, but also to the possible presence of outliers.

[2] Corradi and Swanson (2006) extend the pointwise forecasting accuracy approaches implicit in these two tests by considering evaluation of forecast intervals and densities. For detailed discussion of recent developments in the forecasting evaluation methods, refer to Corradi and Swanson (2013).

The latter feature is shared by Corradi and Swanson (2013).

The main goal of this paper is to assess the trade-offs associated with using the different loss function specific (DM and data snooping) and loss function free (GL and CL forecast superiority) predictive accuracy tests discussed above. This is done by carrying out an extensive set of empirical experiments using the factor augmented forecasting models discussed in Stock and Watson (2002, 2012) and Bai and Liao (2016) as a starting point. In our models, unobserved factors (diffusion indexes) estimated via implementation of principal component analysis, are used to forecast a scalar target variable, Y_{t+h}. However, rather than estimating the number of factors to use in prediction models via a standard testing (e.g., Bai and Ng (2002)), and in order to add functional flexibility, I implement prediction models in which factors are selected using a variety of shrinkage methods. In this sense, I add to the recent work of Bai and Ng (2002, 2008), Stock and Watson (2002), and Kim and Swanson (2014), who survey several methods for shrinkage in the context of factor augmented autoregression models.

The shrinkage methods considered in this paper follow those examined in Kim and Swanson (2014), and include bagging, boosting, Bayesian model averaging, simple model averaging, ridge regression, least angle regression, elastic net (EN) and the non-negative garotte. In addition, I evaluate various linear models and pure shrinkage models that do not incorporate unobserved factor estimates. The variables that I predict in empirical experiments include a variety of macroeconomic variables that are useful for evaluating the state of the economy. Specifically, forecasts are constructed for 11 series, including: the unemployment rate, personal income less transfer payments, the 10-year Treasury-bond yield, the consumer price index, the producer price index, non-farm payroll employment, housing starts, industrial production, M2, the S&P 500 index, and gross domestic product.

Analysis and ranking of the models and methods analyzed in our experi-

ments is found to depend crucially on whether loss function dependent evaluation of their accuracy is carried out, or not. Namely, there tend to be differences between the best models chosen using distributional comparison based on JCS tests, and the best models selected by using reality check (or DM) tests. This is not surprising, given our above discussion, but is noteworthy, as the implications for model building are immediate. In addition, it is found that loss function specific test results are sensitive to the specification of loss function, and in particular to whether MSFE or MAFE loss is used. Again, this is not surprising, but is a key element to understanding the importance of using loss function robust tests. With regard to the actual prediction models and methods utilized in our experiments, it is not possible to claim that a specific forecasting model "uniformly" dominates, across different forecast horizons and specification methods, for a given variable. However, it is noteworthy that hybrid models that combine shrinkage estimation with diffusion indexes perform better than benchmark models including linear models, purely factor-based models, and model averaging. Thus, there appears to be useful information in "big data", and the manner in which this information is extracted matters.

Broadly speaking, the best shrinkage and machine learning methods in our experiments such as non-negative garotte, lasso, ridge, bagging and boosting are most useful when coupled with diffusion indexes, for most of target forecasting variables. Also, when carrying out loss function free model selection, the number of "wins" for hybrid models increases appreciably, relative to when loss function dependent model selection is carried out. For example, hybrid models and methods "win" around 50 % of the time when selection is based on the JCS test, and only around 1/3 of the time based on the application of DM tests. More specifically, based on model selection using first order stochastic dominance comparisons, hybrid models and methods "win" for 11 variables, at 4 different forecast horizons most of the time; and fail to "win"

only in cases where it is hard to rank models because empirical distributions of forecast errors almost coincide. In conclusion, I find that loss function specific and loss function robust tests are complementary, and much can be learned by utilizing both types of tests for empirical forecast model selection.

This rest of the paper is organized as follows. In Section 2, GL and CL forecast superiority tests are discussed. Section 3 briefly surveys diffusion index construction in our setup. Section 4 outlines the robust dimension reduction, shrinkage and machine learning methods used in the construction of our forecasting models. In Section 5, the results of our empirical experiments are reported. Section 6 concludes.

2 Forecast Superiority Testing

Linton et al. (2005) proposed a consistent test that compares the empirical distributions of random variables for making a decision, rather than confining the discussion into certain moments of random variables under general time series settings. Jin et al. (2016) generalize Linton et al. (2005)'s stochastic dominance-based testing into an out-of-sample forecasting framework.

In details, Jin et al. (2016) propose a proposition that maps stochastic dominance principles into optimization problems, which is an application of Klecan et al. (1991) into an out-of-sample forecasting framework. (Propositions 2.2 & 2.3 under Assumption A.0); Klecan et al. (1991) establish the equivalence between first-order stochastic maximality and a utility maximization problem in which the set of utility functions considered are continuous and increasing functions. [3] In an analogous way, the authors also connect second-order stochastic maximality with utility maximization, but under a smaller

[3] Klecan et al. (1991) define stochastic dominance as when one of random variables in a set $\chi = \{X_1, X_2, ..., X_k\}$, stochastically dominates all the other random variables in the set, while stochastic maximality refers to a situation in which some pair of random variables in the set have that kind of relationship.

subset of utility functions, continuous, increasing, and strictly concave ones. Along the same line, Jin et al. (2016) apply this discussion to an out-of-sample forecasting context, by linking the first order (FO) stochastic dominance with General-Loss (GL) forecast superiority test, also second order (SO) stochastic dominance principles with Convex-Loss (CL) forecast superiority test. While Klecan et al. (1991) deals with a situation in which the problem is to maximize expected utilities, since Jin et al. (2016)'s problem is to find the best forecasting model that minimizes expected forecasting errors the most, the underlying function spaces are different: the first-order stochastic dominance principle is equivalent to an expected loss-minimization problem under weakly increasing functions (general loss (GL or L_G) function) and second-order stochastic dominance pertains to a subset of GL functions that are weakly increasing and also convex (convex loss (CL or L_C) function).

An essential mathematical fact that holds in stochastic dominance analysis is that lower order stochastic dominance implies higher order stochastic dominance, while the converse is not necessarily true: based on the equivalence proposition mentioned above, GL outperformance implies CL outperformance, but not necessarily the other way around. That's because the $(s+1)^{th}$ order stochastic dominance compares the areas under the curve of $D^S(y) = \int_0^y D^{(S-1)}(z)dz = \int_0^y F^{(S-1)}(z)dz$ where $D^1(y) = F(y)$ and $D^{(S+1)}(y) = \int_0^y D^S(z)dz$. Hence, even when ordinal rankings are unclear at the s^{th} order because the distributions of forecast errors almost coincide or cross over each other, one can compare the areas under the curve of $D^S(y)$, hoping that the magnitudes of the areas can order random variables more clearly in a $(s+1)^{th}$ order stochastic dominance sense. In particular, if the distribution comparison is conducted in a multiple comparison context, there is expected to be a low possibility of finding a clear stochastic dominance relationship—the case in which the whole CDF of forecast error from a model k clearly lies over other CDFs over the joint support. The problem becomes more serious if the performances of compared

models are quite close to each other and, in that case, it is more likely that many crossing points over the CDFs of forecasting errors are observed.

2.1 Loss function-free predictive accuracy test

Jin et al. (2016) called their study as "robust forecast comparison", in the sense that their test is robust to the choice of loss functions. JCS test is based on the comparison of $\bar{F}(e_{k,t})$, empirical distributions of raw forecast errors, rather than $\bar{F}(L(e_{k,t}))$, so they are robust not only to the choice of the loss function, but also to the possible presence of outliers. Consequently, forecast evaluation results are not be subject to a certain choice of a loss function even under the presumption that all forecasting models are mere approximations of the true underlying DGP. However, as mentioned above, this distributional comparison, by which raw forecast errors themselves are taken as random variables not involving the choice of a loss function, can be translated into a forecast superiority problem that holds under generic loss functions.

As forecasting deals with time series, Jin et al. (2016)'s framework is set to work with a dependent data set, but the possibly dependent data has to be stationary. Here, random variables to be compared are raw forecasting errors, $e_{k,t+h}$, which are defined as, for $k = 1, ..., l$ forecasting model and h forecasting horizon,

$$\begin{aligned} e_{k,t+h} &= Y_{t+h} - g_k(Z_{k,t+h}, \beta_{k0}) \\ &= Y_{t+h} - \tilde{g}_k(Z_{t+h}, \beta_0), \end{aligned} \tag{1}$$

and will be estimated as

$$\widehat{e}_{k,t+h} = \widehat{Y}_{t+h} - \widehat{g}_k(Z_{k,t+h}, \widehat{\beta}_{k,t}) \tag{2}$$

where $\Theta = \prod_{k=1}^{l} \Theta_k$, $g_k : R^{P_k} \times \Theta_k \to R$, $\tilde{g}_k : R^{\bar{P}} \times \Theta \to R$, and Z_t is a $\bar{P} \times 1$ random vector.

The true underlying distribution of forecasting errors evaluated at the point, x, is defined as

$$F_k(x, \beta_k) = P(e_{k,t+h}(\beta_k) \leq x) \tag{3}$$

and its empirical sampling distribution as

$$\bar{F}_{k,n}(x, \widehat{\beta}_{k,R:T}) = n^{-1} \sum_{t=R}^{T} 1(\widehat{e}_{k,t+h}(\widehat{\beta}_{k,t}) \leq x) \tag{4}$$

where $\underline{\widehat{\beta}}_{k,R:T} = (\widehat{\beta}'_{k,R}, ..., \widehat{\beta}'_{k,T})'$ and $n = T - R + 1$. So in this paper, contrary to their earlier version of this paper, Corradi and Swanson (2013), Jin et al. (2016) takes into account parameter estimation errors whose impact will not vanish even in the limit. For simplicity, $F_k(x, \beta_k)$ will be denoted as $F_k(x)$ from now on.

Now, define $G_k(x)$, a functional of the differential between $CDFs$ and $C_k(x)$, of the differential between integrated $CDF(SDF)s$,

$$G_k(x) = (F_k(x) - F_1(x))sgn(x), \tag{5}$$

$$C_k(x) = \int_{-\infty}^{x} (F_1(s) - F_k(s))ds1(x < 0) + \int_{x}^{\infty} (F_k(s) - F_1(s))ds1(x \geq 0) \tag{6}$$

where $sgn(x)$ function will have a value equal to 1 if $x \geq 0$ and -1 otherwise and TG^+ and TC^+ as,

$$TG^+ = \max_{k=2,...,l} \sup_{x \in \chi^+} G_k(x), \quad TG^- = \max_{k=2,...,l} \sup_{x \in \chi^-} G_k(x) \tag{7}$$

$$TC^+ = \max_{k=2,...,l} \sup_{x \in \chi^+} C_k(x), \quad TC^- = \max_{k=2,...,l} \sup_{x \in \chi^-} C_k(x). \tag{8}$$

where $\chi^+ = \chi \cap R^+$, $\chi^- = \chi \cap R^-$ and $R^+ \equiv \{x \in R, x \geq 0\}$, $R^- = R \backslash R^+$. Then, the corresponding hypothesis of testing GL/CL outperformance will be specified as,

$$H_0^{TG} : \ TG^+ \leq 0 \ \cap \ TG^- \leq 0 \ \ vs. \ \ H_A^{TG} : \ TG^+ > 0 \ \cup \ TG^- > 0 \tag{9}$$

$$H_0^{TC} : \ TC^+ \leq 0 \ \cap \ TC^- \leq 0 \ \ vs. \ \ H_A^{TC} : \ TC^+ > 0 \ \cup \ TC^- > 0. \tag{10}$$

If one cannot reject the null, at a prespecified significance level, it means that e_1, a sequence of forecast errors from the benchmark model, say 1, GL/CL outperforms, e_k, sequence(s) of forecast errors from alternatives where $k = 2, ..., l$, and e_1 first-order/second-order stochastically dominates e_k: if that's the case, one can discard all other alternative forecasting methods, $k = 2, \ldots, l$. On the contrary, rejecting the null implies e_1 cannot GL/CL outperform e_k: i.e. e_1 is first-order/second-order stochastically dominated. [4]

The sample counterparts of (7) and (8), expressed as empirical distributions, are the JCS test statistics,

$$TG_n^+ = \max_{k=2,\ldots,l} \sup_{x \in \chi^+} \sqrt{n} G_{k,n}(x), \ \ TG_n^- = \max_{k=2,\ldots,l} \sup_{x \in \chi^-} \sqrt{n} G_{k,n}(x) \tag{11}$$

$$TC_n^+ = \max_{k=2,\ldots,l} \sup_{x \in \chi^+} \sqrt{n} C_{k,n}(x), \ \ TC_n^- = \max_{k=2,\ldots,l} \sup_{x \in \chi^-} \sqrt{n} C_{k,n}(x) \tag{12}$$

where

$$G_{k,n}(x) = (\bar{F}_{k,n}(x, \hat{\underline{\beta}}_{k,R:T}) - \bar{F}_{1,n}(x, \hat{\underline{\beta}}_{1,R:T}))sgn(x) \tag{13}$$

[4] In chapter 3 and 4, Jin et al. (2016) establish the asymptotic null distribution of their test statistics and its asymptotic power properties. And in chapter 6, they study the finite sample properties, size and power, of the tests, by conducting Monte Carlo simulations under a variety of DGP scenarios.

$$C_{k,n}(x) = \int_{-\infty}^{x} (\bar{F}_{1,n}(s, \hat{\underline{\beta}}_{1,R:T}) - \bar{F}_{k,n}(s, \hat{\underline{\beta}}_{k,R:T}))ds1(x < 0) + \int_{x}^{\infty} (\bar{F}_{k,n}(s, \hat{\underline{\beta}}_{k,R:T}) - \bar{F}_{1,n}(s, \hat{\underline{\beta}}_{1,R:T})ds1(x \geq 0). \tag{14}$$

Since the null is composed of composite hypotheses and underlying random variables and data set have dependent structure, the asymptotic null distributions of the test statistics cannot be established in a conventional way. Here, following Jin et al. (2016), I use stationary bootstrap (Politis and Romano (1994)) to derive the sampling distribution of test statistics and construct critical values. Since this sampling method requires the underlying DGP of data set to be stationary, I transform all variables as specified in Table 3.A.1 to induce stationarity.

Specifically, the bootstrap statistics will be constructed as,

$$TG_n^{*+} = \max_{k=2,\ldots,l} \sup_{x \in \chi^+} \sqrt{n}(G_{k,n}^*(x) - G_{k,n}(x)) \tag{15}$$

where

$$G_{k,n}^*(x) = (\bar{F}_{k,n}(x, \hat{\underline{\beta}}_{k,\theta(R):\theta(T)}) - \bar{F}_{1,n}(x, \hat{\underline{\beta}}_{1,\theta(R):\theta(T)}))sgn(x) \tag{16}$$

and

$$\bar{F}_{k,n}(x, \hat{\beta}_{k,\theta(R):\theta(T)}) = n^{-1} \sum_{t=R}^{T} 1(e_{k,\theta(t)+\tau}(\beta_{k,\theta(t)}) \leq x). \tag{17}$$

TG_n^{*-} can be defined in an analogous way for the case where $x \in \chi^-$. Then, one can compute bootstrap p-values as, $p_{B,n,S_n}^{G+} = \frac{1}{B}\sum_{s=1}^{B} 1(TG_n^{*+} \geq TG_n^{+})$, $p_{B,n,S_n}^{G-} = \frac{1}{B}\sum_{s=1}^{B} 1(TG_n^{*-} \geq TG_n^{-})$, $p_{B,n,S_n}^{C+} = \frac{1}{B}\sum_{s=1}^{B} 1(TC_n^{*+} \geq TC_n^{+})$, and $p_{B,n,S_n}^{C-} = \frac{1}{B}\sum_{s=1}^{B} 1(TC_n^{*-} \geq TC_n^{-})$, and make a decision on the hypothesis testing based on these rules,

$$\text{Reject } H_0^{TG} \text{ at level } \alpha \text{ if } \min\{p_{B,n,S_n}^{G+}, p_{B,n,S_n}^{G-}\} \leq \tfrac{\alpha}{2}$$
$$\text{Reject } H_0^{TC} \text{ at level } \alpha \text{ if } \min\{p_{B,n,S_n}^{C+}, p_{B,n,S_n}^{C-}\} \leq \tfrac{\alpha}{2}.$$

Before proceeding the resampling, the mean block length, $W_n = \frac{1}{S_n}$, has to be pre-specified, which works as a smoothing parameter. The choice of it should be based on the dependent structure of time series: the more dependent data is, the longer the mean block length, W_n, should be. In my empirical application, I try a set of different mean block lengths, $W_n = \frac{1}{S_n} = [2, 5, 7, 10]$, and see whether the test results are sensitive to the different values of W_n.

As an extension, Jin et al. (2016) relax the initial stationarity requirement and develop their discussion into heterogeneous environments, which can be seen as a natural extension to deal with more general time series settings (section 5). As for the resampling scheme, since the stationary requirement doesn't hold anymore under heterogeneous settings, Jin et al. (2016) use block bootstrap instead of stationary bootstrapping and find it works well.

3 Factor model structure

A factor model provides a feasible and effective framework to handle with a high-dimensional data set and its empirical performances have been quite competitive against those of other theoretical models. So, this diffusion index approach has been widely used especially when there is no specific prior knowledge on the underlying structure of the state of the economy and enables to incorporate a high-dimensional information set. Depending on whether the covariance matrix of error terms is diagonal or not, a factor model can be classified as strict or approximate factor model. If it has dynamic structures, that is, lagged terms, such as y_{t-h} or $\widehat{f}_{t-h}$, are allowed to enter, then it is called a dynamic factor model, otherwise a static factor model. The basic factor model framework can be represented as follows,

$$x_{it} = f_t \lambda_i' + e_{it} \tag{18}$$

where x_{it} represents observable datum for $t = 1, .., T$ and $i = 1, ..., N$, but both, f_t, a $1 \times r$ common factor, and λ_i, a $1 \times r$ factor loading, are unobservable and have to be estimated.

A conventional way to estimate factors and factor loadings is to employ an ordinary principal component (OPC) estimator. The OPC estimation is based on the optimization problem that minimizes the sum of squared residuals (SSR) as,

$$V(r) = \min_{\wedge, F} (NT)^{-1} \sum_{i=1}^{N} \sum_{t=1}^{T} (X_{it} - \lambda_i' F_t)^2$$

under the normalization constraints, either $\frac{\wedge'\wedge}{N} = I_r$ when $T > N$, or $\frac{F'F}{T} = I_r$ when $T < N$. Under the identification restrictions, the pair of factors and factor loadings is defined as a function of eigenvectors of r largest eigenvalues of XX', which is expected to span the covariation of the data set. Thus, it can be said that the PC estimator is a consistent estimator for the factor "space" subject to the restrictions, not for the underlying true factors themselves. As an estimator of the number of factors, r, I will use one variant of $SIC(BIC)$, following Kim and Swanson (2014). [5]

The underlying assumption of the OPC estimator (OPCE) requires the covariance matrix to be diagonal, which is quite an unrealistic assumption in a time series setting. However, when there is more prior information on the underlying DGP, it is possible to achieve more efficient estimation results exploiting the information at hand, which can also possibly lead to better forecasting performances. Recently, Bai and Liao (2016) develop an efficient estimator using a penalized maximum likelihood (ML) approach that takes cross-sectional and serial correlation into account. Their simulation results with a data set in which the errors terms are both serially and cross-sectionally correlated show

[5] The choice for the estimator of the number of factors is one of the important issues in a factor model specification. In my another paper, I use Bai and Ng (2002)'s IC estimator, Ahn and Horenstein (2013)'s ER/GR estimator, and Onatski (2010)'s estimator to observe how different choices of estimator can affect factor estimation and forecasting performances.

that, when N is small, maximum likelihood estimation (MLE) outperforms other principal component-based methods, but, when N is large, it is difficult to distinctively rank the performances of those different approaches.

Meanwhile, Choi (2012) introduces the generalized principal component estimator (GPCE), $(\hat{F}^G, \hat{\Lambda}^G)$,

$$(\hat{F}^G, \hat{\Lambda}^G) = argmin_{F,\Lambda}(NT)^{-1} \sum_{i=1}^{N} \sum_{t=1}^{T} (X_{it} - \lambda_i^{'} F_t)' \Sigma_{u0}^{-1} (X_{it} - \lambda_i^{'} F_t). \tag{19}$$

Thus, $\hat{F}^G$ and $\hat{\Lambda}^G$ are a function of the eigenvalues of $X\Sigma_{u0}X'$, rather than of XX' like in the usual PC estimation. This method taking the dependence structure of the error matrix into account turns out to be more efficient than the ordinary PC estimator in the same way GLS works.

4 Robust shrinkage estimation

I'll employ robust estimation techniques that can be largely categorized into three main types: learning algorithms (bagging and boosting), penalized regression methods (ridge, least absolute shrinkage selection operator (lasso), least angle regression (LAR), EN (EN), non-negative garotte (NNG)), and Bayesian model averaging (BMA) with two different priors.

Alternatively, these approaches also can be classified as hard-thresholding and soft thresholding approaches (Bai and Ng (2008)). The first one, hard thresholding, orders regressors based on the marginal predictability power of each single regressor, regardless of what other predictors are included in the regression: by checking whether the t-statistic of each regressor surpasses a given critical value, the data matrix will be composed of a compact set of these filtered regressors. The critical drawback of this approach is that regressors that might contain overlapping information could end up being chosen, since the approach concerns only the marginal predictability of each variable.

On the other hand, soft-thresholding takes into account what other regressors are included in the set, which is based on estimating the matrix, Γ,

$$Y_{t+h} = \alpha W_t + \Gamma X_t + \epsilon_{t+h} \tag{20}$$

where $\epsilon_{t+h} \sim N(0,\ \sigma^2)$. Through a specified loss function, a shrinkage mechanism is imposed by putting a smaller weight or zeroing the coefficients of regressors that have weak explanatory powers, which can be thought of as a model specification as well. Ridge, lasso, EN, and LARS belong to this category.

4.1 Learning algorithm

4.1.1 Bagging (bootstrap aggregation)

Bootstrap aggregation (Bagging, Breiman (1996)) is a learning algorithm in which bootstrap samples $\{X^*, Y^*\}_{b=1}^{B}$ are drawn from the original training sample $\{X, Y\}$ and a shrinkage mechanism is imposed on the bootstrapped samples repeatedly. At the final stage, the bagging values $\widehat{Y}_b^* = \widehat{\beta}_b^* X_b^*$ for $b = 1, ..., B$, are averaged as

$$\widehat{Y}^{Bagging} = \frac{1}{B}\sum_{b=1}^{B} \widehat{Y}_b^*(X_b^*),$$

which alleviate prediction variation. Following Stock and Watson (2002), bagging estimation is based on this shrinkage equation,

$$\widehat{Y}_{t+h}^{Bagging} = W_t\widehat{\beta}_w + \sum_{j=1}^{r} \varphi(g_j)\widehat{\beta}_{F_j}\widehat{F}_{t,j} \tag{21}$$

where

$$\varphi(g) = 1 - \Phi(g+c) + \Phi(g-c) + g^{-1}[\phi(g-c) - \phi(g+c)]$$

ϕ stands for the standard normal PDF, Φ for the standard normal CDF, $g = \sqrt{T}\frac{\widehat{\beta}_{F_j}}{s_e}$, and c is a critical value (here we set as $c = 1.96$). The estimation proceeds by regressing Y_t on W_t first and then regressing the residual term, $Y_{t+h} - W_t\widehat{\beta}_w$, on the estimated factors $\widehat{F}_{t,j}$.

4.1.2 Boosting

Boosting is another forward stage-wise learning algorithm introduced by Freund and Schapire (1995). By setting the sample average as an initial value, boosting sequentially updates and aggregates estimates obtained up to the current steps with a weight. The first variant of Boosting algorithm,"AdaBoost", was proposed by Freund and Schapire (1995). Then, "Real AdaBoost", which involves an exponential loss function, was suggested by Friedman et al. (2000), after which "L_2 Boosting," using a quadratic loss function, was proposed by Friedman (2001).

Next, Bai and Ng (2009) proposed two boosting algorithms applicable to an out-of-sample forecasting framework, component-wise L_2 boosting and block-wise L_2 boosting. First, Component-wise L_2 boosting considers each variable as a separate potential regressor and attempts to minimize $SSR(i)$ that is left after regressing with the i^{th} regressor. By combining principal component analysis with this Boosting method (component + boosting), the drawback that comes from using the pure principal component approach solely can be avoided. Next, block-wise L_2 boosting is an algorithm that treats a block of lagged terms as a regressor. Their simulation results show that component-wise L_2 boosting is consistent and results in a relatively parsimonious model, while block-wise L_2 boosting is more efficient. In this paper, I use the component-wise L_2 Boosting method as one of the shrinkage estimation methods. For more detailed information on the procedure, refer to Bai and Ng (2009).

4.2 Penalized regression methodology

A penalized regression refers to a regression method bounded by a certain penalty term, $G(\beta)$. A generic penalized regression can be represented as

$$\min_{\beta} RSS + \lambda G(\beta).$$

where RSS is the sum of squared residuals and λ is a shrinkage parameter $(0 \leq \lambda \leq \infty)$. When $\lambda = 0$, it just reduces to the ordinary least squares (OLS) estimation. Therefore, one can interpret a penalization regression as a constrained OLS regression problem. Here, the penalty term, $G(\beta)$, is devised to punish the act of adding too many variables, implying that the virtue of parsimony is valued. By shrinking or zeroing the coefficients of uninformative variables, this penalized regression specifies a model. Traditional variable selection approaches, such as AIC, BIC, and HQIC, are computationally infeasible when N and/or T grow(s) overwhelmingly large. In this subsection, as alternatives to these IC approaches, I consider ridge, lasso, NNG, LARS, and the EN.

If $G(\beta) = \sum_J |\beta_J|^\gamma$, it becomes a bridge regression, a generalized version of ridge and lasso,

$$\min_{\beta} RSS + \lambda \sum_J |\beta_J|^\gamma$$

where γ is a tuning parameter ($\gamma \geq 0$). When $\gamma = 1$ it becomes lasso, and ridge when $\gamma = 2$. Alternatively, the bridge estimator, $\widehat{\beta}^{Bridge}$, can be expressed as,

$$\widehat{\beta}^{Bridge} = \underset{\beta}{\text{argmin}} \ \ RSS \ \ \text{subject to} \ \sum_J |\beta_J|^\gamma \leq c$$

where c adjusts the degree of shrinkage. [6]

[6] Fu (1998) pointed out that bridge regression can have a Bayesian interpretation: the whole Bridge regression problem can be interpreted as minimizing a posterior distribution, with the bridge penalty part, $\sum_J |\beta_J|^\gamma$, as a prior. By plugging in different values for parameters, λ and γ, a prior distribution can be adjusted, which reflects different preferences and can lead to different conclusions on a model specification problem in the end.

4.2.1 Ridge and least absolute shrinkage operator

A ridge estimator is a bridge estimator with a L_2 penalty,

$$\widehat{\beta}^{Ridge} = \underset{\beta}{\text{argmin}} \ RSS + \lambda \sum_{J} \beta_j^2$$

where $\lambda \ (> 0)$ is a penalty parameter that controls the degree of shrinkage (Hoerl and Kennard (1970)). The ridge problem also can be represented in matrix,

$$\min (Y - X\beta)'(Y - X\beta) + \lambda \beta'\beta.$$

and the ridge estimator as,

$$\widehat{\beta}^{Ridge} = (X'X + \lambda I)^{-1} X'Y$$

where λ is a shrinkage parameter. After obtaining the ridge estimates, the forecast value is computed as,

$$\widehat{Y}_{t+h} = W_t \widehat{\beta}_w + \widehat{f}_t \widehat{\beta}_f^{Ridge}.$$

However, it has been pointed out that, owing to the property of L_2 penalty, ridge ends up retaining most of regressors in the original data matrix, which detracts from its support as it lacks the virtue of parsimony. Then, Tibshirani (1996) introduced a lasso estimator that replaces $L_2$with L_1 penalty,

$$\widehat{\beta}^{lasso} = \underset{\beta}{\text{argmin}} \ RSS + \lambda \sum_{J} |\beta_j|.$$

Contrary to ridge, lasso enables uninformative regressors to be dropped. That is why it is considered to be a stabilized version of Ridge. Although it has not been clearly confirmed that lasso dominates ridge and bridge uniformly, lasso has been widely used because of its parsimony.

