Mining Big Data Using Parsimonious Factor and Shrinkage Methods

Hyun Hak Kim[1] and Norman R. Swanson[2]

[1]Bank of Korea and [2]Rutgers University

July 2013

Abstract

A number of recent studies in the economics literature have focused on the usefulness of factor models in the context of prediction using "big data" (see Bai and Ng (2008), Dufour and Stevanovic (2010), Forni et al. (2000, 2005), Kim and Swanson (2013), Stock and Watson (2002b, 2006, 2012), and the references cited therein). In this paper, our over-arching question is whether such "big data" are useful for modelling low frequency macroeconomic variables such as unemployment, inflation and GDP. In particular, we analyze the predictive benefits associated with the use dimension reducing independent component analysis (ICA) and sparse principal component analysis (SPCA), coupled with a variety of other factor estimation as well as data shrinkage methods, including bagging, boosting, and the elastic net, among others. We do so by carrying out a forecasting "horse-race", involving the estimation of 28 different baseline model types, each constructed using a variety of specification approaches, estimation approaches, and benchmark econometric models; and all used in the prediction of 11 key macroeconomic variables relevant for monetary policy assessment. In many instances, we find that various of our benchmark specifications, including autoregressive (AR) models, AR models with exogenous variables, and (Bayesian) model averaging, do not dominate more complicated nonlinear methods, and that using a combination of factor and other shrinkage methods often yields superior predictions. For example, simple averaging methods are mean square forecast error (MSFE) "best" in only 9 of 33 key cases considered. This is rather surprising new evidence that model averaging methods do not necessarily yield MSFE-best predictions. However, in order to "beat" model averaging methods, including arithmetic mean and Bayesian averaging approaches, we have introduced into our "horse-race" numerous complex new models involve combining complicated factor estimation methods with interesting new forms of shrinkage. For example, SPCA yields MSFE-best prediction models in many cases, particularly when coupled with shrinkage. This result provides strong new evidence of the usefulness of sophisticated factor based forecasting, and therefore, of the use of "big data" in macroeconometric forecasting.

Keywords: prediction, independent component analysis, sparse principal component analysis, bagging, boosting, Bayesian model averaging, ridge regression, least angle regression, elastic net and non-negative garotte.

JEL Classification: C32, C53, G17.

---

[1] Hyun Hak Kim (khdouble@bok.or.kr), The Bank of Korea, 55 Namdaemunno, Jung-Gu, Seoul 100-794, Korea.

[2]corresponding author: Norman R. Swanson (nswanson@econ.rutgers.edu), Department of Economics, Rutgers University, 75 Hamilton Street, New Brunswick, NJ 08901, USA.

The authors owe many thank to Nii Armah, Valentina Corradi, David Hendry, Gary Koop, John Landon-Lane, Fuchun Li, Greg Tkacz, Hiroki Tsurumi and Halbert White for useful comments made on earlier drafts of this paper. Additional thanks are owed to participants at the Sir Clive W.J. Granger Memorial Conference held at Nottingham University in May 2010, the 2012 Korean International Economics Association Conference, the 2013 Eastern Economic Association Conference, the 2013 International Symposium of Forecasting and at seminars at the Bank of Canada, the Bank of Korea, and Rutgers University. The views stated herein are those of authors and are not necessarily those of the Bank of Korea.

1 Introduction

In macroeconomics and financial economics, researchers benefit from the availability of "big data", in the sense that the plethora of information currently available to applied practitioners certainly cannot worsen previous achievements in the area of macroeconometric forecasting, and may indeed improve upon them. Our over-arching question in this paper is whether such "big data" are useful for modelling low frequency macroeconomic variables such as unemployment, inflation and GDP. We begin our examination of this question by noting that available datasets are sometimes so large as to make dimension reduction an important consideration, both in empirical as well as theoretical contexts. One dimension reduction technique, involving the construction of diffusion indices, has received considerable attention in the recent econometrics literature, particularly in the context of forecasting (see e.g. Armah and Swanson (2010a,b), Artis et al. (2005), Bai and Ng (2002, 2006b, 2008), Boivin and Ng (2005, 2006), Ding and Hwang (1999), Stock and Watson (2002a, 2005, 2006, 2012)). Other recent important papers which extend the discussion in the above papers to vector and error-correction type models include Banerjee and Marcellino (2008), Dufour and Stevanovic (2010).

Our effort at connecting the current discussion of "big data" in economics with the extant literature on diffusion index forecasting is based on an examination of a number of novel factor estimation methods within the framework of diffusion index forecasting. In particular, we analyze the empirical usefulness of independent component analysis (ICA) and sparse principal component analysis (SPCA), coupled with a variety of other factor estimation as well as data shrinkage methods, including bagging, boosting, least angle regression, the elastic net, and the nonnegative garotte. We do so by carrying out a large number of real-time out-of-sample forecasting experiments; and our venue for this "horse-race" is the prediction of 11 key macroeconomic variables relevant for monetary policy assessment. These variables include the unemployment, personal income, 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; and as noted in Kim and Swanson (2013) are discussed on the Federal Reserve Bank of New York's website, where it is stated that "In formulating the nation's monetary policy, the Federal Reserve considers a number of factors, including the economic and financial indicators[1], as well as the anecdotal reports compiled in the Beige Book."

The notion of a diffusion index is to use appropriately "distilled" latent common factors extracted from a large number of variables as inputs in the specification of subsequent parsimonious (yet "information rich") models. More specifically, let X be an $T \times N$-dimensional matrix of observations, and define an $T \times r$-dimensional matrix of dynamic factors, F. Namely, let

$$X = F\Lambda' + e \tag{1}$$

where e is a disturbance matrix and Λ is an $N \times r$ coefficient matrix. Once F is extracted using one of the estimation methods examined in this paper, we construct the following forecasting model based on Stock and Watson (2002a,b), Bai and Ng (2006a) and Kim and Swanson (2013):

$$Y_{t+h} = W_t\beta_W + F_t\beta_F + \varepsilon_{t+h}, \tag{2}$$

[1] See below for a list of 11 of these indicators.

where Y_t, is the target variable to be predicted, h is the prediction horizon, W_t is a $1 \times s$ vector of "additional" explanatory variables, and F_t is a $1 \times r$ vector of factors, extracted from F. The parameters, β_W and β_F, are defined conformably, and ε_{t+h} is a disturbance term. In empirical contexts such as that considered herein, we first estimate r unobserved (latent) factors, say $\hat{F}$, from the N observable predictors, X. To achieve useful dimension reduction, r is assumed to be much less than N, (i.e. $r << N$) Then, parameter estimates, $\hat{\beta}_W$ and $\hat{\beta}_F$ are constructed using an in-sample dataset with Y_{t+h}, W_t, and $\hat{F}_t$. Finally, ex-ante forecasts based on rolling or recursive estimation schemes are formed.

In Kim and Swanson (2013), principal component analysis (PCA) is used in obtaining estimates of the latent factors, called principal components. PCA yields "uncorrelated" latent principal components via the use of data projection in the direction of the maximum variance; and principal components (PCs) are naturally ordered in terms of their variance contribution. The first PC defines the direction that captures the maximum variance possible, the second PC defines the direction of maximum variance in the remaining orthogonal subspace, and so forth. Perhaps because derivation of PCs is easily done via use of singular value decompositions, it is the most frequently used method in factor analysis (see e.g. Bai and Ng (2002, 2006b) and Stock and Watson (2002a) for details). In this paper, we additionally implement two novel new methods for estimating latent factors, including ICA and SPCA. These nonlinear methods are used in the statistics discipline in a variety of contexts. However, economists have yet to explore their usefulness in forecasting contexts, to the best of our knowledge. ICA (see e.g. Comon (1994) and Lee (1998)) uses so-called "negentropy", which is a measure of entropy, to construct independent factors. SPCA is designed to uncover *uncorrelated* components and ultimately factors, just like PCA. However, the method also searches for components whose factor loading coefficient matrices are "sparse" (i.e., the matrices can contain zeros). Since PCA yields nonzero loadings for entire set of variables, practical interpretation thereof is difficult, and estimation efficiency may become an issue. Because it allows for "sparsity", SPCA addresses these issues, leading to the estimation of more parsimonious latent factors than PCA or ICA (for further discussion, see Vines (2000), Jolliffe et al. (2003), and Zou et al. (2006)).

In order to add functional flexibility to our forecasting models, we additionally implement versions of (2) where the numbers and functions of factors used are specified via implementation of a variety of shrinkage methods, including boosting, bagging, and related methods (as discussed above). The key feature of our shrinkage methods is that they are used for targeted regressor and factor selection. Related research that focuses on shrinkage and related forecast combination methods is discussed in Stock and Watson (2012), Aiolfi and Timmermann (2006), and Bai and Ng (2008); and our discussion of shrinkage is meant to add to the recent work reported in Stock and Watson (2012) and Kim and Swanson (2013), who survey and analyze several methods for shrinkage that are based on factor augmented autoregression models of the variety given in equation (2). Finally, in our experiments, we also consider various linear benchmark forecasting models including autoregressive (AR) models, AR models with exogenous variables, and combined autoregressive distributed lag models.

Our findings can be summarized as follows. In many instances, simple benchmark approaches based on the use of various AR type models, including Bayesian model averaging, do not dominate more complicated nonlinear methods that involve the use of factors, particularly when the factors are constructed using nonlinear estimation methods including ICA and SPCA. For example, simple averaging methods are mean square forecast error "best" (MSFE-best) in

only 9 of 33 key cases considered. This is rather surprising new evidence that model averaging methods do not necessarily yield MSFE-best predictions. However, in order to "beat" model averaging methods, including arithmetic mean and Bayesian averaging approaches, we have introduced into our "horse-race" numerous complex new models involve combining complicated factor estimation methods with interesting new forms of shrinkage. For example, SPCA yields MSFE-best prediction models in many cases, particularly when coupled with shrinkage. This result provides strong new evidence of the usefulness of sophisticated factor based forecasting, and therefore, of the use of "big data" in macroeconometric forecasting. It is also noteworthy that pure shrinkage-based prediction models are never MSFE-best when not based on the use of factors constructed using either PCA, ICA or SPCA analysis. This result provides strong new evidence of the usefulness of factor based forecasting, although it should be stressed that factor estimation alone does not yield this clear-cut result. Rather, it is usually ICA and SPCA type factor estimation approaches, coupled with shrinkage, that yield the "best" models. Taken together, the above results provide strong new evidence of the usefulness of sophisticated factor based forecasting, and therefore, of the use of "big data" in macroeconometric forecasting. Two additional rather surprising conclusions that we also draw from our empirical investigation include the following. First, recursive estimation window strategies only dominate rolling strategies at the 1-step ahead forecast horizon. Second, including lags in factor model approaches does not generally yield improved predictions.

The rest of the paper is organized as follows. In the next section we provide a survey of dynamic factor models, independent component analysis, and sparse principal component analysis. In Section 3, we survey the robust shrinkage estimation methods used in our prediction experiments. Data, forecasting methods, and baseline forecasting models are discussed in Section 4, and empirical results are presented in Section 5. Concluding remarks are given in Section 6.

2 Diffusion Index Models

Recent forecasting studies using large-scale datasets and pseudo out-of-sample forecasting include: Armah and Swanson (2010a,b), Artis et al. (2005), Boivin and Ng (2005, 2006), Forni et al. (2005), and Stock and Watson (1999, 2002a, 2005, 2006, 2012). Stock and Watson (2006) additionally discuss in some detail the literature on the use of diffusion indices for forecasting. In this section, we begin by outlining the basic factor model framework which we use (see e.g. Stock and Watson (2002a,b) and Kim and Swanson (2013)). Thereafter, we discuss independent component analysis and sparse principal component analysis.

2.1 Factor Models: Basic Framework

Let X_{tj} be the observed datum for the $j-$th cross-sectional unit at time t, for $t = 1, ..., T$ and $j = 1, ..., N$. Recall that we shall consider the following model:

$$X_{tj} = \Lambda_j' F_t + e_{tj}, \tag{3}$$

where F_t is a $r \times 1$ vector of common factors, Λ_j is an $r \times 1$ vector of factor loadings associated with F_t, and e_{tj} is the idiosyncratic component of X_{tj}. The product $\Lambda_j' F_t$ is called the common

component of X_{tj}. This is the dimension reducing factor representation of the data. More specifically, With $r < N$, a factor analysis model has the form

$$\begin{aligned} X_1 &= \lambda_{11}F_1 + \cdots + \lambda_{1r}F_r + e_1 \\ X_2 &= \lambda_{21}F_1 + \cdots + \lambda_{2r}F_r + e_2 \\ &\vdots \\ X_N &= \lambda_{N1}F_1 + \cdots + \lambda_{Nr}F_r + e_N. \end{aligned} \tag{4}$$

Here, F is a vector of $r < N$ underlying latent variables or factors, λ_{ij} is an element of an $N \times r$ matrix, Λ, of factor loadings, and the e are uncorrelated zero-mean disturbances. Many economic analyses fit naturally into the above framework. For example, Stock and Watson (1999) consider inflation forecasting with diffusion indices constructed from a large number of macroeconomic variables. Recall also that our generic forecasting equation is:

$$Y_{t+h} = W_t\beta_W + F_t\beta_F + \varepsilon_{t+h}, \tag{5}$$

where h is the forecast horizon, W_t is a $1 \times s$ vector (possibly including lags of Y), and F_t is a $1 \times r$ vector of factors, extracted from F. The parameters, β_W and β_F are defined conformably, and ε_{t+h} is a disturbance term. Following Bai and Ng (2002, 2006b, 2008, 2009), the whole panel of data $X = (X_1, ..., X_N)$ can be represented as (3). We then estimate the factors, F_t, via principal components analysis, independent component analysis, or sparse principal component analysis. In particular, forecasts of Y_{t+h} based on (5) involve a two step procedure because both the regressors and the coefficients in the forecasting equation are unknown. The data, X_t, are first used to estimate the factors, yielding $\hat{F}_t$. With the estimated factors in hand, we obtain the estimators $\hat{\beta}_F$ and $\hat{\beta}_W$ by regressing Y_{t+h} on $\hat{F}_t$ and W_t. Of note is that if $\sqrt{T}/N \to 0$, then the usual generated regressor problem does not arise, in the sense that least squares estimates of $\hat{\beta}_F$ and $\hat{\beta}_W$ are $\sqrt{T}$ consistent and asymptotically normal (see Bai and Ng (2008)). In this paper, we try different methods for estimating $\hat{\beta}_F$ and then compare the predictive accuracy of the resultant forecasting models.[2]

In the following sections, we begin by introducing ICA and SPCA and underscoring the difference between these methods and PCA. We omit detailed discussion of principal component analysis, given the extensive discussion thereof in the literature (see e.g. Stock and Watson (1999, 2002a, 2005, 2012), Bai and Ng (2002, 2008, 2009), and Kim and Swanson (2013)).[3]

2.2 Independent Component Analysis

Independent Component Analysis (ICA) is of relevance in a variety of disciplines, since it is predicated on the idea of "opening" the black box in which principal components often reside. A few uses of ICA include mobile phone signal processing, brain imaging, voice signal extraction and stock price modeling. In all cases, there is a large set of observed individual signals, and it is assumed that each signal depends on several factors, which are unobserved.

[2] We refer the reader to Stock and Watson (1999, 2002a, 2005, 2012) and Bai and Ng (2002, 2008, 2009) for a detailed explanation of this procedure, and to Connor and Korajczyk (1986, 1988, 1993), Forni et al. (2005) and Armah and Swanson (2010b) for further detailed discussion of generic diffusion index models.

[3] In the sequel, we assume that all variables are standardized, as is customary in this literature.

The starting point for ICA is the very simple assumption that the components, F, are statistically independent in equation (3). The key is the measurement of this independence between components. The method can be graphically depicted as follows:

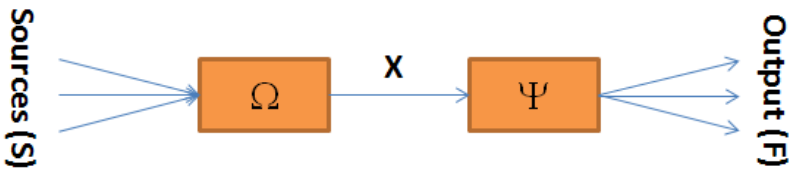

Figure 1: Schematic representation of ICA

More specifically, ICA begins with statistical independent source data, S, which are mixed according to Ω; and X, which is observed, is a mixture of S weighted by Ω. For simplicity, we assume that the unknown mixing matrix, Ω, is square, although this assumption can be relaxed (see Hyvärinen and Oja (2000)). Using matrix notation, we have that

$$X = S\Omega \tag{6}$$

We can rewrite (6) as follows,

$$\begin{aligned} X_1 &= \omega_{11}S_1 + \cdots + \omega_{1N}S_N \\ X_2 &= \omega_{21}S_1 + \cdots + \omega_{2N}S_N \\ &\vdots \\ X_N &= \omega_{1N}S_1 + \cdots + \omega_{NN}S_N, \end{aligned} \tag{7}$$

where ω_{ij} is the (i, j) element of Ω. Since Ω and S are unobserved, we have to estimate the demixing matrix Ψ which transforms the observed X into the independent components, F. That is,

$$F = X\Psi$$

or

$$F = S\Omega\Psi.$$

Since we assume that the mixing matrix, Ω is square, Ψ is also square, and $\Psi = \Omega^{-1}$, so that F is exactly same as S, and perfect separation occurs. In general, it is only possible to find Ψ such that $\Omega\Psi = PD$ where P is a permutation matrix and D is a diagonal scaling matrix. The independent components, F are latent variables, just the same as principal components, meaning that they cannot be directly observed. Also, the mixing matrix, Ω is assumed to be unknown. All we observe is data, X, and we must estimate both Ω and S using it. Only then can we estimate the demixing matrix Ψ, and the independent components, F. However (7) is not identified unless several assumptions are made. The first assumption is that the sources, S, are statistically independent. Since various sources of information (for example, consumer's behavior, political decisions, etc.) may have an impact on the values of macroeconomic variables, this assumption is not strong. The second assumption is that the signals are stationary. For further details, see Tong et al. (1991).

ICA under (7) assumes that N components of F exist. However, we can simply construct factors using up to r ($< N$) components, without loss of generality. In practice, we can construct r independent components by preprocessing with r principal components. See chapter 6 and 10 of Stone (2004) for further details. In general, the above model would be more realistic if there were noise terms added. For simplicity, however, noise terms are omitted; and indeed the estimation of the noise-free model is already computationally difficult (see Hyvärinen and Oja (2000) for a discussion of the noise-free model, and Hyvärinen (1998, 1999a) for a discussion of the model with noise added).

2.2.1 Comparison with Principal Component Analysis

As is evident from Figure 1, ICA is exactly the same as PCA, if we let the demixing matrix be the factor loading coefficients associated with principal components analysis. The key difference between ICA and PCA is in the properties of the factors obtained. Principal components are uncorrelated and have descending variance so that they can easily be ordered in terms of their variances. Moreover, those components explaining the largest share of the variance are often assumed to be the "relevant" ones for subsequent use in diffusion index forecasting. In particular, the first principal component captures the maximum variance possible, the second component also capture the maximum variance but in an orthogonal subspace, and is thus uncorrelated with the first component, and so on.

For simplicity, consider two observables, $X = (X_1, X_2)$. PCA finds a matrix which transforms X into uncorrelated components $F = (F_1, F_2)$, such that the uncorrelated components have a joint probability density function, $p_F(F)$ with

$$E(F_1 F_2) = E(F_1) E(F_2). \tag{8}$$

On the other hand, ICA finds a demixing matrix which transforms the observed $X = (X_1, X_2)$ into independent components $F^* = (F_1^*, F_2^*)$, such that the independent components have a joint pdf $p_{F^*}(F^*)$ with

$$E[F_1^{*p} F_2^{*q}] = E[F_1^{*p}] E[F_2^{*q}], \tag{9}$$

for every positive integer value of p and q. That is, the condition holds for all moments.

Evidently, PCA estimation is much simpler than ICA, since it just involves finding a linear transformation of components which are uncorrelated. Moreover, PCA ranks components using their variances or correlations, so that components associated with higher variance or correlation are assumed to have more explanatory power than those with lower variance or correlation. On the other hand, ICA is unable to find the variance associated with each independent component since both S and Ω in (6) are unknown, so that any scalar multiplier in one of the sources, S_j, could be cancelled by dividing the corresponding mixing vector, ω_j by the same scalar. Therefore, we can randomly change the order of X in (6) so that we cannot determine the order of the independent components. From the perspective of forecasting, this is probably a good thing, since there is no *a priori* reason to believe that "largest variance" PCA components are the most relevant for predicting any particular target variable. Moreover, this feature of ICA is the reason for using PCA for pre-processing in ICA algorithms. For further details about preprocessing, see Appendix F of Stone (2004).

