International Journal of Forecasting 32 (2016) 736–753

Contents lists available at ScienceDirect

International Journal of Forecasting

journal homepage: www.elsevier.com/locate/ijforecast

Nonlinear forecasting with many predictors using kernel ridge regression

Peter Exterkate [a,b,*], Patrick J.F. Groenen [c,d], Christiaan Heij [c], Dick van Dijk [c,d,e]

[a] *School of Economics, University of Sydney, Australia*
[b] *CREATES, Aarhus University, Denmark*
[c] *Econometric Institute, Erasmus University Rotterdam, The Netherlands*
[d] *Erasmus Research Institute of Management, Erasmus University Rotterdam, The Netherlands*
[e] *Tinbergen Institute, Erasmus University Rotterdam, The Netherlands*

ARTICLE INFO

Keywords:
High dimensionality
Nonlinear forecasting
Ridge regression
Kernel methods

ABSTRACT

This paper puts forward kernel ridge regression as an approach for forecasting with many predictors that are related to the target variable nonlinearly. In kernel ridge regression, the observed predictor variables are mapped nonlinearly into a high-dimensional space, where estimation of the predictive regression model is based on a shrinkage estimator in order to avoid overfitting. We extend the kernel ridge regression methodology to enable its use for economic time series forecasting, by including lags of the dependent variable or other individual variables as predictors, as is typically desired in macroeconomic and financial applications. Both Monte Carlo simulations and an empirical application to various key measures of real economic activity confirm that kernel ridge regression can produce more accurate forecasts than traditional linear and nonlinear methods for dealing with many predictors based on principal components.

1. Introduction

Current practice involves forecasters in macroeconomics and finance facing a trade-off between model complexity and forecast accuracy. The uncertainty associated with model choice and parameter estimation means that a highly complex predictive regression model based on many variables or intricate nonlinear structures is often found to produce forecasts that are less accurate than those from a simpler model that discards some of the information that is at the researcher's disposal.

Various methods for working with many predictors while circumventing this *curse of dimensionality* in a linear framework have been applied in the recent forecasting literature, as was surveyed by Stock and Watson (2006). Most prominently, Stock and Watson (2002) advocate summarizing large panels of predictor variables into a small number of principal components, which are then used in a dynamic factor model for forecasting purposes. Alternative approaches include combining forecasts based on multiple models, where each includes only a relatively small number of variables (Aiolfi & Favero, 2005; Faust & Wright, 2009; Huang & Lee, 2010; Rapach, Strauss, & Zhou, 2010; Wright, 2009), partial least squares (Groen & Kapetanios, 2008), and Bayesian regression (Bańbura, Giannone, & Reichlin, 2010; Carriero, Kapetanios, & Marcellino, 2011; De Mol, Giannone, & Reichlin, 2008). Stock and Watson (2012) find that the dynamic factor model approach is preferable to these alternatives for forecasting macroeconomic time series; see also Çakmaklı and van Dijk (2010) and Ludvigson and Ng (2007, 2009) for successful applications in finance.

[*] Correspondence to: School of Economics, Merewether Building H04, The University of Sydney, NSW 2006, Australia. Tel.: +61 2 9351 8532.
E-mail address: peter.exterkate@sydney.edu.au (P. Exterkate).

http://dx.doi.org/10.1016/j.ijforecast.2015.11.017
0169-2070/© 2016 International Institute of Forecasters. Published by Elsevier B.V. All rights reserved.

The possibility of the existence of nonlinear relationships among macroeconomic and financial time series has also received ample attention over the last two decades. The most popular nonlinear forecast methods include regime-switching models and neural networks, see the surveys by Teräsvirta (2006) and White (2006), respectively, and the comprehensive overview by Kock and Teräsvirta (2011). Alternative approaches include sieve estimation (Chen, 2007) and nonparametric regression (Pagan & Ullah, 1999). Typically, these approaches are suitable only for small numbers of predictors, and their ability to improve upon the predictive accuracy of linear forecasting techniques seems limited, see Medeiros, Teräsvirta, and Rech (2006), Stock and Watson (1999), and Teräsvirta, van Dijk, and Medeiros (2005), among others. Giovannetti (2013) proposes a hybrid approach, where a nonlinear model is estimated using principal components extracted from a large set of predictors.