However, as Zou and Hastie (2005) pointed out, it is found that, lasso works poorly when $N >> T$, while it is dominated by Ridge when $T >> N$. In addition, lasso is lack of the 'grouping effect'. [7] In my empirical application, I will implement ridge, a bridge regression with L_2 penalty term.

4.2.2 Non-negative Garrote (NNG)

Breiman (1995) proposed Non-negative Garrote (NNG), an alternative way of selecting a subset of regressors. According to Breiman (1995)'s simulation and empirical results, NNG is somewhere between an unstable ordinary subset selection method and a stable ridge method, and leads to better than or at least comparable forecasting accuracies to those two approaches. Specifically, Breiman (1995)'s idea is to estimate a shrinkage factor, $s(\lambda) = (s_1(\lambda), \ldots, s_p(\lambda))$, based on the optimiaztion problem,

$$s(\lambda) = \underset{s}{\text{argmin}} \ \frac{1}{2} \| \ Y - Bs \ \|^2 + T\lambda \sum_{j=1}^{N} s_j, \quad s.t. \quad s_j > 0 \ \ for \ all \ j$$

where $\lambda \ (> 0)$ is a tuning parameter, $B = (B_1, ..., B_N)'$, each $B_j = X_j \widehat{\beta}_j^{LS}$, and $\widehat{\beta}_j^{LS}$ is the usual OLS estimator. Under the restriction that $XX = I$, $s_j(\lambda)$ is determined as,

$$s_j(\lambda) = (1 - \frac{\lambda}{(\widehat{\beta}_j^{LS})^2}) \tag{22}$$

and then, the NNG estimator is derived as,

$$\widehat{\beta}_j^{NNG}(\lambda) = s_j(\lambda) \widehat{\beta}_j^{LS}. \tag{23}$$

Thus, $\widehat{\beta}_j^{NNG}$ is essentially dependent on $\widehat{\beta}_j^{LS}$, which implies that its computation burden is only as much as that of OLS. At the final stage, the predicted

[7] A regression is said to have the grouping effect if the coefficients of highly correlated variables in a group result in being equal. For more rigorous mathematical discussion on this issue, refer to Zou and Hastie (2005)'s Lemma 2.

value of Y is computed as,

$$\widehat{Y}^{NNG} = X\widehat{\beta}^{NNG}(\lambda). \tag{24}$$

As the magnitude of $\widehat{\beta}_j^{LS}$ value gets larger, $s_j(\lambda)$ moves closer to 1 (equation (22)). Then, it is less likely that $\widehat{\beta}_j^{NNG}$ becomes insignificant so that the corresponding j^{th} variable is less likely to be dropped out. On the other hand, if $\widehat{\beta}_j^{LS}$ moves closer to 0, then $\widehat{\beta}_j^{NNG}$ will also converge to 0 for a given λ and a model without the j^{th} variable will be chosen.

Yuan and Lin (2007) showed that NNG has the path consistency property as long as the initial estimate is consistent, which cannot be guaranteed all the time when it comes to LARS, lasso, and the EN. [8] In addition, Yuan and Lin (2007) proposed an efficient algorithm for the complete NNG solution path that uses the OLS estimate as an initial value. In this paper, I follow Yuan and Lin (2007)'s algorithm. For the detailed procedure, it is recommend to refer to Yuan and Lin (2007).

4.2.3 Least angle regression

Efron et al. (2004) proposed Least Angle Regression (LARS), a stylized algorithm of a forward-stagewise regression. Its algorithm starts to find x_1 that is most correlated with a response variable and then sequentially adds one more covariate to find the most correlated one with the residual left at that time. Thus, LARS first remains with x_1 until it is claimed that another covariate has more correlated information with the current residual. Once one finds x_2, one moves in the direction of equiangular degree between x_1 and x_2 until one finds another x_3. Then, one keeps moving along the "least angle" direction until one finds the next highly-contributing predictor. For the detailed algorithm, see Efron et al. (2004).

[8] A solution path is called to be "path consistent" if the path produces "desirable" estimates, resulting in consistent estimation and variable selection.(Yuan and Lin (2007))

Gelper and Croux (2008) modified this methodology so that it can be applied to a time series setting, times-series least angle regression (TS-LARS). TS LARS allows for the inclusion of lagged terms so that no information that may be contained therein would be lost. To preserve the original dependent structure of times series, the authors used blocks of regressors, not just a single regressor at a time. For the detailed computational procedure of TS LARS, refer to Gelper and Croux (2008).

4.2.4 The elastic net

To overcome the limitations of ridge and lasso that are mentioned above, Zou and Hastie (2005) proposed naive EN, a variant of the EN approach, which is basically a convex combination of ridge and lasso,

$$\hat{\beta}^{EN} = \underset{\beta}{\operatorname{argmin}} \; RSS + \lambda_1 \sum_j |\beta_j| + \lambda_2 \sum_j \beta_j^2 \tag{25}$$

where each λ_1 and λ_2 are positive fixed values. When $\lambda_1 = 0$ it just reduces to ridge and, and it becomes lasso when $\lambda_2 = 0$.

Defining an augmented data set, $\{X^*, Y^*\}$, as

$$X^*_{(T+N)\times N} = (1+\lambda_2)^{(-1/2)} \begin{pmatrix} X \\ \sqrt{\lambda_2} I_N \end{pmatrix} \quad \text{and} \quad Y^*_{(T+N)\times 1} = \begin{pmatrix} Y \\ 0_N \end{pmatrix},$$

$\widehat{\beta}^{naiveEN}$ can be expressed as

$$\widehat{\beta}^{naiveEN} = \frac{1}{\sqrt{1+\lambda_2}} \widehat{\beta}^*, \tag{26}$$

where

$$\widehat{\beta}^* = \underset{\beta^*}{\operatorname{argmin}} \; |Y^* - X^*\beta^*|^2 + \frac{\lambda_1}{\sqrt{1+\lambda_2}} |\beta^*|. \tag{27}$$

Here, with the augmented data set, one is not confined to choose predictors only up to T, but freely up to N even when $N >> T$. Secondly, not only

does the naive EN enjoy the computational easiness of lasso, but it also has the grouping effect that lasso doesn't have, since the naive EN is strictly convex.

One noteworthy point is that, unless it is close enough to either ridge or lasso, it performs poorly, which is the very reason why it is called "naive". In addition, since the naive EN can be interpreted as a Ridge regression with lasso-type shrinkage, it suffers from a double amount of shrinkage. To deal with the double-shrinkage problem, Zou and Hastie (2005) propose the EN as an alternative to the naive EN. $\widehat{\beta}^{EN}$ can be defined as,

$$\widehat{\beta}^{EN} = \sqrt{1+\lambda_2}\widehat{\beta}^{*}, \tag{28}$$

so the EN can be thought of as a re-scaled version of the naive EN,

$$\widehat{\beta}^{EN} = (1+\lambda_2)\widehat{\beta}^{naiveEN}. \tag{29}$$

It holds good properties of the naive approach while solving the bias problem coming from the double-shrinkage. In addition, Zou and Hastie (2005) showed that the EN can be viewed as a stabilized version of lasso (see Theorem 2). Lastly, Zou and Hastie (2005) proposed LARS-EN, a more efficient version of the algorithm. In this paper, I will implement Bai and Ng (2008)'s version of LARS-EN, which applied LARS-EN to a time-series context.

4.3 Model Averaging Approach

The average of forecasting values computed from a set of alternative models delivers an aggregated insight for the unknown true DGP that cannot be obtained if just a single chosen model is used. First, even if a chosen model is the best based on a certain criterion or evaluation method within a set of candidate models, there could be still valuable information left out that could have been

gathered from dropped alternative models. [9] Secondly, the usual PC approach associates factors with the eigenvectors corresponding to the r largest eigenvalues of $X'X$, with its whole focus on summarizing the information contained in the big data set. Since this dimension reduction approach does not take into account which variable is set to forecast, it is not guaranteed that those factors will have the highest predictive power for a particular forecasting variable. In this paper, I use simple mean average model that assigns a uniform weight, $w = \frac{1}{l}$, to each forecasting model, $k = 1, \ldots, l$, and BMA (Bayesian model averaging) with two different prior settings. These model averaging approaches can be thought of as an alternative method that could prevent the model misspecification problem that arises from choosing a single approximation model.

4.3.1 Bayesian Model Averaging (BMA)

BMA derives the expected value of Y_{T+h} as,

$$E(Y_{T+h} \mid Data) = \sum_{k=1}^{l} p(M_k \mid Data) E(Y_{T+h} \mid Data, M_k). \tag{30}$$

where $p(M_k \mid Data)$ is the posterior model probability of model k, M_k, and it is defined as

$$p(M_k \mid Data) = \frac{p(Data \mid M_k) p(M_k)}{\sum_k p(Data \mid M_k) p(M_k)} \tag{31}$$

and the likelihood of M_r as,

$$p(Data \mid M_r) = \int p(Data|\theta_r, M_r) p(\theta_r|M_r) d\theta_r. \tag{32}$$

[9] The phenomenon that the model averaging approach outperforms a single model chosen based on an evaluation criterion is called the forecast combination puzzle.

[10] It requires to specify $p(M_r)$, model space priors, and $p(\theta_r|M_r)$, parameter priors. Rather than fixing a single forecasting model as the best one, here each model M_r itself is considered as a random variable and the notion of model uncertainty is incorporated via prior model probabilities. BMA averages over the set of models, $r = 1, \ldots, R$, with its posterior model probability as a weight (30) and, by increasing r to infinity, it is expected that the averaged value will converge to the true expected value of Y_t in the limit.

To deal with the linear effects of W_t in (20), I will redefine the data set as $Y^*_{t+h} = [I_t - W_t(W_t'W_t)W_t']Y_{t+h}$ and $F^*_t = [I_t - W_t(W_t'W_t)W_T]F_t$ and forecasting framework will be based on,

$$Y^*_{t+h} = \beta^* F^*_t + e^*_t. \tag{33}$$

where $e_t \sim N(0, \sigma^2)$.

First, the model space prior is set as,

$$p(M_p) = p(\Gamma) = \prod_{k=1}^{l} \theta_k^{\Gamma_k}(1-\theta_k)^{\Gamma_k}.$$

The marginal probability that whether a specific series enters or not, is specified as $\theta_k = p(\Gamma_k = 1) = 1 - p(\Gamma_k = 0)$ and, depending on whether β^* is ignorable or significant, Γ_k is assigned as 0 or 1. This Bayesian variable selection serves a role as a 'model identifier'. Among many ways to set the prior, fixing θ_k as $\frac{1}{2}$, a non-informative prior, is the easiest and safest way. Alternatively, the prior probability can be related to the eigenvalues of data set by setting

$$\theta_i = \frac{\lambda_i}{\lambda_1}$$

[10] Bayesian model comparison is based on $\frac{p(M_1|Data)}{p(M_2|Data)}$, the posterior odds, which is equivalent to $\frac{p(Data|M_1)}{p(Data|M_2)} \times \frac{p(M_1)}{p(M_2)}$, the product of Bayes factor and the prior odds. Here, model priors represent an initial representation of model uncertainty and data updates the prior odds via the Bayes factor.

where λ_1 is the smallest eigenvalue and λ_l is the largest.

Next, as for the priors for the parameters, β_Γ^* and σ^{-2}, I will set a natural conjugate prior as,

$$\beta_\Gamma^* \mid \sigma^2, \Gamma \sim N(\underline{\beta}_\Gamma^*, \sigma^2 \underline{V}_\Gamma) \tag{34}$$

$$\sigma^{-2} \mid \Gamma \sim IG(\underline{s}^{-2}, \underline{v}).$$

which features analytical tractability. To proceed, one has to specify hyperparameters, $\underline{\beta}_\Gamma^*, \underline{V}_\Gamma, \underline{s}^{-2}$, and $\underline{v}$ and, given that specific information nor subjective point of view is not available, I will construct non-informative priors for those to 'minimize prior influence'(Chipman et al. (2001)). First, I will use a non-informative prior for σ^{-2} by setting $\underline{v} = 0$ such that the prior information doesn't influence the posterior probabilities at all (Koop and Potter (2004)). $\underline{\beta}_\Gamma^*$ is set as 0. However, when it comes to $\underline{V}$, I do not use non-informative priors, that's because otherwise its posterior distribution would end up either degenerating or being dependent on arbitrary normalizing factors. Following Fernandez et al. (2001), I will adopt a g-prior for $\underline{V}$ (Zellner (1986) and Fernandez et al. (2001)). Thus, the matrix, $\underline{V}$, is fixed by the specification of g through this relationship,

$$\underline{V} = (g(F^*)'F^*)^{-1},$$

which means that the level of prior precision is to be synced with sample variance-covariance matrix information divided by the g-prior chosen. In this study, I use two g priors suggested by Koop and Potter (2004),

$$g = \frac{1}{T}$$

$$g = \frac{1}{l^2}$$

where T is the size of a time series vector and l ($k = 1, ..., l$) is the number of forecasting models considered. Then, $\widehat{Y}_{t+h}^* = \widehat{\beta}_F F_t^*$, finally, $\widehat{Y}_{t+h}$ is computed as $[I_t - W_t(W_t'W_t)W_t']^{-1}\hat{Y}_{t+h}^*$.

Theoretically, BMA is considered to be the optimal approach by selecting a correct model in the limit, which can be done by assigning a higher weight to the correct model via posterior probability. However, this holds under the assumption that the correct set of models and proper prior information is specified, which is hard to be satisfied in reality. [11]

5 Empirical Application

14 different shrinkage and machine learning algorithms will process total 144 macroeconomic and financial variables to predict 11 major macroeconomic variables. Forecasting results will be evaluated using Jin et al. (2016)' forecast superiority tests that are based on the application of first and second order stochastic dominance principles, as discussed in Linton et al. (2005). Also as an attempt to empirically assess the trade-offs associated with using the different loss function specific (DM and data snooping) and loss function free (GL and CL forecast superiority) predictive accuracy tests in big data environment, Jin et al. (2016)'s general loss (GL) forecast superiority and convex loss (CL) forecast superiority test results will be compared to the case when loss function specific approaches are used.

5.1 Data Description

The data set to be used in this empirical application is Kim and Swanson (2014)'s updated and expanded dataset of Stock and Watson (2002, 2012)'s. 144 US macroeconomics time series that spans from January 1960 to May 2009 (monthly) will be our data set, X_t, whose full list is available in Kim and Swan-

[11] In practice, BMA acts more like model selection rather than model averaging by assigning non-uniform weights through posterior probabilities and it is found that this model searching does not make BMA immune from the overfitting issue because of the likelihood's sensitivity to random variations (Domingos (2000)).

son (2013). Using this big dimensional data set, I forecast 11 macro variables, Y_{t+h}, which are the core economic variables that the Federal Reserve Bank pays close attention into for formulating monetary policies: unemployment rate(UR), personal income less transfer payments (PILT), 10 year Treasury-bond yield (TB10Y), consumer price index (CPI), producer price index (PPI), non-farm payroll employment (NPE), housing starts (HS), industrial production (IPX), M2, S&P 500 index (SNP), and gross domestic product (GDP). Before proceeding with the empirical analysis, all forecasting variables, Y_{t+h}, are transformed to make each time series stationary: basically all level variables are log-differenced and other variables already in log form are just taken as the first difference between t and $t-1$. For detailed descriptions on how each variable is transformed, refer to Table 3.A.1.

5.2 Forecasting methods and models

In total, I attempted four different specifications. First, the baseline is Specification Type 1 (SP1), comprising 14 different forecasting and shrinkage methods (Table 3.A.3) that are horse-race compared, including benchmark models (AR, ARX, and CADL), purely factor-based approaches (PCR and FAAR), some hybrid methods combining shrinkage methods and factor analysis, and model-averaging approaches (simple mean averaging and Bayesian averaging). Specification Type 1L (SP1L) is basically the same as SP1, except that y_{t-h} or $\widehat{f}_{t-h}$ are allowed to enter as regressors. Specification Type 2 (SP2) filters the high-dimensional data set based on the equation (20) rather than using the original data set, X_t. Lastly, Specification Type 3 (SP3) conducts only shrinkage methods, not using the factor approach in any stage. For SP1 and SP1L, I horse-race compare the whole set of 14 forecasting methodologies listed in Table 3.A.3, while, as for SP2 and SP3, PCR and FAAR are dropped. [12]

[12] SP2 uses shrinkage methods to screen regressors at the first stage and PCR and FAAR is for when using the original data set. SP3 does not involve factor analysis at any stage.

As for forecasting evaluation methods, I will order the forecasting performances of 14 different forecasting methods using both Jin et al. (2016)'s stochastic dominance approach and White (2000)'s moment-based approach. JCS test will be testing the null when one of 14 forecasting methods is set as a benchmark model, 1, against the rest of 13 alternatives each time. The interpretation of the hypothesis testing will be like this, failing to reject the null means e_1, a sequence of forecast errors from benchmark, 1, CL outperforms e_k, a sequence of forecast errors from alternatives, for $k = 2, \dots l$, while rejecting the null means that e_1 does not CL outperform, e_k. For practical implementation, I replace the supreme in TG_n and TC_n with the maximum, by computing the statistics over densely partitioned points of the joint support, $\chi_n = \{x_1, ..., x_n\}$, with an evenly spaced grid of size, $1.5 * p^{0.6}$. Plus, I use 98 % empirical distribution of forecasting errors pooled from all forecasting models by dropping 1% outliers on each side. For a fixed Y_t and h, if there are several p-value results failing to reject the null, that means those are comparably performing well. However, to choose a single best model in that case, I compare the magnitudes of p-values and choose a model with a higher p-value as done in Jin et al. (2016)'s empirical application. [13]

White's Reality Check tests,

$$H_0 : S_p = \max_{k=2,...,l} S_p(1,k) \leq 0 \tag{35}$$

where

$$S_p(1,k) = \frac{1}{\sqrt{p}} \sum_{t=R}^{T-1} (g(\widehat{u}_{1,t+1}) - g(\widehat{u}_{i,t+1})), \quad k = 2, ..., l \tag{36}$$

with a chosen loss function, g. Here, to see the implications of the choice of a loss function, I will run a data snooping testing by setting g as MAFE (Mean Absolute Forecast Error) and MSFE both. Critical values will be constructed

[13] Having larger p-value means the evidence against the null is weaker in a probabilistic sense (Stock and Watson (2007)). In the meanwhile, Having p-value closer to 0 means it is very unlikely the sample test statistic would have been drawn if the null is true, which implies it makes sense to conclude that the null is not true.

via Politis and Romano (1994)(PS)'s stationary bootstrap. [14] White test is also a multiple forecast comparison testing the outperformance of a fixed benchmark against the rest of 13 alternatives. Failing to reject the null means the outperformance of a benchmark, while rejecting the null implies there is at least one alternative that outperform the benchmark.

Lastly, for comparison, I also report MSFE results and DM test results, which are the basis on which Kim and Swanson (2014) made conclusions. MSFE is defined as

$$MSFE_{k,h} = \sum_{t=R-h+2}^{T-h+1} (Y_{t+h} - \hat{Y}_{k,t+h})^2. \tag{37}$$

for $k = 1, \ldots, l$. DM test is a pairwise comparison between a benchmark, here AR, and an alternative, specifically setting the null as, $H_0 : E[g(e_{1,t}) - g(e_{k,t})] = 0$ for $k = 2, \ldots, l$. The test statistic is defined as $DM_{1,k} = \frac{1}{P} \sum_{t=R+1}^{T} \frac{d_t}{\widehat{\sigma}}$ where $d_t = g(\widehat{e}_{1,t}) - g(\widehat{e}_{k,t})$ for $k = 2, \ldots, l$ and $\widehat{\sigma}$ is a heteroskadasticity and autocorrelation consistent estimator of the standard error of $\bar{d}$, the mean of d_t. A loss function, g, is fixed as MSFE.

5.3 Empirical Results

The total 144 macroeconomic variables are employed as explanatory variables to predict 11 forecasting target variables and it ranges from 1960 Jan to 2009 May. Four different forecasting horizons ($h = 1, 3, 6, 12$), four different mean block lengths ($W_n = \frac{1}{S_n} = \{2, 5, 7, 10\}$), two different estimation strategies (recursive or rolling), and four different specifications (SP1/SP1L/SP2/SP3) will be attempted.

[14] In Jin et al. (2016)'s simulations, they used PS's stationary bootstrap for stationary DGPs and block bootstrapping for heterogeneous DGPs. In this paper, since the data sets are transformed as specified in Table 3.A.1 to induce stationarity, I will employ PS's stationary bootstrap for constructing critical values with four different mean block length, $\{2, 5, 7, 10\}$. Since the results turn out to be robust to the four different block length tried, I just report test results when the mean block length is set as 5.

Forecast evaluation results are ordered in this way: first, Table (3.A.5) to Table (3.A.9) summarize best performance forecasting methods of different evaluation methods: from Jin et al. (2016)'s GL/CL forecasting superiority test results, to White's Reality Check test results (when a loss function is set as MSFE and MAFE both), lastly to MSFE results. Table (3.A.10) to (2.A.13) report detailed JCS's and White's test results all throughout different specifications but just for $h = 1$ case, for the sake of brevity. [15] Table (2.A.14) covers MSFE results and also contains DM test results, a forecast equality test between AR, a fixed benchmark here, and an alternative, with $*(**)$ denoting the rejection of the null of the benchmark's superiority at the $10(5)\%$ significance level. Therefore, largely, the forecasting results can be categorized into two ways, distributional comparison vs. moment-based comparison and multiple comparison (JCS's GL/CL test and White's Reality Check test) vs. pairwise comparison (Diebold and Mariano (2002)'s test).

As a tool of interpreting the distributional comparison results, I also provide two types of plots as a eyeball test, a plot of CDFs/SDFs comparison and plot of GL/CL test results, along with tabulated GL/CL test results (Figure 3.A.1 and 2.A.4). Throughout different configurations, there are many crossings observed in both types of plots as expectedly, which is reasonable since 14 different forecasting methods are being compared simultaneously. Secondly, depending on different configurations, the results tend to fluctuate. It is quite understandable that the total number of possible permutations in this setting is 11 (different forecasting variables) $*$ 2 (different estimation window types, recursive or rolling) $*$ 14 (different forecasting methods) $*$ 4 (different values of bandwidth size) $*$ 3 (different data split points) $*$ 4 (different specifications, $SP1/SP1L/SP2/SP3$) is 14,784. However, within a fixed setting, (whether RE/RO, $p = 200/300/400$, SP1/SP1L/SP2/SP3), if (a) forecasting model(s) is/are obviously performing better than other(s), then it seems to be that JCS

[15] The detailed results for other settings are available upon request.

test results are robust over different values of mean block length, W_n. [16] This implies that, in practice, given a forecasting problem, fixed Y_t, h, and P (where $P = R + T$), if one is sure about which method is more appropriate, recursive or rolling and SP1 or SP1L or SP2 or SP3, mostly, one can sort out best/worst (group of) forecasting method(s) in a GL/CL outperformance sense.

The big pictures that I can get from these results as a whole can be summarized like these: first, when it comes to GL outperformance test results, there are some cases I cannot point out which exact model wins since, for a given Y_t and h, all test results when one of 14 models fixed as a benchmark at a time result in rejecting the null. However, when moving to a higher order stochastic dominance comparison (CL outperformance test), there are some cases where I can judge (a) clearer stochastic dominance relationship(s) (Table 3.A.10 and 2.A.14). The underlying theoretical reasoning is that, as pointed out in section 2, at a higher order stochastic dominance, the magnitude of the difference of the areas under CDFs is more amplified by the power multiplication of $F(x)$ itself, which can be interpreted as a weight. [17] Jin et al. (2016)'s simulation results also show that the probability of correctly rejecting the null in a CL outperformance sense is higher than the probability of rejecting the null when the null is not true in a GL sense. It is also related to the theory that first order stochastic dominance implies second order stochastic dominance.

Secondly, even though the exact hybrid method that wins differs across

[16] When it comes to the choice of different estimation windows, a rolling scheme, a limited memory estimator, will be more appropriate if using information set limited to relatively recent time periods is more pertaining to explaining the variable of interest, rather than recursive scheme, an expansionary memory estimator, that includes even all distant past data that may not have informative power for forecasting Y_{t+h} anymore. When it comes to rolling window, the parameter estimation errors doesn't vanish even in the limit, since the coefficient estimates don't converge to fixed population parameter values, remaining as strong mixing random variables (Giacomini and White (2006)) and Corradi and Swanson (2013)).

[17] The $(s+1)^{th}$ order stochastic dominance compares the areas under the curve of $D^S(y) = \int_0^y D^{(S-1)}(z)dz = \int_0^y F^{(S-1)}(z)dz$. It can also be viewed in the way that stochastic dominance of higher order reflects the depth of distributions, compared to the first order stochastic dominance which just counts the number of realizations of random variables up to a certain value in the support.

different forecasting variables and forecasting horizons, the forecasting results from loss function-robust JCS test and loss function-dependent moment based tests both indicate that hybrid methodologies, combining shrinkage and factor approach, perform better than benchmark models (AR & ARX & CADL), purely factor-based models (FAAR & PCR), and model averaging methods (simple mean averaging & Bayesian averaging). For example, hybrid models and methods "win" around 50 % when selection is based on the JCS test, and only around 1/2 of the time based on the application of DM tests.

Specifically, when it comes to first order distributional comparison, hybrid methods predominate for SP1L case and Shrinkage methods for SP3, across the 11 different forecasting variables and the 3 different forecasting horizons. As for the SP1 case, the hybrid methods and FAAR seem to be outstanding. When moving to second-order distributional comparison, one can observe that hybrid methods dominates across the SP1, SP1L, and SP2 cases, and shrinkage methods and Mean are performing well for SP3. In addition, I observe that, compared to the first-order stochastic dominance test results, the second-order stochastic dominance results report that the simple averaging method wins more often. White's test results describe a similar picture. The overall results indicate that, as for the SP1/SP1L/SP2, hybrid methods are doing well, while, as for the SP3, benchmark models and shrinkage methods show good performances. In addition, especially as for the SP2 case, the simple mean average method seems to be quite competitive.

Here the caveat is that, although the two evaluation approaches, distributional comparison and moment-based comparison, describe the similar pictures in overall favoring hybrid methods (and shrinkage when it comes to the SP2 case), there are differences in details among the results of the different evaluation methods. And, it should be remembered that the power performances of these two different evaluation methods, JCS and White's cannot be formally compared, since their statistics are different asymptotically in the null

(Jin et al. (2016)). Therefore, I recommend that it is better to consider both approaches as complementary, since the distributional comparison has its own merits as emphasized in this paper but the moment-based tests can have better finite sample performances as shown in Jin et al. (2016)'s simulations. [18]

Thirdly, it is interesting to find that first, SP1L, a specification that allows the inclusion of lagged terms, does not necessarily result in better results all the time. Although, one cannot flatly state that it is not being helpful at all for improving forecasting performances, but, neither can one state that it effectively improves results, in terms of MSFEs (Table 2.A.14). Thus, given that adding the lagged terms does not change the underlying forecasting structure fundamentally, what matters to forecasting performances seems to be a specific shrinkage method that effectively weights and selects relevant information. In addition, when it comes to the SP1L specification, it seems to be more difficult to compare forecasting performances in a GL outperformance sense, with p-values as 0 in most of cases. This phenomenon stems from the fact that the forecasting performances are so close to the extent that the empirical CDFs of forecasting errors almost coincide, which makes it harder to rank the results in a GL outperformance sense. However, the CL outperformance results, comparing integrated CDFs, SDFs, provides a better picture to enable the discernment of better/worse methodologies. In addition, the SP3 case, which does not involve a factor approach in any stage, does not necessarily lead to worse results in an MSFE sense. FOSD (first-order stochastic dominance) (first-order stochastic dominance) results says shrinkage methodologies replace the role of hybrid methodologies, while SOSD (second-order stochastic dominance) results side more with Mean and White's results with benchmark models and shrinkage methodologies.