2.2.2 Estimation of ICA

Estimation of independent components is done by estimating the demixing matrix iteratively, systematically increasing the degree of independence of the components. As noted above, uncorrelated components are not independent (except under Gaussianity). However, there is no direct measure for independence. The standard approach is instead to use so-called "nongaussianity" as a measure of independence. In contrast, Gaussian variables cannot produce independent components. This is a straightforward result, since the distribution of any orthogonal transformation of two independent and Gaussian random variables, say X_1 and X_2, has the same distribution as that of X_1 and X_2, in turn implying that the mixing matrix, Ω cannot be identified.

For simplicity, let all independent components have the same distribution. For the first independent component, consider a linear combination of X_j, $j = 1, ..., N$, so that $F_j = X\Psi_j$, where Ψ_j is a vector to be estimated. If Ψ_j were one of the rows of the inverse of Ω, this linear combination would equal one of the independent components. In practice, it is not possible to obtain Ψ_j exactly, since Ω is not observed. Instead, let $\Xi = \Psi\Omega$. Then, we can express F_j as a linear combination of the unobserved source S because

$$F_j = X\Psi_j = S\Omega\Psi_j = S\boldsymbol{\Xi}_j,$$

and combination weights are given by $\boldsymbol{\Xi}_j$. Note also that a sum of two independent random variables is in a concrete sense "more" Gaussian than the original variables, given a central limit theorem. Therefore, $S\boldsymbol{\Xi}_j$ is "more" Gaussian than any of the $S's$. In practice, the objective is to extract Ψ_j as a vector maximizing the nongaussianity of $X\Psi_j$. This in turn implies that $X\Psi_j = S\boldsymbol{\Xi}_j$ is an independent component.

Measuring Nongaussianity In this section we discuss how we measure nongaussianity. The easiest way is via the use of kurtosis.

1. Kurtosis: $kurt(F) = E\left[F^4\right] - 3\left(E\left[F^2\right]\right)^2$, which is zero under Gaussianity. However, this measure is very sensitive to outliers, and so is not particularly useful for measuring nongaussianity.

2. Entropy−Negentropy: Another way of measuring nongaussianity or independence is entropy. The differential entropy H of a random variable F with pdf, p_F is defined as

$$H(F) = -\int p_F(f) \ln p_F(f)\, dF \tag{10}$$

   Note that a moment of a pdf can be expressed as an expectation, and (10) can thus be expressed as

$$H(F) = -E\left[\ln p_F(f)\right]. \tag{11}$$

   A fundamental result of information theory is that a Gaussian variable has the largest entropy among all random variables of equal variance. This supports the use of entropy as a measure of nongaussianity. Moreover, entropy tends to be smaller when the distribution is dense around a certain value. Based on these results, one often uses a modified version

of entropy, so called negentropy, N,where:

$$N(F) = H(F_{gauss}) - H(F). \tag{12}$$

Here, F_{gauss} is a Gaussian random variable with the same covariance matrix as F. The negentropy, $N(\cdot)$, as a measure of nongaussianity, is zero for a Gaussian variable and is always nonnegative. Comon (1994), Hyvärinen (1999b) and Hyvärinen and Oja (2000) note that negentropy has additional interesting properties, including the fact that it is invariant under invertible linear transformations.

3. Mutual Information: This measure is of the amount of information each variable contains about each other variable. Namely, it is the difference between the sum of individual entropies and the joint entropy of two variables, and is defined as follows:

$$I(F) = \sum_{i=1}^{n} H(F_i) - H(F), \tag{13}$$

for n random variables. The quantity $I(F)$ is equivalent to the Kullback-Leibler distance between density $g(F)$ of F and its independence version $\prod_{i=1}^{n} g_i(F_i)$, where $g_i(F_i)$ is the marginal density of F_i. The mutual information becomes zero if the variables are statistically independent. This is somewhat similar to negentropy. If we have an invertible linear transformation $F = X\Psi$, then

$$I(F) = \sum_{i=1}^{n} H(F_i) - H(X) - \ln|\det \Psi| \tag{14}$$

becomes

$$I(F) = \sum_{i=1}^{n} H(F_i) - H(X). \tag{15}$$

Finding Ψ to minimize $I(F) = I(X\Psi)$ involves looking for the orthogonal transformation that leads to the "most" independence between its components; and this is equivalent to minimizing the sum of the entropies of the separate components of F. That is, minimization of mutual information is equivalent to finding directions where negentropy is maximized.

Estimation of Entropy Negentropy is well known and understood in the statistics literature, and is the optimal estimator of nongaussianity in contexts such as that considered here. A classical approximation of negentropy using higher-order moments is the following:

$$N(F) \approx \frac{1}{12} E\left[F^3\right]^2 + \frac{1}{48} kurt(F)^2. \tag{16}$$

Another approximation from Hyvärinen (1998) is based on the maximum-entropy principle, does not explicitly include a measure of kurtosis, and is defined as follows:

$$N(F) \approx \sum_j k_j \left[E\left\{G_j(F)\right\} - E\left\{G_j(Z)\right\}\right]^2, \tag{17}$$

where the k_j are positive constants, Z is a standardized Gaussian variable, F is standardized, and the G_j are nonquadratic functions. Note that (17) can be used consistently, in the sense that it is always non-negative, and equals zero if F has a Gaussian distribution. Simple versions of this approximation use only one nonquadratic function, G, leading to:

$$N(F) \propto \left[E\left\{G(F)\right\} - E\left\{G(\nu)\right\}\right]^2. \tag{18}$$

This equation is a generalization of (16), when F is symmetric. If one sets G as the quartic, (18) becomes (16). Therefore, choosing an appropriate G function is important. If we pick non-fast growing G, we may have more robust estimators. Hyvärinen and Oja (2000) suggest two Gs, and they show that these functions yield good approximations. They are:

$$G_1(y) = \frac{1}{a_1} \log \cosh a_1 y \tag{19}$$

and

$$G_2(y) = -\exp\left(-u^2/2\right), \tag{20}$$

where $1 \leq a_1 \leq 2$ is some suitable constant.

2.2.3 ICA Algorithm: FastICA

ICA implementation involves finding a direction for a unit vector, Ψ_j, such that the component projection matrix, $X\Psi_j$, maximizes nongaussianity. In this paper, we estimate negentropy to measure nongaussianity via the "FastICA" algorithm which efficiently minimizes negentropy. FastICA is a popular ICA algorithm which is based on a fixed-point scheme for tracking maximal nongaussianity of the projection matrix, $X\Psi = \{X\Psi_1, ..., X\Psi_n\}$, where the Ψ_j are the column vectors of Ψ, and are not correlated with each other. Simply put, the algorithm finds a unit vector Ψ_j such that $X\Psi_j$ maximizes nongaussianity, as measured by negentropy (see equation (18)). Note that the variance of $X\Psi_j$ is constrained to be unity, so that the norm of Ψ_j is constrained to be unity, since we use standardized data.

Let g be the derivative of the nonquadratic function G used in (19) and (20). FastICA is the fixed-point algorithm which maximizes (18), and maxima are obtained at optima $E\left\{G(F)\right\} = E\left\{G(X\Psi)\right\}$. Let us explain with one unit for simplicity. Using Kuhn-Tucker conditions, the optima of $E\left\{G(X\Psi_j)\right\}$ under the constraint $E\left\{G(X\Psi_j)^2\right\} = \|\Psi_j\| = 1$ can be obtained at $E\left\{G(X\Psi_j)\right\} - \lambda\Psi_j = 0$. One can solve this equation using the Newton-Raphson method. See Hyvärinen and Oja (2000) for computational details. Thus, we have the following iterative procedure:

$$\Psi^* = \Psi - \frac{\left[E\left\{Xg(X\Psi_j)\right\} - \lambda\Psi_j\right]}{\left[E\left\{g(X\Psi_j)\right\} - \lambda\right]}. \tag{21}$$

Multiplying both sides by $\lambda - E\left\{g'(X\Psi_j)\right\}$ yields

$$\Psi^* = E\left\{Xg\left(X\Psi_j\right)\right\} - E\left\{g'\left(X\Psi_j\right)\right\}\Psi_j. \tag{22}$$

The basic FastICA algorithm is as follows.

1. Choose an initial weight vector Ψ.
2. For $j = 1, ..., r$, find mixing vectors yielding components with minimized negentropy. Let $\Psi_j^* = E\left\{Xg\left(X\Psi_j\right)\right\} - E\left\{g'\left(X\Psi_j\right)\right\}\Psi_j$.
3. Set $\Psi_j^+ = \Psi_j^* / \left\|\Psi_j^*\right\|$. If convergence is not achieved, go back to Step 2.
4. To decorrelate j independent components, for $j \geq 2$, set (a) $\Psi_j^+ = \Psi_j^+ - \Sigma_{h=1}^{j-1}\Psi_j^{+\prime}\Psi_h\Psi_h$ and then (b) $\Psi_j^+ = \Psi_j^+ / \sqrt{\left\|\Psi_j^{+\prime}\Psi_j^+\right\|}$.

The initial vector, Ψ, is given from the loadings of the r ordinary principal components (Penny et al. (2001) and Stone (2004)) Once the final Ψ is estimated, $X\Psi$ are the independent components. In this paper, we choose G as in (19), and accordingly g is defined as $\tanh\left(u\right)$, if we set $a_1 = 1$.

2.3 Sparse Principal Component Analysis

As was explained in the previous section, principal components are linear combinations of variables that are ordered by covariance contributions, and selection is of a small number of components which maximize the variance that is explained. However, factor loading coefficients are all typically nonzero, making interpretation of estimated components difficult. SPCA aids in the interpretation of principal components by placing (zero) restrictions on various factor loading coefficients.

For example, Jolliffe (1995) modifies loadings to be values such as 1, -1 and 0. Another approach is setting thresholds for the absolute value of the loadings, below which loadings are set to zero. Jolliffe et al. (2003) suggest using so-called "SCoTLASS" to construct modified principal components with possible zero loadings, λ, by solving

$$\max \lambda'(X'X)\lambda, \text{ subject to } \sum_{j=1}^{N}|\lambda_j| \leq \varphi, \ \lambda'\lambda = 1,$$

for some tuning parameter, φ. The absolute value threshold results in (various) zero loadings, hence inducing sparseness. However, the SCoTLASS constraint does not ensure convexity, and therefore the approach may be computationally expensive. As an alternative, Zou et al. (2006) develop a regression optimization framework. Namely, they assume that the X are dependent variables, F are explanatory variables, and the loadings are coefficients. They then use the lasso (and elastic net) to derive a sparse loading matrix. Other recent approaches include those discussed in Leng and Wang (2009) and Guo et al. (2010), both of which are based on Zou et al. (2006). We follow the approach of Zou et al. (2006), and readers are referred to Sections 3.3-3.5 of their paper for complete details. As an introduction to the method, the following paragraphs summarize key discussions from their paper.

2.3.1 Estimation of Sparse Principal Components

Suppose that we derive principal components (PCs), F, via ordinary PCA. In particular, our standardized data matrix, X is identical to UDV' by the singular value decomposition. The PCs, F, are defined as UD, and V are the factor coefficient loadings. Then, let the estimated j-th principal component , F_j be the dependent variable and X be independent variables. Suppose that $\hat{\lambda}_j^{Ridge}$ is the ridge estimator[4] of the loading for the j-th principal component. We solve the following problem to obtain the ridge estimator,

$$\hat{\lambda}_j^{Ridge} = \arg\min_{\lambda_j} \|F_j - X\lambda_j\|^2 + \eta \|\lambda_j\|^2 . \tag{23}$$

Note that after normalization, the coefficients are independent of η. Therefore, the ridge penalty term, $\eta \|\lambda_j\|^2$, is not used to penalize the regression coefficients but rather in the construction of the principal components. Add an L_1 penalty to (23) and solve the following optimization problem; namely, solve the so-called naïve elastic net (NEN) (see Section 3.4 for details on the NEN), as follows:

$$\hat{\lambda}_j^{NEN} = \arg\min_{\lambda_j} \|F_j - X\lambda_j\|^2 + \eta \|\lambda_j\| + \eta_1 \|\lambda_j\|_1 , \tag{24}$$

where $\|\lambda_j\|_1 = \sum_{i=1}^{N} |\lambda_{ij}|$. Here, $X\hat{\lambda}_j$ is the j-th principal component. In this problem, large enough η_1 guarantees a sparse λ, and hence a sparse loading matrix. With a fixed value of η, the problem given by equation (24) can be solved using the LARS-EN algorithm[5] proposed by Zou and Hastie (2005). Zou et al. (2006) modify this idea to a more general lasso regression type problem. In particular, they use a two-stage analysis in which they first estimate the principal components by the ordinary PCA, and thereafter find sparse loadings using (24). This type of SPCA is predicated on the fact that PCA can be written as a penalized regression problem[6], and thus the lasso, or the elastic net, can be directly integrated into the regression criterion such that the resulting modified PCA produces sparse loadings.

Continuing the above discussion, note that Zou et al. (2006) suggest using the following penalized regression type criterion. Let X_t denote the t-th row vector of the matrix X. For any positive value of η, let

$$\left(\hat{\delta}_j, \hat{\lambda}_j\right) = \arg\min_{\delta_j, \lambda_j} \sum_{t=1}^{T} \left\|X_t - \delta_j \lambda_j' X_t\right\|^2 + \eta \|\lambda_j\|^2 , \tag{25}$$
$$subject\ to \quad \|\delta_j\|^2 = 1.$$

Then, $\hat{\lambda}_j$ becomes the approximation to the j-th factor loadings, λ_j. If we let λ equal δ, then $\sum_{t=1}^{T} \left\|X_t - \delta_j \lambda_j' X_t\right\|^2 = \sum_{t=1}^{T} \left\|X_t - \delta_j \delta_j' X_t\right\|^2$. Therefore, $\hat{\lambda}(= \hat{\delta})$ becomes the j-th ordinary principal component loading (see Hastie et al. (2009) for details). Equation (25) can be easily extended to derive the whole sequence of PCs. Let there be r components. Set Δ and Λ to be

[4] See Section 3.3 for further details about the ridge estimator.

[5] See Section 3.5 for details about the LARS-EN algorithm.

[6] See Section 3.2 of Kim and Swanson (2013) for a discussion of penalized regression

$N \times r$ matrices. For any positive value of η, let

$$\left(\hat{\Delta}, \hat{\Lambda}\right) = \arg\min_{\Delta,\Lambda} \sum_{t=1}^{T} \|X_t - \Delta\Lambda' X_t\|^2 + \eta \sum_{j=1}^{r} \|\lambda_j\|^2 \quad (26)$$
$$subject\ to\ \ \Delta'\Delta = I_r.$$

Here, Λ is an $N \times r$ matrix with column λ_j and Δ is also an $N \times r$ orthonormal constraint, so that $\hat{\lambda}_j$ is the approximation to the j-th factor loadings, λ_j, for $j = 1, ..., r$. As we see in the above expression, by setting Δ and Λ to be equal, $\hat{\Lambda}$ becomes the exact r factor loadings of ordinary principal components. Equation (26) is the generalized derivation of principal components and enables us to obtain sparse loadings by modifying the original PCA problem. The penalty parameter in the above expression is applied for all variables, and so we do not yet have sparse loadings, however. To construct sparsity, add a lasso penalty into equation (26), and consider the following penalized regression problem,

$$\left(\hat{\Delta}, \hat{\Lambda}\right) = \arg\min_{\Delta,\Lambda} \sum_{t=1}^{T} \|X_t - \Delta\Lambda' X_t\|^2 + \eta \sum_{j=1}^{r} \|\lambda_j\|^2 + \sum_{j=1}^{r} \eta_{1,j} \|\lambda_j\|_1, \quad (27)$$
$$subject\ to\ \ \Delta'\Delta = I_r.$$

Here, $\hat{\Lambda}$ is the approximation of the factor loadings. This problem has two penalties; the first term, η is applied to all possible r components, and the second term, $\eta_{1,j}$ is applied to individual components to penalize their loadings. As in the estimation of a single component in equation (25), if we set $\Lambda = \Delta$, then we have $\sum_{t=1}^{T} \|X_t - \Delta\Lambda' X_t\|^2 = \sum_{t=1}^{T} \|X_t - \Delta\Delta' X_t\|^2$, and so $\hat{\Lambda}(= \hat{\Delta})$ becomes the ordinary principal component loading matrix. Since equation (27) is not jointly convex for Δ and Λ, two steps to solve this problem are needed. The first one involves fixing Δ, and then minimizing over Λ, which leads to a problem involving r elastic nets. In particular, since Δ is orthonormal, let $\Delta^\dagger$ be any orthonormal matrix such that $\left[\Delta; \Delta^\dagger\right]$ is an $r \times r$ orthonormal matrix. Then we have

$$\begin{aligned}
\sum_{t=1}^{T} \|X_t - \Delta\Lambda' X_t\|^2 &= \|X - X\Lambda\Delta'\|^2 \\
&= \left\|X\Delta^\dagger\right\|^2 + \|X\Delta - X\Lambda\|^2 \\
&= \left\|X\Delta^\dagger\right\|^2 + \sum_{j=1}^{r} \|X\delta_j - X\lambda_j\|^2
\end{aligned}$$

That is, let Δ be given. The optimal solution for Λ is based on minimizing

$$\arg\min_{\Lambda} \sum_{j=1}^{r} \left[\|X\delta_j - X\lambda_j\|^2 + \eta \|\lambda_j\|^2 + \eta_{1,j} \|\lambda_j\|_1\right]. \quad (28)$$

This is equivalent to r independent elastic net problems. If we rewrite (28) for a single loading, we have

$$\hat{\lambda}_j = \arg\min_{\lambda_j} \left\|F_j^* - X\lambda_j\right\|^2 + \eta \|\lambda_j\|^2 + \eta_{1,j} \|\lambda_j\|_1, \quad (29)$$

where $F_j^* = X\delta_j$. Now, (29) is identical to

$$(\delta_j - \lambda_j)' X'X (\delta_j - \lambda_j) + \eta \|\lambda_j\|^2 + \eta_{1,j} \|\lambda_j\|_1 . \tag{30}$$

Here, we only need to calculate the correlation matrix, since we already standardized X. In the end, we solve these elastic nets efficiently via the LARS-EN algorithm discussed below.

The next step involves minimizing (27) over Δ, with fixed Λ. Then penalty term in this problem is now meaningless, and so the problem is solved by minimizing

$$\begin{aligned} &\sum_{t=1}^{T} \|X_t - \Delta\Lambda' X_t\|^2 , \\ &subject\ to\ \ \Delta'\Delta = I_r. \end{aligned} \tag{31}$$

This problem can be solved by the so called "Procrustes" transformation. (see Chapter 14.5 of Hastie et al. (2009) for details). Since $\sum_{t=1}^{T} \|X_t - \Delta\Lambda' X_t\|^2 = \|X - X\Lambda\Delta'\|^2$, using an appropriate transformation, we have the following singular value decomposition

$$X'X\Lambda = UDV',$$

where $\hat{\Delta} = UV'$. In practice, we let Δ be the factor loading matrix associated with ordinary PCs, then we estimate Λ as a sparse factor loading matrix. In this variant of the problem, the LARS-EN algorithm discussed below delivers a whole sequence of sparse approximations for each PC and the corresponding values of $\eta_{1,j}$.

2.3.2 SPCA algorithm

The numerical solution for the SPCA criterion to obtain sparse principal components is given as follows.

1. Let Δ be the loadings of the first r ordinary principal components.
2. Given Δ, solve the following problem for $j = 1, 2, ..., r$.

$$\lambda_j = \arg\min_{\lambda} (\delta_j - \lambda)' X'X (\delta_j - \lambda) + \eta \|\lambda\|^2 + \eta_{1,j} \|\lambda\|_1 .$$

3. For each fixed $\Lambda = [\lambda_1, ..., \lambda_r]$, do the singular vector decomposition on $X'X\Lambda = UDV'$, then update $\Delta^* = UV'$.
4. Repeat steps 2-3, until convergence.

In practice, the choice of η does not change the result very much, particularly in the case where X is a full rank matrix, in which case zero is a reasonable value to use. Additionally, one may try picking η_1 using cross-validation, or a related method. Moreover, the LARS-EN algorithm efficiently solves this problem for all possible values of η_1 (see Zou and Hastie (2005) or Kim and Swanson (2013) for computational details). Since the tuning parameter, η_1, affects

the sparsity and variance of the components simultaneously, the algorithm is designed to give more weight to variance.