In this paper, we introduce a forecasting technique that can deal with high-dimensionality and nonlinearity simultaneously. The central idea is to employ a flexible set of nonlinear prediction functions and to prevent overfitting by penalization, in a way that limits the computational complexity. In this approach, which is known as *kernel ridge regression*, the set of predictors is mapped into a high-dimensional (often even infinite-dimensional) space of nonlinear functions of the predictors. A forecast equation is estimated in this infinite-dimensional space, using a penalty (or shrinkage, or ridge) term to avoid overfitting. This allows kernel ridge regression to avoid the curse of dimensionality, which plagues alternative nonparametric approaches when allowing for flexible types of nonlinearity. Computational tractability is achieved by choosing the kernel in a convenient way, so that calculations in the infinite-dimensional space are actually prevented. This approach also avoids the computational difficulties that are encountered in standard linear ridge regression when the number of predictor variables is large relative to the number of time series observations. These properties mean that kernel ridge regression provides an attractive framework for estimating nonlinear predictive relationships in a data-rich environment.

The kernel methodology was developed in the machine learning community, an area which often involves large data sets. The terminology originates from operator theory, as computations are performed in terms of the kernel of a positive integral operator, see Vapnik (1995). We use the term *kernel* in this sense because it is the established term for this method in machine learning. This meaning should not be confused with other uses of the word, such as in kernel smoothing methods for local regression.

A typical application of kernel methods is classification; for example, in the optical recognition of pixel-by-pixel scans of handwritten characters. Schölkopf, Smola, and Müller (1998) document the outstanding performance of kernel methods for this classification task. Kernel ridge regression has also been found to work well in many other applications. Time series applications are scarce, and seem to be limited to deterministic (that is, non-stochastic) time series (Müller et al., 1997). On the other hand, linear penalized regression methods, including linear ridge regression, are used widely in economic forecasting; see the recent overview by Kim and Swanson (2014). To the best of our knowledge, kernel ridge regression has not yet been applied in the context of macroeconomic or financial time series forecasting.

This paper makes two methodological contributions to kernel ridge regression. First, we extend the approach to enable the use of models that include lags of the dependent variable or other individual variables as predictors, as is typically desired in economic and financial forecasting applications. Second, we derive a computationally efficient cross-validation procedure for selecting the tuning parameters involved in kernel ridge regression, including the shrinkage parameter that determines the strength of the penalization.

We provide simulation evidence demonstrating that kernel ridge regression delivers more accurate forecasts than conventional principal components methods in the presence of many predictors that are related nonlinearly to the target variable. These conventional methods include threshold autoregressions, extensions of principal component regressions to accommodate nonlinearity, as put forward by Bai and Ng (2008), sieves, and standard nonparametric regression techniques. The practical usefulness of kernel methods is confirmed in an empirical application to the forecasting of four key measures of US macroeconomic activity over the period 1970–2009: industrial production, personal income, manufacturing and trade sales, and employment. When traditional methods perform poorly, kernel ridge regression yields substantial improvements. This result holds for all series at a one-year horizon, and also at shorter horizons for production and income. When traditional forecasts are of good quality, as is the case for sales and employment, kernel-based forecasts remain competitive. We also find that kernel ridge regression is affected less by the 2008–09 financial and economic crisis than traditional methods.

The remainder of this paper is organized as follows. Section 2 describes the kernel methodology. The Monte Carlo simulation is presented in Section 3, and Section 4 discusses the empirical application. Conclusions are provided in Section 5. Details of the technical results are collected in Appendix A.

2. Methodology

The technique of kernel ridge regression (KRR) is based on ordinary least squares (OLS) regression and ridge regression. Therefore, we begin this section by providing a brief review of these methods, highlighting their limitations for dealing with nonlinearity and high-dimensionality. Next, we show how kernel ridge regression overcomes these drawbacks by means of the so-called *kernel trick*. We extend the KRR methodology to allow for "preferred" predictors, in order to enable it to be used in time series contexts. We also present the properties of some kernel functions that are popular because of their computational efficiency. As will become clear below, kernel ridge regression involves the use of tuning parameters. In Section 2.5 we propose a computationally efficient cross-validation procedure for selecting values for these parameters.