Fourth, it turns out that the two White's Reality test results from the MAFE

[18] Even though Jin et al. (2016)'s simulation results have pointed out the moment-based tests have better powers in small sample sizes given a loss function, it implies that one surely knows which loss function is appropriate to use, which is an additional decision problem.

loss function case and the MSFE loss function case do not necessarily result in the same conclusions all the time, which implies that the choice of a loss function matters. The most noticeable difference is that, when NNG is set as a benchmark, White's reality check test with MAFE always rejects the null for all specifications and for variables, while White's reality check test with MSFE tells that NNG performs well, especially for SP2 and SP3, or even picks NNG as the best one. This finding once again highlights the loss function-robust characteristic of the JCS approach.

Lastly, throughout different estimation schemes, specifications, and forecasting horizons, it is more difficult to forecast TB (10-year treasury bond) and HS (Housing starts) variables accurately than the other variables, as evidenced from the MSFE results. In addition, I can observe that the magnitudes of MSFE generally increase as h increases for both recursive and rolling schemes and all specifications, which is an expected result in the forecasting literature since uncertainties for a more distant future time period are higher.

Now, for the sake of brevity, I will focus on interpreting the results for $Y_t = UR$ and $h = 1$ case using plots, specifically the plots of the empirical CDFs/SDFs comparison and the plots of the p-values of GL and CL forecast superiority test (Figure (3.A.1) to 2.A.4). Tabulated results are be found in Tables 3.A.10 to 2.A.14 as well.

First, as an eyeball test, I present the plots of (1) the probability difference between empirical CDFs of an alternative and a benchmark against over the joint support of forecasting errors and the integrated probability difference between the empirical CDFs (i.e. probability difference between the empirical SDFs) and (2) p-values for GL/CL forecasting superiority test over the different values of bandwidth. First, as for the plots of the comparison over empirical CDFs(SDFs), I set an AR (autoregressive) model as the benchmark model, whereby the values on the y-axis indicate the (integrated) probability

difference of empirical CDFs (SDFs) between an alternative model and AR. On the other hand, the plots of p-values for GL and CL forecast superiority test are when one particular method is set as a benchmark against all different alternative forecasting methods. Thus, the legend for the lines in the empirical CDFs/SDF difference plot indicates a particular alternative method compared with the autoregressive model and the plot describes a glimpse of pairwise comparison between the benchmark, 1, and the specific alternative, k, over the joint support. On the other hand, the legend in the p-value plot for GL and CL indicates the benchmark model fixed on and it plots the formal JCS results, p-values, over different bandwidth values tried.

As for the interpretations of the plots of GL/CL test results, since I compare 13 alternative methods against a benchmark model, it is quite less likely to have perfectly non-overlapping plots among the 14 forecasting methods: if that is the case, it is likely more difficult to pick up a model that is clearly dominant or dominated. And, as expected, I can observe this kind of phenomenon throughout most of all different variables, forecasting horizons, and block windows. However, when moving to a higher order stochastic dominance, there are some cases in which I can judge (a) clearer stochastic dominance relationship(s). In addition, even though there are some crossings observed, if one(some) of methods obviously dominate(s) other(s), it is possible to observe the stochastic dominance relationship is robust to different choice of bandwidth values, enabling the sorting out of better and/or poor forecasting methodologies in a given forecasting framework.

As a way of dealing with overlapping graphs, one may attempt restricted stochastic dominance approach (Davidson (2009)) whereby the forecasting comparison is conducted over a restricted set of the original domain. That is, if the distribution functions are too close to discern each other, one can restrict the domain of forecast errors and compare empirical CDFs/SDFs over the restricted range. Or, for instance, if people are more concerned about having

more big forecasting errors, then comparing the forecast results over a bigger magnitude range of forecast errors could provide a clearer picture.

Now, I deal with the detailed interpretation of $Y_t = UR$ and $h = 1$ case. From this specific case, foremost, I can confirm the pattern that hybrid methods perform relatively better than do benchmark models (AR and ARX and $CADL$) and purely factor-based approaches (PCR and $FAAR$), from this specific case as well. Even though specific forecasting models that outperform are moderately different across different forecasting variables and forecasting horizons, the forecasting results seems to indicate that, to deal with big data, shrinkage methodologies in general outperform relatively simple econometric models that do not involve weighting and shrinking information given.

Figure (3.A.1) plots the probability difference between empirical CDFs of an alternative and a benchmark, AR here, and integrated probability difference between them (i.e., probability difference between SDFs). This provides a general overview of pairwise comparison between an alternative and AR over the joint support. Since no line clearly lies over 0 or below 0 for all x but all of them pass across the zero line, one cannot declare GL or CL outperformance of a model based on this visual representation. Figure (2.A.2) plots the p-values of the GL and CL outperformance test over the values of bandwidth attempted, $W_n = \{2, 5, 7, 10\}$. It can be observed that, even though there are some crossings since in total 14 forecasting methodologies are compared, better forecasting methodologies dominate worse forecasting methodologies, whose results are robust to different values of W_n. Figure (2.A.3) shows the results of the GL test for other specifications, $SP1L$, $SP2$, and $SP3$ and Figure (2.A.4) contains the results of CL tests for those specifications.

If one is to make inferences based on the CL outperformance test, when it comes to the SP1 case, PCR, Bagging, Boosting, BMA2, Ridge, LAR, and, Mean seem to outperform. As for the SP1L case, Ridge, LAR, Boost, and PCR seem to be outstanding, while for the SP2 case, AR, BMA 1 & 2, Ridge, LAR,

EN, NNG, Mean are superior to others. Next, when it comes to the SP3 case, it is difficult to single out the best method(s), since all p-values of the GL test results are equal to 0 and the CL test results are quite indistinguishable. Thus, one can say that, when it comes to SP3, there are no forecasting methods that can explicitly dominate over the alternatives in the GL outperformance sense. However, second-order stochastic dominance results, which compare the integrated probability difference between CDFs, provides a slightly better picture that can differentiate the performances (Figure 2.A.4).

6 Conclusion

The notion of a loss function is also closely related to the decision problem associated with how one chooses to weight and summarize forecast errors. Thus, the choice between using loss function dependent evaluation and loss function free forecast evaluation involves assessing whether the relevant decision problem involves directly addressing loss function dependence, or eschewing the use of loss functions altogether. In this paper, I will empirically assess the trade-offs associated with using the different loss function specific (DM and data snooping) and loss function free (GL and CL forecast superiority) predictive accuracy tests. This will be done by carrying out an extensive set of empirical experiments using the factor augmented forecasting models and a variety of shrinkage and machine learning algorithms.

Analysis and ranking of the models and methods analyzed in our experiments is found to depend crucially on whether loss function dependent evaluation of their accuracy is carried out, or not. Namely, there tend to be differences between the best models chosen using distributional comparison based on JCS tests, and the best models selected by using reality check (or DM) tests. In addition, it is found that loss function specific test results are sensitive to

the specification of loss function, and in particular to whether MSFE or MAFE loss is used. With regard to the actual prediction models and methods utilized in our experiments, it is not possible to claim that a specific forecasting model "uniformly" dominates, across different forecast horizons and specification methods, for a given variable. However, it is noteworthy that hybrid models that combine shrinkage estimation with diffusion indexes perform better than benchmark models including linear models, purely factor-based models, and model averaging. Thus, there appears to be useful information in "big data", and the manner in which this information is extracted matters. Also, when carrying out loss function free model selection, the number of "wins" for hybrid models increases appreciably, relative to when loss function dependent model selection is carried out. In conclusion, we find that loss function specific and loss function robust tests are complementary, and much can be learned by utilizing both types of tests for empirical forecast model selection.

7 Appendix

Table 2.A.1: Target forecasting variables

Series Name	Abbreviation	Transformed Y_{t+h}
Unemployment rate	UR	$Y_{t+1} - Y_t$
Personal income less transfer payments	PI	$ln(Y_{t+1}/Y_t)$
10-year treasury bond	TB	$Y_{t+1} - Y_t$
Consumer price index	CPI	$ln(Y_{t+1}/Y_t)$
Producer price index	PPI	$ln(Y_{t+1}/Y_t)$
Nonfarm payroll employment	NPE	$ln(Y_{t+1}/Y_t)$
Housing starts	HS	$ln(Y_t)$
Industrial production	IPX	$ln(Y_{t+1}/Y_t)$
M2	M2	$ln(Y_{t+1}/Y_t)$
S&P 500 index	SNP	$ln(Y_{t+1}/Y_t)$
Gross domestic product	GNP	$ln(Y_{t+1}/Y_t)$

Table 2.A.2: Specifications

Specification	Description
SP1 (baseline)	horse-race comparison of 14 forecasting methods specified in Table 3
SP1l	lagged terms, y_{t-h} or $\widehat{f}_{t-h}$, allowed to enter
SP2	prescreening regressors before estimation and forecasting
SP3	not involving factor analysis at any stage

Table 2.A.3: 14 forecasting models and shrinkage methods employed

Names in abbreviation	Description
AR(SIC)	Autoregressive model with lags selected by SIC
ARX	Autoregressive model with exogenous regressors
CADL	Combined AR distributed lag model
FAAR	Factor augmented AR model
PCR	Principal components regression
Bagging	Bagging with shrinkage, $c = 1.96$
Boosting	Component Boosting, $M = 50$
BMA1	Bayesian model averaging with g-prior=$\frac{1}{T}$
BMA2	Bayesian model averaging with g-prior=$(\frac{1}{P})^2$
Ridge	Ridge regression
LAR	Least angle regression
EN	Elastic net
NNG	Non-negative garotte
Mean	Arithmetic mean

Table 2.A.4: Abbreviation

1	AR(SIC)	B (Benchmark)	
2	ARX		
3	CADL		
4	FAAR	F (Factor)	
5	PCR		
6	Bagging	H (Hybrid) (SP1/1L/2)	S (Shrinkage) (SP3)
7	Boosting		
8	BMA1		
9	BMA2		
10	Ridge		
11	LAR		
12	EN		
13	NNG		
14	Mean	M (Mean)	

Table 2.A.5: Best FOSD models

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	AR	BMA2	Boost	FAAR	AR	Bagging	BMA1	Mean	FAAR	BMA2/Ridge	BMA2
h=3	NNG	Ridge	Bag	FAAR	Boost	BMA1	BMA1	FAAR	NNG	FAAR	Boost
h=12	AR	PCR	Bag	FAAR	ARX	Boost	BMA1	PCR	Mean	NNG	Mean
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	EN	N\A	N\A	N\A	N\A	FAAR	FAAR	N\A	FAAR	N\A	N\A
h=3	Bag	N\A	N\A	N\A	N\A	FAAR	N\A	N\A	FAAR	N\A	N\A
h=12	CADL	N\A	N\A	N\A	N\A	FAAR	N\A	N\A	FAAR	N\A	N\A
SP2	N\A	N\A	N\A	N\A	N\A	N\A	N\A	N\A	N\A	N\A	N\A
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	EN	Mean	EN	Ridge	NNG	Boost	BMA2	EN	BMA1	LAR	Ridge
h=3	AR	Boost	NNG	AR	NNG	LAR	BMA2	LAR	AR	BMA1	Bag
h=12	AR	LAR, EN	1,5,10,11	Boost	Ridge	BMA2	BMA2	BMA1	LAR	BMA1	NNG
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	N\A	Bag	Ridge	AR, NNG	BMA2	N\A	ARX	Mean	N\A	Ridge	Ridge
h=3	Mean	Bag	N\A	N\A	Bag	N\A	BMA2	N\A	Boost	N\A	Ridge
h=12	N\A	Ridge	BMA1	AR	BMA2	Ridge	BMA1	BMA1, Ridge	BMA1	Bag	Ridge

Notes: The null of GL outperformance test is when e_1, a sequence of forecast errors from benchmark, 1, GL outperforms e_k, a sequence of forecast errors from alternatives, for $k = 2, \ldots l$, while rejecting the null means that e_1 does not GL outperform, e_k.

† N\A refers to the situation where the p-values of the null are all close to zero when one of 14 forecasting methods is set as a benchmark model, 1, against the rest of 13 alternatives each time, so that one cannot pick up a forecasting method that can claim GL outperformance over the alternatives within a fixed h and Y_t.

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	B	H	H	F	B	H	H	Mean	F	H	H
h=3	H	H	H	F	H	H	H	F	H	F	H
h=12	B	F	H	F	B	H	H	F	Mean	H	Mean
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	H	N\A	N\A	N\A	N\A	F	F	N\A	F	N\A	N\A
h=3	H	N\A	N\A	N\A	N\A	F	N\A	N\A	F	N\A	N\A
h=12	B	N\A	N\A	N\A	N\A	F	N\A	N\A	F	N\A	N\A
SP2											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	H	Mean	H	H	H	H	H	H	H	H	H
h=3	B	H	H	B	H	H	H	H	B	H	H
h=12	B	H	B, H	H	H	H	H	H	H	H	H
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	N\A	S	S	B, S	S	N\A	B	Mean	N\A	S	S
h=3	Mean	S	N\A	N\A	S	N\A	S	N\A	S	N\A	S
h=12	N\A	S	S	B	S	S	S	S	S	S	S

Notes: Notations as specified in Table 4.

Table 2.A.6: Best SOSD models

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	BMA2	AR	Mean	N\A	ARX	1,7	Boost	ARX	N\A	EN	NNG
h=3	Ridge	Bag	Mean	N\A	BMA1	AR	PCR	AR	N\A	Bag	Bag
h=12	EN	Bag	1,2,4,5,7,8,9,10,12,14	N\A	NNG	Bag	Boost	NNG	N\A	LAR	N\A
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	Ridge	2,4,8,9,11	N\A	FAAR	FAAR	BMA2	Ridge	ARX	BMA2	ARX	N\A
h=3	EN	N\A	N\A	FAAR	AR	PCR	PCR	BMA1	N\A	PCR	BMA1
h=12	LAR	N\A	N\A	FAAR	FAAR	FAAR	LAR	ARX	BMA1	Ridge	N\A
SP2											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	AR	N\A	LAR	N\A	Mean	ARX	AR	BMA2	N\A	Mean	N\A
h=3	Ridge	Mean	Bag	N\A	Mean	N\A	Mean	Bag	N\A	Boost	N\A
h=12	N\A	Mean	Bag	N\A	Bag	AR	Boost	EN	N\A	Mean	N\A
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	LAR, Mean	BMA2	Mean	Mean	Ridge	Mean	CADL	LAR	N\A	Mean	Mean
h=3	Mean	Mean	Mean	BMA2	N\A	Mean	LAR	BMA2	BMA1	N\A	Boost
h=12	BMA1	Mean	Mean	N\A	CADL	Boost	LAR	3,14	Mean	CADL	Mean

Notes: The null of CL outperformance test is when e_1, a sequence of forecast errors from benchmark, 1, CL outperforms e_k, a sequence of forecast errors from alternatives, for $k = 2, \dots l$, while rejecting the null means that e_1 does not CL outperform, e_k.

† N\A refers to the situation where the p-values of the null are all close to zero when one of 14 forecasting methods is set as a benchmark model, 1, against the rest of 13 alternatives each time, so that one cannot pick up a forecasting method that can claim GL outperformance over the alternatives within a fixed h and Y_t.

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	H	B	M	N\A	B	B,H	H	B	N\A	H	H
h=3	H	H	M	N\A	H	B	N\A	B	N\A	H	H
h=12	H	H	B, H, M	N\A	H	H	H	H	N\A	H	N\A
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	H	B,F,H	N\A	F	F	H	H	B	H	B	N\A
h=3	H	N\A	N\A	F	B	F	F	H	N\A	F	H
h=12	H	N\A	N\A	F	F	H	H	B	H	H	N\A
SP2											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	B	N\A	H	N\A	M	B	B	H	N\A	M	N\A
h=3	H	M	H	N\A	M	N\A	M	H	N\A	H	N\A
h=12	N\A	M	H	N\A	H	B	H	H	N\A	M	N\A
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	S, M	S	M	M	S	M	B	S	N\A	M	M
h=3	M	M	M	S	N\A	M	S	S	S	N\A	S
h=12	S	M	M	N\A	S	S	S	B, M	M	B	M

Notations as specified in Table 4.

Table 2.A.7: Best White's Reality Check models ($g = MSFE$)

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	AR	AR	LAR	ARX	Mean	EN	NNG	ARX	NNG	Bag	ARX
h=3	NNG	ARX	CADL	Mean	ARX	EN	ARX	ARX	FAAR	Boost	EN
h=12	Mean	ARX	BMA1	Mean	CADL	Bag	ARX	CADL	Bag	Ridge	Bag
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	LAR	FAAR	PCR	Boost	PCR,EN	Bag	FAAR	AR	EN	BMA2	AR,EN
h=3	Ridge	EN	ARX	EN	Ridge	AR	AR	AR	BMA2	LAR	ARX
h=12	PCR	EN	BMA1	Bag	Bag	EN	FAAR	EN	LAR	BMA1	EN
SP2											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	AR	Mean	ARX	Mean	Boost	Ridge	BMA2	Mean	Mean	ARX	Bag
h=3	NNG	AR	ARX	Mean	ARX	BMA2	AR	Ridge	Mean	Mean	AR
h=12	CADL,Bag	BMA1	CADL	Mean	Bag	AR	AR	Bag	AR	Mean	Ridge
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	CADL	ARX	AR	ARX,BMA1	Boost,Mean	CADL	AR	ARX	AR	3,12,13	ARX
h=3	EN	Boost	ARX,NNG	BMA2	ARX	CADL	Boost	CADL	Mean	AR	ARX
h=12	Boost	ARX	EN	Mean	ARX	CADL	Boost	NNG	Boost	Boost	ARX

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	B	B	H	B	M	H	H	B	H	H	B
h=3	H	B	B	M	B	H	B	B	F	H	H
h=12	M	B	H	M	B	H	B	B	H	H	H
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	H	F	F	H	F,H	H	F	B	H	H	B,H
h=3	H	H	B	H	H	B	B	B	H	H	B
h=12	F	H	H	H	H	H	F	H	H	H	H
SP2											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	B	M	B	M	H	H	H	M	M	B	H
h=3	H	B	B	M	B	H	B	H	M	M	B
h=12	B, H	H	B	M	H	B	B	H	B	M	H
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	B	B	B	B,S	S,M	B	B	B	B	B, S	B
h=3	S	S	B, S	S	B	B	S	B	M	B	B
h=12	S	B	S	M	B	B	S	S	S	S	B

Notations as specified in Table 4.

Table 2.A.8: Best White's Reality Check models ($g = MAFE$)

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	AR	Mean	FAAR,LAR	EN	FAAR	EN	EN	ARX,BMA2	FAAR	Bag	ARX,Mean
h=3	EN	Mean	LAR	FAAR	Ridge	EN	ARX	Mean	FAAR	FAAR	ARX
h=12	Mean	EN	Mean	CADL	Bag	Bag	Mean	Bag	PCR	BMA2	BMA2
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	LAR	FAAR	PCR	Boost	PCR,EN	Bag	FAAR	AR	EN	BMA2	AR,EN
h=3	Ridge	EN	ARX	EN	Ridge	AR	AR	AR	BMA2	BMA1	ARX
h=12	PCR	EN	Bag	Bag	Bag	EN	FAAR	EN	LAR	BMA1	EN
SP2											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	AR	Mean	ARX,CADL	Bag	Mean	AR	Boost	LAR	EN	ARX	Mean
h=3	LAR	Boost	ARX	Mean	Mean	EN	AR,EN	Mean	Mean	LAR	Mean
h=12	CADL	Mean	CADL	Mean	Mean	Mean	Mean	Mean	Mean	EN	AR,Boost
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	CADL	Mean	Boost	Mean	EN	CADL	CADL	Boost	Mean	AR	BMA2
h=3	EN	Boost	CADL	BMA2	Mean	CADL	AR	CADL	BMA1	ARX	LAR
h=12	CADL	ARX	CADL	Mean	Boost	CADL	LAR	EN	BMA1	ARX	AR

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	B	M	F,H	H	F	H	H	B,H	F	H	B,M
h=3	H	M	H	F	H	H	B	M	F	F	B
h=12	M	H	M	B	H	H	M	H	F	H	H
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	H	F	F	H	F,H	H	F	B	H	H	B, H
h=3	H	H	B	H	H	B	B	B	H	H	B
h=12	F	H	H	H	H	H	F	H	H	H	H
SP2											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	B	M	B	H	M	B	H	H	H	B	M
h=3	H	H	B	M	M	H	B, H	M	M	H	M
h=12	B	M	B	M	M	M	M	M	M	H	B, H
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	B	M	S	M	S	B	B	S	M	B	S
h=3	S	S	B	S	M	B	B	B	S	B	S
h=12	B	B	B	M	S	B	S	S	S	B	B

Notations as specified in Table 4.

Table 2.A.9: Best MSFE models

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	FAAR	PCR	Mean	PCR	PCR	CADL	Mean	Ridge	FAAR	AR	ARX
h=3	PCR	ARX	Bag	Mean	PCR	CADL	Bag	Mean	Mean	Bag	NNG
h=12	PCR	AR	PCR	Mean	Mean	CADL	Bag	CADL	ARX	Mean	Mean
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	LAR	Mean	CADL	ARX	Mean	CADL	CADL	ARX	ARX	AR	Mean
h=3	CADL	ARX	Mean	Mean	Mean	CADL	CADL	CADL	Mean	AR	AR
h=12	Boost	Mean	CADL	Mean	Mean	CADL	ARX	CADL	ARX	AR	Mean
SP2											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	LAR	EN	CADL	ARX	Bag	Mean	CADL	ARX	Mean	AR	ARX
h=3	BMA2	Mean	Mean	NNG	EN	CADL	CADL	Mean	AR	AR	ARX
h=12	Mean	Mean	CADL	ARX	AR	CADL	Mean	LAR	Mean	BMA1	Mean
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	CADL	ARX	Boost	Mean	Mean	CADL	CADL	Boost	LAR	LAR	Boost
h=3	CADL	Boost	CADL	ARX	ARX	CADL	Boost	EN	Mean	EN	EN
h=12	Boost	Boost	CADL	Mean	EN	CADL	Mean	CADL	ARX	EN	NNG

SP1											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	F	F	M	F	F	B	M	H	F	B	B
h=3	F	B	H	M	F	B	H	M	M	H	H
h=12	F	B	F	M	M	B	H	B	B	M	M
SP1L											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	H	M	B	B	M	B	B	B	B	B	M
h=3	B	B	M	M	M	B	B	B	M	B	B
h=12	H	M	B	M	M	B	B	B	B	B	M
SP2											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	H	H	B	B	H	M	B	B	M	B	B
h=3	H	M	M	H	H	B	B	M	B	B	B
h=12	M	M	B	B	B	B	M	H	M	H	M
SP3											
Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
h=1	B	B	S	M	M	B	B	S	S	S	S
h=3	B	S	B	B	B	B	S	S	M	S	S
h=12	S	S	B	M	S	B	M	B	B	S	S

Notations as specified in Table 4.

Table 2.A.10: SP1 Results

Benchmark	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
FOSD, Recursive, h=1											
AR	**0.9458**	0.3883	0.9725	0.0008	**0.6025**	0.0292	0.0767	0.1642	0.0008	0.9550	0.2067
ARX(SIC)	0.9108	0.3067	0.0892	0.0000	0.0467	0.4300	0.1408	0.0483	0.0017	0.9817	0.8033
CADL	0.6542	0.0000	0.6717	0.0000	0.0000	0.0000	0.0192	0.0000	0.0000	0.0142	0.0000
FAAR	0.9008	0.4883	0.9675	**0.0133**	0.4508	0.0317	0.0083	0.0400	**0.0075**	0.8633	0.8083
PCR	0.3675	0.3658	0.4158	0.0067	0.1158	0.9508	0.0200	0.0692	0.0008	0.9592	0.5817
Bagging	0.2867	0.3308	0.3750	0.0067	0.4542	**0.9558**	0.0275	0.0675	0.0000	0.9650	0.4367
Boosting	0.2283	0.3417	**0.9742**	0.0008	0.5875	0.2717	0.1258	0.3300	0.0000	0.9617	0.4942
BMA1	0.0075	0.5900	0.9150	0.0000	0.0167	0.3500	**0.2633**	0.0025	0.0017	0.9767	0.7967
BMA2	0.2400	**0.6592**	0.9083	0.0000	0.0867	0.5875	0.1508	0.2325	0.0000	**0.9850**	**0.8575**
Rridge	0.5333	0.5408	0.9092	0.0000	0.1383	0.3658	0.1417	0.2858	0.0008	**0.9850**	0.8017
LAR	0.1075	0.4650	0.3025	0.0000	0.0150	0.8650	0.0542	0.0075	0.0008	0.9825	0.7083
EN	0.7825	0.1767	0.3917	0.0117	0.2942	0.3308	0.0158	0.1433	0.0000	0.9150	0.3225
NNG	0.7358	0.1892	0.9192	0.0050	0.1542	0.2608	0.0075	0.1758	0.0000	0.9767	0.3067
Mean	0.6200	0.3117	0.5367	0.0000	0.0675	0.3250	0.0383	**0.3558**	0.0008	0.9833	0.3158
SOSD, Recursive, h=1											
AR	0.2600	**0.0008**	0.2042	0.0000	0.0067	**0.0017**	0.8667	0.0325	0.0000	0.2192	0.0000
ARX(SIC)	0.1617	0.0000	0.4575	0.0000	**0.2308**	0.0008	0.7158	**0.0467**	0.0000	0.0633	0.0000
CADL	0.2150	0.0000	0.4675	0.0000	0.0042	0.0000	0.1650	0.0100	0.0000	0.1425	0.0000
FAAR	0.3967	0.0000	0.4217	0.0000	0.0000	0.0000	0.0242	0.0000	0.0000	0.1825	0.0000
PCR	0.8400	0.0000	0.6133	0.0000	0.0242	0.0000	0.9192	0.0008	0.0000	0.2208	0.0000
Bagging	0.8158	0.0000	0.5975	0.0000	0.0067	0.0008	0.9217	0.0042	0.0000	0.2142	0.0000
Boosting	0.9067	0.0008	0.2233	0.0000	0.0150	**0.0017**	**0.9500**	0.0067	0.0000	0.2275	0.0000
BMA1	0.2450	0.0000	0.3817	0.0000	0.0958	0.0000	0.6283	0.0000	0.0000	0.2075	0.0000
BMA2	**0.9117**	0.0000	0.4292	0.0000	0.0625	0.0000	0.8808	0.0025	0.0000	0.1683	0.0000
Rridge	0.8258	0.0000	0.5183	0.0000	0.0542	0.0000	0.9292	0.0050	0.0000	0.1983	0.0000
LAR	0.6333	0.0000	0.6583	0.0000	0.0608	0.0000	0.9125	0.0000	0.0000	0.1725	0.0000
EN	0.1225	0.0000	0.7233	0.0000	0.0200	0.0008	0.7642	0.0117	0.0000	**0.2925**	0.0000
NNG	0.1525	0.0000	0.1833	0.0000	0.0300	0.0000	0.7325	0.0083	0.0000	0.1225	**0.0017**
Mean	0.7717	0.0000	**0.7950**	0.0000	0.0467	0.0000	0.8717	0.0075	0.0000	0.2058	0.0000
Reality Check, $g = MSFE$, Recursive, h=1											
AR	**0.9667**	**0.9667**	0.3833	0.4733	0.7867	0.8867	0.8433	0.9300	0.8967	0.7333	0.4433
ARX(SIC)	0.8500	0.9133	0.8767	**0.9867**	0.9333	0.5267	0.9567	**0.9800**	0.6400	0.6900	**0.9600**
CADL	0.5900	0.8400	0.9700	0.6033	0.9367	0.0933	0.1433	0.7533	0.1133	0.8767	0.6633
FAAR	0.8267	0.7433	0.9567	0.9700	0.9300	0.9300	0.9700	0.7500	0.9433	0.9333	0.6533
PCR	0.4067	0.8933	0.7567	0.4933	0.7900	0.0133	0.0167	0.4600	0.2667	0.7800	0.9000
Bagging	0.8333	0.7133	0.9800	0.9633	0.9167	0.9400	0.9867	0.7700	0.8833	**0.9567**	0.7100
Boosting	0.9233	0.9600	0.3567	0.5033	0.8000	0.8833	0.8400	0.8967	0.9133	0.6933	0.3933
BMA1	0.6700	0.9433	0.3567	0.4767	0.7967	0.6533	0.8000	0.7900	0.9267	0.7167	0.3367
BMA2	0.9033	0.9567	0.3967	0.5200	0.7533	0.8133	0.8200	0.8733	0.8567	0.7633	0.4333
Rridge	0.8900	0.9367	0.3633	0.5300	0.8200	0.8533	0.8167	0.8967	0.8600	0.7433	0.4167
LAR	0.7867	0.7333	**0.9833**	0.9633	0.9267	0.9000	0.9800	0.7000	0.8967	0.9133	0.6200
EN	0.8900	0.8000	0.9767	0.9400	0.9067	**0.9867**	0.9833	0.7933	0.9133	0.9267	0.6767
NNG	0.8900	0.7800	0.1733	0.9500	0.9300	0.9833	**0.9900**	0.7567	0.9300	0.9533	0.6400
Mean	0.9433	0.9267	0.9533	0.9267	**0.9600**	0.8667	0.8133	0.9133	**0.9867**	0.8900	0.8433
Reality Check, $g = MAFE$, Recursive, h=1											
AR	**0.5267**	0.5133	0.3967	0.1900	0.4067	0.3300	0.1233	0.5133	0.1933	0.2533	0.2600
ARX(SIC)	0.4700	0.5400	0.4733	0.4833	0.4067	0.0067	0.3733	**0.5400**	0.4033	0.4433	**0.5167**
CADL	0.4500	0.2933	0.4933	0.0100	0.4400	0.0000	0.0000	0.3167	0.0000	0.4700	0.2367
FAAR	0.5100	0.5267	**0.5133**	0.5167	**0.5067**	0.2667	0.4033	0.4400	**0.4767**	0.4467	0.4700
PCR	0.2867	0.3667	0.4767	0.1067	0.3800	0.0067	0.0000	0.2600	0.0033	0.4067	0.5000
Bagging	0.4533	0.4967	0.5067	0.4900	0.5000	0.2733	0.4200	0.5333	0.4567	**0.5267**	0.4833
Boosting	0.4733	0.5400	0.4200	0.2100	0.3500	0.3067	0.1400	0.5133	0.1767	0.3067	0.2700
BMA1	0.4200	0.4800	0.4067	0.1933	0.2833	0.0867	0.0833	0.4433	0.2000	0.2800	0.1400
BMA2	0.4733	0.5267	0.4200	0.2300	0.3700	0.2700	0.1333	**0.5400**	0.1800	0.2700	0.2200
Rridge	0.4900	0.5367	0.3967	0.2467	0.3300	0.2733	0.1433	0.5167	0.1900	0.2967	0.2167
LAR	0.4233	0.4700	**0.5133**	0.4667	0.4767	0.2200	0.4200	0.4633	0.4633	0.4867	0.4167
EN	0.5233	0.5067	0.4767	**0.5900**	0.4567	**0.5200**	**0.5100**	0.5367	0.4667	0.4333	0.4800
NNG	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Mean	0.4967	**0.5533**	0.4967	0.4967	0.4767	0.3167	0.1900	0.5167	0.4500	0.4267	**0.5167**

Notes: Best models, within a fixed h, Y_t and estimation window scheme, are marked in bold. Entries are p-values of forecasting superiority test. The null of GL\CL outperformance test is when e_1, a sequence of forecast errors from benchmark, 1, GL/CL outperforms e_k, a sequence of forecast errors from alternatives, for $k = 2, \dots l$, while rejecting the null means that e_1 does not GL/CL outperform, e_k. † White's Reality Check test compares $g(e_1)$ against $g(e_k)$ for $k = 2, \dots l$ and here two different loss functions, $MSFE$ and $MAFE$, are set as g. The interpretations of the results can be done in an analogous way, if one fails to reject the null, that means, given a loss function, a benchmark outperforms all alternatives, but if not, it refers to the situation where there is at least one model that outperforms the benchmark.