Note that if $\hat{F}$, are factors estimated by ordinary PCA, then they are uncorrelated so that we can compute the total variance explained by $\hat{F}$ as $tr\left(\hat{F}'\hat{F}\right)$. However, two conditions for principal components, uncorrelatedness and orthogonality, are not guaranteed in the case of sparse principal components. Still, it is necessary to derive the total variance in order to explain how much the components explain, even when the above two conditions are not satisfied. Zou et al. (2006) proposes a new way to compute the variance explained by the components, accounting for any correlation among the components. Since variance is given as $tr\left(\hat{F}'\hat{F}\right)$ for total variance if sparse principal components are already uncorrelated, this formula can be used more generally to compute the total variance of sparse principal components. Let $\tilde{F} = \left[\tilde{F}_1, ..., \tilde{F}_r\right]$ be the r components constructed via sparse principal component analysis. Denote $\hat{r}_j$ as the residual after regressing $\tilde{F}_j$ on $\tilde{F}_1, ..., \tilde{F}_{j-1}$, so that

$$\hat{r}_j = \tilde{F}_j - \mathbf{P}_{1,...,j-1}\tilde{F}_j,$$

where $\mathbf{P}_{1,...,j-1}$ is the projection matrix on $\tilde{F}_1, ..., \tilde{F}_{j-1}$. Then, the adjusted variance of a single component is $\|\hat{r}_j\|^2$ and the total variance is $\sum_{j=1}^{r} \|\hat{r}_j\|^2$. In practice, computation is easily done by QR factorization. If we let $\tilde{F} = QR$, then $\|\hat{r}_j\|^2 = R_{jj}^2$, so that total variance is $\sum_{j=1}^{r} R_{jj}^2$. Since the above computation is sequential, the order of components matters. However, in the current paper, we derive sparse PCs based on ordinary PCs, which are in turn already ordered by the size of the variance.

3 Robust Estimation Techniques

We consider a variety of "robust" shrinkage techniques in our forecasting experiments. The methods considered include bagging, boosting, ridge regression, least angle regression, the elastic net, the non-negative garotte and Bayesian model averaging. Here, we briefly summarize the shrinkage methods, and provide relevant citations to detailed discussions thereof.

Bagging, which was introduced by Breiman (1996), is a machine based learning algorithm whereby outputs from different predictors of bootstrap samples are combined in order to improve overall forecasting accuracy. Bühlmann and Yu (2002) use bagging in order to improve forecast accuracy when data are *iid*. Inoue and Kilian (2008) and Stock and Watson (2012) extend bagging to time series models. Stock and Watson (2012) consider "bagging" as a form of shrinkage, when constructing prediction models. In this paper, we use the same algorithm that they do when constructing bagging estimators. This allows us to avoid time intensive bootstrap computation done elsewhere in the bagging literature. Boosting, a close relative of bagging, is another statistical learning algorithm, was originally designed for classification problems in the context of Probability Approximate Correct (PAC) learning (see Schapire (1990)), and is implemented in Freund and Schapire (1997) using the algorithm called "AdaBoost.M1". Hastie et al. (2009) apply it to classification, and argue that "boosting" is one of the most powerful learning algorithms currently available. The method has been extended to regression problems

in Ridgeway et al. (1999) and Shrestha and Solomatine (2006). In the economics literature, Bai and Ng (2009) use a boosting for selecting the predictors in factor augmented autoregressions. We implement a boosting algorithm that mirrors that used by these authors.

The other shrinkage methods implemented herein are basically regression with regression coefficient penalization. First, we consider ridge regression, which is a well known linear regression shrinkage method which modifies sum of square residual computations to include a penalty for inclusion of larger numbers of parameters. Conveniently, ridge regression uses a quadratic penalty term, and has a closed form solution. Second, we implement the "least absolute shrinkage and selection operator" (lasso), which was introduced by Tibshirani (1996), and is another attractive technique for variable selection using high-dimensional datasets, especially when N is greater than T. This method doesn't yield a closed form solution, and it needs to be estimated numerically. Third, we examine "Least Angle Regression" (LARS), which is introduced in Efron et al. (2004), and is a method for choosing a linear model using the same set of data as that used to evaluate and implement the model. LARS can be interpreted as the algorithm which finds a solution path for the lasso. Moreover, LARS is based on a well known model-selection approach known as "forward-selection", which has been extensively used to examine cross-sectional data (for further details, see Efron et al. (2004)). Bai and Ng (2008) show how to apply LARS and the lasso in the context of time series data, and Gelper and Croux (2008) extend Bai and Ng (2008)'s work to time series forecasting with many predictors. We implement Gelper and Croux (2008)'s algorithm when constructing the LARS estimator. Fourth, we consider a related method called the "Elastic Net", which is proposed by Zou and Hastie (2005), and which is also similar to the lasso, as it simultaneously carries out automatic variable selection and continuous shrinkage. Its name comes from the notion that it is similar in structure to a stretchable fishing net that retains "all the big fish". LARS-Elastic Net (LARS-EN) is proposed by Zou and Hastie (2005) for computing entire elastic net regularization paths using only a single least squares model, for the case where the number of variables is greater than the number of observations. Bai and Ng (2008) apply the elastic net method to time series using the approach of Zou and Hastie (2005). We also follow their approach when implementing the elastic net. Finally, we consider the so-called, "non-negative garotte", originally introduced by Breiman (1995). This method is a scaled version of the least square estimator with shrinkage factors. Yuan and Lin (2007) develop an efficient garrotte algorithm and prove consistency in variable selection. We follow Yuan and Lin (2007) in the sequel.

In addition to the above shrinkage methods, we consider Bayesian model averaging (henceforth, BMA), as it is one of the most attractive methods for model selection currently available (see Fernandez et al. (2001b), Koop and Potter (2004) and Ravazzolo et al. (2008)). The concept of Bayesian model averaging can be described with simple probability rules. If we consider R different models, each model has a parameter vector and is represented by its prior probability, likelihood function and posterior probability. Given this information, using Bayesian inference, we can obtain model averaging weights based on the posterior probabilities of the alternative models. Koop and Potter (2004) consider BMA in the context of many predictors and evaluate its performance. We follow their approach.

In the following subsections, we explain the intuition behind the above methods, and how they are used in our forecasting framework.

3.1 Bagging

Bagging, which is short for "bootstrap aggregation", was introduced by Breiman (1996) as a device for reducing prediction error in learning algorithms. Bagging involves drawing bootstrap samples from the training sample (i.e. in-sample), applying a learning algorithm (prediction model) to each bootstrap sample, and averaging the predicted values. Consider the regression problem with the training sample $\{Y, X\}$. Generate B bootstrap samples from the dataset and form predictions, $\hat{Y}_b^*(X_b^*)$, say, using each bootstrap sample, $b = 1, ..., B$. Bagging averages these predictions across bootstrap samples in order to reduce prediction variation. In particular, for each bootstrap sample, $\{Y_b^*, X_b^*\}$, regress Y_b^* on X_b^* and construct the fitted value $\hat{Y}_b^*(X_b^*)$. The bagging predictor is defined as follows:

$$\hat{Y}^{Bagging} = \frac{1}{B}\sum_{b=1}^{B}\hat{Y}_b^*(X_b^*) \tag{32}$$

Inoue and Kilian (2008) apply this bagging predictor in a time series context. Bühlmann and Yu (2002) consider bagging with a fixed number of strictly exogenous regressors and *iid* errors, and show that, asymptotically, the bagging estimator can be represented in shrinkage form. Namely:

$$\hat{Y}_{T+h}^{Bagging} = \sum_{j=1}^{N}\psi(\omega_j)\hat{\beta}_j X_{Tj} + o_p(1), \tag{33}$$

where $\hat{Y}_{T+h}^{Bagging}$ is the forecast of Y_{t+h} made using data through time T, $\hat{\beta}_j$ is the least squares estimator of β_j under $Y = X\beta$ and $\omega_j = \sqrt{T}\hat{\beta}_j/s_e$, with $s_e^2 = \Sigma_{t=1}^{T}(Y_{t+h} - X_t\hat{\beta}')^2/(T-N)$, where $\hat{\beta} = \left(\hat{\beta}_1, ..., \hat{\beta}_N\right)'$. Also, ψ is

$$\psi(\omega) = 1 - \Phi(\omega + c) + \Phi(\omega - c) + \omega^{-1}[\phi(\omega - c) - \phi(\omega + c)], \tag{34}$$

where c is the pre-test critical value, ϕ is the standard normal density and Φ is the standard normal CDF.

Now, following Stock and Watson (2012), define the forecasting model using bagging as follows:

$$\hat{Y}_{T+h}^{Bagging} = W_T\hat{\beta}_W + \sum_{j=1}^{r}\psi(\omega_j)\hat{\beta}_{Fj}\hat{F}_{Tj}, \tag{35}$$

where $\hat{\beta}_W$ is the LS estimator of β_W, W_T is a vector of observed variables (e.g. lags of Y) as in (5), and $\hat{\beta}_{Fj}$ is estimated using residuals, $Y_{T+h} - W_T\hat{\beta}_W$. The t-statistics used for shrinkage (i.e. the ω_j) are computed using least squares and Newey-West standard errors. Further, the pretest critical value for bagging in this paper is set at $c = 1.96$.

3.2 Boosting

Boosting (see Freund and Schapire (1997)) is a procedure that combines the outputs of many "weak learners" (models) to produce a "committee" (prediction). In this sense, boosting bears a resemblance to bagging and other "committee-based" shrinkage approaches. Conceptually, the boosting method builds on a user-determined set of many weak learners (for example, least square estimators) and uses the set repeatedly on modified data which are typically outputs

from previous iterations of the algorithm. Typically this output comes from minimizing a loss function averaged over training data. In this sense, boosting has something in common with forward stagewise regression. The final boosted procedure takes the form of linear combinations of weak learners. Freund and Schapire (1997) propose the so-called "adaBoost" algorithm. AdaBoost and other boosting algorithms have attracted a lot of attention due to their success in data modeling.

Friedman et al. (2000) extend AdaBoost to "Real AdaBoost", which focuses on the construction of real-valued predictions. Suppose that we have a training sample of data, (Y, X), and let $\hat{\mu}(X)$ be a function (learner) defined on $\mathbb{R}^n$. Also, let $L(Y_t, \mu(X_t))$ be the loss function that penalizes deviations of $\hat{\mu}(X)$ from Y, at time t. The objective is to estimate $\mu(\cdot)$ to minimize expected loss, $E[L(Y_t, \hat{\mu}(X_t))]$. Popular "learners" include smoothing splines, kernel regressions and least squares. Additionally, in AdaBoost, an exponential loss function is used.

Friedman (2001) introduces "L_2Boosting", which takes the simple approach of refitting base learners to residuals from previous iterations under quadratic loss. Bühlmann and Yu (2003) suggest another boosting algorithm, fitting learners using one predictor at one time when large numbers of predictors exist. Bai and Ng (2009) modify this algorithm to handle time-series. We use their "Component-Wise L_2Boosting" algorithm in the sequel.

Boosting Algorithm Let $Z = Y - \hat{Y}^W$, which is obtained in a first step by fitting an autoregressive model to the target variable using W_t as regressors. Then, using estimated factors:

1. Initialize : $\hat{\mu}^0(F_t) = \bar{Z}$, for each t.
2. For $i = 1, ..., M$ iterations, carry out the following procedure. For $t = 1, ..., T$, let $u_t = Z_t - \hat{\mu}^{i-1}(D_t)$ be the "current residual". For each $j = 1, .., r$, regress the current $T \times 1$ residuals, u on $\hat{F}_j$ (the j-th factor) to obtain $\hat{\beta}_j$.
3. Compute $\hat{d}_j = u - \hat{F}_j \hat{\beta}_j$ for $j = 1, .., r$, and the sum of squared residuals, $SSR_j = \hat{d}_j' \hat{d}_j$. Let j_*^i denote the column selected at the i^{th} iteration, say, such that $SSR_{j_*^i} = \min_{j \in [1,...,r]} SSR_j$, and let $\hat{g}_*^i(F) = \hat{F}_{j_*^i} \hat{\beta}_{j_*^i}$.
4. For $t = 1, ..., T$, update $\hat{\mu}^i = \hat{\mu}^{i-1} + \nu \hat{g}_*^i$, where $0 \leq \nu \leq 1$ is the step length.

Over-fitting may arise if this algorithm is iterated too many times. Therefore, selecting the number of iterations, M is crucial. Bai and Ng (2009) define the stopping parameter M using an information criterion of the form:

$$IC(i) = \log\left[\hat{\sigma}^{i^2}\right] + \frac{A_T \cdot df^i}{T} \tag{36}$$

where $\hat{\sigma}^{i^2} = \Sigma_{t=1}^T \left(Y_t - \hat{\mu}^i\left(\hat{F}_t\right)\right)^2$ and $A_T = \log(T)$. Evidently,

$$M = \arg\min_i IC(i). \tag{37}$$

Here, the degrees of freedom is defined as $df^i = trace(B^i)$, where $B^i = B^{i-1}\nu \mathbf{P}^{(i)}(I_T - B_{i-1}) = I_T - \Pi_{h=0}^i \left(I_T - \nu \mathbf{P}^{(h)}\right)$, with $\mathbf{P}^{(i)} = \hat{F}_{j_*^i}\left(\hat{F}_{j_*^i}' \hat{F}_{j_*^i}\right)^{-1} \hat{F}_{j_*^i}$. Starting values for B^i are given as

$B^0 = \frac{1}{\nu}P^{(0)} = \mathbf{1}_T'\mathbf{1}_T/T$, where $\mathbf{1}_T$ is a $T \times 1$ vector of 1's. Our boosting estimation uses this criterion. Finally, we have:

$$\hat{Y}_{t+h}^{Boosting} = W_t\hat{\beta}_W + \hat{\mu}^M\left(\hat{F}_t\right), \tag{38}$$

where $\hat{\beta}_W$ is defined above.

3.3 Ridge Regression

In the following three subsections, we discuss penalized regression approaches, including ridge regression, least angle regression, the elastic net and the nonnegative garotte. These methods shrink regression coefficients by only retaining a subset of potential predictor variables. Ridge regression, as introduced by Hoerl and Kennard (1970), is the classical penalized regression method, and is introduced here in order to place the methods discussed thereafter in context. Consider explanatory variables that are stacked in an $T \times N$ matrix, and a univariate response or target variable, Y. Coefficients are estimated by minimizing a penalized residual sum of squares criterion. Namely, define:

$$\hat{\beta}^{Ridge} = \arg\min_{\beta}\left[\sum_{t=1}^{T}\left(Y_t - \sum_{i=1}^{N}X_{ti}\beta_i\right)^2 + \eta\sum_{i=1}^{N}\beta_i^2\right], \tag{39}$$

where η is a positive penalty parameter. The larger is η, the more we penalize coefficients, and the smaller the eventual subset of possible predictors that is used. The ridge regression estimator in (39) can be restated in the context of constrained regression, as follows:

$$\hat{\beta}^{Ridge} = \arg\min_{\beta}\left[\sum_{t=1}^{T}\left(Y_t - \sum_{i=1}^{N}X_{ti}\beta_i\right)^2\right], \tag{40}$$

$$\text{subject to } \sum_{i=1}^{N}\beta_i^2 \leq m,$$

where m is a positive number which corresponds to η. (Note that all observable predictors are standardized here, as elsewhere in this paper.) The ridge criterion (39) picks coefficients to minimize the residual sum of squares, and can conveniently be written in matrix form, as follows:

$$RSS(\eta) = (Y - X\beta)'(Y - X\beta) + \eta\beta'\beta, \tag{41}$$

where RSS denotes the residual sum of squares. Thus,

$$\hat{\beta}^{Ridge} = (X'X + \eta\mathbf{I})^{-1}X'Y, \tag{42}$$

where $\mathbf{I}$ is the $N \times N$ identity matrix. In our experiments, we use the following model for forecasting:

$$\hat{Y}_{t+h}^{Ridge} = W_t\hat{\beta}_W + \hat{F}_t\hat{\beta}_F^{Ridge}. \tag{43}$$

Note that there is another penalized regression method that is similar to ridge regression,

which is called the lasso (i.e. least absolute shrinkage selection operator). The key difference between two methods is the penalty function. The lasso estimator is defined as follows:

$$\hat{\beta}^{Lasso} = \arg\min_{\beta} \left[\sum_{t=1}^{T} \left(Y_t - \sum_{i=1}^{N} X_{ti}\beta_i \right)^2 \right], \tag{44}$$

$$\text{subject to} \sum_{i=1}^{N} |\beta_i| \leq m$$

That is, the L_2 ridge penalty is replaced by an L_1 lasso penalty. Accordingly, the lasso does not have a closed form solution like the ridge estimator. Although we report findings based upon ridge regression type models, we no not estimate the lasso, as it can be interpreted as a special case of least angle regression, which is discussed in the next subsection.

3.4 Least Angle Regression (LARS)

Least Angle Regression (LARS) is proposed in Efron et al. (2004), and can be viewed as an application of forward stagewise regression. In forward stagewise regression, predictor sets are constructed by adding one new predictor at a time, based upon the explanatory context of each new candidate predictor in the context of a continually updated least squares estimator. For details, see Efron et al. (2004).

Like many other stagewise regression approaches, start with $\hat{\mu}^0 = \bar{Y}$, the mean of the target variable, use the residuals after fitting W_t to the target variable, and construct a first estimate, $\hat{\mu} = X_t\hat{\beta}$, in stepwise fashion, using standardized data. Define $\hat{\mu}_{\mathcal{G}}$ to be the current LARS estimator, where $\mathcal{G}$ is a set of variables that is incrementally increased according to the relevance of each variable examined. Define $c\left(\hat{\mu}_{\mathcal{G}}\right) = \hat{c} = X'\left(Y - \hat{\mu}_{\mathcal{G}}\right)$, where X is the "current" set of regressors, to be the "current correlation" vector of length N. In particular, define the set $\mathcal{G}$ to be the set including covariates with the largest absolute correlations; so that we can define $\hat{C} = \max_j \{\hat{c}_j\}$ and $\mathcal{G} = \left\{ j : |\hat{c}_j| = \left|\hat{C}\right| \right\}$, by letting $s_j = sign\left(\hat{c}_j\right)$ (i.e. ± 1), for $j \in \mathcal{G}$, and defining the active matrix corresponding to $\mathcal{G}$ as $\mathcal{X}_{\mathcal{G}} = \left(...s_j X_j...\right)_{j\in\mathcal{G}}$. The objective is to find the predictor, X_j, that is most highly correlated with the residual. Let

$$\mathcal{D}_{\mathcal{G}} = \mathcal{X}'_{\mathcal{G}}\mathcal{X}_{\mathcal{G}} \text{ and } A_{\mathcal{G}} = \left(\mathbf{1}'_{\mathcal{G}}\mathcal{D}_{\mathcal{G}}^{-1}\mathbf{1}_{\mathcal{G}}\right)^{-\frac{1}{2}}, \tag{45}$$

where $\mathbf{1}_{\mathcal{G}}$ is a vector of ones equal in length to the rank of $\mathcal{G}$. A unit equiangular vector with columns of $\mathcal{X}_{\mathcal{G}}$ can be defined as $u_{\mathcal{G}} = \mathcal{X}_{\mathcal{G}} w_{\mathcal{G}}$, where $w_{\mathcal{G}} = A_{\mathcal{G}}\mathcal{D}_{\mathcal{G}}^{-1}\mathbf{1}_{\mathcal{G}}$ so that $\mathcal{X}'_{\mathcal{G}}u_{\mathcal{G}} = A_{\mathcal{G}}\mathbf{1}_{\mathcal{G}}$. LARS then updates $\hat{\mu}$ as

$$\hat{\mu}_{\mathcal{G}^+} = \hat{\mu}_{\mathcal{G}} + \hat{\gamma}u_{\mathcal{G}}, \tag{46}$$

where

$$\hat{\gamma} = \min_{j\in\mathcal{G}^c}{}^{+} \left(\frac{\hat{C} - \hat{c}_j}{A_{\mathcal{G}} - a_j} \right) \left(\frac{\hat{C} + \hat{c}_j}{A_{\mathcal{G}} + a_j} \right), \tag{47}$$

with $a_j = X'w_j$ for $j \in \mathcal{G}^c$. Efron et al. (2004) show that the lasso is in fact a special case of LARS that imposes specific sign restrictions. In summary, LARS is a procedure that simply

seeks new predictors that have the highest correlation with the current residual.

In order to apply LARS to time series data, Gelper and Croux (2008) revise the basic algorithm described here. They start by fitting an autoregressive model to the target variable, excluding predictor variables, using least squares. The corresponding residual series is retained and its standardized version is denoted by Z. The time-series LARS (henceforth, TS-LARS) procedure ranks the predictors according to how much they contribute to improving upon autoregressive fit. Using estimated factors as regressors, the following is the "LARS" algorithm of Gelper and Croux (2008):

LARS Algorithm

1. Fit an autoregressive model to the dependent variable without factors and retain the corresponding residuals. The objective is to forecast these residuals. Begin by setting $\hat{\mu}^0 = \hat{\mu}^0\left(\hat{F}\right) = \bar{Z}$, as done in the boosting algorithm above, and using standardized data.

2. For $i = 1, 2, ..., r$:

   (a) Pick j_*^i from $j = 1, 2, ..., r$ $(\leq N)$ which has the highest R^2 value, $R^2\left(\hat{\mu}^{i-1} \sim \hat{F}_j\right)$, where R^2 is a measure of least square regression fit, and where "$\backsim$" denotes horizontal concatenation. The predictor with highest R^2 is denoted $\hat{F}_{(i)} = \hat{F}_{j_*^i}$, and this predictor will be included in the active set $\mathcal{G}^i$. That is, $\hat{F}_{(i)}$ denotes the i^{th} ranked predictor, the active set $\mathcal{G}^i$ will contain $\hat{F}_{(1)}, \hat{F}_{(2)}, ..., \hat{F}_{(i)}$, and j_*^i is excluded in next iteration.

   (b) Denote the matrix corresponding to the i^{th} ranked active predictor by $H_{(i)}$, which is the projection matrix on the space spanned by the columns of $\hat{F}_{(i)}$. That is, $H_{(i)} = \hat{F}_{(i)}\left(\hat{F}'_{(i)}\hat{F}^{-1}_{(i)}\right)\hat{F}'_{(i)}$.

   (c) Let $\tilde{F}_{(i)} = H_{(i)}\hat{\mu}^{i-1}$ be the $T \times 1$ standardized vector of values, $\hat{F}$, at the i^{th} iteration. Then, find the equiangular vector u^i, where $u^i = \left(\tilde{F}_{(1)}, \tilde{F}_{(2)}, ..., \tilde{F}_{(i)}\right) w^i$, $w^i = \frac{\mathcal{D}_{\mathcal{G}^i}^{-1}\mathbf{1}_i}{\sqrt{\mathbf{1}_i'\mathcal{D}_{\mathcal{G}^i}^{-1}\mathbf{1}_i}}$, $\mathcal{D}_{\mathcal{G}^i} = \mathcal{F}'_{\mathcal{G}^i}\mathcal{F}_{\mathcal{G}^i}$, $\mathcal{F}_{\mathcal{G}^i} = \left(...s_j\hat{F}^j...\right)_{j\in\mathcal{G}^i}$, $s_j = sign\left(\hat{c}_j\right)$, and $\hat{c} = \hat{F}'\left(\bar{Z} - \hat{\mu}^i\right)$.

3. (iii) Update the response $\hat{\mu}^i = \hat{\mu}^{i-1} - \hat{\gamma}^i u^i$, where $\hat{\gamma}^i$ is the smallest positive solution for a predictor $\hat{F}_j$ which is not already in the active set, and is defined in (47).Then go back to Step 2, where $\hat{F}_{(i+1)}$ is added to the active set and the new response is standardized and denoted by $\hat{\mu}^{i+1}$ (see Gelper and Croux (2008) for further computational details).

After ranking the predictors, $\hat{F}$, the highest ranked will be included in the final model. Now, the only choice remaining is how many predictors to include in the model. Finally, construct

$$\hat{Y}_{t+h}^{LARS^+} = W_t\hat{\beta}_W + \hat{\mu}^{LARS}(\hat{F}_t) \tag{48}$$

where $\hat{\mu}^{LARS}(\hat{F}_t)$ is the optimal value of the LARS estimator. The final predictor of Y is formed by adding back the mean to $\hat{Y}_{t+h}^{LARS^+}$.

3.5 Elastic Net (EN)

The elastic net (EN) is proposed by Zou and Hastie (2005), who point out various limitations of the lasso. Since it is a modification of the lasso, it can be viewed as a type of LARS, and indeed, their algorithm is sometimes called "LARS-EN". In order to motivate the LARS-EN algorithm, we begin with a generic discussion of the "naïve elastic net" (NEN). Assume again that we are interested in X and Y, and that the variables in X are standardized. For any fixed non-negative η_1 and η_2, the naive elastic net criterion is defined as:

$$L(\eta_1, \eta_2, \beta) = |Y - X\beta|^2 + \eta_2 |\beta|^2 + \eta_1 |\beta|_1, \tag{49}$$

where $|\beta|^2 = \sum_j^N (\beta_j)^2$ and $|\beta|_1 = \sum_j^N |\beta_j|$. The naïve elastic net estimator is $\hat{\beta}^{NEN} = \arg\min_\beta \{L(\eta_1, \eta_2, \beta)\}$. This problem is equivalent to the optimization problem:

$$\hat{\beta}^{NEN} = \arg\min_\beta |Y - X\beta|^2, \quad \text{subject to } (1-\alpha)|\beta|_1 + \alpha|\beta|^2, \tag{50}$$

where $\alpha = \frac{\eta_2}{\eta_1 + \eta_2}$. The term $(1-\alpha)|\beta|_1 + \alpha|\beta|^2$ is called the elastic net penalty, and leads to the lasso or ridge estimator, depending on the value of α. (If $\alpha = 1$, it becomes ridge regression; if $\alpha = 0$, it is the lasso, and if $\alpha \in (0,1)$, it has properties of both methods.) The solution to the naïve elastic involves defining a new dataset (X^+, Y^+), where

$$X^+_{(T+N)\times N} = (1+\eta_2)^{-1/2} \begin{pmatrix} X \\ \sqrt{\eta_2}\mathbf{I}_N \end{pmatrix} \quad \text{and} \quad Y^+_{(T+N)\times 1} = \begin{pmatrix} Y \\ \mathbf{0}_N \end{pmatrix}. \tag{51}$$

Now, rewrite the naive elastic criterion as:

$$L\left(\frac{\eta_1}{\sqrt{1+\eta_2}}, \beta\right) = L\left(\frac{\eta_1}{\sqrt{1+\eta_2}}, \beta^+\right) = \left|Y^+ - D^+\beta^+\right|^2 + \frac{\eta_1}{\sqrt{1+\eta_2}}\left|\beta^+\right|_1. \tag{52}$$

If we let

$$\hat{\beta}^+ = \arg\min_{\beta^+} L\left(\frac{\eta_1}{\sqrt{1+\eta_2}}, \beta^+\right), \tag{53}$$

then the NEN estimator $\hat{\beta}^{NEN}$ is:

$$\hat{\beta}^{NEN} = \frac{1}{\sqrt{1+\eta_2}}\hat{\beta}^+. \tag{54}$$

In this orthogonal setting, the naïve elastic net can be represented as combination of ordinary least squares and the parameters (η_1, η_2). Namely:

$$\hat{\beta}^{NEN} = \frac{\left(\left|\hat{\beta}^{LS}\right| - \eta_1/2\right)_{pos}}{1+\eta_2} sign\left\{\hat{\beta}^{LS}\right\}, \tag{55}$$

where $\hat{\beta}^{LS}$ is the least squares estimator of β and $sign(\cdot)$ equals ± 1. Here, "*pos*" denotes the positive part of the term in parentheses. Using these expressions, the ridge estimator can be

written as

$$\hat{\beta}^{Ridge} = \frac{\hat{\beta}^{LS}}{1+\eta_2} \tag{56}$$

and the lasso estimator is

$$\hat{\beta}^{Lasso} = \left(\left|\hat{\beta}^{LS}\right| - \eta_1/2\right)_{pos} sign\left\{\hat{\beta}^{LS}\right\}. \tag{57}$$

Zou and Hastie (2005), in the context of the above naive elastic net, point out that there is double shrinkage, which does not help to reduce the variance and may lead to unnecessary bias, and they propose the elastic net, in which this double shrinkage is corrected. Given equation (51), the naive elastic net solves the regularization problem of the type:

$$\hat{\beta}^{+} = \arg\min_{\beta^+} \left|Y^+ - X^+\beta^+\right|^2 + \frac{\eta_1}{\sqrt{1+\eta_2}} \left|\beta^+\right|_1. \tag{58}$$

In this context, the elastic net estimator, $\hat{\beta}^{EN}$, is defined as:

$$\hat{\beta}^{EN} = \sqrt{1+\eta_2}\hat{\beta}^{+}. \tag{59}$$

Thus ,

$$\hat{\beta}^{EN} = (1+\eta_2)\,\hat{\beta}^{NEN}. \tag{60}$$

Via this rescaling, the estimator preserves the properties of naive elastic net. Moreover, by Theorem 2 in Zou and Hastie (2005), is can be seen that the elastic net is a stabilized version of the lasso. Namely,

$$\hat{\beta}^{EN} = \arg\min_{\beta}\beta' \left(\frac{X'X + \eta_2\mathbf{I}_N}{1+\eta_2}\right)\beta - 2Y'X\beta + \eta_1\left|\beta\right|_1, \tag{61}$$

which is the estimator that we use in the forecasting model given as (5) when carrying out our prediction experiments.

Zou and Hastie (2005) propose an algorithm called the LARS-EN to estimate $\hat{\beta}^{EN}$ using LARS, as discussed above. With fixed η_2, the elastic net problem is equivalent to the lasso problem on the augmented dataset (X^+, Y^+), where $\mathcal{D}_{\mathcal{G}}$ in (45) is equal to $\frac{1}{1+\eta_2}\left(\mathcal{X}_{\mathcal{G}}'\mathcal{X}_{\mathcal{G}} + \eta_2\mathbf{I}_{\mathcal{G}}\right)$ for any active set $\mathcal{G}$. The LARS-EN algorithm updates the elastic net estimator sequentially.

Choosing tuning parameters, η_1 and η_2, is a critical issue in the current context. Hastie et al. (2009) discuss some popular ways to choose tuning parameters, and Zou and Hastie (2005) use tenfold cross-validation (CV). Since there are two tuning parameters, it is necessary to cross-validate in two dimensions. We do this by picking a small grid of values for η_2, say $(0, 0.01, 0.1, 1, 10, 100)$. LARS-EN selects the η_2 value that yields the smallest CV error. We follow this approach when implementing LARS-EN.

3.6 Non-Negative Garotte (NNG)

The NNG estimator of Breiman (1995) is a scaled version of the least squares estimator. As in the previous section, we begin by considering generic X and Y. Assume that the follow-

ing shrinkage factors are given: $q(\zeta) = (q_1(\zeta), q_2(\zeta), ..., q_N(\zeta))'$. The objective is to choose shrinkage factors in order to minimize:

$$\frac{1}{2}\|Y - Gq\|^2 + T\zeta\sum_{j=1}^{N} q_j, \qquad \text{subject to } q_j > 0,\ j = 1, .., N, \tag{62}$$

where $G = (G_1, .., G_N)'$, $G_j = X_j\widehat{\beta}_j^{LS}$, and $\widehat{\beta}^{LS}$ is the least squares estimator. Here $\zeta > 0$ is the tuning parameter. The NNG estimator of the regression coefficient vector is defined as $\hat{\beta}_j^{NNG}(\zeta) = q_j(\zeta)\hat{\beta}_j^{LS}$, and the estimate of Y is defined as $\widehat{\mu} = X\hat{\beta}^{NNG}(\zeta)$. Assuming, for example, that $X'X = I$, the minimizer of expression (62) has the following explicit form: $q_j(\zeta) = \left(1 - \frac{\zeta}{(\hat{\beta}_j^{LS})^2}\right)_+, \quad j = 1, ..., N$. This ensures that the shrinking factor may be identically zero for redundant predictors. The disadvantage of the NNG is its dependence on the ordinary least squares estimator, which can be especially problematic in small samples. Accordingly, Yuan and Lin (2007) consider lasso, ridge regression, and the elastic net as alternatives for providing an initial estimate for use in the NNG; and they prove that if the initial estimate is consistent, the non-negative garotte is a consistent estimator, given that the tuning parameter, ζ, is chosen appropriately. Zou (2006) shows that the original non-negative garotte with ordinary least squares is also consistent, if N is fixed, as $T \to \infty$. Our approach is to start the algorithm with the least squares estimator, as in Yuan (2007), who outline the following algorithm for the non-negative garotte that we use in the sequel:

Non-Negative Garotte Algorithm

1. First, set $i = 1$, $q^0 = 0$, $\hat{\mu}^0 = \bar{Z}$. Then compute the current active set

$$\mathcal{G}^i = \arg\max_j \left(G_j'\hat{\mu}^{i-1}\right),$$

   where $G_j = \hat{F}_j\hat{\beta}_j$, is the j^{th} element of the $T \times r$ matrix G; and the initial $\hat{\beta}$ is obtained by regressing $\hat{F}$ on Z, using least squares.

2. Compute the current direction γ, which is an r dimensional vector defined by $\gamma_{(\mathcal{G}^i)^c} = 0$ and

$$\gamma_{\mathcal{G}^i} = \left(G_{\mathcal{G}^i}'G_{\mathcal{G}^i}'\right)^{-1} G_{\mathcal{G}^i}'\hat{\mu}^{i-1}.$$

3. For every $j' \notin \mathcal{G}^i$, compute how far the non-negative garotte will progress in direction γ before $\hat{F}_j$ enters the active set. This can be measured by a α_j such that

$$G_{j'}'\left(\hat{\mu}^{i-1} - \alpha_j G'\gamma\right) = G_j'\left(\hat{\mu}^{i-1} - \alpha_j G'\gamma\right)$$

   where j is arbitrarily chosen from $\mathcal{G}^i$. Now, for every $j \in \mathcal{G}^i$, compute $\alpha_j = \min\left(\beta_j, 1\right)$, where $\beta_j = -q_j^{i-1}/\gamma_j$, if nonnegative, measures how far the group non-negative garotte will "progress" before q_j becomes zero.

4. If $\alpha_j \leq 0$, $\forall j$ or $\min_{j,\alpha_j>0}\{\alpha_j\} > 1$, set $\alpha = 1$. Otherwise, denote $\alpha = \min_{j,\alpha_j>0}\{\alpha_j\} \equiv \alpha_{j^*}$ Set $q^i = q^{i-1} + \alpha'\gamma$. If $j^* \notin \mathcal{G}^i$, update $\mathcal{G}^{i+1}$ by adding j^* to the set $\mathcal{G}^i$; else update $\mathcal{G}^{i+1}$ by

taking out j^* from the set $\mathcal{G}^i$.

5. Set $\hat{\mu}^i = Y - G'q^i$ and $i = i + 1$. Go back to Step 1 repeat until $\alpha = 1$, yielding $\hat{\mu}^{final} = \hat{\mu}^{NNG}$. Finally, form

$$\hat{Y}_{t+h}^{NNG^+} = W_t\hat{\beta}_W + \hat{\mu}^{NNG}, \tag{63}$$

and construct the prediction $\hat{Y}_{t+h}^{NNG}$ by adding back the mean to $\hat{Y}_{t+h}^{NNG^+}$.

3.7 Bayesian Model Averaging (BMA)

Bayesian Model Averaging (BMA) has received considerable attention in recent years in the forecasting literature (see e.g. Koop and Potter (2004) and Wright (2008, 2009)) For a concise discussion of BMA, see Hoeting et al. (1999) and Chipman et al. (2001). The basic idea of BMA starts with supposing that interest focuses on Q possible models, denoted by $M_1, ..., M_Q$, say. In forecasting contexts, BMA involves averaging target predictions, Y_{t+h} from the candidate models, with weights appropriately chosen. In a very real sense, thus, it resembles bagging. One might also select a model by choosing M_{q^*} which maximizes $p\left(M_q|Data\right)$, but model averaging is generally preferred. If we denote ω as a particular parameter vector, then BMA begins by noting that:

$$p\left(\omega|Data\right) = \sum_{q=1}^{Q} p\left(\omega|Data, M_q\right) p\left(M_q|Data\right). \tag{64}$$

If $g\left(\omega\right)$ is a function of ω, then without loss of generality, the conditional expectation is given as:

$$E\left[g\left(\omega\right)|Data\right] = \sum_{q=1}^{Q} E\left[g\left(\omega\right)|Data, M_q\right] p\left(M_q|Data\right). \tag{65}$$

This means that we can compute the variance of the parameter for quadratic g. Accordingly, BMA involves obtaining results for all candidate models and averaging them with weights determined by the posterior model probabilities. That is, BMA, puts little weight on implausible models, as opposed to other varieties of shrinkage discussed above that operate directly on regressors. As we have 144 variables in our empirical work, we have 2^{144} possible models. This means that we must estimate more than 10^{43} models at every forecasting horizon, and prior to the construction of each new prediction in this paper. Though there has been a quantum leap in computing technology in recent years, it would take several years to do this. Koop and Potter (2004) use the approach of Clyde (1999) for dealing with this problem, and take posterior draws of the parameters and associated variances using Gibbs sampling. The algorithm they use is somewhat different from the popular Markov Chain Monte Carlo algorithm in that draws are taken directly from the conditional probability of the parameters, given the data and the variance. In this paper, we use the algorithm given in Koop and Potter (2004). However, we additionally require a slightly different setup from that discussed above, in order to handle W_t in (2). Accordingly, we follow Chipman et al. (2001). Specifically, we transform our forecasting framework as follows. Let:

$$Y_{t+h}^* = \beta^* F_t^* + \varepsilon_t^*, \tag{66}$$

where $Y^*_{t+h} = [I_T - W_t (W_t'W_t) W_t'] Y_{t+h}$, $F^*_t = [I_T - W_t (W_t'W_t) W_t'] \hat{F}_t$, W_t, and $\hat{F}_t$ are defined in (5), and $\varepsilon_{t+h} \sim N(0, \sigma^2)$. This setup leads to a natural conjugate prior (i.e. $\beta^*|\sigma^{-2} \sim N\left(\underline{\beta^*}, \sigma^2 \underline{V}\right)$) and $\sigma^{-2} \sim G(\underline{s}^{-2}, \underline{\varpi})$, where $G(\underline{s}^{-2}, \underline{\varpi})$ denotes the gamma distribution with mean $\underline{s}^{-2}$ and degrees of freedom $\underline{\varpi}$. Each candidate model is described with U, which is an $r \times 1$ vector which shows whether each column of explanatory variables is included in current model, with a one or a zero. In this sense, U is similar to the current set in penalized regression. Moreover, U gives the prior model probability, $p(M_q)$, as the prior for U is equivalent to $p(M_q)$. According to Koop and Potter (2004), $p(U|Y^*)$ is drawn directly, since our explanatory variables are orthogonal. We set $p(Y^*|U, \sigma^2)$ to be the marginal likelihood for the normal regression model defined by U, and derive $P(U|Y^*, \sigma^2)$, given a prior, $p(U)$. Here, $p(\sigma^2|Y^*, U)$ takes the inverted-Gamma form as usual. The next step involves specifying the prior model probability, $p(M_q)$ or equivalently, a prior for $p(U)$:

$$p(U) = \prod_{j=1}^{R} \upsilon_j^{U_j} (1 - \upsilon_j)^{U_j}, \tag{67}$$

where υ_j is the prior probability that each potential factor enters the model. A common benchmark case sets $\upsilon_j = \frac{1}{2}$, equivalently, $P(M_q) = \frac{1}{Q}$ for $q = 1, ..., Q$. Other choices are also possible. For example, we could allow υ_j to depend on the j-th largest eigenvalue of $\hat{F}'\hat{F}$.

Using the strategy described in Fernandez et al. (2001a) and Kass and Raftery (1995), we use a noninformative improper prior over parameters for lagged variables in all models; and in particular we follow Koop and Potter (2004), who suggest a noninformative prior for σ^{-2}. Namely, if $\underline{\varpi} = 0$, s^{-2} does not enter the marginal likelihood or posterior. Following Fernandez et al. (2001a), we set $\underline{\beta^*} = \mathbf{0}_R$ and use a g-prior form for $\underline{V}$ by setting

$$\underline{V}_r = [g_r F_r^{*\prime} F_r^*]^{-1} \tag{68}$$

(see Fernandez et al. (2001a) and Zellner (1986) for more details on the use of g-priors). Finally, we are left with the issue of specification of g. Fernandez et al. (2001a) examine the properties of many possible choices for g and Koop and Potter (2004), in an objective Bayesian spirit, focus on values for g including $g = \frac{1}{T}$ and $g = \frac{1}{Q^2}$. We specify the same functions for g. Using the above approach, we form:

$$\hat{Y}^{*,BMA}_{t+h} = \hat{\beta}_F F^*_t \tag{69}$$

and our forecast, $\hat{Y}^{BMA}_{t+h}$ is defined as $[I_T - W_t (W_t'W_t) W_t']^{-1} \hat{Y}^{*,BMA}_{t+h}$.