Table 2.A.11: SP1L Results

Benchmark	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
FOSD, Recursive, h=1											
AR	0.7750	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
ARX(SIC)	0.0175	0.0000	0.0000	0.0000	0.0000	0.0242	0.0000	0.0000	0.0000	0.0000	0.0000
CADL	0.4600	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
FAAR	0.3750	0.0000	0.0000	0.0000	0.0000	**0.0625**	**0.0008**	0.0000	**0.3383**	0.0000	0.0000
PCR	0.0900	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Bagging	0.6967	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Boosting	0.2133	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
BMA1	0.0042	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
BMA2	0.0017	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0008	0.0000	0.0000
Rridge	0.1458	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0008	0.0000	0.0000
LAR	0.2758	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
EN	**0.7850**	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
NNG	0.6358	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Mean	0.5475	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
SOSD, Recursive, h=1											
AR	0.1650	0.0000	0.0000	0.1417	0.1408	0.1150	0.0275	0.0175	0.2133	0.0017	0.0000
ARX(SIC)	0.0200	**0.0008**	0.0000	0.1700	0.1075	0.0167	**0.0867**	0.0250	0.1558	**0.0050**	0.0000
CADL	0.0208	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
FAAR	0.1750	**0.0008**	0.0000	**0.2642**	**0.1983**	0.0958	0.0175	0.0400	0.1433	0.0025	0.0000
PCR	0.5208	0.0000	0.0000	0.1500	0.1133	0.2517	0.0467	0.0392	0.1967	0.0033	0.0000
Bagging	0.0358	0.0000	0.0000	0.1425	0.1550	0.0950	0.0367	0.0033	**0.2292**	0.0008	0.0000
Boosting	0.4942	0.0000	0.0000	0.1383	0.1375	0.1517	0.0408	0.0308	0.2117	0.0017	0.0000
BMA1	0.0808	**0.0008**	0.0000	0.1342	0.0958	0.4042	0.0858	0.0517	0.2142	0.0017	0.0000
BMA2	0.0858	**0.0008**	0.0000	0.1325	0.0942	**0.4192**	0.0700	**0.0567**	0.2267	0.0025	0.0000
Rridge	**0.5950**	0.0000	0.0000	0.1308	0.0983	0.1983	0.0508	0.0300	0.2200	0.0017	0.0000
LAR	0.5767	**0.0008**	0.0000	0.1367	0.1225	0.1217	0.0400	0.0242	0.2058	0.0025	0.0000
EN	0.1750	0.0000	0.0000	0.1342	0.1275	0.1067	0.0367	0.0167	0.2267	0.0025	0.0000
NNG	0.2592	0.0000	0.0000	0.1367	0.1208	0.1467	0.0433	0.0167	0.2033	0.0033	0.0000
Mean	0.1775	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Reality Check, $g = MSFE$, Recursive, h=1											
AR	0.5067	0.4800	0.4433	0.4667	0.5133	0.5067	0.4700	**0.5500**	0.5033	0.4467	**0.5167**
ARX(SIC)	0.0033	0.0000	0.0033	0.0333	0.0233	0.0000	0.0067	0.0033	0.1733	0.0033	0.0000
CADL	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
FAAR	0.2833	**0.5300**	0.4333	0.4400	0.4600	0.4133	**0.5267**	0.4300	0.4667	0.4600	0.4133
PCR	0.2933	0.3800	**0.5067**	0.2733	**0.5200**	0.0133	0.0000	0.2833	0.1233	0.4900	0.2600
Bagging	0.4867	0.4733	0.3233	0.4900	0.4933	**0.5333**	0.5167	0.5100	0.4333	0.4767	0.4267
Boosting	0.4267	0.5233	0.4867	**0.5000**	0.4867	0.4733	0.4833	0.4167	0.5133	0.4533	0.4900
BMA1	0.2733	0.3900	0.4900	0.4967	0.4433	0.2933	0.4700	0.2400	0.5000	0.4567	0.2533
BMA2	0.2800	0.4100	0.4800	0.4633	0.4567	0.3033	0.4700	0.2200	0.4867	**0.5100**	0.2700
Rridge	0.4567	0.4700	0.4900	0.4700	0.4767	0.4367	0.4567	0.4800	0.5067	0.4967	0.5067
LAR	**0.5167**	0.4800	0.4400	0.4833	0.5000	0.4867	0.5000	0.5100	0.5200	0.4500	0.4633
EN	0.4933	0.4867	0.4967	0.4500	**0.5200**	0.4600	0.4767	0.5100	**0.5233**	0.4467	**0.5167**
NNG	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Mean	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Reality Check, $g = MAFE$, Recursive, h=1											
AR	0.5067	0.4800	0.4433	0.4667	0.5133	0.5067	0.4700	**0.5500**	0.5033	0.4467	**0.5167**
ARX(SIC)	0.0033	0.0000	0.0033	0.0333	0.0233	0.0000	0.0067	0.0033	0.1733	0.0033	0.0000
CADL	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
FAAR	0.2833	**0.5300**	0.4333	0.4400	0.4600	0.4133	**0.5267**	0.4300	0.4667	0.4600	0.4133
PCR	0.2933	0.3800	**0.5067**	0.2733	**0.5200**	0.0133	0.0000	0.2833	0.1233	0.4900	0.2600
Bagging	0.4867	0.4733	0.3233	0.4900	0.4933	**0.5333**	0.5167	0.5100	0.4333	0.4767	0.4267
Boosting	0.4267	0.5233	0.4867	**0.5000**	0.4867	0.4733	0.4833	0.4167	0.5133	0.4533	0.4900
BMA1	0.2733	0.3900	0.4900	0.4967	0.4433	0.2933	0.4700	0.2400	0.5000	0.4567	0.2533
BMA2	0.2800	0.4100	0.4800	0.4633	0.4567	0.3033	0.4700	0.2200	0.4867	**0.5100**	0.2700
Rridge	0.4567	0.4700	0.4900	0.4700	0.4767	0.4367	0.4567	0.4800	0.5067	0.4967	0.5067
LAR	**0.5167**	0.4800	0.4400	0.4833	0.5000	0.4867	0.5000	0.5100	0.5200	0.4500	0.4633
EN	0.4933	0.4867	0.4967	0.4500	**0.5200**	0.4600	0.4767	0.5100	**0.5233**	0.4467	**0.5167**
NNG	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Mean	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

Refer to notes to Table 10.

Table 2.A.12: SP2 Results

Benchmark	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
FOSD, Recursive, h=1											
AR	0.9067	0.4333	0.2500	0.2517	0.2525	0.6133	0.3700	0.7467	0.3525	0.9533	0.6033
ARX(SIC)	0.6625	0.0000	0.0817	0.0000	0.0000	0.0000	0.0100	0.0000	0.0000	0.0042	0.0000
CADL	0.6358	0.0000	0.0900	0.0000	0.0000	0.0000	0.0042	0.0000	0.0000	0.0033	0.0000
Bagging	0.0200	0.4817	0.0000	0.1133	0.2858	0.1375	0.0817	0.5200	0.2358	0.0017	0.0642
Boosting	0.4100	0.1542	0.0025	0.1142	0.0342	**0.9967**	0.7742	0.0225	0.1792	0.7667	0.3775
BMA1	0.5933	0.3242	0.0000	0.0617	0.0258	0.2025	0.9750	0.0925	**0.4717**	0.9325	0.6175
BMA2	0.8692	0.3883	0.0008	0.1692	0.1533	0.2167	**0.9825**	0.1292	0.4233	0.9783	0.5208
Rridge	0.9075	0.0042	0.0533	**0.3742**	0.1550	0.6633	0.4050	0.6442	0.0292	0.7942	**0.8575**
LAR	0.5117	0.5525	0.0225	0.2775	0.0783	0.6000	0.9300	0.5550	0.2842	**0.9842**	0.7183
EN	**0.9625**	0.4725	**0.3483**	0.3458	0.2392	0.5850	0.3667	**0.7608**	0.2500	0.9583	0.6892
NNG	0.9500	0.4275	0.2767	0.2958	**0.3817**	0.6492	0.4242	0.7125	0.3025	0.8058	0.8192
Mean	0.7933	**0.5892**	0.0600	0.0042	0.0433	0.0100	0.4217	0.1983	0.0000	0.7575	0.1817
SOSD, Recursive, h=1											
AR	**1.0000**	0.0000	0.1792	0.0000	0.0000	0.0000	**0.7417**	0.0025	0.0000	0.1142	0.0000
ARX(SIC)	0.6992	0.0000	0.2283	0.0000	0.0025	**0.0008**	0.1525	0.0242	0.0000	0.1150	0.0000
CADL	0.7008	0.0000	0.2375	0.0000	0.0000	0.0000	0.1742	0.0308	0.0000	0.1067	0.0000
Bagging	0.4292	0.0000	0.0000	0.0000	0.0042	0.0000	0.5183	0.0050	0.0000	0.0642	0.0000
Boosting	0.9567	0.0000	0.2275	0.0000	0.0000	0.0000	0.5292	0.0000	0.0000	0.1483	0.0000
BMA1	0.9992	0.0000	0.1400	0.0000	0.0100	0.0000	0.2792	0.0825	0.0000	0.0250	0.0000
BMA2	0.9983	0.0000	0.1267	0.0000	0.0075	0.0000	0.3242	**0.0925**	0.0000	0.0642	0.0000
Rridge	0.9908	0.0000	0.2417	0.0000	0.0000	0.0000	0.7350	0.0067	0.0000	0.0333	0.0000
LAR	0.9983	0.0000	**0.4083**	0.0000	0.0000	0.0000	0.4867	0.0292	0.0000	0.0742	0.0000
EN	0.9983	0.0000	0.1642	0.0000	0.0000	0.0000	0.7217	0.0050	0.0000	0.1042	0.0000
NNG	0.9917	0.0000	0.1683	0.0000	0.0008	0.0000	0.7250	0.0058	0.0000	0.1833	0.0000
Mean	0.9992	0.0000	0.3650	0.0000	**0.0200**	0.0000	0.7333	0.0392	0.0000	**0.2183**	0.0000
Reality Check, $g = MSFE$, Recursive, h=1											
AR	**0.9667**	0.8900	0.6367	0.7900	0.7233	0.9500	0.9333	0.9133	0.9000	0.9033	0.7000
ARX(SIC)	0.5367	0.7067	**0.9900**	0.2233	0.8733	0.0000	0.0367	0.5567	0.0000	**0.9667**	0.7967
CADL	0.5533	0.7033	0.9533	0.2133	0.8767	0.0000	0.0333	0.5367	0.0000	0.9500	**0.8333**
Bagging	0.5100	0.6300	0.5867	0.2833	0.8767	0.0000	0.0000	0.5367	0.0000	0.6067	0.7567
Boosting	0.5500	0.6833	0.9433	0.2733	**0.8933**	0.0000	0.0000	0.5267	0.0000	0.9167	0.7033
BMA1	0.0000	0.7867	0.0000	0.8000	0.8033	0.2367	0.7133	0.6367	0.3933	0.0067	0.4800
BMA2	0.1533	0.8533	0.5767	0.7633	0.7467	0.9400	**0.9767**	0.3567	0.9267	0.9067	0.2333
Rridge	0.8400	0.7900	0.6767	0.7733	0.8167	**0.9600**	0.8967	0.4300	0.8333	0.8100	0.5300
LAR	0.8933	0.8267	0.5067	0.7867	0.8233	0.9133	0.8733	0.5833	0.8700	0.8800	0.5967
EN	0.9000	0.6433	0.6000	0.7867	0.6533	0.5867	0.9733	0.7667	0.5167	0.7633	0.6433
NNG	0.9300	0.9067	0.7967	0.7467	0.7767	0.9467	0.9333	0.9233	0.9200	0.9233	0.6133
Mean	0.9533	**0.9133**	0.6800	**0.8467**	0.8233	0.8900	0.8633	**0.9467**	**0.9300**	0.9000	0.7667
Reality Check, $g = MAFE$, Recursive, h=1											
AR	**0.5000**	0.5333	0.1200	0.3500	0.4233	**0.5133**	0.4000	0.4833	0.4900	0.4233	0.5567
ARX(SIC)	0.3467	0.1233	0.4833	0.0000	0.3900	0.0000	0.0000	0.0800	0.0000	**0.5033**	0.3300
CADL	0.3233	0.0933	**0.4833**	0.0000	0.4100	0.0000	0.0000	0.0767	0.0000	0.4733	0.3300
Bagging	0.0033	0.4167	0.0000	**0.5167**	0.4267	0.0033	0.0833	0.1400	0.0900	0.0067	0.2900
Boosting	0.0100	0.5500	0.1267	0.3367	0.3233	0.1900	**0.5467**	0.0367	0.4867	0.4733	0.1467
BMA1	0.3867	0.4400	0.1433	0.3867	0.5200	0.5000	0.1800	0.0300	0.3967	0.2367	0.3500
BMA2	0.2700	0.4767	0.0600	0.4100	0.4933	0.2433	0.2100	0.0533	0.3333	0.2833	0.3633
Rridge	0.3900	0.1533	0.0867	0.4167	0.3233	0.0567	0.4200	0.2033	0.1133	0.2700	0.4633
LAR	0.4833	0.5000	0.2367	0.3900	0.4033	0.3700	0.3067	**0.5100**	0.4833	0.4133	0.3733
EN	0.4300	0.5000	0.0900	0.3533	0.4300	0.3733	0.4000	0.4767	**0.4933**	0.4000	0.5333
NNG	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Mean	0.4767	**1.0000**	0.3500	0.4833	**1.0000**	0.0433	0.0033	0.3600	0.4600	0.4067	**1.0000**

Refer to notes to Table 10.

Table 2.A.13: SP3 Results

Benchmark	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
FOSD, Recursive, h=1											
AR	0.0000	0.0000	0.0125	**1.0000**	0.8217	0.0000	0.0117	0.0017	0.0000	0.0000	0.0008
ARX(SIC)	0.0000	0.0008	0.0050	0.6158	0.2417	0.0000	**0.6067**	0.0000	0.0000	0.0000	0.0000
CADL	0.0000	0.0000	0.0108	0.2117	0.2483	0.0000	0.0108	0.0000	0.0000	0.0000	0.0000
Bagging	0.0000	**0.0133**	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0025
Boosting	0.0000	0.0000	0.0042	0.9883	0.7992	0.0000	0.0142	0.0000	0.0000	0.0000	0.0000
BMA1	0.0000	0.0000	0.0133	0.9033	0.9892	0.0000	0.3408	0.0000	0.0000	0.0000	0.0000
BMA2	0.0000	0.0000	0.0133	0.9733	**0.9925**	0.0000	0.5608	0.0000	0.0000	0.0000	0.0000
Rridge	0.0000	0.0000	**0.9542**	0.0000	0.0025	0.0000	0.0867	0.0000	0.0000	**0.0717**	**0.1733**
LAR	0.0000	0.0008	0.0108	0.9992	0.7758	0.0000	0.0200	0.0042	0.0000	0.0000	0.0000
EN	0.0000	0.0008	0.0083	0.9992	0.8208	0.0000	0.0117	0.0025	0.0000	0.0000	0.0000
NNG	0.0000	0.0000	0.0125	**1.0000**	0.8367	0.0000	0.0150	0.0025	0.0000	0.0000	0.0000
Mean	0.0000	0.0000	0.0142	0.3100	0.2733	0.0000	0.0142	**0.0200**	0.0000	0.0000	0.0000
SOSD, Recursive, h=1											
AR	0.0225	0.0533	0.0008	0.0017	0.0000	0.0025	0.2425	0.2925	0.0000	0.0142	0.2483
ARX(SIC)	0.0258	0.0358	0.0017	0.0075	0.0050	0.0025	0.0500	0.1933	0.0000	0.0092	0.1825
CADL	0.0225	0.0125	0.0017	0.0150	0.0042	0.0017	**0.3025**	0.0392	0.0000	0.0275	0.0925
Bagging	0.0000	0.0775	0.0000	0.0042	0.0008	0.0000	0.0450	0.0017	0.0000	0.0008	0.0042
Boosting	0.0175	0.0675	0.0017	0.0033	0.0000	0.0017	0.1700	0.1417	0.0000	0.0175	0.1942
BMA1	0.0200	0.1500	0.0000	0.0092	0.0000	0.0000	0.0583	0.0467	0.0000	0.0175	0.3608
BMA2	0.0267	**0.1725**	0.0008	0.0033	0.0000	0.0000	0.0392	0.1150	0.0000	0.0183	0.4625
Rridge	0.0000	0.0000	0.0000	0.0125	**0.0092**	0.0000	0.2525	0.0000	0.0000	0.0000	0.0250
LAR	**0.0275**	0.0458	0.0017	0.0000	0.0000	0.0017	0.2258	**0.3058**	0.0000	0.0117	0.2033
EN	0.0267	0.0417	0.0017	0.0017	0.0000	0.0017	0.2583	0.2742	0.0000	0.0158	0.2450
NNG	0.0258	0.0475	0.0017	0.0008	0.0000	0.0017	0.2458	0.2900	0.0000	0.0158	0.2442
Mean	0.0225	0.1033	**0.0042**	**0.0158**	0.0067	**0.0075**	0.2875	0.2642	0.0000	**0.0283**	**0.4833**
Reality Check, $g = MSFE$, Recursive, h=1											
AR	0.9600	0.8200	**0.9800**	0.8933	0.8767	0.9533	**0.9800**	0.8833	**0.9733**	0.9633	0.8233
ARX(SIC)	0.9567	**0.9600**	0.8533	**0.9867**	0.9200	0.7900	0.8933	**0.9767**	0.7500	0.8833	**0.9700**
CADL	**0.9800**	0.7333	0.9700	0.8933	0.8800	**0.9833**	0.9700	0.9300	0.9100	**0.9767**	0.6533
Bagging	0.0000	0.8400	0.0000	0.0767	0.0267	0.0000	0.0067	0.0967	0.0000	0.0000	0.0000
Boosting	0.9167	0.8733	0.9533	0.9533	**0.9533**	0.8667	0.9567	0.9567	0.8767	0.9667	0.9567
BMA1	0.9000	0.8200	0.8767	**0.9867**	0.8567	0.6933	0.6233	0.9533	0.8067	0.8967	0.9600
BMA2	0.9700	0.8833	0.9267	0.9267	0.8900	0.7933	0.7100	0.9533	0.9233	0.9000	0.9667
Rridge	0.0067	0.0100	0.5067	0.3633	0.0033	0.2733	0.4000	0.0033	0.0000	0.0567	0.0067
LAR	0.9600	0.8433	0.9700	0.9333	0.9033	0.9700	0.9700	0.9000	0.9600	0.9667	0.8500
EN	0.9567	0.8533	0.9600	0.9167	0.9133	0.9467	0.9667	0.9100	0.9633	**0.9767**	0.8267
NNG	0.9467	0.8667	0.9533	0.9267	0.9100	0.9500	0.9700	0.9067	0.9533	**0.9767**	0.8167
Mean	0.9233	0.9067	0.6167	0.9733	**0.9533**	0.7267	0.9167	0.9533	0.9333	0.9367	0.8967
Reality Check, $g = MAFE$, Recursive, h=1											
AR	0.4200	0.4433	0.4767	0.3867	0.5200	0.3100	0.4800	0.2633	0.4567	**0.5567**	0.3033
ARX(SIC)	0.4800	0.4867	0.3333	0.4467	0.4133	0.0033	0.3300	0.5000	0.3967	0.4533	0.4067
CADL	**0.5067**	0.3333	0.4567	0.4000	0.4367	**0.5233**	**0.5267**	0.2767	0.4433	0.4400	0.1500
Bagging	0.0000	0.2333	0.0000	0.0067	0.0000	0.0000	0.0000	0.0367	0.0000	0.0000	0.0000
Boosting	0.4333	0.4533	**0.4967**	0.4333	0.5133	0.0500	0.5233	**0.5333**	0.4533	0.4600	0.4733
BMA1	0.2067	0.3967	0.2533	0.4367	0.2833	0.0000	0.0500	0.4033	0.3967	0.3433	0.4733
BMA2	0.4867	0.4600	0.3667	0.3900	0.4667	0.0000	0.0533	0.4867	0.4333	0.4000	**0.5267**
Rridge	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
LAR	0.4700	0.4867	0.4800	0.4267	0.4933	0.3433	0.4500	0.2633	0.4500	0.5000	0.2967
EN	0.3933	0.4567	0.4900	0.4133	**0.5400**	0.2933	0.4267	0.2667	0.4667	0.5133	0.2767
NNG	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
Mean	0.3867	**1.0000**	0.0967	**1.0000**	0.4533	0.0000	0.5000	0.3333	**1.0000**	0.3633	0.3300

Refer to notes to Table 10.