4 Data, Forecasting Methods, and Baseline Forecasting Models

4.1 Data

The data that we use are monthly observations on 144 U.S. macroeconomic time series for the period 1960:01 - 2009:5 ($N = 144, T = 593$)[7]. Forecasts are constructed for eleven variables, including: the unemployment rate, personal income less transfer payments, the 10 year

[7]This is an updated and expanded version of the Stock and Watson (2005a,b) dataset.

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.[8] Table 1 lists the eleven variables. The third row of the table gives the transformation of the variable used in order to induce stationarity. In general, logarithmic differences were taken for all nonnegative series that were not already in rates (see Stock and Watson (2002a, 2012) for complete details). Note that a full list of the 144 predictor variables is provided in an appendix to an earlier version of this paper which is available upon request from the authors.

4.2 Forecasting Methods

Using the transformed dataset, denoted by X, factors are estimated using linear and nonlinear factor estimation methods, as discussed above. Thereafter,the robust estimation methods outlined in the previous sections are used to form forecasting models and predictions. In particular, we consider three specification types, as follows.

Specification Type 1: Factors are first constructed using the large-scale dataset and each of PCA, ICA, and SPCA; and then prediction models are formed using the robust shrinkage methods of Section 3 to select functions of and weights for the factors to be used in prediction models of the variety given in (5). This specification type is estimated with and without lags of factors.

Specification Type 2: Factors are first constructed using subsets of variables from the large-scale dataset and each of PCA, ICA, and SPCA. Variables used in factor calculations are pre-selected via application of the robust shrinkage methods discussed in Section 3. Thereafter, prediction models of the variety given in (5) are estimated. This is different from the above approach of estimating factors using all of the variables. Note that forecasting models are estimated with and without lags of factors.

Specification Type 3: Prediction models are constructed using only the shrinkage methods discussed in Section 3, without use of factor analysis at any stage.

Specification Type 4: Prediction models are constructed using only shrinkage methods, and only with variables which have nonzero coefficients, as specified via pre-selection using SPCA.

In Specification Types 3 and 4, factor augmented autoregressions (FAAR) and pure factor based models (such as principal component regression - see next subsection for complete details) are not used as candidate forecasting models, since models with these specification types do not include factors or any type.

In our prediction experiments, pseudo out-of-sample forecasts are calculated for each variable, model variety, and specification type, for prediction horizons $h = 1, 3$, and 12. All estimation, including lag selection, shrinkage method application, and factor selection is done anew, at each point in time, prior to the construction of each new prediction, using both recursive and rolling data window strategies. Note that at each estimation period, the number of factors included will be different, following the testing approach discussed in Section 2. Note also that lags of the target predictor variables are also included in the set of explanatory variables, in all cases. Selection of the number of lagged variable to include is done using the SIC. Out-of-sample forecasts begin after 13 years (e.g. the initial in-sample estimation period is

[8]Note that gross domestic product is reported quaterly. We interpolate these data to a monthly frequency following Chow and Lin (1971),

R =156 observations, and the out-of-sample period consists of $P = T - R = 593 - 156 = 437$ observations, for $h = 1$). Moreover, the initial in-sample estimation period is adjusted so that the ex ante prediction sample length, P, remains fixed, regardless of the forecast horizon. For example, when forecasting the unemployment rate, when $h = 1$, the first forecast will be $\hat{Y}_{157}^{h=1} = \hat{\beta}_W W_{156} + \hat{\beta}_F \tilde{F}_{156}$, while in the case where $h = 12$, the first forecast will be $\hat{Y}_{157}^{h=12} = \hat{\beta}_W W_{145} + \hat{\beta}_F \tilde{F}_{145}$. In our rolling estimation scheme, the in-sample estimation period used to calibrate our prediction models is fixed at length 12 years. The recursive estimation scheme begins with the same in-sample period of 12 years (when $h = 12$), but a new observation is added to this sample prior to the re-estimation and construction of each new forecast, as we iterate through the ex-ante prediction period. Note, thus, that the actual observations being predicted as well as the number of predictions in our ex-ante prediction period remains fixed, regardless of forecast horizon, in order to facilitate comparison across forecast horizons as well as models.

Forecast performance is evaluated using mean square forecast error (MSFE), defined as:

$$MSFE_{i,h} = \sum_{t=R-h+2}^{T-h+1} \left(Y_{t+h} - \hat{Y}_{i,t+h} \right)^2, \tag{70}$$

where $\widehat{Y}_{i,t+h}$ is the forecast for horizon h. Forecast accuracy is evaluated using the above point MSFE measure as well as the predictive accuracy test statistic (called "DM" hereafter) of Diebold and Mariano (1995), which is implemented using quadratic loss, and which has a null hypothesis that the two models being compared have equal predictive accuracy (see Clark and McCracken (2001), McCracken (2000), McCracken (2007), and McCracken (2004)for details describing the importqance of accounting for parameter estimation error and nonestedness in the DM and related predictive accuracy tests).[9] In the simplest case, the DM test statistic has an asymptotic $N(0,1)$ limiting distribution, under the assumption that parameter estimation error vanishes as $T, P, R \to \infty$, and assuming that each pair of models being compared is nonnested. The null hypothesis of the test is $H_0 : E\left[l\left(\varepsilon_{t+h|t}^1\right)\right] - E\left[l\left(\varepsilon_{t+h|t}^2\right)\right] = 0$, where $\varepsilon_{t+h|t}^i$ is $i-$th model's prediction error and $l(\cdot)$ is the quadratic loss function. The actual statistic in this case is constructed as: $DM = P^{-1}\sum_{i=1}^{P} d_t/\hat{\sigma}_{\overline{d}}$, where $d_t = \left(\widehat{\varepsilon_{t+h|t}^1}\right)^2 - \left(\widehat{\varepsilon_{t+h|t}^2}\right)^2$, $\overline{d}$ is the mean of d_t, $\hat{\sigma}_{\overline{d}}$ is a heteroskedasticity and autocorrelation robust estimator of the standard deviation of $\overline{d}$, and $\widehat{\varepsilon_{t+h|t}^1}$and $\widehat{\varepsilon_{t+h|t}^2}$ are estimates of the true prediction errors $\varepsilon_{t+h|t}^1$and $\varepsilon_{t+h|t}^2$. Thus, if the statistic is negative and significantly different from zero, then Model 2 is preferred over Model 1.

4.3 Baseline Forecasting Models

In conjunction with the various forecast model specification approaches discussed above, we also form predictions using the following benchmark models, all of which are estimated using least squares.

[9] In the context of the experiments carried out in this paper, we no not consider so-called real-time data. However, it is worth noting that the use of real-time datasets in macroeconometrics, and in particular in forecasting and policy analysis, has received considerable attention in the literature in recent years. For a discussion of DM and related tests using real-time data, the reader is referred to Clark and McCracken (2009a).

Univariate Autoregression: Forecasts from a univariate AR(p) model are computed as $\hat{Y}_{t+h}^{AR} = \hat{\alpha} + \hat{\phi}(L) Y_t$, with lags , p, selected using of the SIC.

Multivariate Autoregression: Forecasts from an ARX(p) model are computed as $Y_{t+h}^{ARX} = \hat{\alpha} + \hat{\beta} Z_t + \hat{\phi}(L) Y_t$, where Z_t is a set of lagged predictor variables selected using the SIC. Dependent variable lags are also selected using the SIC. Selection of the exogenous predictors includes choosing up to six variables prior to the construction of each new prediction model, as the recursive or rolling samples iterate forward over time.

Principal Component Regression: Forecasts from principal component regression are computed as $\hat{Y}_{t+h}^{PCR} = \hat{\alpha} + \hat{\gamma}\hat{F}_t$, where $\hat{F}_t$ is estimated via principal components using X, as in equation (5).

Factor Augmented Autoregression: Based on equations (5), forecasts are computed as $Y_{t+h}^{h} = \hat{\alpha} + \hat{\beta}_F \hat{F}_t + \hat{\beta}_W(L) Y_t$. This model combines an AR(p) model, with lags selected using the SIC, and the above principal component regression (PCR) model. PCR and factor augmented autoregressive (FAAR) models are estimated using ordinary least squares. Factors in the above models are constructed using PCA, ICA and SPCA.

Combined Bivariate ADL Model: Following Stock and Watson (2012), we implement a combined bivariate autoregressive distributed lag (ADL) model. Forecasts are constructed by combining individual forecasts computed from bivariate ADL models. The i-th ADL model includes $p_{i,x}$ lags of $X_{i,t}$, and $p_{i,y}$ lags of Y_t, and has the form $\hat{Y}_{t+h}^{ADL} = \hat{\alpha} + \hat{\beta}_i(L) X_{i,t} + \hat{\phi}_i(L) Y_t$. The combined forecast is $\hat{Y}_{T+h|T}^{Comb,h} = \sum_{t=1}^{N} w_i \hat{Y}_{T+h|T}^{ADL,h}$. Here, we set $(w_i = 1/N)$, where $N = 144$. There are a number of studies that compare the performance of combining methods in controlled experiments, including: Clemen (1989), Diebold and Lopez (1996), Newbold and Harvey (2002), and Timmermann (2005); and in the literature on factor models, Stock and Watson (2004, 2006, 2012), and the references cited therein. In this literature, combination methods typically outperform individual forecasts. This stylized fact is sometimes called the "forecast combining puzzle."

Mean Forecast Combination: To further examine the issue of forecast combination, and in addition to the Bayesian model averaging methods discussed in the previous section, we form forecasts as the simple average of the thirteen forecasting models summarized in Table 2.

5 Empirical Results

In this section, we summarize the results of our prediction experiments. Target variable mnemonics are given in Table 1, and forecasting models used are summarized in Panel A of Table 2. There are 6 different specification "permutations". Specification Types 1 and 2 (estimated with and without lags) are estimated via PCA, ICA and SPCA, so that there $4 \times 3 = 12$ permutations of these two specifications. Adding Specification Types 3 and 4, and multiplying by two (for recursive and rolling windowing strategies) yields a total of $(12 + 2) \times 2 = 28$ specification types for each target variable and each forecast horizon. Forecast models and modelling methods are summarized in Panel B of Table 2. For the sake of brevity, we eschew reporting the entirety of our experimental findings, instead focusing on key findings and results. Complete details are available upon request from the authors.

Table 3 summarizes point MSFEs "best" models, relative to the AR(SIC) model, where the AR(SIC) MSFE is normalized to unity. Results are reported in two panels, with the first

panel summarizing findings across recursively estimated prediction models, and the second panel likewise reporting findings based on models estimated using rolling windows of data. Entries in bold denote MSFE-best models from among the factor estimation methods, for a given model specifications, estimation windows, and forecast horizons. Since the benchmark models, including AR(SIC), ARX, etc., are included as candidate models under specification type, there are some cases where the lowest relative MSFEs are same across factor estimation methods, for a given specification type. For example, in the case of Specification Type 1 and $h = 1$, GDP MSFEs are 0.916 for all three factor estimation methods. This is because ARX, one of benchmark models, yields a lower MSFE than any other model used in conjunction with the factor estimation methods. Moreover, since Specification Types 3 and 4 do not involve use of a factors, there are no bold entries in rows corresponding to these specification types.

Although there are a limited number of exceptions, most of the entries in Table 3 are less than unity, indicating that our factor based forecasting models dominate the autoregressive model. For example, note that the relative MSFE value for IPX, when using Specification Type 1 (SP1) and $h = 1$, is 0.268. Other bold entries can be seen to range from the low 0.80s to the mid 0.90s. Almost all of these entries are associated with models in which the DM null hypothesis of equal predictive accuracy is rejected.

Entries in the Table 4 show which forecast modelling method from Panel A of Table 2 has the lowest relative MSFE, for each target variable, and for each specification type, factor estimation method, and forecast horizon, by estimation window (Panel A summarizes results for recursive window estimation, and Panel B does the same for rolling window estimation). These entries, thus, report the forecast modelling methods associated with each MSFE value given in Table 3. For example, in the leftmost three entries of Panel A of Table 3, we see that for unemployment, the FAAR, ARX, and FAAR models resulted in the MSFE-best predictions, under SP1 and for each of PCA, ICA, and SPCA, respectively, where these MSFEs, as reported in Table 3, are 0.780, 0.897 and 0.827, respectively. Bold entries in Panels A and B of the table denote forecasting method yielding MSFE-best predictions, for a given specification type, forecast horizon, and target variable. Panel C of Table 4 summarizes the number of "wins" across 6 main specification types[10] for the 11 target variables, by forecast horizon (i.e. reports the number of bold entries by forecast modelling method in Panels A and B). Note that FAAR and PCR are methods that are not used in Specification Types 3 and 4, since these specifications do not use factors. Accordingly, mean forecasts in Specification Types 3 and 4 are constructed using the arithmetic mean of all forecast modelling methods except these two.

Notice also, in Table 4, that ARX appears in multiple entries. For example, for HS and $h = 1$, ARX appears as the "winner" in numerous cases. The reason for this is that each specification type has the same ARX model as one of the baseline models, and so correct interpretation of this finding is that the *same* ARX model dominates for a couple of variables (i.e. HS and GDP), when $h = 1$, regardless of factor estimation method used for specification of factor models. However, note that for HS, the FAAR model "wins" under SP1 and SPCA for $h = 1$, and has a relative MSFE (from Table 3) of 0.542, which is substantially lower than the value of 0.901 that applies to all of the cases where ARX "wins". Thus, care must be taken when interpreting the results of Table 4; inasmuch as the ARX model is much less dominant than may appear to be the case upon cursory inspection of entries. Interestingly, boosting and

[10] In Specification Type 1 and 2 without lags and with lags, we pick the best model amongst the three factor estimation methods so that we have 6 specifications in this analysis, and not 14.

LARS perform well in several specifications and forecast horizons. This is particularly true for higher forecast horizons, where the only method to "win" more frequently involves simply using the arithmetic mean.

Entries in Panel A of Table 5 report which factor estimation method yields the lowest MSFE for each specification type, forecast horizon and target variable, when models are estimated using recursive data windows. (Since Specification Types 3 and 4 do not use factors, they are excluded in this table.) Panel B is the same as Panel A, except that results are for models estimated using rolling windows of data. Panel C of the table summarizes the result in Panels A and B across target variables, thus reporting counts of the number of times each factor estimation method "wins" by specification type, forecast horizon, and estimation window. For example, upon inspection of Panel C, we see that for Specification Type 2 without lags, PCA, ICA and SPCA win 7, 2 and 1 times, respectively, for $h = 1$. Notice that SPCA performs very well under Specification 1, when $h = 1$, although PCA "wins" the most across all other specification types, regardless of forecast horizon. Moreover, ICA performs much worse than either other factor estimation method. However, this result does not directly imply that PCA is a better method for factor analysis, since these results are based on complex hybrid forecasting modelling strategies coupling factor estimation methods with shrinkage and other regression modelling strategies.

Entries in Panel A of Table 6 report which estimation window method yields the lowest MSFE for each specification type, factor estimation method, forecast horizon and target variable. Again, since Specification Types 3 and 4 do use factors, they are excluded in this table. Panel B of the table summarizes results in Panel A across specification types. Here, 'Recur' stands for recursive window estimation and 'Roll' for rolling window estimation. Recursive window estimation "wins" in 93 out of 154 cases, when $h = 1$. On the other hand, it is interesting to note that rolling window estimation dominates at the $h = 12$ horizon, winning in 119 of 154 cases. Thus, the trade-off between using less data (and hence inducing increased parameter uncertainty in order to benefit from quicker adjustment for structural breaks) and using more data (and hence failing to account for breaks), appears to depend on forecast horizon. For further discussion of data windowing, including a discussion of window combination, see Clark and McCracken (2009b).

Panels A, B, and C of Table 7 summarize results reported in Table 3 and 4. Entries in Panel A report the "best" MSFEs for each target variable, by specification type and forecast horizon. Further, the window estimation scheme / factor estimation method / winning model combinations associated with the lowest MSFE associated with each target variable in Panel A are given in Panel B of the table. Finally, the specification type / window estimation scheme / factor estimation method / winning model combinations associated with each bold MSFE entry in Panel A are given in Panel C of the table. In Panel A, note that SP1 and SP1L yield the MSFE-best prediction models in 15 of 33 possible cases, across forecast horizon, with more than one half of these "wins" arising for the case where $h = 1$. Thus, just as estimation window selection seems to require differentiating across forecast horizon, so too does forecast horizon make a difference when ranking specification types. However, recall from the results reported in Table 5 that although PCA "wins" quite frequently, the "wins" accorded to ICA and SPCA arise rather uniformly across forecast horizon. Upon inspection of Panel B of the table, the following conclusions emerge. First, of the window estimation scheme / factor estimation method / winning model combinations, recursive windowing "wins" 17 of 33 times. Thus, over

all permutations and variables, the evidence suggests that there is little to choose between the two schemes. This points to a need to carefully consider the methods discussed in Clark and McCracken (2009) when estimating prediction models. Second, PCA actually "wins" in only 14 of 33 possible cases, overall. This suggests that although PCA "wins" in many more cases when disaggregating our findings, as reported earlier, when we actually summarize across the very best models, it wins less than one half of the time. We thus have interesting new evidence suggesting that ICA and SPCA are very useful factor modelling tools; and in particular, we have seen from earlier discussion that SPCA is the clear winner from amongst non-PCA factor methods. Thus, as discussed in numerous papers, imposing parsimony on our factor modelling methods is quite useful. This in turn points to the fact that there is much remaining to be done in the area of parsimonious diffusion index modelling, given the novel nature and relative inexperience that economists have with the methods used herein. Third, we see that the arithmetic mean forecasting model "wins" in only 9 of 33 cases. This is rather surprising new evidence that simple model averaging does not necessarily yield MSFE-best predictions. However, in order to "beat" model averaging methods, including arithmetic mean and Bayesian averaging approaches, we have needed to introduce into our horse-race numerous complex new models. Indeed, we see from further inspection of this table that most of the winning models involve combining complicated factor estimation methods with interesting new forms of shrinkage. It is really the combination of factors and shrinkage that is delivering our results that model averaging does not always "win". Finally, turning to Panel C of Table 7, note that hybrid methods that couple factor estimation methods with shrinkage "win" in 9 of 33 cases, while simpler factor modelling approaches that do not additionally use shrinkage "win" in 10 of 33 cases. Pure shrinkage methods (i.e. SP3 and SP4 with shrinkage) "win" in 3 cases, while Bayesian model averaging and simpler arithmetic mean combination methods "win" the remaining 11 cases. Simple linear autoregressive type models never win. We take these final results as further evidence of the usefulness of new methods in factor modelling and shrinkage, when the objective is prediction of macroeconomic time series variables.

In a final twist on our results, please refer to Table 8, where we summarize whether or not including lags in our specification types yields MSFE-best models, or not, for each target variable and factor estimation method. Interesting, it is immediately apparent, upon inspection of the entries in the table, that "No Lag" specification types dominate. This finding again points to the need for parsimonious data reduction methods when using "big data".

6 Concluding Remarks

In this paper we outline and discuss a number of interesting new forecasting methods that have recently been developed in the statistics and econometrics literatures. We focus in particular on the examination of a variety of factor estimation methods, including principal components as discussed by Stock and Watson (2002a,b), independent component analysis (ICA) and sparse principal component analysis (SPCA); as well as hybrid forecasting methods that use these factor estimation methods in conjunction with various types of shrinkage, such as bagging, boosting, least angle regression, the elastic net, and the nonnegative garotte. Finally, we carry out a series of real-time prediction experiments evaluating all of these methods against a number of benchmark linear models and forecast combination approaches. Our experiments are carried out in the context of predicting 11 key macroeconomic indicators at various forecast horizons.

We find that model simple time series models and model averaging methods do not dominate hybrid methods that couple factor estimation methods with shrinkage. However, pure shrinkage methods do not fare well, when implemented in isolation, with the use of latent factors. We take these final results as further evidence of the usefulness of new methods in factor modelling and shrinkage, when the objective is prediction of macroeconomic time series variables.

References

Aiolfi, M. and Timmermann, A. (2006). Persistence in forecasting performance and conditional combination strategies. *Journal of Econometrics*, 135(1-2):31–53.

Armah, N. A. and Swanson, N. R. (2010a). Diffusion index models and index proxies: Recent results and new direction. *European Journal of Pure and Applied Mathematics*, 3:478–501.

Armah, N. A. and Swanson, N. R. (2010b). Seeing inside the black box: Using diffusion index methodology to construct factor proxies in large scale macroeconomic time series environments. *Econometric Reviews*, 29:476–510.