Table 2.A.14: MSFE results

Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
SP1, Recursive, h=1											
AR	6.257	0.007	18.650	0.002	0.010	0.000	1.316	0.010	0.003	**0.400**	0.006
ARX(SIC)	1.011	0.941	1.136	0.966	0.992	1.347**	1.088	0.913	1.181	1.135	**0.836**
CADL	0.976**	1.031	0.991	1.026	1.102	**0.920**	0.988	0.975	1.060	1.009	1.075
FAAR	**0.889**	0.907	1.029	0.897	0.941	1.014	1.054	0.865*	**0.953**	1.046	0.964
PCR	0.935	**0.856**	1.046	**0.858**	**0.906**	1.276**	2.084**	0.864	1.472**	1.046	0.875
Bagging	0.917	1.039	0.962	1.129	1.039	1.373**	1.147	1.126	0.963	1.010	1.001
Boosting	0.972	0.994	0.929	0.961	0.990	0.965	1.003	0.882**	0.997	1.008	1.032
BMA1	0.974	0.984	0.941	0.961	0.994	0.974	1.019	0.875**	1.014	1.027	1.047
BMA2	0.980	0.990	0.952	0,955	0.991	1.002	1.012	0.870**	1.009	1.028	1.040
Rridge	0.984	0.990	0.959	0.969	0.988	1.052	1.000	**0.864****	1.030	1.035	1.013
LAR	0.963*	0.984	0.957**	0.991	0.991	0.970**	1.003	0.950**	0.987	1.005	1.021
EN	0.963*	0.984	0.957**	0.991	0.991	0.970**	1.003	0.950**	0.987	1.005	1.021
NNG	0.987*	0.992	0.995	0.991	0.993	0.978**	0.999	0.986**	0.995	1.003	1.005
Mean	0.924**	0.952*	**0.924**	0.927	0.953	0.939	**0.974**	0.884**	0.969	1.006	0.939**
SP1L, Recursive, h=1											
AR	6.257	0.007	18.650	0.002	0.010	0.000	1.316	0.010	**0.003**	**0.400**	0.006
ARX(SIC)	1.011	0.941	1.136	**0.966**	0.992	1.347**	1.088	**0.913**	1.181	1.135	0.836
CADL	0.976**	1.031	**0.991**	1.026	1.018	**0.920**	**0.988**	0.975	1.060	1.009	1.075
FAAR	1.069	0.882	1.851**	1.392	1.121	1.985**	3.631**	1.209	1.454	1.390**	0.799
PCR	1.013	0.881	1.676**	1.359	1.120	1.837**	4.869**	1.163	1.700**	1.371	0.889
Bagging	1.310**	0.984	1.762**	1.477**	1.120	3.494**	3.905**	1.225	1.229	1.131	0.804
Boosting	1.038	0.897	1.456**	1.310	1.101	1.758**	3.714**	1.104	1.419	1.224**	0.795
BMA1	1.042	0.890	1.544**	1.328	1.090	1.758**	3.711**	1.124	1.446	1.253**	0.802
BMA2	1.028	0.888	1.538**	1.344	1.103	1.762**	3.720**	1.111	1.430	1.242**	0.805
Rridge	1.033	0.891	1.636**	1.272	1.060	1.735**	3.664**	1.125	1.478**	1.270**	0.784
LAR	1.068	0.901	1.395**	1.276	1.094	1.782**	3.823**	1.072	1.360	1.210**	0.791
EN	1.069	0.902	1.396**	1.275	1.094	1.777**	3.825**	1.073	1.360	1.210**	0.788
NNG	1.070	0.907	1.408**	1.275	1.097	1.779**	3.844**	1.072	1.362	1.207**	0.784
Mean	**0.966**	**0.876**	1.259**	1.046	**0.973**	1.441**	2.681**	0.986	1.143	1.150**	**0.745****
SP2, Recursive, h=1											
AR	6.257	0.007	18.650	0.002	0.010	0.000	1.316	0.010	0.003	**0.400**	0.006
ARX(SIC)	1.011	0.941	1.136	**0.966**	0.992	1.347**	1.088	**0.913**	1.181	1.135	**0.836**
CADL	0.976**	1.031	**0.991**	1.026	1.018	0.920	**0.988**	0.975	1.060	1.009	1.075
Bagging	1.492**	1.194	1.282**	1.266	**0.952**	1.823**	9.795**	1.316**	1.655	1.081	1.089
Boosting	1.465**	0.940	1.545**	1.606	1.206	1.196	1.282	1.287	1.157	1.119	1.257
BMA1	1.544**	1.012	1.595**	1.707	1.057	2.558**	1.337**	1.529**	1.246	1.049	1.793**
BMA2	1.021	0.934	1.603**	1.416	0.444	1.219	1.229	1.570**	1.237	1.638	1.093
Rridge	1.089	0.996	1.654**	1.447	1.477	1.329**	1.206	1.363**	1.157	1.734	1.077
LAR	**0.891****	0.890	1.388**	1.402**	1.119	1.173**	1.368**	0.961	1.108	1.083	1.332**
EN	0.893**	**0.885**	1.441**	1.402**	1.100	1.172**	1.367**	0.961	1.115	1.123	1.289**
NNG	0.959	0.925	1.394**	1.404**	1.118	1.114**	1.383**	0.943	1.142	1.976	1.083
Mean	0.965	0.917	1.131	1.142	0.973	**0.864****	1.229	1.008	**0.996**	1.021	1.000
SP3, Recursive, h=1											
AR	6.257	0.007	18.650	0.002	0.010	0.000	1.316	0.010	0.003	0.400	0.006
ARX(SIC)	1.011	**0.941**	1.136	0.966	0.992	1.347**	1.088	0.913	1.181	1.135	0.836
CADL	**0.976****	1.031	0.991	1.026	1.018	**0.920**	**0.988**	0.975	1.060	1.009	1.075
Bagging	6.843**	1.142	5.235**	5.337	2.120	13.78**	10.14**	3.521**	4.217	11.140	3.637
Boosting	1.010**	0.985	**0.991****	0.947	0.983	1.143	1.011	**0.893**	1.118	1.024	**0.826**
BMA1	1.137**	1.010	1.062**	0.989	1.069	2.150**	1.409**	0.966**	1.080	1.162	0.842**
BMA2	0.980	0.998	1.081**	0.996	1.027	1.440	1.301	0.905**	1.115	1.121	0.834
Rridge	1.564	1.274	1.354**	1.051	1.186	2.258**	1.231	1.322**	1.334	1.417	1.079
LAR	0.990**	0.998	1.000**	1.017**	1.004	0.981**	1.003**	1.000	**0.992**	**0.998**	0.847**
EN	0.990**	0.996	1.000**	1.016**	1.004	0.982**	1.001**	1.000	0.993	0.998	0.850**
NNG	0.996	0.998	0.998**	0.999**	1.000	0.989**	1.003**	0.995	1.001	1.001	1.000
Mean	1.012	0.954	1.365	**0.928**	**0.937**	1.822**	1.013	0.928	1.001	1.102	0.865

Notes: Only AR results are expressed in MSFE, and the results of other methods are expressed in RMSFE against the AR's MSFE. So, when the ratio is less than 1, it refers to the situation where the alternative performs better than AR, and vice versa. Best MSFE models are in bold and the DM test results testing equal forecast accuracy between a benchmark, here AR, and an alternative are expresseed with * denoting the rejection of the null at the 10% significance level and ** at the 5 % significance level.

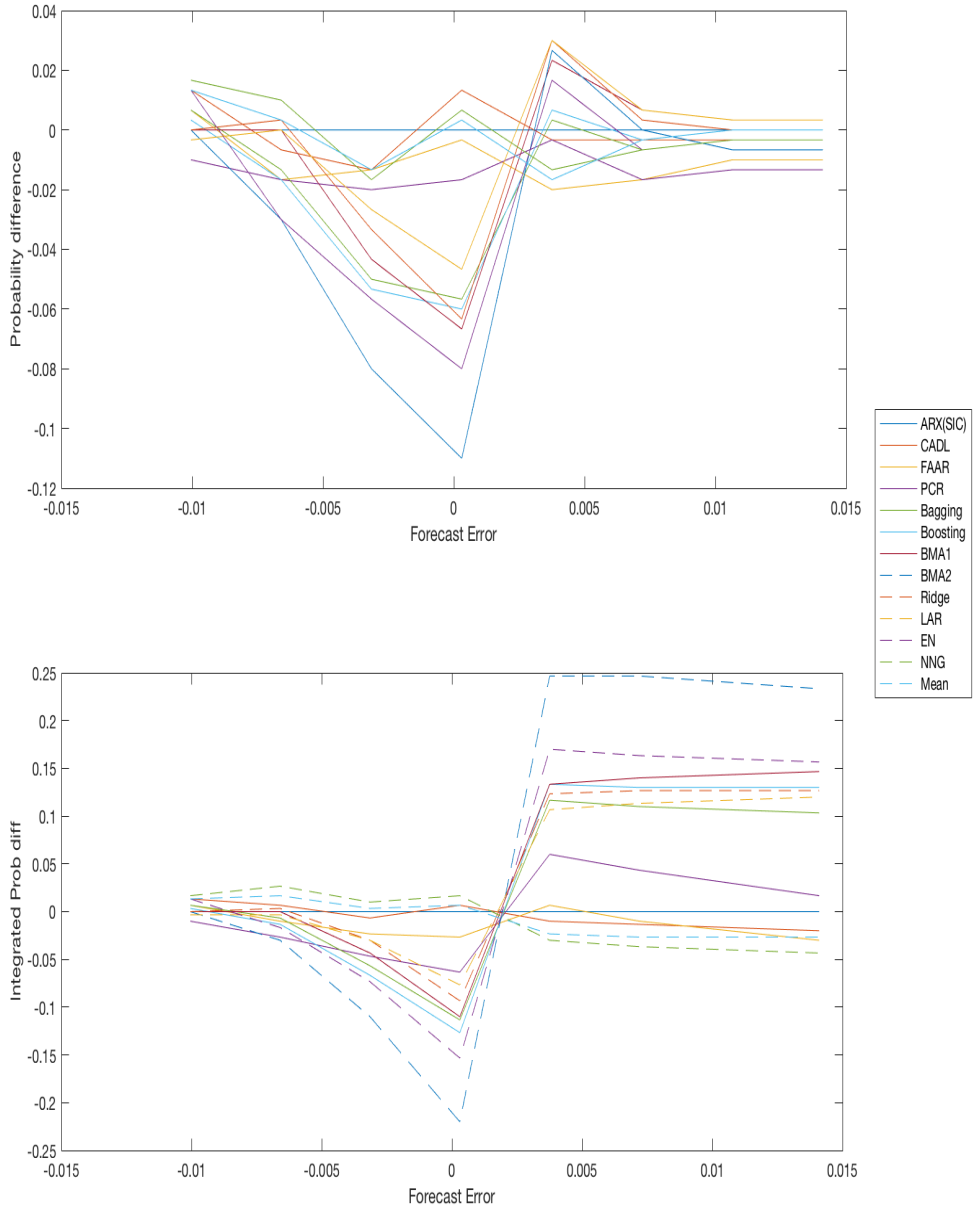

Figure 2.A.1: Plots of G_n and C_n statistics (SP1)

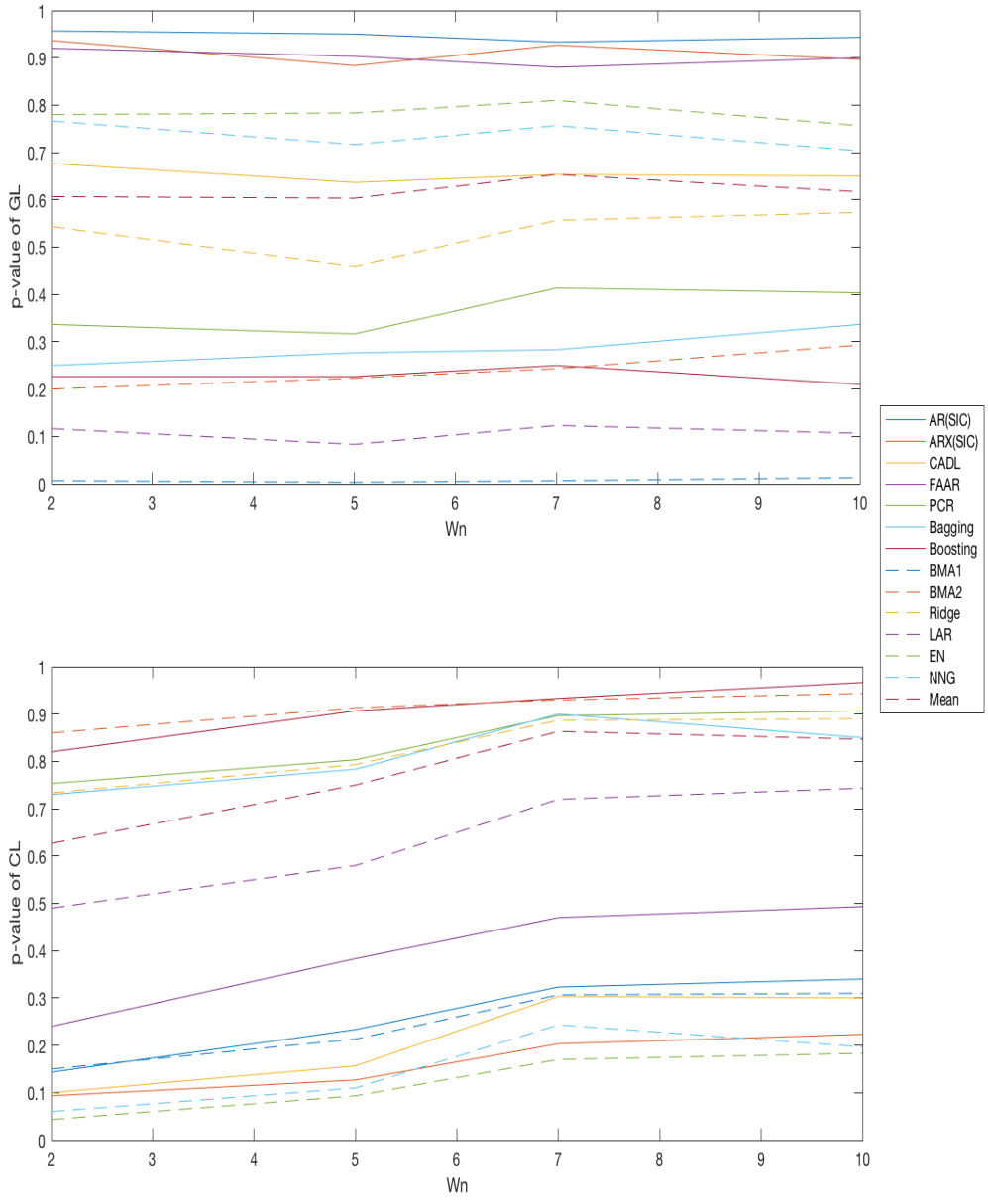

Figure 2.A.2: Plots of p-values of GL and CL test (SP1)

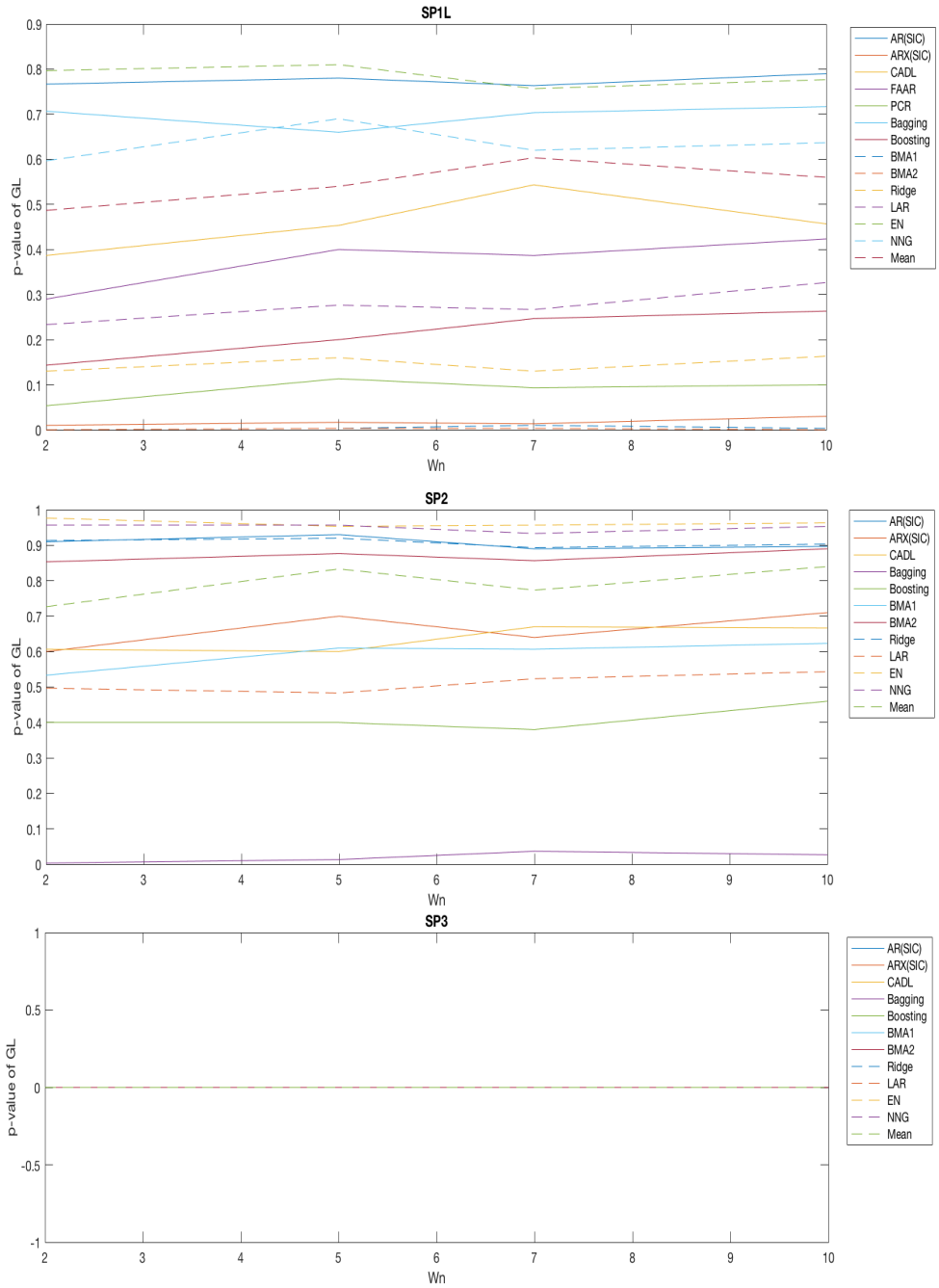

Figure 2.A.3: Plots of p-values of GL test (SP1L/SP2/SP3)

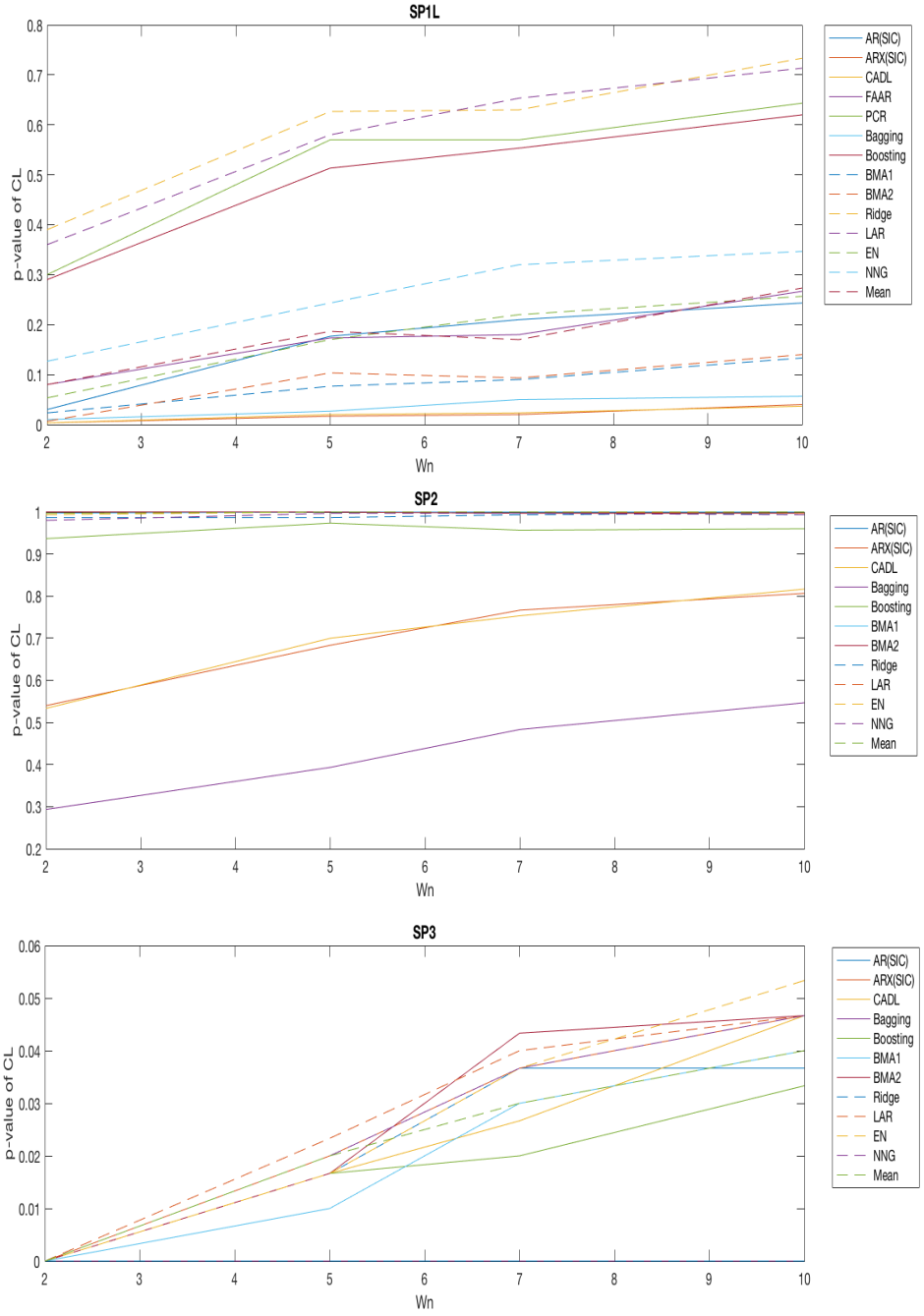

Figure 2.A.4: Plots of p-values of CL test (SP1L/SP2/SP3)

Chapter 3
Predictive Gains Associated with Using Supervised Model Specification Method in Big Data Environments

1 Introduction

This paper examines alternative approaches to how to construct a data set, select factors, and specify a forecasting model in big data environments. Since these choices can affect the efficiency and robustness of factor estimation and forecasting performance, it is crucial to choose a proper estimator and shrinkage method so that one can construct a feasible framework with a set of useful regressors "targeted" for the forecasting purpose.

Specifically, this paper assesses the efficacy of "supervised" approaches by carrying out Monte-Carlo simulations under different factor structures associated with a variety of data generating processes and empirical exercises. Traditional variable selection methods, factor estimators and forecasting models are "naive" in the sense that they extract a handful of factors that summarize the variability of the high-dimensional data set, without any consideration to which variable is being forecasted. On the other hand, supervised methods "train" the data set for a foresting variable, specifically by taking into account the correlations between data set and the variable being predicted. This paper brings the notion of supervision throughout all three stages of a model building process, the choice of regressors, the choice of the number of factors, and the specification of forecasting models. (Figure 3.A.1)

First, regarding the construction of the data set, Bai and Ng (2008) empirical results showed that pre-screening predictors rather than using the original

high-dimensional data set leads to more efficient factor estimation and better forecasting results and they proposed two ways to select 'targeted predictors' on the basis of the predictability for y_t, the hard and soft thresholding. Next, Jacobs et al. (2012) introduced a new information-based measure, Inf_{n+1}^R, which enables to order the series in the data set, X_t, and decide whether to add $x_{t,n+1}$, a $t \times 1$ vector, into X_t, a $T \times N$ data matrix or not. Their empirical results that used Stock and Watson (2005)'s data set also found that employing a moderate number of series such as around 40 to 50 out of 144 series results in better performances in terms of all three measures, (1) Inf_{n+1}^R; (2) the conventional information-based measures, such as AIC; and (3) the commonality ratio that measures the variation of the common component over the whole variation of the data set. In this paper, I empirically assess how pre-screening predictors affects forecasting performances (supervised principal component).

Second, as an exercise to see the implication of correct factor estimation, I compare three kinds of factor estimators, first, Bai and Ng (2002)'s IC estimator as IC-based approach, next, eigenvalue-based estimators such as, Ahn and Horenstein (2013)'s eigenvalue ratio (ER) and growth ratio (GR) estimators, Onatski (2010)'s ED (Edge Distribution) estimator, and lastly, supervised version of estimators, such as generalized cross-validation (GCV) and Mallows' C_L ((Li, 1986, 1987)). The comparison will be focused on the consistency and robustness of factor estimation and forecasting performances and will be tested under different factor structures. [1]

When constructing forecasting models using latent factor variables that are designed to condense information from large datasets into a small set of useful explanatory variables, standard approaches involve extracting information relevant to the entire dataset, and not targeted to a particular variable be-

[1] Boivin and Ng (2006)'s simulation results showed that, when the number of factors is underestimated, $\hat{r} < r$, factors are less precisely estimated and forecasting results are also worse compared to the case in which the correct number of factors is specified. On the other hand, when it comes to the overestimated case, $\hat{r} > r$, the efficiency of factor estimation improves compared to the underestimated case but the forecasting results are still worse than the correctly specified case.

ing forecasted. Supervised approaches to model specification do not do this, but instead penalize model specifications according to metrics designed to focus on the particular target variable(s) of interest. For example, regardless of which variable is being forecasted, standard principal component methods involve constructing diffusion indexes based on eigenvectors corresponding to the r largest eigenvalues of the correlation matrix of a data set. However, this practice is questionable, in the sense that it is not clear whether such latent factors (i.e., diffusion indexes) are effective in terms of predicting a specific variable. Supervised methods tackle this issue by "training" involving simple approaches such as taking into account the correlations between big data and the variable being predicted.

Thirdly, as for a forecasting model specification, the conventional principal component analysis relates common components with the marginal distribution of X_t. That is, the usual factor approach deals with $f(y, X) = f(y|X)f(X)$ and factors are distilled from $f(X)$. However, as pointed by Cox (1968), there is no reason why principal components as regressors should not be connected with the variable to be predicted and carry the information of it. Here, I introduce the supervised regression that hinges on $f(y, X) = f(X|y)f(y)$, which achieves the task of dimension reduction by utilizing the information contained in the dependent variable, $f(X|y)$. Specifically, while the usual PC (principal component) regression imposes a factor structure on the relationship between X_t and F_t, extracting a handful of factors as the summary information of a big data set, the supervised forecasting models internalize the correlation between the forecasting variable and the data set inside its forecasting scheme. In this study, three supervised forecasting models will be considered, such as a boosting, PLS (Partial Least Squares) regression, and CFPC (combining forecast-principal component).

In order to analyze the efficacy of such approaches, I carry out a Monte-Carlo simulation study in which I simulate different factor structures associ-

ated with a variety of data generating processes (with and without heterogeneity, for example), including cases where: (i) there are very few latent factors; (ii) there are many (up to 50) underlying factors; (iii) there are very few factors, and only 1 or 2 are relevant for forecasting the target variable; and (iv) no factors are relevant for forecasting. I compare factor estimation and forecasting results when supervised approaches are used. The simulation results indicate that, while IC-based estimator and eigenvalue-based approaches can consistently estimate r, supervised estimators can produce more robust forecasting results throughout all different types of DGPs. In addition, I found that the predictive gains when switching from PCR to PLS are quite substantial than the improvement from using supervised factor estimators, implying that what makes significant differences is a regression type, rather than configuration of regressors. Empirical exercise results that horse-race compare 16 forecasting models also indicate that supervised forecasting schemes outperform other un-supervised forecasting models, proving the efficacy of supervision. Especially, the predictive gains when using PLS are quite substantial and I observe that CFPC outperforms the simple average model most of time. Therefore, I conclude that supervised approaches that are geared for the purpose of forecasting do serve its own purpose.

This rest of the paper is organized as follows. In Section 2, I cover conventional approaches to variable selection and factor estimation in more details, and, in section 3, alternatively, I introduce supervised approaches. In Section 4, I simulate 8 different types of stationary environments and evaluate the performances of 6 different factor estimators including 2 supervised estimators. In Section 5, I conduct empirical exercises to see the efficacy of supervised approaches. Section 6 concludes.

2 Factor Model Specification

2.1 Do more data necessarily lead to better factor analysis ?

The question whether more data necessarily lead to better factor analysis is related to the task of finding the optimal number of regressors, N. Boivin and Ng (2006) concluded that, after trying different combinations of N & T, the data configuration matters and employing a moderate number of data series as few as 47 works better than using all available 147 series. Caggiano et al. (2011) also pointed out that, using 7 big European countries, extracting factors from the pre-selected predictors results in significantly better forecasting performances, rather than imposing the factor structure on the original high dimensional data. den Reijer (2013)'s empirical findings also emphasized the necessity of constructing a compact data set by considering the relationship between a forecasting variable and predictors.

As ways to shrink the original high-dimensional set, I implement a variety of robust shrinkage methods that are used in Lee (2017), which includes ridge, NNG, LARS, and the EN. In more details, refer to Lee (2017).

2.2 Estimator for the number of latent factors

Traditional variable selection approaches, such as AIC, BIC, and HQIC, are computationally infeasible when N and/or T grow(s) overwhelmingly large. As alternatives, first, Bai and Ng (2002) proposed PC and IC estimator that can consistently estimate the true underlying number of factors, r, in approximate factor models. [2] Although some papers still adopt this BN estimator, two critical weaknesses have been pointed out: one has to choose a threshold function and also has to specify the upper bound for the number of factors, r_{max}, which

[2]The difference between those two estimators, PC and IC, is that, as for the IC estimator, the first SSR term is log-transformed and Bai and Ng (2002)'s simulation results show its performance is more robust to the different choices of r_{max}.

makes the finite-sample properties of BN estimator unstable subject to the two choices.

Alternatively, estimators that associate with eigenvalues of the correlation matrix of X_t, were proposed. This approach is based on the idea that the first k^{th} eigenvalues explode when the cross-sectional dimension, N, goes to infinity, while the rest of eigenvalues are bounded. Thus, when the Frobenius norm of X_t is decomposed into,

$$\{||X_t^2||\} = tr(E(X_t X_t')) = \sum_{j=1}^{r} \lambda_j + \sum_{j=r+1}^{m} \lambda_j, \tag{38}$$

a factor analysis is to explain the variation of X_t by the eigenvectors corresponding to the first r largest eigenvalues of the correlation matrix of the data matrix, which will consist of the common component of the factor model. Here, estimating the threshold value, λ_{r+1}, is the task (de Reijer et al. (2014)). [3]

First, Onatski (2010) proposed an estimator that is a function of the difference between adjoining eigenvalues of the correlation matrix, specifically as,

$$\hat{r}(\delta) = max\{i \leq r_{max}^n : \lambda_i - \lambda_{i+1} \geq \delta\}. \tag{39}$$

Here, under the assumption that the upper bound that r can take grows as $n \to \infty$, it is denoted as r_{max}^n. One distinctive feature of Onatski (2010)'s estimator, compared to Bai and Ng (2002)'s, is that its threshold function is deterministic, not subject to an arbitrary scaling. As for the choice of δ, Onatski (2010) suggested an iterative procedure in which a calibrated value of δ, $\hat{\delta}$, will be continuously plugged into the recursive algorithm for $\hat{r}$ until convergence. On the other hand, de Reijer et al. (2014)'s estimator that is in the family of Onatski (2010)'s estimator sets $\widehat{\delta}_{r+1} = \frac{n}{r}\widehat{\lambda}_{r+1}$. Thus, if r_{max} is chosen, $\widehat{\delta}$ is

[3] Before embarking eigenvalue tests, one can visually check by drawing the Scree plot (Cattell (1966)), which detects a point where the slope of the line changes abruptly and that point will be the estimate of the number of latent factors, r. However, the decision can be subjective and ambiguous sometimes, but it is obviously a good eyeball test to begin with.

fixed as $\widehat{\delta} = \frac{n}{k}\widehat{\lambda}_{r_{max}+1}$. de Reijer et al. (2014)'s simulation results showed that their estimator performs better than Onatski (2010)' estimator in small sample cases especially under the situation where there exists neither significant cross-sectional nor serial correlation.

Next, Ahn and Horenstein (2013) proposed two eigenvalue-ratio estimators, eigenvalue ratio (ER) and growth ratio (GR) estimators that are based on the criterion functions,

$$ER(r) \equiv \frac{\tilde{\lambda}_{NT,r}}{\tilde{\lambda}_{NT,r+1}} \tag{40}$$

$$GR(r) \equiv \frac{\ln[V(r-1)/V(r)]}{\ln[V(r)/V(r+1)]} = \frac{\ln(1+\tilde{\lambda}^{*}_{NT,r})}{\ln(1+\tilde{\lambda}^{*}_{NT,r+1})} \tag{41}$$

where $\tilde{\lambda}_{NT,r} \equiv \psi_r[XX'/(NT)] = \psi_r[X'X/(NT)]$ and $\psi_r(A)$ denotes the r^{th} largest eigenvalues of a positive semi-definite matrix A. Here, $V(r) \equiv \sum_{j=r+1}^{m} \tilde{\lambda}_{NT,j}$ where $\tilde{\lambda}^{*}_{NT,r} \equiv \tilde{\lambda}_{NT,r}/V(r)$ and $V(r)$ also can be expressed as the sample mean of the squared residuals left from a regression of the forecasting variable Y on $\tilde{\lambda}_{NT,1}, \tilde{\lambda}_{NT,2}, ...,$
$\tilde{\lambda}_{NT,r}$. Then, ER estimator, $\hat{r}_{ER}$, and GR estimator, $\hat{r}_{GR}$, are defined as

$$\widehat{r}_{ER} = \max_{1\leq r\leq r_{max}} ER(r) \text{ and } \widehat{r}_{GR} = \max_{1\leq r\leq r_{max}} GR(r). \tag{42}$$

[4] Their simulation results showed that ER and GR estimators perform better than Bai and Ng (2002)'s BN estimator and Onatski (2010)'s ED (Edge Distribution) estimator under a general times series context when error terms are serially or/and cross-sectionally correlated and also under the situations where dominant/weak factors exist. In addition, the simulation results confirmed the finite-sample outperformance of ER and GR estimators when different values of r_{max} are tried. [5]

[4] Ahn and Horenstein (2013)'s simulation results indicate that, GR estimator works better than ER estimator especially when there are few dominating factors.

[5] Ahn and Horenstein (2013) suggested two ways to set a r_{max}. First, if one has a prior information on r_{max}, they suggested to set r_{max_1}=$2r_{max}$. On the other hand, if it is not the case, then they proposed to use r_{max_2} = min(r_{max^*},0.1m), where $m = min(N,T)$ and

3 Supervised approaches

3.1 Targeted variables

As evidenced from many simulation and empirical studies, the quality of data is critical as much as the quantity of data. Especially in big data environments, it is suggested that pre-screening regressors based on the predictability for Y_t (targeted) provides feasible forecasting framework and leads to better forecasting results as well. First, Bair et al. (2006) suggested to make a marginal decision to include a particular, X_j, based on,

$$c_j = \frac{X_j^T y}{\| X_j \|} > \theta,$$

given a fixed threshold number, θ, which takes into account the correlation between y_t and X_t.

Next, Bai and Ng (2008) suggested two screening mechanisms for selecting predictors, the hard and soft thresholding. The hard thresholding selects predictors solely based on the marginal predictability power of each single regressor, regardless of what other predictors are included in the regression: by checking whether the t-statistic of each regressor surpasses a given critical value, the data matrix will be composed of a compact set of these filtered regressors. The critical drawback of this approach is that regressors that might contain overlapping information could end up being chosen, since the approach concerns only the marginal predictability of each variable. On the other hand, the soft-thresholding approach takes into account what other regressors are included in the set and ridge, least absolute shrinkage operator

$r_{max^*} = \#\{r|\tilde{\mu}_{NT,r} \geq V(0)/m,\ r \geq 1\}$. Their simulation results show that, when the idiosyncratic component does not have any extreme dominant/weak factors and strong cross-sectional correlation in the idiosyncratic component, r_{max_2} leads to good performances of ER & GR estimators.

(lasso), elastic net (EN), and least angle regression (LARS) belong to this category. Specifically, when the unrestricted original factor analysis is specified as,

$$Y_{t+h} = \alpha W_t + \beta F_t + \epsilon_{t+h} \tag{43}$$

where the data matrix, X_t, and factors, F_t, are related linearly as,

$$X_t = \wedge F_t + e_t. \tag{44}$$

the factor analysis with targeted predictors,

$$Y_{t+h} = \alpha W_t + \bar{\Gamma} F_t + \epsilon_{t+h}, \tag{45}$$

is to estimate $\bar{\Gamma}$ which determines the entry of each predictor by shrinking and zeroing the coefficients of uninformative series.

3.2 Supervised estimators for the number of factors

As alternatives to IC-based and the eigenvalue-based estimators, I employ two supervised factor estimators, which internalize the correlation between the forecasting variable and the data set inside its optimization scheme. Specifically, following Carrasco and Rossi (2016), I try two cross-validation criteria, such as generalized cross-validation (GCV) and Mallows' C_L ((Li, 1986, 1987)). Suppose Φ is the matrix that consists of r eigenvectors associated with the eigenvalues that are greater than say η and define $\hat{\alpha}^{\eta}$ and $M_T^{\eta} y$ as,

$$\hat{\alpha}^{\eta} = (\hat{\Phi}'\hat{\Phi}')^{-1}\hat{\Phi}' y \tag{46}$$

$$M_T^{\eta} y = \hat{\Phi}\hat{\alpha}^{\eta}. \tag{47}$$

Then, the two cross-validation criteria are devised as to minimize $\parallel x_t'\hat{\alpha}^{\eta} - x_t'\alpha \parallel$ where $y_t = x_t'\alpha + e_t$ First, GCV estimates r based on,

$$\hat{r} = \underset{r \in R_T}{\operatorname{argmin}} \frac{T^{-1} \parallel y - M_T^{\eta} y \parallel^2}{(1 - T^{-1} tr(M_T^{\eta}))^2)} \tag{48}$$

and Mallows' C_L criterion that has $\hat{\sigma}^2$ in it is based on,

$$\hat{r} = \underset{r \in R_T}{\operatorname{argmin}} T^{-1} \parallel y - M_T^{\eta} y \parallel^2 + 2\hat{\sigma}_{\epsilon}^2 T^{-1} tr(M_T^{\eta}). \tag{49}$$

where R_T is the set which r can be chosen from and $\hat{\sigma}^2$ is a consistent estimator of σ^2, the variance of error term. Both approaches consider having no factors at all, $r = 0$, as a possible scenario.

3.3 Supervised forecasting models

The conventional principal component analysis relates the components with the marginal distribution of X_t. However, as pointed by Cox (1968), there is no reason why principal components as regressors should not be connected with the dependent variable to be predicted and carry the information of it.

As an alternative, Cook et al. (2008) suggested Principal Fitted Components (PFC), which uses an "inverse regression X on Y to gain reductive information for the forward regression of Y on X." The usual factor approach deals with $f(y, X) = f(y|X)f(X)$ and factors are distilled from the marginal distribution of X, $f(X)$. On the other hand, the inverse regression hinges on $f(y, X) = f(X|y)f(y)$, achieving the task of dimension reduction by utilizing the information of the dependent variable, $f(X|y)$. (Giovannelli and Proietti (2015))

In the following subsection, I will brief four supervised forecasting models, Boosting (Freund and Schapire (1995)), PLS (partial least squares) regression

(De Jong (1993), Garthwaite (1994), and Groen and Kapetanios (2016)), PCovR (Principal Covariate Regression, De Jong and Kiers (1992) and and Heij et al. (2007)), and CFPC (Combining forecast PC, Huang and Lee (2010) and Chan et al. (1999)).

3.3.1 Boosting

Boosting is a forward stage-wise learning algorithm introduced by Freund and Schapire (1995). By setting the sample average as an initial value, boosting sequentially updates and aggregates estimates obtained up to the current steps with a weight. The first variant of Boosting algorithm,"AdaBoost", was developed by Freund and Schapire (1995). Then, "Real AdaBoost", which involves an exponential loss function, was developed by Friedman et al. (2000), after which "L_2 Boosting," using a quadratic loss function, was proposed by Friedman (2001).

Next, Bai and Ng (2009) came up with two boosting algorithms applicable to an out-of-sample forecasting framework, component-wise L_2 boosting and block-wise L_2 boosting. First, Component-wise L_2 boosting considers each variable as a separate potential regressor and attempts to minimize $SSR(i)$ that are left after regressing with the i^{th} regressor. By combining principal component analysis with this Boosting method (component + boosting), the drawback that comes from using the pure principal component approach solely can be avoided. Next, block-wise L_2 boosting is an algorithm that treats a block of lagged terms as a regressor. Their simulation results show that component-wise L_2 boosting is consistent and results in a parsimonious model relatively, while block-wise L_2 boosting turns out to be more efficient. In this paper, I use the component-wise L_2 Boosting method as one of the shrinkage estimation methods. For more detailed information on the procedure, refer to Bai and Ng (2009) and Kim and Swanson (2014).

3.3.2 PLS (partial least squares) regression

PLS trains factors that are linear combinations of X_t, ΓX_t, by maximizing the correlation between Y_t and ΓX_t (Wold (1982)). The consistency of the ordinary PC estimators is established under the assumption that common components dominate the co-movement of the underlying data set, $X'X$, and Groen and Kapetanios (2016)'s simulation and empirical results confirm that, when it comes to weak factor models, PLS outperforms the ordinary PC estimator. For practical implementations, I follow the algorithm introduced in Groen and Kapetanios (2016) (Algorithm 1. in chapter 2.1.).

3.3.3 Principal Covariate Regression (PCovR)

A forecasting exercise using the usual PC can be viewed as the two-stage optimization problem: the first step is to estimate r factors, $\hat{r}$, that minimizes

$$\| X_T - \hat{X}_T \| \tag{50}$$

where $X_T = \Lambda F_k + e_T$ and $\widehat{X}_T = \widehat{\Lambda}\widehat{F}_k + \widehat{e}_T$ and, then the forecasting method that minimize forecasting error, $\hat{e}_T = y_T - \hat{y}_T$, the most will be selected based on a particular evaluation method. On the other hand, Principal Covariate Regression (PCovR) combines them into a one-stage optimization problem that incorporates the notion of supervision by minimizing,

$$w_1 \| X_T - \hat{X}_T \|^2 + w_2 \| y_T - \hat{y}_T \|^2 \tag{51}$$

for a fixed w_1 and w_2. When $w_1 = 0$, the emphasize is fully imposed on fitting y_t, while when $w_2 = 0$, it is just the usual PC approach. For the detailed procedure, I will follow Heij et al. (2007).

3.3.4 Combining forecast PC (CFPC)

Combining forecast PC (CFPC) motivates from the ideas that it can be better to combine forecasts (CF) than combining information (CI) under certain situations and that, even if a specific CI model turns out to be the best in-sample, it is not guaranteed it will be the best out-of-sample as well. That is, CF is designed to incorporate the notion of supervision by combining m different $\hat{y}_i$ where $i = 1, ..., m$ also, not only extracting information of data set, X_t, from each model.

As ways to fix the weights, w_i, Huang and Lee (2010) suggested (1) simple average, $w_i = \frac{1}{m}$ (CF-mean), (2) regression-based approach, and (3) CF-PC. Let $\hat{y}_{t+h} = (\hat{y}^1_{t+h}, \hat{y}^2_{t+h}, ..., \hat{y}^m_{t+h})$, a vector of forecasts from m different models, then based on the equation,

$$\hat{y}_{t+h} = \Lambda F_{t+h} + v_{t+h} \tag{52}$$

factors, $\hat{F} = (\hat{F}_{1+h}, \hat{F}_{2+h}, ..., \hat{F}_{T+h})'$, are estimated as

$$\widehat{F} = \widehat{Y}\widehat{\Lambda}/N \tag{53}$$

where $\hat{Y} = (\hat{y}_{1+h}, \hat{y}_{2+h}, ..., \hat{y}_{T+h})'$ and $\widehat{\Lambda}$ is set as the corresponding eigenvectors of the r largest eigenvalues of $\hat{Y}'\hat{Y}$. Then, a CF-PC forecast, $\widehat{y}^{CF-PC}_{t+h}$, is computed based on the equation,

$$y_{t+h} = \gamma_0 + \sum_{i=1}^{r} \gamma_i \widehat{F}^{(i)}_{t+h} + e_{t+h} \tag{54}$$

where r, the number of factors, is estimated using one of estimators for the number of factors. Thus, when it comes to CF-PC, factors are a function of forecast values from m different forecasting models.

Huang and Lee (2010)'s empirical results indicate that, when N is relatively small, CF-mean often outperforms CF-PC, since the consistency of factor es-

timation requires $N \to \infty$. In this paper, I also conduct arithmetic mean and Bayesian model averaging (BMA) and compare the forecasting performances of different model averaging approaches. For the detailed procedures on BMA, refer to Lee (2017).

4 Monte-Carlo Simulation

4.1 Description of Data Generating Process

In order to assess the efficacy of supervised approaches, I carry out a Monte-Carlo simulation study in which I simulate different factor structures associated with a variety of data generating processes (with and without heterogeneity, for example), including cases where: (i) there are very few latent factors; (ii) there are many (up to 50) underlying factors; (iii) there are very few factors, and only 1 is relevant for forecasting the target variable; and (iv) no factors are relevant for forecasting (Carrasco and Rossi (2016)). And (v) to (viii) are the extension of (i) to (v) to a heterogeneous setting. The underlying DGP assumes a baseline factor structure as,

$$y_{t+h} = \sum_{j=1}^{r} \beta_j f_{jt} + \varepsilon_{t+h} \tag{55}$$

$$x_{it} = \sum_{j=1}^{r} \lambda_{ij} f_{jt} + \sqrt{\theta} u_{it} \tag{56}$$

where

$$u_{it} = \sqrt{\frac{1-\sigma^2}{1+2H\gamma^2}} e_{it} \tag{57}$$

(Ahn and Horenstein (2013)). Throughout DGP 1 to DGP 4, ε_{t+h} and u_{it} are assumed to be iid $N(0,1)$, and then the restriction is relaxed for DGP 5 to DGP 8, by allowing both serial correlation and cross-sectional correlation to be pos-

sible ($\sigma = 0.5$, $\gamma = 0.2$, and $H = max(10, \frac{N}{20})$). Here, the number of factors, r, will be estimated using Bai and Ng (2002)'s IC approach, eigenvalue-based approaches such as Onatski (2010), Ahn and Horenstein (2013), and the supervised version of estimators, such as generalized cross-validation (GCV) and Mallows' C_L and will compare their factor estimation and forecasting performance results under these 8 different circumstances.

In details, DGP 1 has a few factor structure where $r = 4$ and $r_{max} = r + 10$, with β, a $(r \times 1)$ vector of ones under *iid* error assumptions. DGP 2 describes the situation in which there are as many factors as $r = 50$ and r_{max} is set as $min(N, \frac{t}{2})$. DGP 3 considers the case when not all of factors are relevant for predicting y_{t+h} with $r = 5$, $r_{max} = min(r + 10, min(N, \frac{T}{2}))$, and $\beta = (1, 0_{1\times 4})'$. Here, factors can be fractioned into two parts, $f = [f_1, f_2']'$, where f_1 is a relevant factor and f_2 refers to a vector of four irrelevant factors. Specifically, f will have a structure as, $f \sim N(0_{1\times 5}, \Omega)$ where $\Omega = \begin{pmatrix} 1&0&0&0&0 \\ 0&2&0&0&0 \\ 0&0&3&0&0 \\ 0&0&0&4&0 \\ 0&0&0&0&5 \end{pmatrix}$. Thus, given the structure of Ω, f_1 is a weak factor and f_2 is a group of dominant factors. Lastly, DGP 4 refers to the case when all factors are unrelated to y_{t+h}, with β fixed as a vector of five (r) zeros and $r_{max} = r + 10$. Factors will have a similar structure as that of DGP 3. DGP 5 is the application of DGP1 to a heterogeneity case ($\sigma = 0.5$, $\gamma = 0.2$, and $H = max(10, \frac{N}{20})$): in the same way, DGP 6 is a heterogeneous extension from DGP 2, DGP 7 from DGP 3, and DGP 8 from DGP 4. I set $R = 300$, in-sample period, and, $P = 300$, out-of-sample period, then $T = R + P = 600$. The out-of-sample forecasting exercise is based on PCR and PLS and I use the recursive estimation window.

4.2 Simulation Results

Table (3.A.5) reports the Monte-Carlo simulation results. The third column indicates the true underlying number of factors, r, that I fixed when simulating each DGP. For every DGP, I report the estimated number of latent factors, $\widehat{r}$,

using 6 different factor estimators, including IC-based approach, 3 eigenvalue-based approaches, and 2 supervised approaches (Table (3.*A*.3)), which is the average of estimated number of factors computed over when the recursive estimation window moves. I also report the standard errors involved in the factor estimation. The out-of-sample forecasting results are evaluated in terms of MSFE (mean squared forecast error). $MSFE^1$ refers to when out-of-sample forecasting is carried out using PCR, while $MSFE^2$ is the results when using PLS.

Throughout different DGPs, I observe that IC estimator and three eigenvalue-based estimators can consistently estimate the number of factors most of the time. Specifically, under i.i.d. settings from DGP 1 to DGP 4, the four estimators are able to estimate r consistently all the time. When it comes to heterogeneity cases, ER & GR estimators can estimate the correct number of factors always, while, as for BN & ED, they tend to overestimate sometimes. However, the deviations of $\widehat{r}$ from r are always less than 1 and the forecasting results are also quite comparable each other.

On the other hand, supervised estimators sometimes tend to underestimate and overestimate r. Interestingly, under DGP 3, 4, 7, & 8, where not all of factors are relevant for y_t, supervised estimators report estimates that are closer to the number of relevant factors, r^*, rather than to the true underlying factor numbers, r, but their forecasting results are markedly better than the case when the eigenvalue-based approaches and IC-based estimator are used. For example, as for DGP 3 case in which r is set as 5 but the number of relevant factors is 1, then all of the eigenvalue-based estimators report $\hat{r}$ as 5, but GCV gives 1.6067 and Mallow's Cl 1.62. And the forecasting results are better when using the supervised estimators. The same story goes for DGP 4 and 8 as well, except DGP 7 in which the supervised estimators result in higher MSFEs than un-supervised estimators.

Another noticeable finding is that when the underlying number of factors

are as many as $r = 50$, (DGP 2 & 6) and forecasting regression is carried out using PCR, the associated forecasting uncertainties are much higher compared to other DGPs. For example, the $MSFE^1$s of DGP 2 & 6 are above 60, while the $MSFE^1$s results for all other DGPs are less than 5. In addition, when the number of relevant factors are as small as 1 or 0 (DGP 3, 4, 5, & 8), the $MSFE^1$s results are always less than 2. It indicate that estimation uncertainty largely determines forecasting performances, regardless of whether it is i.i.d. or heterogeneous.

However, when PLS is used for out-of-sample forecasting, the forecasting results, $MSFE^2$, are always better and more robust to different DGPs than the results when using PCR. The effects of using the supervised forecasting model are more visible and robust than those of using the supervised factor estimator. This may indicate that what makes significant differences is a regression type, rather than configuration of regressors. Interestingly, the results make hard to say there is double-supervision effect. That is, when both supervised estimator and supervised forecasting scheme are used, the results are not way better than when the supervised forecasting model solely is used, or sometimes they are worse.

In conclusion, the simulation results indicate that, first, the IC-based approach and eigenvalue-based approaches can consistently estimate the number of factors most of time and their performances are robust to different types of DGPs, while supervised approaches occasionally over- and underestimate r. Especially when the number of underlying factors and the number of relevant factors for y_t are different, the supervised approach estimates are closer to the number of relevant factors, which results in better forecasting results. However, I can observe more substantial improvements when switching from PCR to PLS, indicating that supervised forecasting models can contribute to forecasting accuracy more effectively than supervised factor estimators. Also, the predictive gains are robust throughout different DGPs.

5 Empirical Application

5.1 Description of Data and Forecasting methods

The data set to be used in this empirical application is Kim and Swanson (2014)'s updated and expanded dataset of Stock and Watson (2002, 2012)'s. 144 US macroeconomics time series that spans from January 1960 to May 2009 (monthly) will be our data set, X_t, whose full list is available in Kim and Swanson (2013). Using this big dimensional data set, I forecast 11 macro variables, Y_{t+h}, which are the core economic variables that the Federal Reserve Bank pays close attention into for formulating monetary policies: unemployment rate(UR), personal income less transfer payments (PILT), 10 year Treasury-bond yield (TB10Y), consumer price index (CPI), producer price index (PPI), non-farm payroll employment (NPE), housing starts (HS), industrial production (IPX), M2, S&P 500 index (SNP), and gross domestic product (GDP). Before proceeding with the empirical analysis, all forecasting variables, Y_{t+h}, are transformed to make each time series stationary: basically all level variables are log-differenced and other variables already in log form are just taken as the first difference between t and $t-1$. For detailed descriptions on how each variable is transformed, refer to Table 3.A.1.

I will horse-race compare the total 16 forecasting models that includes basic econometric models such as AR & CADL, factor models, shrinkage and machine learning approaches, and 3 supervised forecasting models, boosting, PLS, and $CFPC$. (Table 3.A.1) As for the estimator of the number of latent factors, I compare 6 different estimators, such as BN estimator, ER and GR estimator, ON estimator, and supervised estimators, such as GCV criterion and Mallows' C_L. (Table 3.A.3)

5.2 Empirical Results

Table (3.A.6) to (3.A.11) report the MSFE results of 16 forecasting methods (Table 3.A.1) combined with each different factor estimator (Table 3.A.3). First, within this fixed data environment, I cannot conclude that the supervised factors estimators uniformly outperform the un-supervised factor estimators across different forecasting horizon and different forecasting variables nor can I decide a specific estimator is dominant. These results are expected, since unless under special circumstances where the true number of factors is not the same with the number of relevant factors, the simulation results already indicate that the 6 different estimators perform well, resulting in consistent estimation and comparable forecasting performances most of the time. However, although the MSFE results do not show much difference among different factor estimators, but they do differ over different forecasting models, which can be confirmed from simulation results as well. That is, in this forecasting exercise, what makes visible difference is a specific shrinkage or supervising method that will determine a way to shrink and train the set.

The most notable result is PLS (partial least squares) "uniformly" and overwhelmingly outperforms other forecasting models across different forecasting variables, forecasting horizons, and 6 different factor estimators tried. Within a fixed Y_t and h, one can see that PLS "wins" all the time, proving the efficacy of the supervised forecasting model. The performance of CFPC is also impressive, resulting in better performances than the simple mean average model most of time, but not enough to outperform other alternative models all the time. It implies that the notion of model averaging seems to work better when factors are trained for the forecasting values from the competing forecasting models, compared to the case in which an uniform weight is assigned to each forecasting model. However, when it comes to boosting, it is difficult to say its performance is outstanding compared to un-supervised forecasting models.

In addition, regardless of the discussion of the efficacy of supervised approaches, the MSFE results show that, in general, it is more difficult to forecast UR (unemployment rate), TB (10-year treasury bond), and HS (Housing starts) variables accurately than other variables. In addition, I can observe that the magnitudes of MSFE generally increase as h increases, which is an expected result in the forecasting literature since uncertainties for a more distant future time period are higher. Plus, as confirmed from Lee (2017), across different Y_t and h, 6 different factor estimators, if one is to compare the forecasting results of un-supervised forecasting models, then hybrid methods, shrinkage approaches combined with diffusion index, outperform benchmark models such as AR and CADL and purely factor-based approaches most of the time.

6 Conclusion

As a sequence of studies on forecasting model specification in big data environments, this paper attempts to verify the efficacy of supervised approaches by carrying out Monte-Carlo simulations and empirical exercises. Traditional factor estimators and forecasting models are "naive" in the sense that the optimization processes do not take into account the correlations between big data and the variable being predicted. On the other hand, "supervised" methods "train" the data set for a foresting variable so that it can be targeted for the purpose of forecasting the variable.

The Monte-Carlo simulation results indicate that three different factor estimators, IC-based approach, eigenvalue-based approach, and supervised approach can consistently estimate the true number of latent factor, r, most of the time except the case when the number of factors, r, and the number of relevant factors for Y_t, r^*, are different. In that case, I can observe that supervised factor estimators outperform un-supervised estimators. Interestingly, I can observe

more substantial improvements when switching from PCR to PLS, indicating that supervised forecasting models can contribute to forecasting accuracy more effectively than supervised factor estimators do. Also, the predictive gains are robust throughout different DGPs. Empirical exercise results that horse-race compare 16 forecasting models also indicate that supervised forecasting schemes outperform other un-supervised forecasting models, proving the efficacy of supervision. Especially, the predictive gains when using PLS are quite substantial and I observe that CFPC outperforms the simple average model most of time. Therefore, I conclude that supervised approaches that are geared for the purpose of forecasting do serve its own purpose.