Artis, M. J., Banerjee, A., and Marcellino, M. (2005). Factor forecasts for the uk. *Journal of Forecasting*, 24(4):279–298.

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

Bai, J. and Ng, S. (2006a). Confidence intervals for diffusion index forecasts and inference for factor-augmented regressions. *Econometrica*, 74(4):1133–1150.

Bai, J. and Ng, S. (2006b). Evaluating latent and observed factors in macroeconomics and finance. *Journal of Econometrics*, 131(1-2):507–537.

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

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

Banerjee, A. and Marcellino, M. (2008). Factor-augmented error correction models. CEPR Discussion Papers 6707, C.E.P.R. Discussion Papers.

Boivin, J. and Ng, S. (2005). Understanding and comparing factor-based forecasts. *International Journal of Central Banking*, 1(3):117–152.

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

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

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

Bühlmann, P. and Yu, B. (2002). Analyzing bagging. *Annals of Statistics*, 30:927–961.

Bühlmann, P. and Yu, B. (2003). Boosting with the l_2 loss: Regression and classification. *Journal of the American Statistical Association*, 98:324–339.

Chipman, H., George, E. I., and Mcculloch, R. E. (2001). The practical implementation of bayesian model selection. In *Institute of Mathematical Statistics*, pages 65–134.

Chow, G. C. and Lin, A.-l. (1971). Best linear unbiased interpolation, distribution, and extrapolation of time series by related series. *The Review of Economics and Statistics*, 53(4):372–75.

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

Clark, T. and McCracken, M. W. (2009a). Tests of equal predictive ability with real-time data. *Journal of Business and Economic Statistics*, 27:441–454.

Clark, T. E. and McCracken, M. W. (2009b). Improving forecast accuracy by combining recursive and rolling forecasts. *International Economic Review*, 50(2):363–395.

Clemen, R. T. (1989). Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5(4):559–583.

Clyde, M. (1999). Bayesian model averaging and model search strategies. In J. M. Bernardo, J. O. Berger, A. P. D. and Smith, A., editors, *Bayesian Statistics 6*, pages 157–185. Oxford University Press.

Comon, P. (1994). Independent component analysis - a new concept? *Signal Processing*, 36:287–314.

Connor, G. and Korajczyk, R. A. (1986). Performance measurement with the arbitrage pricing theory : A new framework for analysis. *Journal of Financial Economics*, 15(3):373–394.

Connor, G. and Korajczyk, R. A. (1988). Risk and return in an equilibrium apt : Application of a new test methodology. *Journal of Financial Economics*, 21(2):255–289.

Connor, G. and Korajczyk, R. A. (1993). A test for the number of factors in an approximate factor model. *Journal of Finance*, 48(4):1263–91.

Diebold, F. X. and Lopez, J. A. (1996). Forecast evaluation and combination. NBER Technical Working Papers 0192, National Bureau of Economic Research, Inc.

Diebold, F. X. and Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business & Economic Statistics*, 13(3):253–63.

Ding, A. A. and Hwang, J. T. G. (1999). Prediction intervals, factor analysis models, and high-dimensional empirical linear prediction. *Journal of the American Statistical Association*, 94(446):446–455.

Dufour, J.-M. and Stevanovic, D. (2010). Factor-augmented varma models: Identification, estimation, forecasting and impulse responses. Working paper, McGill University.

Efron, B., Hastie, T., Johnstone, L., and Tibshirani, R. (2004). Least angle regression. *Annals of Statistics*, 32:407–499.

Fernandez, C., Ley, E., and Steel, M. F. J. (2001a). Benchmark priors for bayesian model averaging. *Journal of Econometrics*, 100(2):381–427.

Fernandez, C., Ley, E., and Steel, M. F. J. (2001b). Model uncertainty in cross-country growth regressions. *Journal of Applied Econometrics*, 16(5):563–576.

Forni, M., Hallin, M., Lippi, M., and Reichlin, L. (2000). The generalized dynamic-factor model: Identification and estimation. *The Review of Economics and Statistics*, 82(4):540–554.

Forni, M., Hallin, M., Lippi, M., and Reichlin, L. (2005). The generalized dynamic factor model: One-sided estimation and forecasting. *Journal of the American Statistical Association*, 100:830–840.

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139.

Friedman, J., Hastie, T., and Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28:2000.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5):1189–1232.

Gelper, S. and Croux, C. (2008). Least angle regression for time series forecasting with many predictors, working paper. Technical report, Katholieke Universiteit Leuven.

Guo, J., James, G., Levina, E., Michailidis, G., and Zhu, J. (2010). Principal component analysis with sparse fused loadings. *Journal of Computational and Graphical Statistics*, 19(4):947–962.

Hastie, T., Tibshirani, R., and Friedman, J. H. (2009). *The Elements of Statistical Learning*. Springer, 2nd edition.

Hoerl, A. and Kennard, R. (1970). Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67.

Hoeting, J. A., Madigan, D., Raftery, A. E., and Volinsky, C. T. (1999). Bayesian model averaging: a tutorial. *Statistical Science*, 14:382–417.

Hyvärinen, A. (1998). Independent component analysis in the presence of gaussian noise by maximizing joint likelihood. *Neurocomputing*, 22:49–67.

Hyvärinen, A. (1999a). Gaussian moments for noisy independent component analysis. *IEEE Signal Processing Letters*, 6(6):145–147.

Hyvärinen, A. (1999b). Survey on independent component analysis. *Neural Computing Surveys*, 2:94–128.

Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural Networks*, 13(4-5):411–430.

Inoue, A. and Kilian, L. (2008). How useful is bagging in forecasting economic time series? a case study of us cpi inflation. *Journal of the American Statistical Association*, 103(482):511–522.

Jolliffe, I., Trendafilov, N., and Uddin, M. (2003). A modified principal component technique based on the lasso. *Journal of Computational and Graphical Statistics*, 12:531–547.

Jolliffe, I. T. (1995). Rotation of principal components: choice of normalization constraints. *Journal of Applied Statistics*, 22:29–35.

Kass, R. and Raftery, A. (1995). Bayes factors. *Journal of the American Statistical Association*, 90:773–795.

Kim, H. H. and Swanson, N. R. (2013). Forecasting financial and macroeconomic variables using data reduction methods: New empirical evidence. *Journal of Econometrics*, Forthcoming.

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

Lee, T.-W. (1998). *Independent Component Analysis - Theory and Applications*. Springer, Boston, Massachusetts, 1 edition.

Leng, C. and Wang, H. (2009). On general adaptive sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 18(1):201–215.

McCracken, M. W. (2000). Robust out-of-sample inference. *Journal of Econometrics*, 99:195–223.

McCracken, M. W. (2004). Parameter estimation error and tests of equal forecast accuracy between non-nested models. *International Journal of Forecasting*, 20:503–514.

McCracken, M. W. (2007). Asymptotics for out-of-sample tests of granger causality. *Journal of Econometrics*, 140:719–752.

Newbold, P. and Harvey, D. I. (2002). Forecast combination and encompassing. In Clements, M. P. and Hendry, D. F., editors, *A Companion to Economic Forecasting*, pages 268–283. Blackwell Press, Oxford.

Penny, W., Robert, S., and Everson, R. (2001). Ica: Model order selection and dynamic source models. In Roberts, S. and Everson, R., editors, *Independent Component Analysis: Principles and Practice*, pages 299–314. Cambridge University Press, Cambridge, UK.

Ravazzolo, F., Paap, R., van Dijk, D., and Franses, P. H. (2008). *Bayesian Model Averaging in the Presence of Strutural Breaks*, chapter 15. Frontier of Economics and Globalization.

Ridgeway, G., Madigan, D., and Richardson, T. (1999). Boosting methodology for regression problems. In *The Seventh International Workshop on Artificial Intelligence and Statistics (Uncertainty '99)*, pages 152–161. Morgan Kaufmann.

Schapire, R. E. (1990). The strength of weak learnability. *Machine Learning*, 5(2):197–227.

Shrestha, D. L. and Solomatine, D. P. (2006). Experiments with adaboost.rt, an improved boosting scheme for regression. *Neural Computation*, 18(7):1678–1710.

Stock, J. H. and Watson, M. W. (1999). Forecasting inflation. *Journal of Monetary Economics*, 44(2):293–335.

Stock, J. H. and Watson, M. W. (2002a). Forecasting using principal components from a large number of predictors. *Journal of the American Statistical Association*, 97:1167–1179.

Stock, J. H. and Watson, M. W. (2002b). Macroeconomic forecasting using diffusion indexes. *Journal of Business & Economic Statistics*, 20(2):147–62.

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

Stock, J. H. and Watson, M. W. (2005). Implications of dynamic factor models for var analysis. NBER Working Papers 11467, National Bureau of Economic Research, Inc.

Stock, J. H. and Watson, M. W. (2006). Forecasting with many predictors. In Elliott, G., Granger, C., and Timmermann, A., editors, *Handbook of Economic Forecasting*, volume 1, chapter 10, pages 515–554. Elsevier.

Stock, J. H. and Watson, M. W. (2012). Generalized shrinkage methods for forecasting using many predictors. *Journal of Business and Economic Statistics*, forthcoming.

Stone, J. V. (2004). *Independent Component Analysis*. MIT Press.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288.

Timmermann, A. G. (2005). Forecast combinations. CEPR Discussion Papers 5361, C.E.P.R. Discussion Papers.

Tong, L., Liu, R.-w., Soon, V., and Huang, Y.-F. (1991). Indeterminacy and identifiability of blind identification. *IEEE Transactions on Circuits and Systems*, 38:499–509.

Vines, S. (2000). Simple principal components. *Applied Statistics*, 49:441–451.

Wright, J. H. (2008). Bayesian model averaging and exchange rate forecasting. *Journal of Econometrics*, 146:329–341.

Wright, J. H. (2009). Forecasting u.s. inflation by bayesian model averaging. *Journal of Forecasting*, 28:131–144.

Yuan, M. (2007). Nonnegative garrote component selection in functional anova models. In *Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics*, pages 660–666. JMLR Workshop and Conference Proceedings.

Yuan, M. and Lin, Y. (2007). On the non-negative garrotte estimator. *Journal of the Royal Statistical Society*, 69(2):143–161.

Zellner (1986). On assessing prior distributions and bayesian regression analysis with g-prior distributions,. In Goel, P. and Zellner, A., editors, *Bayesian Inference and Decision Techniques: Essays in Honour of Bruno de Finetti*. Armsterdam: North-Holland.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101:1418–1429.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal Of The Royal Statistical Society Series B*, 67(2):301–320.

Zou, H., Hastie, T., and Tibshirani, R. (2006). Sparse principal component analysis. *Journal of Computational and Graphical Statistics*, 15(2):262–286.

Table 1: Target Forecast Variables *

Series	Abbreviation	Y_{t+h}
Unemployment Rate	UR	$Z_{t+1} - Z_t$
Personal Income Less transfer payments	PI	$\ln(Z_{t+1}/Z_t)$
10-Year Treasury Bond	TB	$Z_{t+1} - Z_t$
Consumer Price Index	CPI	$\ln(Z_{t+1}/Z_t)$
Producer Price Index	PPI	$\ln(Z_{t+1}/Z_t)$
Nonfarm Payroll Employment	NPE	$\ln(Z_{t+1}/Z_t)$
Housing Starts	HS	$\ln(Z_t)$
Industrial Production	IPX	$\ln(Z_{t+1}/Z_t)$
M2	M2	$\ln(Z_{t+1}/Z_t)$
S&P 500 Index	SNP	$\ln(Z_{t+1}/Z_t)$
Gross Domestic Product	GNP	$\ln(Z_{t+1}/Z_t)$

* Notes: Data used in model estimation and prediction construction are monthly U.S. figures for the period 1960:1-2009:5. Data transformations used in prediction experiments are given in the last column of the table. See Section 4 for further details.

Table 2: Models and Methods Used In Real-Time Forecasting Experiments*

Method	Description
AR(SIC)	Autoregressive model with lags selected by the SIC
ARX	Autoregressive model with exogenous regressors
CADL	Combined autoregressive distributed lag model
FAAR	Factor augmented autoregressive model
PCR	Principal components regression
Bagging	Bagging with shrinkage, $c = 1.96$
Boosting	Component boosting, $M = 50$
BMA1	Bayesian model averaging with g-prior $= 1/T$
BMA2	Bayesian model averaging with g-prior $= 1/N^2$
Ridge	Ridge regression
LARS	Least angle regression
EN	Elastic net
NNG	Non-negative garotte
Mean	Arithmetic mean

* Notes: This table summarizes the prediction model specifications used in all experiments. In addition to directly estimating the above pure linear and factor models (i.e., AR, ARX, CADL, FAAR, PCR), three different combined factor and shrinkage type prediction "specification methods" are used in our forecasting experiments, including: Specification Type 1 - Principal components are first constructed, and then prediction models are formed using the above shrinkage methods (including Bagging, Boosting, Ridge, LARS, EN, and NNG) to select functions of and weights for the factors to be used in our prediction models. Specification Type 2 - Principal component models are constructed using subsets of variables from the large-scale dataset that are first selected via application of the above shrinkage methods (ranging from bagging to NNG). This is different from the above approach of estimating factors using all of the variables. Specification Type 3 - Prediction models are constructed using only the above shrinkage methods (ranging from bagging to NNG), without use of factor analysis at any stage. See Sections 3 and 4 for complete details.

Table 3: Point MSFEs by Forecast Estimation Metho and Specification Type*

Panel A: Recursive Window Estimation

Forecast Horizon	Factor Spec.	Mtd.	UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
$h=1$	SP1	PCA	**0.780**	0.870	0.940	0.875	0.943	0.811	0.901	0.800	**0.939**	0.976	**0.916**
		ICA	0.897	0.920	0.931	**0.840**	**0.843**	0.802	0.901	0.574	0.965	0.920	**0.916**
		SPCA	0.827	**0.789**	**0.409**	0.870	0.858	**0.706**	**0.542**	**0.268**	0.969	**0.897**	**0.916**
	SP1L	PCA	**0.850**	**0.889**	**0.955**	**0.865**	**0.945**	**0.879**	**0.901**	**0.804**	**0.930**	**0.976**	**0.916**
		ICA	0.897	0.966	0.978	0.939	0.960	0.918	**0.901**	0.861	0.991	1.002	**0.916**
		SPCA	0.897	0.954	0.987	0.939	0.972	0.881	**0.901**	0.826	0.954	0.998	**0.916**
	SP2	PCA	**0.861**	**0.950**	**0.965**	**0.933**	0.968	**0.854**	**0.901**	**0.833**	0.942	**0.985**	0.871
		ICA	0.897	0.959	0.971	0.939	**0.965**	0.861	**0.901**	0.874	0.959	0.991	**0.867**
		SPCA	0.897	0.959	0.976	0.939	0.966	0.860	**0.901**	0.873	**0.940**	0.986	0.873
	SP2L	PCA	**0.861**	**0.950**	**0.965**	0.933	0.968	**0.854**	**0.901**	**0.833**	**0.942**	**0.985**	**0.871**
		ICA	0.864	0.957	0.975	**0.923**	0.967	0.862	**0.901**	0.840	0.961	0.993	**0.871**
		SPCA	0.868	0.961	0.974	0.939	**0.963**	0.859	**0.901**	0.874	0.950	0.991	0.879
	SP3		0.897	0.944	0.987	0.933	0.956	0.826	0.901	0.874	0.977	0.989	0.873
	SP4		0.897	0.964	0.979	0.939	0.962	0.865	0.901	0.829	0.971	0.986	0.916
$h=3$	SP1	PCA	**0.913**	**0.866**	0.998	0.929	**0.910**	0.819	**0.852**	0.850	0.977	**0.994**	0.956
		ICA	0.914	0.902	**0.975**	0.922	0.945	0.819	0.917	0.834	0.969	1.002	0.976
		SPCA	0.916	0.892	0.988	**0.895**	0.940	**0.775**	0.862	**0.816**	**0.942**	0.997	**0.944**
	SP1L	PCA	**0.925**	**0.892**	0.988	**0.901**	**0.929**	**0.818**	**0.852**	**0.838**	**0.978**	**0.993**	**0.963**
		ICA	0.963	0.902	0.998	0.967	0.945	0.927	0.948	0.895	0.997	1.007	0.979
		SPCA	0.951	0.902	**0.984**	0.968	0.945	0.924	0.912	0.887	0.990	0.997	0.988
	SP2	PCA	**0.916**	**0.895**	**0.992**	**0.888**	**0.945**	**0.827**	**0.783**	**0.809**	0.967	0.995	**0.954**
		ICA	0.941	0.902	0.995	0.959	**0.945**	0.859	0.824	0.821	0.980	0.997	0.963
		SPCA	0.943	0.902	0.998	0.975	**0.945**	0.894	0.793	0.873	**0.964**	**0.993**	0.963
	SP2L	PCA	**0.916**	**0.895**	**0.992**	**0.888**	**0.945**	**0.827**	**0.783**	**0.809**	**0.967**	0.995	**0.954**
		ICA	**0.916**	0.902	0.998	0.903	**0.945**	**0.827**	0.854	0.812	0.979	0.997	0.967
		SPCA	0.950	0.902	0.994	0.972	**0.945**	0.889	0.803	0.812	0.974	**0.993**	0.962
	SP3		0.943	0.902	0.998	0.926	0.945	0.860	0.723	0.881	0.939	1.001	0.975
	SP4		0.950	0.902	0.986	0.979	0.945	0.898	0.937	0.872	0.990	0.988	0.978
$h=12$	SP1	PCA	0.939	0.956	0.997	**0.886**	**0.939**	0.874	**0.818**	**0.919**	0.958	1.002	0.999
		ICA	0.948	0.944	0.997	0.960	0.977	0.907	0.844	0.952	0.960	1.001	**0.986**
		SPCA	**0.933**	**0.940**	**0.992**	0.928	0.950	**0.845**	0.841	0.932	**0.950**	**0.996**	0.993
	SP1L	PCA	**0.903**	**0.956**	**0.988**	**0.888**	**0.927**	**0.860**	**0.829**	**0.926**	**0.942**	**0.995**	1.000
		ICA	0.943	0.969	0.997	0.961	0.981	0.912	0.912	0.939	0.964	1.002	**0.981**
		SPCA	0.912	0.977	0.997	0.945	0.970	0.879	0.832	0.937	0.981	1.001	0.997
	SP2	PCA	0.926	0.949	**0.992**	**0.891**	**0.950**	**0.816**	**0.749**	**0.916**	0.930	**0.995**	**0.982**
		ICA	0.941	0.949	0.997	0.909	0.960	0.843	0.901	0.942	0.933	0.999	0.991
		SPCA	**0.916**	**0.948**	0.997	0.935	0.957	0.843	0.910	0.919	**0.916**	0.997	0.992
	SP2L	PCA	0.926	**0.949**	**0.992**	**0.891**	**0.950**	**0.816**	**0.749**	**0.916**	**0.930**	0.995	**0.982**
		ICA	0.933	0.953	**0.992**	0.894	0.964	0.853	0.883	0.944	0.942	0.998	0.985
		SPCA	**0.914**	0.950	0.996	0.958	0.968	0.872	0.880	0.938	0.961	**0.994**	0.989
	SP3		0.926	0.961	0.997	0.899	0.953	0.862	0.804	0.890	0.910	1.002	0.982
	SP4		0.926	0.963	0.997	0.943	0.962	0.855	0.886	0.927	0.976	1.001	0.990