7 Appendix

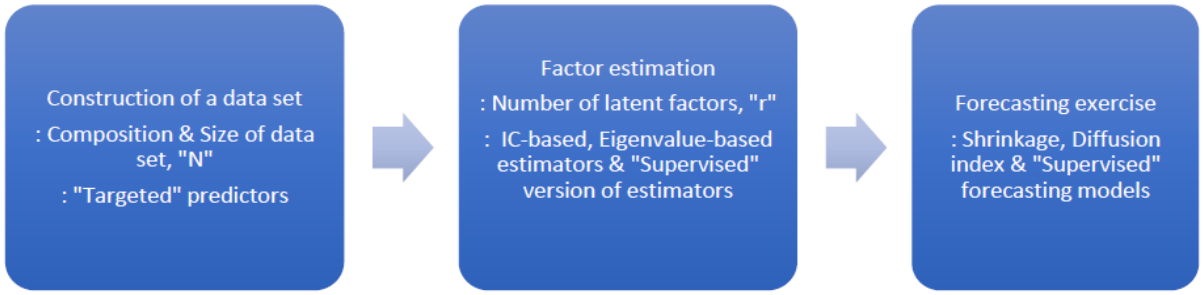

Figure 3.A.1: Factor model specification

Table 3.A.1: 16 forecasting models and shrinkage methods employed

Names in abbreviation	Description
AR(SIC)	Autoregressive model with lags selected by SIC
ARX	Autoregressive model with exogenous regressors
CADL	Combined AR distributed lag model
FAAR	Factor augmented AR model
PCR	Principal components regression
Bagging	Bagging with shrinkage, $c = 1.96$
BMA1	Bayesian model averaging with g-prior=$\frac{1}{T}$
BMA2	Bayesian model averaging with g-prior=$(\frac{1}{P})^2$
Ridge	Ridge regression
LAR	Least angle regression
EN	Elastic net
NNG	Non-negative garotte
Mean	Arithmetic mean
Boosting	Component Boosting, $M = 50$
PLS	Partial Least Squares
CFPC	Combining forecast PC (CFPC)

Table 3.A.2: Abbreviation

1	AR(SIC)	Benchmark
2	ARX	
3	CADL	
4	FAAR	Factor
5	PCR	
6	Bagging	Hybrid
7	BMA1	
8	BMA2	
9	Ridge	
10	LAR	
11	EN	
12	NNG	
13	Mean	Mean
14	Boosting	Supervised
15	PLS	
16	CFPC	

Table 3.A.3: Factor estimator

Bai and Ng (2002)	IC	IC-based approach
Ahn and Horenstein (2013)	ER & GR	eigenvalue-value based appraoch
Onatski (2010)	ED	
Li (1986, 1987)	GCV Mallow's Cl	Supervised

Table 3.A.4: Target forecasting variables

Series Name	Abbreviation	Transformed Y_{t+h}
Unemployment rate	UR	$Y_{t+1} - Y_t$
Personal income less transfer payments	PI	$ln(Y_{t+1}/Y_t)$
10-year treasury bond	TB	$Y_{t+1} - Y_t$
Consumer price index	CPI	$ln(Y_{t+1}/Y_t)$
Producer price index	PPI	$ln(Y_{t+1}/Y_t)$
Nonfarm payroll employment	NPE	$ln(Y_{t+1}/Y_t)$
Housing starts	HS	$ln(Y_t)$
Industrial production	IPX	$ln(Y_{t+1}/Y_t)$
M2	M2	$ln(Y_{t+1}/Y_t)$
S&P 500 index	SNP	$ln(Y_{t+1}/Y_t)$
Gross domestic product	GNP	$ln(Y_{t+1}/Y_t)$

Table 3.A.5: Monte carlo simulation results

		r	r^*	BN	ER	GR	ED	GCV	CL
DGP1	rhat	4	4	4	4	4	4	5.6233	4.8667
	(s.e.)			0	0	0	0	1.5651	0.9514
	$MSFE^1$			4.8811	4.8811	4.8811	4.8811	4.8937	4.8831
	$MSFE^2$			0.6956	0.6956	0.6956	0.6956	0.5523	0.5829
DGP2	rhat	50	50	50	50	50	50	50.7133	50.49
	(s.e.)			0	0	0	0	1.8482	1.7185
	$MSFE^1$			61.823	61.823	61.823	61.823	61.8642	60.9972
	$MSFE^2$			0.3502	0.3502	0.3502	0.3502	0.3526	0.3528
DGP3	rhat	5	1	5	5	5	5	1.3667	2.54
	(s.e.)			0	0	0	0	0.9462	2.0759
	$MSFE^1$			1.1593	1.1593	1.1593	1.1593	1.1339	1.1327
	$MSFE^2$			0.6838	0.6838	0.6838	0.6838	0.9695	0.9104
DGP4	rhat	5	0	5	5	5	5	3.21	3.3167
	(s.e.)			0	0	0	0	0.56	0.5867
	$MSFE^1$			1.4294	1.4294	1.4294	1.4294	1.418	1.4203
	$MSFE^2$			0.7407	0.7407	0.7407	0.7407	1.0736	1.0672
DGP5	rhat	4	4	4.4867	4	4	4	4.6367	4
	(s.e.)			0.7991	0	0	0	2.075	0
	$MSFE^1$			4.9155	4.9266	4.9266	4.9266	4.9085	4.9266
	$MSFE^2$			0.6508	0.7812	0.7812	0.7812	0.7593	0.7812
DGP6	rhat	50	50	50.6867	50	50	50	50.5133	52.3733
	(s.e.)			1.2438	0	0	0	0.5007	2.8415
	$MSFE^1$			59.2928	59.1952	59.1952	59.1952	59.1465	59.2451
	$MSFE^2$			0.3524	0.353	0.353	0.353	0.3526	0.3485
DGP7	rhat	5	1	5.4633	5	5	5	1.3833	1.44
	(s.e.)			0.8782	0	0	0	0.487	0.4972
	$MSFE^1$			1.3127	1.3141	1.3141	1.3141	1.2992	1.2997
	$MSFE^2$			0.7887	0.812	0.812	0.812	1.0358	1.0352
DGP8	rhat	5	0	5.3867	5	5	5	2.7967	3.97
	(s.e.)			0.931	0	0	0	2.6906	3.8824
	$MSFE^1$			1.1294	1.1298	1.1298	1.1298	1.1333	1.1259
	$MSFE^2$			0.8857	0.9577	0.9577	0.9577	0.9545	0.8498

Note: All DGPs assume a factor structure with different numbers of factors, r, and different numbers of relevant factors for Y_t, r^*. The DGP 1 to 4 describe i.i.d. error cases and DGP 5 to 8 are for serially and cross sectionally-correlated errors. $MSFE^1$ refers to when out-of-sample forecasting is carried out using PCR, while $MSFE^2$ is the results when using PLS.

Table 3.A.6: MSFE results with BN estimator

Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
Recursive, h=1											
AR	6.01647	0.00640	25.93355	0.00263	0.01100	0.00032	1.83508	0.00984	0.00351	0.46105	0.00664
ARX(SIC)	6.32510	0.00655	20.41634	0.00183	0.01005	0.00040	1.43406	0.00956	0.00382	0.45194	0.00517
CADL	6.81039	0.00718	19.09536	0.00254	0.00990	0.00070	10.22033	0.01125	0.00557	0.42074	0.00620
FAAR	7.25588	0.00647	20.09121	0.00230	0.01028	0.00082	6.19255	0.01179	0.00469	0.45181	0.00588
PCR	6.86619	0.00690	18.69430	0.00189	0.01000	0.00039	1.53261	0.01090	0.00314	0.42224	0.00608
Bagging	6.96982	0.00718	19.25276	0.00187	0.00981	0.00046	1.62188	0.01190	0.00308	0.40849	0.00603
BMA1	6.45073	0.00630	25.03210	0.00261	0.01102	0.00036	2.20504	0.00947	0.00331	0.46866	0.00689
BMA2	6.42131	0.00627	24.81385	0.00259	0.01103	0.00036	2.20447	0.00949	0.00331	0.46949	0.00682
Rridge	6.01268	0.00637	25.00934	0.00259	0.01099	0.00031	1.87685	0.00953	0.00347	0.46185	0.00669
LAR	6.32654	0.00707	18.07647	0.00190	0.01002	0.00030	1.34765	0.01064	0.00320	0.39898	0.00618
EN	6.26707	0.00703	18.57563	0.00191	0.01003	0.00029	1.32516	0.01067	0.00320	0.39951	0.00607
NNG	6.23623	0.00699	18.42869	0.00190	0.00999	0.00029	1.33771	0.01055	0.00315	0.40026	0.00610
Mean	9.72605	0.00651	28.21763	0.00195	0.00912	0.00045	1.38529	0.01590	0.00371	0.47392	0.00529
Boosting	6.09573	0.00631	25.06456	0.00261	0.01102	0.00032	1.88140	0.00943	0.00333	0.45721	0.00672
PLS	**0.000032**	**0.000000**	**0.000139**	**0.000000**	**0.000000**	**0.000000**	**0.036116**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.03212	0.00627	18.89379	0.00194	0.00960	0.00030	1.66259	0.00955	0.00278	0.41703	0.00553
Recursive, h=3											
AR	6.32542	0.00650	21.73429	0.00239	0.01097	0.00042	3.31169	0.01076	0.00420	0.43161	0.00525
ARX(SIC)	6.84166	0.00604	21.60795	0.00254	0.01064	0.00046	2.55763	0.01074	0.00357	0.43824	0.00524
CADL	6.84350	0.00711	21.02610	0.00280	0.01046	0.00075	11.27095	0.01133	0.00607	0.44135	0.00615
FAAR	7.14151	0.00669	23.50582	0.00248	0.01044	0.00086	8.64967	0.01279	0.00488	0.46570	0.00579
PCR	6.87827	0.00673	23.74844	0.00250	0.01091	0.00049	3.24384	0.01248	0.00388	0.46742	0.00604
Bagging	6.73097	0.00740	24.07297	0.00240	0.01061	0.00071	3.37871	0.01350	0.00367	0.44101	0.00629
BMA1	7.15409	0.00669	21.96361	0.00245	0.01123	0.00046	5.15734	0.01218	0.00431	0.43942	0.00599
BMA2	7.11549	0.00662	21.65526	0.00245	0.01130	0.00046	5.14545	0.01212	0.00431	0.43990	0.00602
Rridge	6.30450	0.00647	21.20795	0.00234	0.01084	0.00040	3.51418	0.01072	0.00424	0.43185	0.00526
LAR	6.42233	0.00664	21.61359	0.00251	0.01111	0.00043	2.68708	0.01085	0.00365	0.43530	0.00540
EN	6.29079	0.00655	21.62100	0.00253	0.01117	0.00039	2.55939	0.01039	0.00358	0.43419	0.00526
NNG	6.31423	0.00650	21.78195	0.00255	0.01122	0.00039	2.57252	0.01033	0.00359	0.43382	0.00530
Mean	8.21350	0.00633	39.96991	0.00245	0.01040	0.00050	2.67249	0.01389	0.00400	0.50263	0.00527
Boosting	6.47873	0.00654	22.18960	0.00237	0.01105	0.00044	3.72695	0.01149	0.00421	0.43094	0.00553
PLS	**0.000026**	**0.000000**	**0.000026**	**0.000000**	**0.000000**	**0.000000**	**0.020333**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.16024	0.00628	21.30342	0.00226	0.01032	0.00040	3.30326	0.01040	0.00355	0.43309	0.00532
Recursive, h=12											
AR	7.52761	0.00669	21.02375	0.00248	0.01102	0.00092	12.45944	0.01197	0.00406	0.43427	0.00530
ARX(SIC)	7.79336	0.00644	21.25878	0.00255	0.01079	0.00092	12.19288	0.01224	0.00409	0.43422	0.00531
CADL	7.31029	0.00721	20.92354	0.00283	0.01050	0.00096	16.04111	0.01213	0.00644	0.44127	0.00621
FAAR	7.81270	0.00681	22.26425	0.00263	0.01087	0.00105	15.84247	0.01282	0.00455	0.44546	0.00564
PCR	7.92898	0.00687	21.99345	0.00254	0.01094	0.00102	13.35102	0.01290	0.00417	0.44654	0.00570
Bagging	7.52654	0.00672	23.35133	0.00256	0.01098	0.00104	10.68486	0.01253	0.00383	0.43440	0.00527
BMA1	8.66192	0.00727	20.93315	0.00237	0.01101	0.00145	20.58426	0.01326	0.00404	0.43535	0.00524
BMA2	8.66014	0.00729	20.97945	0.00236	0.01097	0.00144	20.56385	0.01344	0.00404	0.43671	0.00525
Rridge	7.49785	0.00673	20.76322	0.00243	0.01091	0.00095	13.26538	0.01206	0.00396	0.43549	0.00537
LAR	7.66963	0.00667	21.08653	0.00253	0.01097	0.00102	12.47747	0.01245	0.00399	0.43448	0.00538
EN	7.50109	0.00665	21.06206	0.00256	0.01099	0.00094	12.29468	0.01229	0.00408	0.43432	0.00537
NNG	7.43319	0.00661	21.15658	0.00252	0.01091	0.00090	12.09557	0.01181	0.00405	0.43726	0.00529
Mean	11.86207	0.00694	43.57187	0.00272	0.01208	0.00100	11.30395	0.01637	0.00404	0.50859	0.00547
Boosting	7.43920	0.00678	21.03143	0.00238	0.01109	0.00096	12.69504	0.01209	0.00405	0.43523	0.00533
PLS	**0.000002**	**0.000000**	**0.000022**	**0.000000**	**0.000000**	**0.000000**	**0.000218**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	7.23853	0.00650	21.25086	0.00228	0.01053	0.00088	12.15803	0.01182	0.00374	0.43393	0.00523

Note: The data set spans from January 1960 to May 2009 (Monthly), with total 144 macroeconomic series as X_t to predict the 11 forecasting variables listed on the first row. I set P, out-of-sample period, as 300 where $T = R + P = 260 + 300 = 560$. The first column lists 16 forecasting models that include benchmark, factor, hybrid, and supervised methods. The best MSFE models, for a given Y_t and h, are in bold.

Table 3.A.7: MSFE results with ER estimator

Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
Recursive, h=1											
AR	6.01647	0.00640	25.93355	0.00263	0.01100	0.00032	1.83508	0.00984	0.00351	0.46105	0.00664
ARX(SIC)	6.32510	0.00655	20.41634	0.00183	0.01005	0.00040	1.43406	0.00956	0.00382	0.45194	0.00517
CADL	6.81039	0.00718	19.09536	0.00254	0.00990	0.00070	10.22033	0.01125	0.00557	0.42074	0.00620
FAAR	7.42923	0.00683	21.18732	0.00252	0.01057	0.00102	19.23398	0.01200	0.00491	0.44850	0.00566
PCR	6.44224	0.00710	18.78987	0.00188	0.00997	0.00032	1.39578	0.01089	0.00322	0.41025	0.00609
Bagging	6.47147	0.00718	18.75646	0.00188	0.01005	0.00032	1.40781	0.01082	0.00320	0.41004	0.00605
BMA1	6.59677	0.00650	25.75852	0.00263	0.01107	0.00037	1.99765	0.01056	0.00351	0.46516	0.00712
BMA2	6.19257	0.00642	25.83739	0.00261	0.01101	0.00034	1.96996	0.01006	0.00351	0.46819	0.00690
Rridge	6.05004	0.00644	25.91318	0.00261	0.01097	0.00033	1.88514	0.00992	0.00351	0.46538	0.00676
LAR	6.46096	0.00716	18.72497	0.00190	0.01004	0.00032	1.37314	0.01096	0.00320	0.40510	0.00629
EN	6.26889	0.00704	18.73345	0.00191	0.01003	0.00029	1.32334	0.01066	0.00320	0.40036	0.00609
NNG	6.22577	0.00699	33.96986	0.00191	0.01001	0.00029	1.32728	0.01063	0.00321	0.40475	0.00611
Mean	6.28665	0.00664	23.36673	0.00190	0.00883	0.00055	5.25414	0.01412	0.00393	0.45773	0.00528
Boosting	6.07403	0.00642	25.80672	0.00263	0.01100	0.00033	1.86727	0.00993	0.00351	0.46105	0.00670
PLS	**0.000016**	**0.000000**	**0.000231**	**0.000000**	**0.000000**	**0.000000**	**0.032064**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.02043	0.00640	19.51682	0.00194	0.00964	0.00032	1.98508	0.00983	0.00287	0.41981	0.00557
Recursive, h=3											
AR	6.32542	0.00650	21.73429	0.00239	0.01097	0.00042	3.31169	0.01076	0.00420	0.43161	0.00525
ARX(SIC)	6.84166	0.00604	21.60795	0.00254	0.01064	0.00046	2.55763	0.01074	0.00357	0.43824	0.00524
CADL	6.84350	0.00711	21.02610	0.00280	0.01046	0.00075	11.27095	0.01133	0.00607	0.44135	0.00615
FAAR	7.47966	0.00686	21.52738	0.00257	0.01048	0.00109	19.52803	0.01275	0.00454	0.44392	0.00536
PCR	6.61697	0.00675	21.90197	0.00245	0.01085	0.00045	2.93278	0.01134	0.00350	0.44595	0.00546
Bagging	6.41930	0.00646	22.12916	0.00247	0.01105	0.00044	2.91104	0.01057	0.00351	0.43465	0.00522
BMA1	7.04061	0.00697	21.77508	0.00236	0.01092	0.00059	4.28911	0.01292	0.00415	0.43494	0.00590
BMA2	6.68496	0.00674	21.71088	0.00235	0.01086	0.00050	4.01391	0.01180	0.00414	0.43640	0.00559
Rridge	6.41795	0.00659	21.65800	0.00236	0.01088	0.00046	3.52223	0.01113	0.00415	0.43638	0.00537
LAR	6.52504	0.00677	21.60384	0.00252	0.01110	0.00047	2.85730	0.01135	0.00350	0.43820	0.00549
EN	6.28073	0.00657	21.57041	0.00254	0.01116	0.00039	2.56143	0.01041	0.00357	0.43438	0.00524
NNG	6.33036	0.00658	21.52676	0.00267	0.01112	0.00039	2.58154	0.01045	0.00400	0.43630	0.00527
Mean	6.41029	0.00645	32.76804	0.00236	0.01000	0.00062	6.78013	0.01389	0.00412	0.49195	0.00525
Boosting	6.39164	0.00656	21.73429	0.00239	0.01098	0.00046	3.48502	0.01124	0.00420	0.43161	0.00533
PLS	**0.000008**	**0.000000**	**0.000006**	**0.000000**	**0.000000**	**0.000000**	**0.017241**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.26161	0.00633	21.52808	0.00226	0.01031	0.00044	3.66339	0.01040	0.00352	0.43539	0.00520
Recursive, h=12											
AR	7.52761	0.00669	21.02375	0.00248	0.01102	0.00092	12.45944	0.01197	0.00406	0.43427	0.00530
ARX(SIC)	7.79336	0.00644	21.25878	0.00255	0.01079	0.00092	12.19288	0.01224	0.00409	0.43422	0.00531
CADL	7.31029	0.00721	20.92354	0.00283	0.01050	0.00096	16.04111	0.01213	0.00644	0.44127	0.00621
FAAR	7.50524	0.00666	21.60042	0.00266	0.01071	0.00110	18.25008	0.01222	0.00424	0.43539	0.00524
PCR	7.95423	0.00663	21.63112	0.00256	0.01076	0.00099	12.03773	0.01244	0.00385	0.43684	0.00530
Bagging	7.84308	0.00646	21.47633	0.00257	0.01087	0.00075	11.84091	0.01170	0.00386	0.43155	0.00522
BMA1	8.73941	0.00696	21.06157	0.00248	0.01097	0.00135	14.70127	0.01367	0.00390	0.43451	0.00531
BMA2	8.09715	0.00684	21.09583	0.00247	0.01093	0.00112	14.04921	0.01286	0.00394	0.43541	0.00532
Rridge	7.69084	0.00675	21.06892	0.00248	0.01095	0.00101	12.79921	0.01233	0.00394	0.43605	0.00533
LAR	8.00979	0.00673	21.28872	0.00258	0.01091	0.00112	12.65974	0.01268	0.00389	0.43628	0.00534
EN	7.59289	0.00662	21.33511	0.00256	0.01095	0.00092	12.21581	0.01209	0.00406	0.43487	0.00531
NNG	7.63267	0.00662	31.68031	0.02193	0.01261	0.00092	12.25493	0.01213	0.01527	0.51687	0.00539
Mean	7.93966	0.00676	34.18830	0.00263	0.01151	0.00105	15.48067	0.01645	0.00392	0.49202	0.00536
Boosting	7.77314	0.00674	21.02375	0.00249	0.01102	0.00105	12.75416	0.01231	0.00397	0.43427	0.00530
PLS	**0.000000**	**0.000000**	**0.000062**	**0.000000**	**0.000000**	**0.000000**	**0.000088**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	7.41374	0.00647	21.36235	0.00246	0.01059	0.00091	12.72923	0.01201	0.00377	0.43483	0.00520

Notes: See notes to Table 6.

Table 3.A.8: MSFE results with GR estimator

Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
Recursive, h=1											
AR	6.01647	0.00640	25.93355	0.00263	0.01100	0.00032	1.83508	0.00984	0.00351	0.46105	0.00664
ARX(SIC)	6.32510	0.00655	20.41634	0.00183	0.01005	0.00040	1.43406	0.00956	0.00382	0.45194	0.00517
CADL	6.81039	0.00718	19.09536	0.00254	0.00990	0.00070	10.22033	0.01125	0.00557	0.42074	0.00620
FAAR	6.81148	0.00636	21.45579	0.00252	0.01051	0.00084	11.98468	0.01113	0.00506	0.44231	0.00548
PCR	6.37288	0.00687	19.23184	0.00186	0.00986	0.00032	1.45983	0.01044	0.00317	0.40602	0.00601
Bagging	6.70147	0.00718	18.80973	0.00188	0.00997	0.00035	1.50719	0.01093	0.00313	0.40642	0.00611
BMA1	6.52786	0.00630	25.76117	0.00261	0.01094	0.00035	2.00052	0.01028	0.00351	0.46014	0.00700
BMA2	6.10816	0.00621	25.80104	0.00261	0.01086	0.00032	1.96164	0.00975	0.00351	0.46034	0.00676
Rridge	6.00552	0.00638	25.91223	0.00260	0.01096	0.00032	1.86136	0.00976	0.00350	0.46351	0.00671
LAR	6.46204	0.00716	18.72428	0.00190	0.01004	0.00032	1.37301	0.01097	0.00321	0.40509	0.00629
EN	6.26893	0.00704	18.73289	0.00191	0.01003	0.00029	1.32347	0.01066	0.00320	0.40037	0.00609
NNG	6.22467	0.00700	33.98330	0.00191	0.00996	0.00029	1.32844	0.01063	0.00319	0.40192	0.00608
Mean	13.06306	0.00654	28.87022	0.00188	0.00904	0.00042	1.33308	0.01610	0.00333	0.45710	0.00510
Boosting	6.07962	0.00625	25.88908	0.00264	0.01098	0.00032	1.86309	0.00985	0.00350	0.46013	0.00669
PLS	**0.000021**	**0.000000**	**0.000303**	**0.000000**	**0.000000**	**0.000000**	**0.036434**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.12917	0.00628	19.49962	0.00194	0.00961	0.00030	1.72299	0.00975	0.00285	0.41519	0.00552
Recursive, h=3											
AR	6.32542	0.00650	21.73429	0.00239	0.01097	0.00042	3.31169	0.01076	0.00420	0.43161	0.00525
ARX(SIC)	6.84166	0.00604	21.60795	0.00254	0.01064	0.00046	2.55763	0.01074	0.00357	0.43824	0.00524
CADL	6.84350	0.00711	21.02610	0.00280	0.01046	0.00075	11.27095	0.01133	0.00607	0.44135	0.00615
FAAR	6.82003	0.00668	21.73459	0.00260	0.01045	0.00093	14.00064	0.01184	0.00469	0.44705	0.00523
PCR	6.37089	0.00661	22.07339	0.00249	0.01072	0.00044	3.00225	0.01104	0.00357	0.44936	0.00537
Bagging	6.61658	0.00667	22.41086	0.00246	0.01110	0.00051	3.09148	0.01161	0.00355	0.43681	0.00550
BMA1	6.84005	0.00681	21.80477	0.00238	0.01085	0.00053	4.24730	0.01225	0.00419	0.43229	0.00579
BMA2	6.54725	0.00658	21.77099	0.00239	0.01083	0.00044	3.97330	0.01115	0.00418	0.43189	0.00548
Rridge	6.30478	0.00654	21.63434	0.00236	0.01087	0.00044	3.43193	0.01084	0.00415	0.43508	0.00530
LAR	6.52502	0.00677	21.60170	0.00252	0.01111	0.00047	2.84525	0.01134	0.00350	0.43828	0.00548
EN	6.28141	0.00657	21.57032	0.00254	0.01116	0.00039	2.56226	0.01042	0.00357	0.43440	0.00524
NNG	6.27848	0.00653	21.52832	0.00268	0.01114	0.00039	2.57633	0.01032	0.00401	0.43366	0.00521
Mean	9.45296	0.00642	41.71967	0.00245	0.01052	0.00048	2.54505	0.01392	0.00364	0.52031	0.00517
Boosting	6.29448	0.00652	21.77854	0.00239	0.01091	0.00045	3.48778	0.01099	0.00422	0.43054	0.00528
PLS	**0.000013**	**0.000000**	**0.000024**	**0.000000**	**0.000000**	**0.000000**	**0.020566**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.08945	0.00626	21.40153	0.00227	0.01031	0.00041	3.33461	0.01019	0.00350	0.43256	0.00516
Recursive, h=12											
AR	7.52761	0.00669	21.02375	0.00248	0.01102	0.00092	12.45944	0.01197	0.00406	0.43427	0.00530
ARX(SIC)	7.79336	0.00644	21.25878	0.00255	0.01079	0.00092	12.19288	0.01224	0.00409	0.43422	0.00531
CADL	7.31029	0.00721	20.92354	0.00283	0.01050	0.00096	16.04111	0.01213	0.00644	0.44127	0.00621
FAAR	7.17484	0.00656	21.74582	0.00251	0.01051	0.00092	14.47367	0.01134	0.00428	0.43644	0.00523
PCR	7.49932	0.00657	21.59088	0.00239	0.01046	0.00087	11.16717	0.01155	0.00385	0.43748	0.00530
Bagging	7.97716	0.00659	21.68526	0.00247	0.01060	0.00073	11.54274	0.01181	0.00387	0.43985	0.00524
BMA1	8.48049	0.00683	21.06880	0.00235	0.01092	0.00118	13.81293	0.01269	0.00391	0.43432	0.00524
BMA2	7.87365	0.00669	21.06309	0.00235	0.01095	0.00096	13.25233	0.01199	0.00393	0.43496	0.00525
Rridge	7.58278	0.00672	21.06773	0.00247	0.01092	0.00099	12.68884	0.01216	0.00394	0.43474	0.00533
LAR	7.96242	0.00673	21.28872	0.00258	0.01091	0.00111	12.65974	0.01268	0.00390	0.43628	0.00534
EN	7.59295	0.00662	21.33508	0.00256	0.01095	0.00092	12.21938	0.01209	0.00406	0.43487	0.00531
NNG	7.51066	0.00658	31.67042	0.02192	0.01265	0.00090	12.13772	0.01185	0.01526	0.51680	0.00535
Mean	13.23713	0.00690	45.50544	0.00264	0.01209	0.00096	10.89283	0.01662	0.00393	0.51411	0.00532
Boosting	7.44333	0.00668	21.02375	0.00236	0.01097	0.00093	11.24012	0.01146	0.00395	0.43427	0.00528
PLS	**0.000002**	**0.000000**	**0.000079**	**0.000000**	**0.000000**	**0.000000**	**0.000176**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	7.26267	0.00643	21.50420	0.00240	0.01053	0.00084	11.77505	0.01160	0.00375	0.43517	0.00518

Notes: See notes to Table 6.

Table 3.A.9: MSFE results with ED estimator

Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
Recursive, h=1											
AR	6.01647	0.00640	25.93355	0.00263	0.01100	0.00032	1.83508	0.00984	0.00351	0.46105	0.00664
ARX(SIC)	6.32510	0.00655	20.41634	0.00183	0.01005	0.00040	1.43406	0.00956	0.00382	0.45194	0.00517
CADL	6.81039	0.00718	19.09536	0.00254	0.00990	0.00070	10.22033	0.01125	0.00557	0.42074	0.00620
FAAR	7.75134	0.00698	19.62084	0.00230	0.01009	0.00105	19.74857	0.01249	0.00506	0.44368	0.00582
PCR	6.64691	0.00717	18.00800	0.00186	0.00989	0.00033	1.45782	0.01099	0.00327	0.40613	0.00614
Bagging	6.52502	0.00717	19.01031	0.00186	0.00995	0.00033	1.45431	0.01077	0.00321	0.40832	0.00608
BMA1	6.58072	0.00655	24.71888	0.00261	0.01107	0.00038	2.14192	0.01034	0.00352	0.46195	0.00720
BMA2	6.41176	0.00657	24.93104	0.00260	0.01104	0.00037	2.12273	0.01018	0.00353	0.46710	0.00716
Rridge	6.09639	0.00647	25.08653	0.00258	0.01092	0.00033	1.89846	0.00985	0.00353	0.46085	0.00680
LAR	6.36523	0.00714	18.14595	0.00190	0.01005	0.00031	1.36452	0.01070	0.00323	0.40031	0.00624
EN	6.28206	0.00703	18.66203	0.00191	0.01003	0.00029	1.32553	0.01066	0.00320	0.40026	0.00608
NNG	6.23227	0.00705	25.16222	0.00190	0.01003	0.00029	1.32956	0.01062	0.00325	0.40032	0.00612
Mean	7.17828	0.00662	19.86525	0.00206	0.00926	0.00069	15.12445	0.01445	0.00409	0.45219	0.00528
Boosting	6.16230	0.00643	25.05047	0.00263	0.01102	0.00033	1.87446	0.00977	0.00351	0.45851	0.00668
PLS	**0.000014**	**0.000000**	**0.000170**	**0.000000**	**0.000000**	**0.000000**	**0.028494**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.12283	0.00644	18.90183	0.00195	0.00968	0.00033	2.29756	0.00982	0.00290	0.41670	0.00558
Recursive, h=3											
AR	6.32542	0.00650	21.73429	0.00239	0.01097	0.00042	3.31169	0.01076	0.00420	0.43161	0.00525
ARX(SIC)	6.84166	0.00604	21.60795	0.00254	0.01064	0.00046	2.55763	0.01074	0.00357	0.43824	0.00524
CADL	6.84350	0.00711	21.02610	0.00280	0.01046	0.00075	11.27095	0.01133	0.00607	0.44135	0.00615
FAAR	7.56666	0.00701	21.38425	0.00236	0.01002	0.00111	20.15209	0.01316	0.00485	0.43842	0.00561
PCR	6.73796	0.00690	21.55360	0.00240	0.01065	0.00047	3.19792	0.01186	0.00369	0.44117	0.00577
Bagging	6.53165	0.00661	22.35081	0.00248	0.01082	0.00046	3.10555	0.01089	0.00357	0.43430	0.00540
BMA1	7.39455	0.00721	21.67579	0.00227	0.01085	0.00064	5.14542	0.01390	0.00432	0.43240	0.00623
BMA2	7.28723	0.00714	21.60532	0.00227	0.01079	0.00060	4.92663	0.01350	0.00432	0.43483	0.00620
Rridge	6.48921	0.00666	21.39602	0.00231	0.01076	0.00047	3.61115	0.01137	0.00422	0.43334	0.00546
LAR	6.45756	0.00669	21.64952	0.00250	0.01112	0.00044	2.78725	0.01108	0.00362	0.43629	0.00542
EN	6.27160	0.00655	21.61732	0.00253	0.01117	0.00039	2.57263	0.01038	0.00358	0.43433	0.00524
NNG	6.30439	0.00660	21.30513	0.00252	0.01110	0.00039	2.57022	0.01045	0.00407	0.43472	0.00528
Mean	7.00336	0.00631	23.90331	0.00217	0.00976	0.00073	15.67506	0.01398	0.00420	0.44931	0.00532
Boosting	6.46886	0.00661	21.73429	0.00235	0.01098	0.00046	3.69448	0.01138	0.00424	0.43161	0.00537
PLS	**0.000007**	**0.000000**	**0.000008**	**0.000000**	**0.000000**	**0.000000**	**0.015501**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.32547	0.00639	21.23703	0.00222	0.01026	0.00045	4.03873	0.01057	0.00358	0.43245	0.00526
Recursive, h=12											
AR	7.52761	0.00669	21.02375	0.00248	0.01102	0.00092	12.45944	0.01197	0.00406	0.43427	0.00530
ARX(SIC)	7.79336	0.00644	21.25878	0.00255	0.01079	0.00092	12.19288	0.01224	0.00409	0.43422	0.00531
CADL	7.31029	0.00721	20.92354	0.00283	0.01050	0.00096	16.04111	0.01213	0.00644	0.44127	0.00621
FAAR	7.51748	0.00692	21.23799	0.00242	0.01037	0.00119	19.50051	0.01347	0.00416	0.44049	0.00538
PCR	7.86311	0.00690	21.18798	0.00253	0.01076	0.00110	13.01051	0.01358	0.00381	0.44192	0.00545
Bagging	7.63792	0.00668	22.16873	0.00267	0.01104	0.00082	12.52297	0.01269	0.00388	0.43414	0.00523
BMA1	9.16130	0.00757	20.89296	0.00243	0.01097	0.00182	20.13507	0.01620	0.00388	0.43628	0.00532
BMA2	9.14021	0.00759	20.86946	0.00240	0.01094	0.00177	19.87539	0.01624	0.00385	0.43724	0.00536
Rridge	7.64490	0.00686	20.70073	0.00242	0.01090	0.00108	13.33031	0.01279	0.00387	0.43623	0.00539
LAR	7.85385	0.00664	21.24638	0.00252	0.01093	0.00103	12.30767	0.01226	0.00399	0.43471	0.00533
EN	7.59065	0.00662	21.25774	0.00255	0.01095	0.00092	12.20328	0.01207	0.00409	0.43424	0.00531
NNG	7.60126	0.00664	26.29487	0.00253	0.01099	0.00094	12.18731	0.01224	0.00402	0.43687	0.00540
Mean	7.93966	0.00676	34.18830	0.00263	0.01151	0.00105	15.48067	0.01645	0.00392	0.49202	0.00536
Boosting	7.72542	0.00689	21.03054	0.00249	0.01102	0.00111	13.88304	0.01285	0.00397	0.43427	0.00530
PLS	**0.000001**	**0.000000**	**0.000013**	**0.000000**	**0.000000**	**0.000000**	**0.000137**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	7.32760	0.00653	21.20567	0.00233	0.01056	0.00093	12.86081	0.01231	0.00368	0.43492	0.00522

Notes: See notes to Table 6.

Table 3.A.10: MSFE results with GCV estimator

Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
Recursive, h=1											
AR	6.01647	0.00640	25.93355	0.00263	0.01100	0.00032	1.83508	0.00984	0.00351	0.46105	0.00664
ARX(SIC)	6.32510	0.00655	20.41634	0.00183	0.01005	0.00040	1.43406	0.00956	0.00382	0.45194	0.00517
CADL	6.81039	0.00718	19.09536	0.00254	0.00990	0.00070	10.22033	0.01125	0.00557	0.42074	0.00620
FAAR	7.25588	0.00647	20.09121	0.00230	0.01028	0.00082	6.19255	0.01179	0.00469	0.45181	0.00588
PCR	6.86619	0.00690	18.69430	0.00189	0.01000	0.00039	1.53261	0.01090	0.00314	0.42224	0.00608
Bagging	6.96982	0.00718	19.25276	0.00187	0.00981	0.00046	1.62188	0.01190	0.00308	0.40849	0.00603
BMA1	6.50125	0.00627	24.97246	0.00260	0.01100	0.00036	2.21106	0.00943	0.00331	0.46718	0.00686
BMA2	6.43462	0.00626	25.04126	0.00260	0.01099	0.00036	2.21076	0.00935	0.00330	0.46951	0.00682
Rridge	6.01268	0.00637	25.00934	0.00259	0.01099	0.00031	1.87685	0.00953	0.00347	0.46185	0.00669
LAR	6.32654	0.00707	18.07647	0.00190	0.01002	0.00030	1.34765	0.01064	0.00320	0.39898	0.00618
EN	6.26707	0.00703	18.57563	0.00191	0.01003	0.00029	1.32516	0.01067	0.00320	0.39951	0.00607
NNG	6.23623	0.00699	18.42869	0.00190	0.00999	0.00029	1.33771	0.01055	0.00315	0.40026	0.00610
Mean	9.72605	0.00651	28.21763	0.00195	0.00912	0.00045	1.38529	0.01590	0.00371	0.47392	0.00529
Boosting	6.09573	0.00631	25.06456	0.00261	0.01102	0.00032	1.88140	0.00943	0.00333	0.45721	0.00672
PLS	**0.000032**	**0.000000**	**0.000138**	**0.000000**	**0.000000**	**0.000000**	**0.036096**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.03658	0.00626	18.90306	0.00194	0.00959	0.00030	1.66329	0.00953	0.00278	0.41698	0.00553
Recursive, h=3											
AR	6.32542	0.00650	21.73429	0.00239	0.01097	0.00042	3.31169	0.01076	0.00420	0.43161	0.00525
ARX(SIC)	6.84166	0.00604	21.60795	0.00254	0.01064	0.00046	2.55763	0.01074	0.00357	0.43824	0.00524
CADL	6.84350	0.00711	21.02610	0.00280	0.01046	0.00075	11.27095	0.01133	0.00607	0.44135	0.00615
FAAR	7.14151	0.00669	23.50582	0.00248	0.01044	0.00086	8.64967	0.01279	0.00488	0.46570	0.00579
PCR	6.87827	0.00673	23.74844	0.00250	0.01091	0.00049	3.24384	0.01248	0.00388	0.46742	0.00604
Bagging	6.73097	0.00740	24.07297	0.00240	0.01061	0.00071	3.37871	0.01350	0.00367	0.44101	0.00629
BMA1	7.09148	0.00668	21.58036	0.00244	0.01122	0.00046	5.16264	0.01200	0.00432	0.43914	0.00602
BMA2	7.17627	0.00662	21.99129	0.00245	0.01116	0.00046	5.15540	0.01205	0.00430	0.43960	0.00606
Rridge	6.30450	0.00647	21.20795	0.00234	0.01084	0.00040	3.51418	0.01072	0.00424	0.43185	0.00526
LAR	6.42233	0.00664	21.61359	0.00251	0.01111	0.00043	2.68708	0.01085	0.00365	0.43530	0.00540
EN	6.29079	0.00655	21.62100	0.00253	0.01117	0.00039	2.55939	0.01039	0.00358	0.43419	0.00526
NNG	6.31423	0.00650	21.78195	0.00255	0.01122	0.00039	2.57252	0.01033	0.00359	0.43382	0.00530
Mean	8.21350	0.00633	39.96991	0.00245	0.01040	0.00050	2.67249	0.01389	0.00400	0.50263	0.00527
Boosting	6.47873	0.00654	22.18960	0.00237	0.01105	0.00044	3.72695	0.01149	0.00421	0.43094	0.00553
PLS	**0.000026**	**0.000000**	**0.000026**	**0.000000**	**0.000000**	**0.000000**	**0.020302**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.15848	0.00628	21.29924	0.00226	0.01031	0.00040	3.30410	0.01039	0.00355	0.43306	0.00532
Recursive, h=12											
AR	7.52761	0.00669	21.02375	0.00248	0.01102	0.00092	12.45944	0.01197	0.00406	0.43427	0.00530
ARX(SIC)	7.79336	0.00644	21.25878	0.00255	0.01079	0.00092	12.19288	0.01224	0.00409	0.43422	0.00531
CADL	7.31029	0.00721	20.92354	0.00283	0.01050	0.00096	16.04111	0.01213	0.00644	0.44127	0.00621
FAAR	7.81270	0.00681	22.26425	0.00263	0.01087	0.00105	15.84247	0.01282	0.00455	0.44546	0.00564
PCR	7.92898	0.00687	21.99345	0.00254	0.01094	0.00102	13.35102	0.01290	0.00417	0.44654	0.00570
Bagging	7.52654	0.00672	23.35133	0.00256	0.01098	0.00104	10.68486	0.01253	0.00383	0.43440	0.00527
BMA1	8.70257	0.00729	20.95637	0.00237	0.01100	0.00145	20.63582	0.01315	0.00406	0.43514	0.00525
BMA2	8.64804	0.00731	21.02016	0.00236	0.01099	0.00143	20.59054	0.01331	0.00404	0.43627	0.00526
Rridge	7.49785	0.00673	20.76322	0.00243	0.01091	0.00095	13.26538	0.01206	0.00396	0.43549	0.00537
LAR	7.66963	0.00667	21.08653	0.00253	0.01097	0.00102	12.47747	0.01245	0.00399	0.43448	0.00538
EN	7.50109	0.00665	21.06206	0.00256	0.01099	0.00094	12.29468	0.01229	0.00408	0.43432	0.00537
NNG	7.43319	0.00661	21.15658	0.00252	0.01091	0.00090	12.09557	0.01181	0.00405	0.43726	0.00529
Mean	11.86207	0.00694	43.57187	0.00272	0.01208	0.00100	11.30395	0.01637	0.00404	0.50859	0.00547
Boosting	7.43920	0.00678	21.03143	0.00238	0.01109	0.00096	12.69504	0.01209	0.00405	0.43523	0.00533
PLS	**0.000002**	**0.000000**	**0.000022**	**0.000000**	**0.000000**	**0.000000**	**0.000218**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	7.23823	0.00650	21.25485	0.00228	0.01053	0.00088	12.16092	0.01180	0.00374	0.43389	0.00523

Notes: See notes to Table 6.

Table 3.A.11: MSFE results with CL estimator

Method	UR	PI	TB	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
Recursive, h=1											
AR	6.01647	0.00640	25.93355	0.00263	0.01100	0.00032	1.83508	0.00984	0.00351	0.46105	0.00664
ARX(SIC)	6.32510	0.00655	20.41634	0.00183	0.01005	0.00040	1.43406	0.00956	0.00382	0.45194	0.00517
CADL	6.81039	0.00718	19.09536	0.00254	0.00990	0.00070	10.22033	0.01125	0.00557	0.42074	0.00620
FAAR	7.65883	0.00634	19.64651	0.00258	0.01080	0.00072	4.50174	0.01156	0.00436	0.47106	0.00584
PCR	7.21429	0.00677	18.83189	0.00198	0.01024	0.00040	1.65269	0.01082	0.00308	0.44324	0.00606
Bagging	7.06441	0.00703	19.42780	0.00190	0.00974	0.00044	1.88422	0.01280	0.00306	0.40977	0.00602
BMA1	6.46284	0.00624	24.66451	0.00261	0.01111	0.00037	2.19350	0.00956	0.00328	0.46832	0.00686
BMA2	6.49424	0.00627	25.13570	0.00262	0.01112	0.00037	2.21760	0.00950	0.00329	0.47106	0.00683
Rridge	6.01207	0.00638	24.96197	0.00259	0.01101	0.00031	1.85003	0.00950	0.00348	0.46230	0.00673
LAR	6.32654	0.00707	18.07647	0.00190	0.01002	0.00030	1.34765	0.01064	0.00320	0.39898	0.00618
EN	6.26355	0.00703	18.58874	0.00191	0.01003	0.00029	1.32476	0.01067	0.00320	0.39955	0.00607
NNG	6.25083	0.00699	18.43036	0.00191	0.01002	0.00029	1.34140	0.01053	0.00314	0.40066	0.00611
Mean	13.12719	0.00669	29.31811	0.00191	0.00913	0.00041	1.33319	0.01652	0.00323	0.50326	0.00520
Boosting	6.08626	0.00625	24.88826	0.00262	0.01109	0.00032	1.90426	0.00942	0.00331	0.45757	0.00671
PLS	**0.000022**	**0.000000**	**0.000168**	**0.000000**	**0.000000**	**0.000000**	**0.036781**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.19256	0.00622	18.72398	0.00194	0.00963	0.00030	1.63244	0.00957	0.00275	0.41307	0.00552
Recursive, h=3											
AR	6.32542	0.00650	21.73429	0.00239	0.01097	0.00042	3.31169	0.01076	0.00420	0.43161	0.00525
ARX(SIC)	6.84166	0.00604	21.60795	0.00254	0.01064	0.00046	2.55763	0.01074	0.00357	0.43824	0.00524
CADL	6.84350	0.00711	21.02610	0.00280	0.01046	0.00075	11.27095	0.01133	0.00607	0.44135	0.00615
FAAR	7.31467	0.00674	23.87128	0.00277	0.01089	0.00074	6.84030	0.01309	0.00443	0.47926	0.00586
PCR	7.11771	0.00679	24.04249	0.00263	0.01112	0.00050	3.43636	0.01297	0.00367	0.48247	0.00615
Bagging	6.81990	0.00730	25.25260	0.00236	0.01103	0.00071	4.09277	0.01367	0.00366	0.44591	0.00623
BMA1	7.12599	0.00663	21.51404	0.00249	0.01150	0.00046	5.09558	0.01217	0.00413	0.44298	0.00595
BMA2	7.18842	0.00666	22.08970	0.00250	0.01152	0.00047	5.18187	0.01242	0.00415	0.43879	0.00611
Rridge	6.29623	0.00647	21.05696	0.00232	0.01082	0.00040	3.40512	0.01070	0.00421	0.43295	0.00530
LAR	6.42233	0.00664	21.61359	0.00251	0.01110	0.00043	2.68708	0.01085	0.00365	0.43529	0.00540
EN	6.28773	0.00655	21.62047	0.00253	0.01117	0.00039	2.55785	0.01039	0.00358	0.43429	0.00526
NNG	6.32980	0.00650	21.91931	0.00255	0.01128	0.00039	2.57605	0.01035	0.00356	0.43387	0.00532
Mean	9.46298	0.00670	41.40617	0.00242	0.01037	0.00045	2.38887	0.01427	0.00360	0.64995	0.00525
Boosting	6.52535	0.00656	22.34868	0.00241	0.01112	0.00044	3.81323	0.01149	0.00409	0.43643	0.00550
PLS	**0.000014**	**0.000000**	**0.000031**	**0.000000**	**0.000000**	**0.000000**	**0.021243**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	6.19835	0.00627	21.38016	0.00224	0.01036	0.00040	3.23454	0.01047	0.00344	0.43511	0.00530
Recursive, h=12											
AR	7.52761	0.00669	21.02375	0.00248	0.01102	0.00092	12.45944	0.01197	0.00406	0.43427	0.00530
ARX(SIC)	7.79336	0.00644	21.25878	0.00255	0.01079	0.00092	12.19288	0.01224	0.00409	0.43422	0.00531
CADL	7.31029	0.00721	20.92354	0.00283	0.01050	0.00096	16.04111	0.01213	0.00644	0.44127	0.00621
FAAR	7.72813	0.00687	23.05951	0.00263	0.01081	0.00081	13.14432	0.01196	0.00437	0.44621	0.00591
PCR	7.79466	0.00697	22.84112	0.00246	0.01073	0.00081	12.20396	0.01201	0.00407	0.44785	0.00600
Bagging	7.55733	0.00665	23.30543	0.00252	0.01112	0.00088	11.09928	0.01273	0.00419	0.43643	0.00524
BMA1	8.61414	0.00734	20.86058	0.00235	0.01093	0.00149	19.17361	0.01404	0.00397	0.43388	0.00532
BMA2	8.59307	0.00728	20.88311	0.00233	0.01092	0.00151	19.17948	0.01401	0.00394	0.43418	0.00530
Rridge	7.50775	0.00678	20.69533	0.00246	0.01093	0.00103	13.36515	0.01269	0.00380	0.43675	0.00540
LAR	7.69714	0.00670	21.08400	0.00253	0.01089	0.00103	12.34471	0.01229	0.00398	0.43674	0.00536
EN	7.54091	0.00667	21.06237	0.00256	0.01091	0.00094	12.13013	0.01212	0.00407	0.43652	0.00534
NNG	7.40826	0.00661	21.13969	0.00254	0.01090	0.00090	12.07815	0.01182	0.00392	0.43787	0.00531
Mean	13.23893	0.00705	44.93888	0.00268	0.01195	0.00091	10.48048	0.01723	0.00385	0.59934	0.00542
Boosting	7.42609	0.00680	20.84673	0.00233	0.01096	0.00087	12.42225	0.01178	0.00385	0.43521	0.00535
PLS	**0.000003**	**0.000000**	**0.000026**	**0.000000**	**0.000000**	**0.000000**	**0.000284**	**0.000000**	**0.000000**	**0.000000**	**0.000000**
CFPC	7.18371	0.00650	21.11683	0.00223	0.01041	0.00082	11.72415	0.01168	0.00363	0.43472	0.00526

Notes: See notes to Table 6.

References

Seung C Ahn and Alex R Horenstein. Eigenvalue ratio test for the number of factors. *Econometrica*, 81(3):1203–1227, 2013.

Jushan Bai and Yuan Liao. Efficient estimation of approximate factor models via penalized maximum likelihood. *Journal of Econometrics*, 191(1):1–18, 2016.

Jushan Bai and Serena Ng. Determining the number of factors in approximate factor models. *Econometrica*, 70(1):191–221, 2002.

Jushan Bai and Serena Ng. Forecasting economic time series using targeted predictors. *Journal of Econometrics*, 146(2):304–317, 2008.

Jushan Bai and Serena Ng. Boosting diffusion indices. *Journal of Applied Econometrics*, 24(4):607–629, 2009.

Eric Bair, Trevor Hastie, Debashis Paul, and Robert Tibshirani. Prediction by supervised principal components. *Journal of the American Statistical Association*, 101(473):119–137, 2006.

Jean Boivin and Serena Ng. Are more data always better for factor analysis? *Journal of Econometrics*, 132(1):169–194, 2006.

Leo Breiman. Better subset regression using the nonnegative garrote. *Technometrics*, 37(4):373–384, 1995.

Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

Giovanni Caggiano, George Kapetanios, and Vincent Labhard. Are more data always better for factor analysis? results for the euro area, the six largest euro area countries and the uk. *Journal of Forecasting*, 30(8):736–752, 2011.

Marine Carrasco and Barbara Rossi. In-sample inference and forecasting in misspecified factor models. *Journal of Business & Economic Statistics*, 34(3): 313–338, 2016.

Raymond B Cattell. The scree test for the number of factors. *Multivariate behavioral research*, 1(2):245–276, 1966.

Yeung Lewis Chan, James H Stock, and Mark W Watson. A dynamic factor

model framework for forecast combination. *Spanish Economic Review*, 1(2): 91–121, 1999.

Hugh Chipman, Edward I George, Robert E McCulloch, Merlise Clyde, Dean P Foster, and Robert A Stine. The practical implementation of bayesian model selection. *Lecture Notes-Monograph Series*, pages 65–134, 2001.

In Choi. Efficient estimation of factor models. *Econometric Theory*, 28(2):274–308, 2012.

R Dennis Cook, Liliana Forzani, et al. Principal fitted components for dimension reduction in regression. *Statistical Science*, 23(4):485–501, 2008.

Valentina Corradi and Norman R Swanson. Predictive density and conditional confidence interval accuracy tests. *Journal of Econometrics*, 135(1):187–228, 2006.

Valentina Corradi and Norman R Swanson. A survey of recent advances in forecast accuracy comparison testing, with an extension to stochastic dominance. In *Recent Advances and Future Directions in Causality, Prediction, and Specification Analysis*, pages 121–143. Springer, 2013.

David Roxbee Cox. Notes on some aspects of regression analysis. *Journal of the Royal Statistical Society. Series A (General)*, pages 265–279, 1968.

Russell Davidson. Testing for restricted stochastic dominance: Some further results. 2009.

Sijmen De Jong. Simpls: an alternative approach to partial least squares regression. *Chemometrics and intelligent laboratory systems*, 18(3):251–263, 1993.

Sijmen De Jong and Henk AL Kiers. Principal covariates regression: part i. theory. *Chemometrics and Intelligent Laboratory Systems*, 14(1-3):155–164, 1992.

Ard HJ de Reijer, Jan PAM Jacobs, Pieter W Otter, et al. A criterion for the number of factors in a data-rich environment. Technical report, University of Groningen, Research Institute SOM (Systems, Organisations and Management), 2014.

Ard HJ den Reijer. Forecasting dutch gdp and inflation using alternative factor

model specifications based on large and small datasets. *Empirical Economics*, 44(2):435–453, 2013.

Francis X Diebold and Robert S Mariano. Comparing predictive accuracy. *Journal of Business & economic statistics*, 20(1):134–144, 2002.

Pedro Domingos. Bayesian averaging of classifiers and the overfitting problem. In *ICML*, volume 2000, pages 223–230, 2000.

Bradley Efron, Trevor Hastie, Iain Johnstone, Robert Tibshirani, et al. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

Carmen Fernandez, Eduardo Ley, and Mark FJ Steel. Benchmark priors for bayesian model averaging. *Journal of Econometrics*, 100(2):381–427, 2001.

Yoav Freund and Robert E Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. In *European conference on computational learning theory*, pages 23–37. Springer, 1995.

Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). *The annals of statistics*, 28(2):337–407, 2000.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

Wenjiang J Fu. Penalized regressions: the bridge versus the lasso. *Journal of computational and graphical statistics*, 7(3):397–416, 1998.

Paul H Garthwaite. An interpretation of partial least squares. *Journal of the American Statistical Association*, 89(425):122–127, 1994.

Sarah Gelper and Christophe Croux. Least angle regression for time series forecasting with many predictors. 2008.

Raffaella Giacomini and Halbert White. Tests of conditional predictive ability. *Econometrica*, 74(6):1545–1578, 2006.

Alessandro Giovannelli and Tommaso Proietti. On the selection of common factors for macroeconomic forecasting. 2015.

Clive WJ Granger and Mark J Machina. Forecasting and decision theory. *Handbook of economic forecasting*, 1:81–98, 2006.

Jan JJ Groen and George Kapetanios. Revisiting useful approaches to data-rich macroeconomic forecasting. *Computational Statistics & Data Analysis*, 100: 221–239, 2016.

Christiaan Heij, Patrick JF Groenen, and Dick van Dijk. Forecast comparison of principal component regression and principal covariate regression. *Computational statistics & data analysis*, 51(7):3612–3625, 2007.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Huiyu Huang and Tae-Hwy Lee. To combine forecasts or to combine information? *Econometric Reviews*, 29(5-6):534–570, 2010.

Jan PAM Jacobs, Pieter W Otter, and Ard HJ den Reijer. Information, data dimension and factor structure. *Journal of Multivariate Analysis*, 106:80–91, 2012.

Sainan Jin, Valentina Corradi, and Norman R Swanson. Robust forecast comparison. *Econometric Theory*, pages 1–46, 2016.

Hyun Hak Kim and Norman R Swanson. Mining big data using parsimonious factor and shrinkage methods. 2013.

Hyun Hak Kim and Norman R Swanson. Forecasting financial and macroeconomic variables using data reduction methods: New empirical evidence. *Journal of Econometrics*, 178:352–367, 2014.

Lindsey Klecan, Raymond McFadden, and Daniel McFadden. A robust test for stochastic dominance. Technical report, working paper, Department of Economics, MIT, 1991.

Gary Koop and Simon Potter. Forecasting in dynamic factor models using bayesian model averaging. *The Econometrics Journal*, 7(2):550–565, 2004.

Sungkyung Lee. Stochastic dominance based evaluation of factor and shrinkage based prediction models. *Working Paper*, 2017.

Ker-Chau Li. Asymptotic optimality of cl and generalized cross-validation in ridge regression with application to spline smoothing. *The Annals of Statistics*, pages 1101–1112, 1986.

Ker-Chau Li. Asymptotic optimality for cp, cl, cross-validation and generalized cross-validation: discrete index set. *The Annals of Statistics*, pages 958–975, 1987.

Oliver Linton, Esfandiar Maasoumi, and Yoon-Jae Whang. Consistent testing for stochastic dominance under general sampling schemes. *The Review of Economic Studies*, 72(3):735–765, 2005.

MJ Machina and CWJ Granger. Decision-based forecast evaluation. *Manuscript, Department of Economics, University of California, San Diego*, 2005.

Alexei Onatski. Determining the number of factors from empirical distribution of eigenvalues. *The Review of Economics and Statistics*, 92(4):1004–1016, 2010.

Dimitris N Politis and Joseph P Romano. The stationary bootstrap. *Journal of the American Statistical association*, 89(428):1303–1313, 1994.

James H Stock and Mark W Watson. Macroeconomic forecasting using diffusion indexes. *Journal of Business & Economic Statistics*, 20(2):147–162, 2002.

James H Stock and Mark W Watson. Implications of dynamic factor models for var analysis. Technical report, National Bureau of Economic Research, 2005.

James H Stock and Mark W Watson. Introduction to econometrics. 2007.

James H Stock and Mark W Watson. Generalized shrinkage methods for forecasting using many predictors. *Journal of Business & Economic Statistics*, 30 (4):481–493, 2012.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

Sander Tideman and Allan Timmermann. Economic forecasting. *Journal of Economic Literature*, 46(1):3–56, 2008.

Halbert White. A reality check for data snooping. *Econometrica*, 68(5):1097–1126, 2000.

Herman Wold. Soft modelling: the basic design and some extensions. *Systems under indirect observation, Part II*, pages 36–37, 1982.

Ming Yuan and Yi Lin. On the non-negative garrotte estimator. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(2):143–161, 2007.

Arnold Zellner. On assessing prior distributions and bayesian regression analysis with g-prior distributions. *Bayesian inference and decision techniques: Essays in Honor of Bruno De Finetti*, 6:233–243, 1986.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(2):301–320, 2005.