Panel B: Rolling Window Estimation

Forecast Horizon	Factor Spec.	Mtd.	UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
$h=1$	SP1	PCA	**0.787**	**0.909**	**0.944**	**0.843**	**0.971**	**0.831**	**0.841**	**0.803**	**0.863**	**0.998**	**0.940**
		ICA	0.871	1.014	0.977	0.876	0.973	0.918	**0.841**	0.910	0.918	**0.998**	0.948
		SPCA	0.871	1.023	0.977	0.883	0.996	0.877	**0.841**	0.875	0.869	1.007	0.945
	SP1L	PCA	**0.852**	**0.989**	**0.954**	**0.850**	**0.973**	**0.871**	**0.841**	**0.845**	**0.845**	**1.002**	0.943
		ICA	0.871	1.004	0.982	0.883	0.985	0.924	**0.841**	0.877	0.908	1.008	**0.941**
		SPCA	0.871	1.081	0.992	0.883	1.003	0.911	**0.841**	0.851	0.880	1.008	0.989
	SP2	PCA	**0.871**	**1.085**	**0.963**	0.849	**0.936**	**0.869**	**0.841**	**0.858**	0.889	**0.998**	**0.915**
		ICA	**0.871**	1.114	0.977	0.849	0.941	0.884	**0.841**	**0.858**	0.908	1.006	**0.915**
		SPCA	**0.871**	1.087	0.979	**0.844**	0.949	0.877	**0.841**	0.892	**0.888**	1.007	0.927
	SP2L	PCA	**0.871**	**1.088**	**0.964**	0.850	**0.948**	**0.865**	**0.841**	**0.833**	**0.886**	**0.997**	**0.905**
		ICA	**0.871**	1.100	0.977	0.843	0.953	0.880	**0.841**	0.841	0.909	1.004	**0.905**
		SPCA	**0.871**	1.095	0.979	**0.840**	0.957	0.879	**0.841**	0.864	0.910	1.004	0.915
	SP3		0.871	1.114	0.992	0.858	1.000	0.924	0.841	0.841	0.916	1.008	0.930
	SP4		0.871	1.091	0.977	0.828	0.946	0.872	0.841	0.867	0.899	1.008	0.945
$h=3$	SP1	PCA	**0.882**	**0.872**	1.002	**0.861**	**0.937**	**0.786**	**0.769**	**0.835**	**0.914**	**0.997**	**0.937**
		ICA	0.923	0.925	0.996	0.890	0.941	0.833	0.839	0.854	0.978	1.004	0.957
		SPCA	0.926	0.913	**0.993**	0.870	0.944	0.847	0.807	0.869	0.941	1.003	0.969
	SP1L	PCA	**0.904**	**0.889**	**0.981**	**0.848**	**0.920**	**0.807**	**0.744**	**0.820**	**0.908**	**0.988**	**0.953**
		ICA	0.936	0.925	1.001	0.900	0.951	0.876	0.854	0.877	0.976	1.008	0.957
		SPCA	0.957	0.903	1.002	0.905	0.945	0.905	0.840	0.884	0.981	1.001	0.972
	SP2	PCA	**0.895**	**0.883**	0.998	0.875	0.941	**0.814**	**0.740**	**0.833**	**0.912**	**0.989**	**0.929**
		ICA	0.912	0.899	**0.995**	0.875	**0.939**	0.838	0.743	0.850	0.915	**0.989**	0.950
		SPCA	0.919	0.914	0.997	**0.863**	0.941	0.846	0.785	0.857	0.927	**0.989**	0.947
	SP2L	PCA	0.889	**0.886**	**0.988**	0.864	0.942	**0.792**	**0.738**	**0.823**	**0.911**	**0.985**	**0.938**
		ICA	**0.888**	0.901	0.998	0.865	0.941	**0.792**	0.806	0.838	0.921	**0.985**	0.947
		SPCA	0.927	0.919	1.002	**0.861**	**0.936**	0.843	0.772	0.858	0.929	**0.985**	0.943
	SP3		0.911	0.903	1.002	0.906	0.960	0.839	0.683	0.844	0.950	1.002	0.970
	SP4		0.930	0.903	1.002	0.842	0.925	0.831	0.806	0.858	0.942	0.994	0.960
$h=12$	SP1	PCA	0.897	**0.935**	**0.997**	0.812	**0.891**	0.729	0.723	**0.884**	**0.896**	1.007	1.010
		ICA	0.930	0.944	**0.997**	0.863	0.949	0.779	0.741	0.909	0.937	**0.996**	0.999
		SPCA	**0.879**	0.953	**0.997**	**0.781**	0.920	**0.720**	**0.715**	0.890	0.904	1.006	**0.997**
	SP1L	PCA	**0.864**	**0.946**	0.997	0.819	**0.902**	0.737	0.726	**0.898**	**0.899**	**1.000**	0.996
		ICA	0.908	0.951	0.997	0.872	0.962	**0.730**	0.773	0.902	0.942	1.003	0.987
		SPCA	0.869	0.983	**0.992**	**0.816**	0.938	0.759	**0.712**	0.943	0.960	1.002	**0.984**
	SP2	PCA	**0.893**	**0.929**	**0.997**	**0.818**	**0.912**	0.692	**0.637**	**0.880**	0.884	0.994	0.994
		ICA	0.911	0.932	**0.997**	0.833	0.915	**0.691**	0.726	0.902	0.888	0.994	0.993
		SPCA	0.901	0.935	**0.997**	0.819	0.921	0.692	0.693	0.896	**0.879**	**0.991**	**0.991**
	SP2L	PCA	**0.883**	**0.927**	**0.997**	**0.816**	**0.903**	**0.714**	**0.624**	**0.888**	**0.880**	**0.993**	0.996
		ICA	0.895	0.929	**0.997**	0.835	0.917	0.719	0.695	0.898	0.897	0.994	0.993
		SPCA	0.888	0.935	**0.997**	0.836	0.910	0.722	0.768	0.897	0.905	0.994	**0.991**
	SP3		0.903	0.971	0.997	0.799	0.947	0.690	0.551	0.940	0.891	1.001	0.998
	SP4		0.882	0.937	0.997	0.804	0.912	0.702	0.616	0.886	0.902	0.997	0.985

*Notes: See notes to Tables 1 and 2. Numerical entries in this table are the lowest (relative) mean square forecast errors (MSFEs) based on the use of various "recursively estimated" (Panel A) and "rolling estimated" (Panel B) prediction models using three different factor estimation methods (PCA, ICA and SPCA - see Section 2 for further discussion), for six different specification types. Prediction models and target variables are described in Tables 1 and 2 (see Section 4 for further discussion). Forecasts are monthly, for the period 1974:3-2009:5. Forecast horizons reported on include h=1,3 and 12. Tabulted relative MSFEs are calculated such that numerical values less than unity constitute cases for which the alternative model has lower point MSFE than the AR(SIC) model. Entries in bold denote point-MSFE "best" models among the three factor estimation methods, for a given specification type, estimation window and forecast horizon. See Section 5 for further details.

Table 4: Summary of MSFE "Best" Models*

Panel A: Recursive Window Estimation

Forecast Horizon	Factor Spec.	Mtd.	UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
$h=1$	SP1	PCA	**FAAR**	PCR	Ridge	PCR	PCR	FAAR	ARX	PCR	**Mean**	Mean	**ARX**
		ICA	ARX	FAAR	FAAR	**FAAR**	**FAAR**	Ridge	ARX	FAAR	Mean	Boost	**ARX**
		SPCA	FAAR	**PCR**	**PCR**	BMA1	BMA2	**Mean**	**FAAR**	**FAAR**	Mean	**Boost**	**ARX**
	SP1L	PCA	**FAAR**	**PCR**	**Mean**	**PCR**	**Mean**	**Mean**	**ARX**	**BMA1**	**Mean**	**Boost**	**ARX**
		ICA	ARX	Mean	Mean	ARX	Mean	Mean	**ARX**	Mean	Mean	AR	**ARX**
		SPCA	ARX	Mean	CADL	ARX	Mean	Boost	**ARX**	Mean	Mean	Mean	**ARX**
	SP2	PCA	**Boost**	**Mean**	**Mean**	**Boost**	Mean	**Mean**	**ARX**	**BMA1**	BMA2	**Mean**	Boost
		ICA	ARX	Mean	Mean	ARX	**Mean**	Mean	**ARX**	ARX	EN	Mean	**Boost**
		SPCA	ARX	Mean	Mean	ARX	Mean	Mean	**ARX**	BMA1	**Boost**	Mean	Boost
	SP2L	PCA	**Boost**	**Mean**	**Mean**	Boost	Mean	**Mean**	**ARX**	**BMA1**	**BMA2**	**Mean**	**Boost**
		ICA	Boost	Mean	Mean	**Boost**	Mean	Mean	**ARX**	Boost	EN	Mean	**Boost**
		SPCA	Boost	Mean	Mean	ARX	**Mean**	Mean	**ARX**	ARX	Boost	Mean	Boost
	SP3		ARX	Mean	CADL	Mean	Mean	Mean	ARX	ARX	Mean	Boost	Mean
	SP4		ARX	Mean	Mean	ARX	Mean	Mean	ARX	BMA1	Mean	Mean	ARX
$h=3$	SP1	PCA	**PCR**	**PCR**	CADL	FAAR	**PCR**	FAAR	**Boost**	Mean	Mean	**LARS**	Mean
		ICA	FAAR	ARX	**PCR**	FAAR	ARX	FAAR	LARS	Mean	Bagg	AR	Mean
		SPCA	Mean	PCR	Mean	**FAAR**	Mean	**Ridge**	Mean	**FAAR**	**Mean**	NNG	**Mean**
	SP1L	PCA	**Mean**	**Mean**	Mean	**Mean**	**Mean**	**BMA1**	**Mean**	**Mean**	**Mean**	**NNG**	**Mean**
		ICA	Mean	ARX	CADL	Mean	ARX	Mean	LARS	ARX	NNG	AR	Mean
		SPCA	Mean	ARX	**Mean**	Mean	ARX	BMA2	Mean	Mean	NNG	NNG	NNG
	SP2	PCA	**Boost**	**Mean**	**EN**	**Boost**	**ARX**	**Boost**	**Boost**	**Mean**	Mean	Mean	**Mean**
		ICA	Mean	ARX	LARS	Boost	**ARX**	Boost	Boost	Boost	Mean	Mean	Mean
		SPCA	Mean	ARX	CADL	Mean	**ARX**	Mean	Boost	Mean	**Boost**	**LARS**	Mean
	SP2L	PCA	**Boost**	**Mean**	**EN**	**Boost**	**ARX**	**Boost**	**Boost**	**Mean**	**Mean**	Mean	**Mean**
		ICA	**Boost**	ARX	CADL	Boost	**ARX**	**Boost**	Boost	LARS	Mean	Mean	Mean
		SPCA	Mean	ARX	BMA2	Mean	**ARX**	Mean	Boost	LARS	Boost	**Mean**	Mean
	SP3		Boost	ARX	CADL	Mean	ARX	Mean	Mean	BMA2	Mean	AR	Boost
	SP4		Mean	ARX	Mean	Mean	ARX	Mean	Mean	Mean	NNG	Mean	Mean
$h=12$	SP1	PCA	Ridge	Mean	CADL	**FAAR**	**FAAR**	FAAR	**FAAR**	**Mean**	Mean	AR	Mean
		ICA	Mean	Mean	CADL	Mean	Mean	Mean	FAAR	CADL	Mean	AR	**Bagg**
		SPCA	**Mean**	**Mean**	**NNG**	Mean	Mean	**Mean**	Mean	Mean	**Mean**	**LARS**	Mean
	SP1L	PCA	**Mean**	**Mean**	**Boost**	**Mean**	**Mean**	**Mean**	**Mean**	**Mean**	**Boost**	**LARS**	AR
		ICA	Mean	Bagg	CADL	Mean	Mean	Mean	FAAR	Bagg	Mean	AR	**Bagg**
		SPCA	Mean	Mean	CADL	Mean	BMA2	Mean	Mean	Mean	Mean	AR	Mean
	SP2	PCA	Mean	Mean	**Mean**	**BMA1**	**Mean**	**Boost**	**Boost**	**Mean**	Mean	**LARS**	**LARS**
		ICA	Mean	Mean	CADL	Boost	Mean	EN	Boost	Mean	Mean	LARS	Mean
		SPCA	**Boost**	**Mean**	CADL	Mean	Mean	EN	Boost	Mean	**Mean**	LARS	Mean
	SP2L	PCA	Mean	**Mean**	**Mean**	**BMA1**	**Mean**	**Boost**	**Boost**	**Mean**	**Mean**	LARS	**LARS**
		ICA	Mean	Mean	**BMA2**	Boost	Mean	Boost	Boost	Mean	Mean	Mean	LARS
		SPCA	**Boost**	Mean	Mean	Mean	Mean	Mean	Boost	Mean	Mean	**BMA2**	LARS
	SP3		Boost	Boost	CADL	Mean	Mean	Boost	EN	EN	Mean	AR	EN
	SP4		Mean	Mean	CADL	Mean	Mean	Mean	Boost	Mean	Mean	AR	Mean

Panel B: Rolling Window Estimation

Forecast Horizon	Factor Spec. Mtd.		UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
$h=1$	SP1	PCA	**FAAR**	**PCR**	**Mean**	**FAAR**	**Mean**	**FAAR**	**ARX**	**PCR**	**FAAR**	**LARS**	**Mean**
		ICA	ARX	AR	Mean	Mean	Mean	Mean	**ARX**	ARX	Mean	**NNG**	Mean
		SPCA	ARX	AR	Mean	ARX	LARS	Mean	**ARX**	Mean	Mean	AR	Mean
	SP1L	PCA	**Mean**	**PCR**	**Mean**	**Mean**	**Mean**	**Mean**	**ARX**	**Mean**	**Mean**	**AR**	Mean
		ICA	ARX	AR	Mean	ARX	Mean	Mean	**ARX**	Mean	Mean	AR	**Mean**
		SPCA	ARX	AR	CADL	ARX	AR	Mean	**ARX**	Mean	Mean	AR	LARS
	SP2	PCA	**ARX**	**AR**	**Mean**	Mean	**LARS**	**Mean**	**ARX**	**Boost**	Mean	**EN**	**EN**
		ICA	**ARX**	AR	Mean	Mean	LARS	Mean	**ARX**	**Boost**	Mean	AR	**EN**
		SPCA	**ARX**	AR	Mean	**Boost**	LARS	Mean	**ARX**	Mean	**Mean**	AR	LARS
	SP2L	PCA	**ARX**	**AR**	**Mean**	Mean	**EN**	**Mean**	**ARX**	**BMA2**	**Mean**	**LARS**	**LARS**
		ICA	**ARX**	AR	Mean	Mean	EN	Mean	**ARX**	Boost	Mean	AR	**LARS**
		SPCA	**ARX**	AR	Mean	**Mean**	Mean	Mean	**ARX**	Boost	Mean	AR	LARS
	SP3		ARX	AR	CADL	Boost	AR	Boost	ARX	Boost	LARS	AR	EN
	SP4		ARX	AR	Boost	BMA2	Mean	Mean	ARX	Mean	Boost	AR	Mean
$h=3$	SP1	PCA	**Mean**	**PCR**	AR	**Mean**	**Mean**	**PCR**	**Boost**	**Mean**	**FAAR**	**LARS**	**Boost**
		ICA	Mean	Mean	PCR	Mean	Mean	Mean	Bagg	Mean	Bagg	AR	Mean
		SPCA	Mean	Mean	**BMA2**	BMA1	Mean	Mean	Mean	Mean	Mean	AR	Mean
	SP1L	PCA	**Mean**	**Mean**	**LARS**	**Mean**	**Mean**	**Mean**	**Boost**	**Mean**	**Mean**	**Mean**	**Mean**
		ICA	Mean	Mean	AR	BMA2	Boost	Mean	Boost	Mean	Mean	AR	Mean
		SPCA	Mean	Mean	AR	BMA2	NNG	Mean	Mean	Mean	Mean	AR	LARS
	SP2	PCA	**Mean**	**Mean**	NNG	Mean	Mean	**BMA2**	**Boost**	**Mean**	**EN**	**NNG**	**LARS**
		ICA	Mean	Mean	**BMA2**	Mean	**Mean**	Mean	Boost	Mean	EN	**NNG**	Mean
		SPCA	Boost	Mean	BMA1	**BMA2**	Mean	Mean	Boost	Mean	Mean	**NNG**	Mean
	SP2L	PCA	Boost	**Mean**	**BMA1**	Mean	Mean	**Boost**	**BMA2**	**Mean**	**Mean**	**NNG**	**Mean**
		ICA	**Boost**	Mean	BMA2	Mean	Mean	**Boost**	Boost	Boost	Boost	**NNG**	Mean
		SPCA	Mean	Mean	AR	**Mean**	**Mean**	Mean	Boost	Mean	Boost	**NNG**	Mean
	SP3		Boost	Boost	AR	Boost	NNG	Boost	Boost	Boost	Boost	AR	Boost
	SP4		Mean	Mean	AR	Mean	Mean	Boost	Mean	Boost	Boost	Mean	LARS
$h=12$	SP1	PCA	Mean	**Mean**	**CADL**	Mean	**PCR**	FAAR	Boost	**Mean**	**Mean**	AR	AR
		ICA	Mean	Mean	**CADL**	Ridge	Mean	Mean	FAAR	Mean	Mean	**Bagg**	Mean
		SPCA	**Mean**	Mean	**CADL**	**BMA2**	Mean	**Mean**	**Mean**	Mean	Mean	AR	**Mean**
	SP1L	PCA	**Mean**	**Mean**	CADL	Mean	**Mean**	Mean	Mean	**Mean**	**Mean**	**AR**	NNG
		ICA	Mean	Mean	CADL	Mean	Mean	**Mean**	Mean	Mean	Mean	AR	Bagg
		SPCA	Mean	NNG	**NNG**	**BMA2**	Boost	Mean	**Mean**	LARS	LARS	AR	**LARS**
	SP2	PCA	**Mean**	**Mean**	**CADL**	**Mean**	**Mean**	EN	**Boost**	**Mean**	Boost	NNG	Mean
		ICA	Mean	Mean	**CADL**	Mean	Mean	**EN**	Boost	Mean	Boost	NNG	Mean
		SPCA	Mean	Mean	**CADL**	Mean	Mean	EN	Boost	Mean	**Boost**	**LARS**	**Mean**
	SP2L	PCA	**Mean**	**Mean**	**CADL**	**Mean**	**Mean**	**Boost**	**Boost**	**Mean**	**Mean**	**BMA2**	Mean
		ICA	Mean	Mean	**CADL**	Mean	Mean	Boost	Boost	Mean	Boost	NNG	Mean
		SPCA	Mean	Mean	**CADL**	Mean	LARS	Boost	Boost	Mean	Boost	NNG	**Mean**
	SP3		Boost	Boost	CADL	EN	EN	Boost	Boost	Boost	Boost	AR	NNG
	SP4		Mean	Mean	CADL	Boost	Mean	Mean	Boost	Mean	Mean	NNG	EN

Panel C: Summary of Forecast Model Sepcification "Wins" Reported in Panels A and B

		Recursive Window Estimation							Rolling Window Estimation						
Horizon	Method	SP1	SP1L	SP2	SP2L	SP3	SP4	Total	SP1	SP1L	SP2	SP2L	SP3	SP4	Total
$h=1$	AR	0	1	0	0	0	0	1	3	6	5	5	3	2	24
	ARX	6	10	8	5	3	4	36	7	7	6	6	2	2	30
	CADL	0	1	0	0	1	0	2	0	1	0	0	1	0	2
	FAAR	10	1	0	0	0	0	11	4	0	0	0	0	0	4
	PCR	6	2	0	0	0	0	8	2	1	0	0	0	0	3
	Bagg	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Boost	2	2	6	10	1	0	21	0	0	3	2	3	2	10
	BMA1	1	1	2	1	0	1	6	0	0	0	0	0	0	0
	BMA2	1	0	1	1	0	0	3	0	0	0	1	0	1	2
	Ridge	2	0	0	0	0	0	2	0	0	0	0	0	0	0
	LAR	0	0	0	0	0	0	0	2	1	4	4	1	0	12
	EN	0	0	1	1	0	0	2	0	0	3	2	1	0	6
	NNG	0	0	0	0	0	0	0	1	0	0	0	0	0	1
	Mean	5	15	15	15	6	6	62	14	17	12	13	0	4	60
$h=3$	AR	1	1	0	0	1	0	3	3	4	0	1	2	1	11
	ARX	2	5	5	5	2	2	21	0	0	0	0	0	0	0
	CADL	1	1	1	1	1	0	5	0	0	0	0	0	0	0
	FAAR	7	0	0	0	0	0	7	1	0	0	0	0	0	1
	PCR	5	0	0	0	0	0	5	3	0	0	0	0	0	3
	Bagg	1	0	0	0	0	0	1	2	0	0	0	0	0	2
	Boost	1	0	10	10	2	0	23	2	3	4	9	8	3	29
	BMA1	0	1	0	0	0	0	1	1	0	1	1	0	0	3
	BMA2	0	1	0	1	1	0	3	1	2	3	2	0	0	8
	Ridge	1	0	0	0	0	0	1	0	0	0	0	0	0	0
	LAR	2	1	2	2	0	0	7	1	2	1	0	0	1	5
	EN	0	0	1	1	0	0	2	0	0	2	0	0	0	2
	NNG	1	5	0	0	0	1	7	0	1	4	3	1	0	9
	Mean	11	18	14	13	4	8	68	19	21	18	17	0	6	81
$h=12$	AR	2	3	0	0	1	1	7	3	3	0	0	1	0	7
	ARX	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	CADL	3	2	2	0	1	1	9	3	2	3	3	1	1	13
	FAAR	5	1	0	0	0	0	6	2	0	0	0	0	0	2
	PCR	0	0	0	0	0	0	0	1	0	0	0	0	0	1
	Bagg	1	3	0	0	0	0	4	1	1	0	0	0	0	2
	Boost	0	2	6	7	3	1	19	1	1	6	8	6	2	24
	BMA1	0	0	1	1	0	0	2	0	0	0	0	0	0	0
	BMA2	0	1	0	2	0	0	3	1	1	0	1	0	0	3
	Ridge	1	0	0	0	0	0	1	1	0	0	0	0	0	1
	LAR	1	1	4	4	0	0	10	0	3	1	1	0	0	5
	EN	0	0	2	0	3	0	5	0	0	3	0	2	1	6
	NNG	1	0	0	0	0	0	1	0	3	2	2	1	1	9
	Mean	19	20	18	19	3	8	87	20	19	18	18	0	6	81

*Notes: See notes to Tables 1-3. In Panels A and B, "winning" models, based on results reported in Table 4, for each factor estimation method are tabulated across forecast horizons and target forecast variable. Panel C summarizes results from Panels A and B, reporting the number of "wins" by forecast model specification.

Table 5: Summary of MSFE "Best" Factor Estimation Methods*

Panel A: Recursive Window Estimation

Specification	Horizon	UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
SP1	$h=1$	PCA	SPCA	SPCA	ICA	ICA	SPCA	SPCA	SPCA	PCA	SPCA	ALL
	$h=3$	PCA	PCA	ICA	SPCA	PCA	SPCA	PCA	SPCA	SPCA	PCA	SPCA
	$h=12$	SPCA	SPCA	SPCA	PCA	PCA	SPCA	PCA	PCA	SPCA	SPCA	ICA
SP1L	$h=1$	PCA	PCA	PCA	PCA	PCA	PCA	ALL	PCA	PCA	PCA	PCA
	$h=3$	PCA	PCA	SPCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA
	$h=12$	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	ICA
SP2	$h=1$	PCA	PCA	PCA	PCA	ICA	PCA	ALL	PCA	SPCA	PCA	ICA
	$h=3$	PCA	PCA	PCA	PCA	ALL	PCA	PCA	PCA	SPCA	SPCA	PCA
	$h=12$	SPCA	SPCA	PCA	PCA	PCA	PCA	PCA	PCA	SPCA	PCA	PCA
SP2L	$h=1$	PCA	PCA	PCA	ICA	SPCA	PCA	ALL	PCA	PCA	PCA	ICA
	$h=3$	PCA	PCA	PCA	PCA	ALL	ICA	PCA	SPCA	PCA	SPCA	PCA
	$h=12$	SPCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	SPCA	PCA

Panel B: Rolling Window Estimation

Specification	Horizon	UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
SP1	$h=1$	PCA	PCA	PCA	PCA	PCA	PCA	ALL	PCA	PCA	PCA	PCA
	$h=3$	PCA	PCA	SPCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA
	$h=12$	SPCA	PCA	PCA	SPCA	PCA	SPCA	SPCA	PCA	PCA	ICA	SPCA
SP1L	$h=1$	PCA	PCA	PCA	PCA	PCA	PCA	ALL	PCA	PCA	PCA	ICA
	$h=3$	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA
	$h=12$	PCA	PCA	SPCA	SPCA	PCA	ICA	SPCA	PCA	PCA	PCA	SPCA
SP2	$h=1$	PCA	PCA	PCA	SPCA	PCA	PCA	PCA	PCA	SPCA	PCA	PCA
	$h=3$	PCA	PCA	ICA	SPCA	ICA	PCA	PCA	PCA	PCA	PCA	PCA
	$h=12$	PCA	PCA	PCA	PCA	PCA	ICA	PCA	PCA	SPCA	SPCA	SPCA
SP2L	$h=1$	PCA	PCA	PCA	SPCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA
	$h=3$	ICA	PCA	PCA	SPCA	SPCA	PCA	PCA	PCA	PCA	PCA	PCA
	$h=12$	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	PCA	SPCA

Panel C: Summary of MSFE-best by PC Method

	Recursive Window Estimation									Rolling Window Estimation								
	$h=1$			$h=3$			$h=12$			$h=1$			$h=3$			$h=12$		
	PCA	ICA	SPCA	PCA	ICA	SPCA	PCA	ICA	SPCA	PCA	ICA	SPCA	PCA	ICA	SPCA	PCA	ICA	SPCA
SP1	2	2	6	5	1	5	4	1	6	10	0	0	10	0	1	5	1	5
SP1L	10	0	0	10	0	1	10	1	0	9	1	0	11	0	0	6	1	4
SP2	7	2	1	8	0	2	8	0	3	9	0	2	8	2	1	7	1	3
SP2L	7	2	1	7	1	2	9	0	2	10	0	1	8	1	2	10	0	1

* Notes: See notes to Table 4. Entries in Panels A and B of this table show which factor estimation method yields the lowest MSFE predictions. Cases where a benchmark model (AR,ARX and CADL) is MSFE "better" than PCA, ICA and SPCA in Table 4, are denoted by the entry "ALL"; otherwise, entries correspond to MSFE-best factor estimation methods reported in Table 4. Summarizing results from Panels A and B, entries in Panel C give counts of the number of factor estimation method "wins" by Specification type and forecast horizon, across all forecast target variables. Since there is no column for "ALL", count sums across individual row of entries do not always sum to eleven (the number of target forecast variables).

Table 6: Summary of Estimation Windowing Scheme Yielding MSFE "Best" Models*

Panel A: MSFE "Best" Models by Estimation Windowing Scheme Across Specification Types and Factor Estimation Method

Specification Type	Forecast Horizon	Fac. Est. Mtd.	UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
SP1	$h=1$	PCA	Recur	Recur	Recur	Roll	Recur	Recur	Roll	Recur	Roll	Recur	Recur
		ICA	Roll	Recur	Recur	Recur	Recur	Recur	Roll	Recur	Roll	Recur	Recur
		SPCA	Recur	Recur	Recur	Recur	Recur	Recur	Recur	Recur	Roll	Recur	Recur
	$h=3$	PCA	Roll	Recur	Recur	Roll	Recur	Roll	Roll	Roll	Roll	Recur	Roll
		ICA	Recur	Recur	Recur	Roll	Roll	Recur	Roll	Recur	Recur	Recur	Roll
		SPCA	Recur	Recur	Recur	Roll	Recur	Recur	Roll	Recur	Roll	Recur	Recur
	$h=12$	PCA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Recur	Recur
		ICA	Roll	Recur	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Recur
		SPCA	Roll	Recur	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Recur	Recur
SP1L	$h=1$	PCA	Recur	Recur	Roll	Roll	Recur	Roll	Roll	Recur	Roll	Recur	Recur
		ICA	Roll	Recur	Recur	Roll	Recur	Recur	Roll	Recur	Roll	Recur	Recur
		SPCA	Roll	Recur	Recur	Roll	Recur	Recur	Roll	Recur	Roll	Recur	Recur
	$h=3$	PCA	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Roll
		ICA	Roll	Recur	Recur	Roll	Recur	Roll	Roll	Roll	Roll	Recur	Roll
		SPCA	Recur	Recur	Recur	Roll	Recur	Roll	Roll	Roll	Roll	Recur	Roll
	$h=12$	PCA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Recur	Roll
		ICA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Recur	Recur
		SPCA	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Recur	Roll	Recur	Roll
SP2	$h=1$	PCA	Recur	Recur	Roll	Roll	Roll	Recur	Roll	Recur	Roll	Recur	Recur
		ICA	Roll	Recur	Recur	Roll	Roll	Recur	Roll	Roll	Roll	Recur	Recur
		SPCA	Roll	Recur	Recur	Roll	Roll	Recur	Roll	Recur	Roll	Recur	Recur
	$h=3$	PCA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Recur	Roll	Roll	Roll
		ICA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Recur	Roll	Roll	Roll
		SPCA	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Roll
	$h=12$	PCA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Recur
		ICA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Recur
		SPCA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Roll
SP2L	$h=1$	PCA	Recur	Recur	Roll	Roll	Roll	Recur	Roll	Recur	Roll	Recur	Recur
		ICA	Recur	Recur	Recur	Roll	Roll	Recur	Roll	Recur	Roll	Recur	Recur
		SPCA	Recur	Recur	Recur	Roll	Roll	Recur	Roll	Roll	Roll	Recur	Recur
	$h=3$	PCA	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Recur	Roll	Roll	Roll
		ICA	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Recur	Roll	Roll	Roll
		SPCA	Roll	Recur	Recur	Roll	Roll	Roll	Roll	Recur	Roll	Roll	Roll
	$h=12$	PCA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Recur
		ICA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Recur
		SPCA	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Recur	Recur
SP3	$h=1$		Roll	Recur	Recur	Roll	Recur	Recur	Roll	Roll	Roll	Recur	Recur
	$h=3$		Roll	Recur	Recur	Roll	Recur	Roll	Roll	Roll	Recur	Recur	Roll
	$h=12$		Roll	Recur	Recur	Roll	Roll	Roll	Roll	Recur	Roll	Roll	Recur
SP4	$h=1$		Roll	Recur	Roll	Roll	Roll	Recur	Roll	Recur	Roll	Recur	Recur
	$h=3$		Roll	Recur	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Recur	Roll
	$h=12$		Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Roll

Panel B: Count of MSFE "Best" Models By Estimation Window and Specification Type

	$h=1$		$h=3$		$h=12$	
	Recur	Roll	Recur	Roll	Recur	Roll
SP1	26	7	19	14	10	23
SP1L	20	13	9	24	8	25
SP2	17	16	5	28	5	28
SP2L	19	14	5	28	7	26
SP3	6	5	5	6	4	7
SP4	5	6	3	8	1	10
Total	93	61	46	108	35	119

* Note: See notes to Tables 1-5. Entries in Panel A indicate winning estimation windowing scheme across various measures including forecast horizon and specification type, for each target forecast variable. "Recur" refers recursive window estimation and "Roll" refers to rolling window esimation. Panel B entries are counts of "wins" across specification types, and hence summarize results from Panel A of the table.

Table 7: Prediction "Best" MSFEs By Specification Type and Forecast Horizon*

Panel A: "Best" MSFEs By Specification Type, for Each Target Forecast Variable

Forecast Horizon	Specification Type	UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
$h=1$	SP1	0.780	0.789	0.409	0.84	0.843	0.706	0.542	0.268	0.863	0.897	0.916
	SP1L	0.850	0.889	0.954	0.850	0.945	0.871	0.841	0.804	0.845	0.976	0.916
	SP2	0.861	0.950	0.963	0.844	0.936	0.854	0.841	0.833	0.888	0.985	0.867
	SP2L	0.861	0.950	0.964	0.840	0.948	0.854	0.841	0.833	0.886	0.985	0.871
	SP3	0.871	0.944	0.987	0.858	0.956	0.826	0.841	0.841	0.916	0.989	0.873
	SP4	0.871	0.964	0.977	0.828	0.946	0.865	0.841	0.829	0.899	0.986	0.916
$h=3$	SP1	0.882	0.866	0.975	0.861	0.910	0.775	0.769	0.816	0.914	0.994	0.937
	SP1L	0.904	0.889	0.981	0.848	0.920	0.807	0.744	0.820	0.908	0.988	0.953
	SP2	0.895	0.883	0.992	0.863	0.939	0.814	0.740	0.809	0.912	0.989	0.929
	SP2L	0.888	0.886	0.988	0.861	0.936	0.792	0.738	0.809	0.911	0.985	0.938
	SP3	0.911	0.902	0.998	0.906	0.945	0.839	0.683	0.844	0.939	1.001	0.970
	SP4	0.930	0.902	0.986	0.842	0.925	0.831	0.806	0.858	0.942	0.988	0.960
$h=12$	SP1	0.879	0.935	0.992	0.781	0.891	0.720	0.715	0.884	0.896	0.996	0.986
	SP1L	0.864	0.946	0.988	0.816	0.902	0.730	0.712	0.898	0.899	0.995	0.981
	SP2	0.893	0.929	0.992	0.818	0.912	0.691	0.637	0.880	0.879	0.991	0.982
	SP2L	0.883	0.927	0.992	0.816	0.903	0.714	0.624	0.888	0.880	0.993	0.982
	SP3	0.903	0.961	0.997	0.799	0.947	0.690	0.551	0.890	0.891	1.001	0.982
	SP4	0.882	0.937	0.997	0.804	0.912	0.702	0.616	0.886	0.902	0.997	0.985

Panel B: Summary of Winning Methods and Models by Forecast Horizon and Sepcification Type

Forecast Horizon	Specification Type	UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
$h=1$	SP1	Recur PCA FAAR	Recur SPCA PCR	Recur SPCA PCR	Recur ICA FAAR	Recur ICA FAAR	Recur SPCA Mean	Recur SPCA FAAR	Recur SPCA FAAR	Roll PCA FAAR	Recur SPCA Boost	Recur PCA ARX
	SP1L	Recur PCA FAAR	Recur PCA PCR	Roll PCA Mean	Roll PCA Mean	Recur PCA Mean	Roll PCA Mean	Roll PCA ARX	Recur PCA BMA1	Roll PCA Mean	Recur PCA Boost	Recur PCA ARX
	SP2	Recur PCA Boost	Recur PCA Mean	Roll PCA Mean	Roll SPCA Boost	Roll PCA LARS	Recur PCA Mean	Roll PCA ARX	Recur PCA BMA1	Roll SPCA Mean	Recur PCA Mean	Recur ICA Boost
	SP2L	Recur PCA Boost	Recur PCA Mean	Roll PCA Mean	Roll SPCA Mean	Roll PCA EN	Recur PCA Mean	Roll PCA ARX	Recur PCA BMA1	Roll PCA Mean	Recur PCA Mean	Recur PCA Boost
	SP3	Roll ARX	Recur Mean	Recur CADL	Roll Boost	Recur Mean	Recur Mean	Roll ARX	Roll Boost	Roll LARS	Recur Boost	Recur Mean
	SP4	Roll ARX	Recur Mean	Roll Boost	Roll BMA2	Roll Mean	Recur Mean	Roll ARX	Recur BMA1	Roll Boost	Recur Mean	Recur ARX
$h=3$	SP1	Roll PCA Mean	Recur PCA PCR	Recur ICA PCR	Roll PCA Mean	Recur PCA PCR	Recur SPCA Ridge	Roll PCA Boost	Recur SPCA FAAR	Roll PCA FAAR	Recur PCA LARS	Roll PCA Boost
	SP1L	Roll PCA Mean	Roll PCA Mean	Roll PCA LARS	Roll PCA Mean	Roll PCA Mean	Roll PCA Mean	Roll PCA Boost	Roll PCA Mean	Roll PCA Mean	Roll PCA Mean	Roll PCA Mean
	SP2	Roll PCA Mean	Roll PCA Mean	Recur PCA EN	Roll SPCA BMA2	Roll ICA Mean	Roll PCA BMA2	Roll PCA Boost	Recur PCA Mean	Roll PCA EN	Roll PCA NNG	Roll PCA LARS
	SP2L	Roll ICA Boost	Roll PCA Mean	Roll PCA BMA1	Roll SPCA Mean	Roll SPCA Mean	Roll PCA Boost	Roll PCA BMA2	Recur PCA Mean	Roll PCA Mean	Roll PCA NNG	Roll PCA Mean
	SP3	Roll Boost	Recur ARX	Recur CADL	Roll Boost	Recur ARX	Roll Boost	Roll Boost	Roll Boost	Recur Mean	Recur AR	Roll Boost
	SP4	Roll Mean	Recur ARX	Recur Mean	Roll Mean	Roll Mean	Roll Boost	Roll Mean	Roll Boost	Roll Boost	Recur Mean	Roll LARS
$h=12$	SP1	Roll SPCA Mean	Roll PCA Mean	Recur SPCA NNG	Roll SPCA BMA2	Roll PCA PCR	Roll SPCA Mean	Roll SPCA Mean	Roll PCA Mean	Roll PCA Mean	Recur SPCA LARS	Recur ICA Bagg
	SP1L	Roll PCA Mean	Roll PCA Mean	Recur PCA Boost	Roll SPCA BMA2	Roll PCA Mean	Roll ICA Mean	Roll SPCA Mean	Roll PCA Mean	Roll PCA Mean	Recur PCA LARS	Recur ICA Bagg
	SP2	Roll PCA Mean	Roll PCA Mean	Recur PCA Mean	Roll PCA Mean	Roll PCA Mean	Roll ICA EN	Roll PCA Boost	Roll PCA Mean	Roll SPCA Boost	Roll SPCA LARS	Recur PCA LARS
	SP2L	Roll PCA Mean	Roll PCA Mean	Recur PCA Mean	Roll PCA Mean	Roll PCA Mean	Roll PCA Boost	Roll PCA Boost	Roll PCA Mean	Roll PCA Mean	Roll PCA BMA2	Recur PCA LARS
	SP3	Roll Boost	Recur Boost	Recur CADL	Roll EN	Roll EN	Roll Boost	Roll Boost	Recur EN	Roll Boost	Roll AR	Recur EN
	SP4	Roll Mean	Roll Mean	Recur CADL	Roll Boost	Roll Mean	Roll Mean	Roll Boost	Roll Mean	Roll Mean	Roll NNG	Roll EN

Panel C: Summary of Winning Methods and Models by Forecast Horizon

Forecast Horizon	UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
$h = 1$	SP1	SP1	SP1	SP4	SP1	SP1	SP1	SP1	SP1L	SP1	SP2
	Recur	Recur	Recur	Roll	Recur	Recur	Recur	Recur	Roll	Recur	Recur
	PCA	SPCA	SPCA	N/A	ICA	SPCA	SPCA	SPCA	PCA	SPCA	ICA
	FAAR	PCR	PCR	BMA2	FAAR	Mean	FAAR	FAAR	Mean	Boost	Boost
$h = 3$	SP1	SP1	SP1	SP4	SP1	SP1	SP3	SP2	SP1L	SP2L	SP2
	Roll	Recur	Recur	Roll	Recur	Recur	Roll	Recur	Roll	Roll	Roll
	PCA	PCA	ICA	N/A	PCA	SPCA	N/A	PCA	PCA	PCA	PCA
	Mean	PCR	PCR	Mean	PCR	Ridge	Boost	Mean	Mean	NNG	LARS
$h = 12$	SP1L	SP2L	SP1L	SP1	SP1	SP3	SP3	SP2	SP2	SP2	SP1L
	Roll	Roll	Recur	Roll	Roll	Roll	Roll	Roll	Roll	Roll	Recur
	PCA	PCA	PCA	SPCA	PCA	N/A	N/A	PCA	SPCA	SPCA	ICA
	Mean	Mean	Boost	BMA2	PCR	Boost	Boost	Mean	Boost	LARS	Bagg

* Notes: See notes to Tables 1-6. Entries in Panel A are lowest relative MSFEs across specification type, for each forecast horizon and target forecast variable. Thus, entries report "best" MSFEs across all factor estimation methods. Entries in Panel B break down the information from Panel A by listing, for each MSFE in Panel A, the winning (Prediction Model,Factor Estimation Method,Estimation Windowing Scheme) triple, for each forecast horizon and model specification type. Panel C summarizes the results of Panel A by aggregating over specification types, hence reporting the "ultimate" winning permutations. Since Specification types 3 and 4 do not involve factor estimation, third rows of entries are reported as "N/A" in cases where either of these two specifcation types win. Note that benchmark models, including AR and ARX models, are never MSFE-best across all specification types, for a given forecasting horizon and variable.

Table 8: Comparison of "Best" Factor Estimation Methods With and Without Lags*

Specification		UR	PI	TB10Y	CPI	PPI	NPE	HS	IPX	M2	SNP	GDP
SP1	PCA	No Lag	No Lag	No Lag	Lag	No Lag	No Lag	No Lag	No Lag	Lag	Lag	N/A
	ICA	Lag	No Lag	No Lag	No Lag	No Lag	No Lag	Lag	No Lag	No Lag	No Lag	N/A
	SPCA	No Lag	No Lag	No Lag	No Lag	No Lag	No Lag	No Lag	No Lag	Lag	No Lag	N/A
SP2	PCA	No Lag	No Lag	No Lag	No Lag	No Lag	No Lag	No Lag	No Lag	No Lag	No Lag	No Lag
	ICA	Lag	Lag	No Lag	Lag	No Lag	No Lag	No Lag	Lag	No Lag	No Lag	No Lag
	SPCA	Lag	No Lag	Lag	No Lag	Lag	Lag	No Lag	No Lag	No Lag	No Lag	No Lag

* Notes: Entries denotes which factor estimation method (with lags or without lags) is MSFE "best" under each specification type (i.e., SP1 and SP2). For GDP, results are not available no factor-based forecasting model ever yields the lowest point MSFE.