2.1. Preliminaries

We consider the following general setup for forecasting. At the end of period T, we wish to forecast the value of a target variable y at a specific future date, denoted y_*, given an $N \times 1$ vector of predictors x_*. Historical observations for $t = 1, \ldots, T$ are available for all variables, collected in the $T \times 1$ vector y and the $T \times N$ matrix X. If we assume a linear prediction function $\hat{y}_* = x_*' \hat{\beta}$, we may obtain $\hat{\beta}$ by minimizing the OLS criterion $\|y - X\beta\|^2$, where $\|e\| = \sqrt{e'e}$ denotes the L_2 norm. Provided that X has rank N, the solution is $\hat{\beta} = (X'X)^{-1} X'y$, which leads to the forecast $\hat{y}_* = x_*' (X'X)^{-1} X'y$.

The OLS procedure presupposes that $N \leq T$, and, in practice, $N \ll T$ is required in order to prevent overfitting problems. That is, if N is not small compared to T, although we may obtain a good in-sample fit (indeed, if $N = T$ and X has full rank, the in-sample fit will be perfect), the out-of-sample prediction $\hat{y}_*$ will generally be found to be of poor quality. One possible solution to this problem would be shrinkage estimation or ridge regression, which aims to balance the goodness-of-fit with the magnitude of the coefficient vector β.[1] The ridge criterion is given by $\|y - X\beta\|^2 + \lambda \|\beta\|^2$, where the penalty parameter $\lambda > 0$ is to be specified by the user. As every element of the parameter vector β is penalized equally, the predictors in X should be scaled appropriately. In our applications, we studentize each column of X over the estimation sample, so that each predictor has a zero mean and unit variance. The solution $\hat{\beta}$ that minimizes the ridge criterion can be found most easily by defining the $(T + N) \times 1$ vector $u = \left(y', 0_{N\times 1}'\right)'$ and the $(T + N) \times N$ matrix $V = \left(X', \sqrt{\lambda} I_N\right)'$, where I_N denotes the N-dimensional identity matrix. We may then write $\|y - X\beta\|^2 + \lambda \|\beta\|^2 = \|u - V\beta\|^2$. Minimizing this criterion using OLS yields $\hat{\beta} = (V'V)^{-1} V'u$, or, in terms of the original variables, $\hat{\beta} = (X'X + \lambda I_N)^{-1} X'y$. The resulting forecast $\hat{y}_* = x_*' (X'X + \lambda I_N)^{-1} X'y$ can be computed even if the number of predictors N is larger than the number of observations T. Nevertheless, if N becomes very large, the direct calculation of the ridge forecast may present computational difficulties, as it involves inverting the $N \times N$ matrix $X'X + \lambda I_N$, which becomes demanding when $N \gg T$.

2.2. Kernel ridge regression and the kernel trick

Kernel ridge regression extends the general setup considered above to allow for nonlinear prediction functions $\hat{y}_* = f(x_*)$. At the same time, it also provides a way to avoid the computational complications that are involved in the production of the ridge forecast when the number of predictors becomes very large. As will become clear below, this is particularly relevant in the context of nonlinear forecasting. Henceforth, let $\varphi : \mathbb{R}^N \to \mathbb{R}^M$ be a (possibly nonlinear) mapping of the N observed predictor variables x, resulting in M transformed predictor variables $z = \varphi(x)$. We assume that the prediction function is linear in z, say $\hat{y}_* = z_*' \hat{\gamma}$, where $z_* = \varphi(x_*)$. Collecting the transformed predictor variables in the $T \times M$ matrix Z with rows $z_t' = \varphi(x_t)'$, we may apply ridge regression to obtain $\hat{\gamma} = (Z'Z + \lambda I_M)^{-1} Z'y$, and hence,

$$\hat{y}_* = z_*' (Z'Z + \lambda I_M)^{-1} Z'y. \tag{1}$$

In macroeconomic and financial applications, we are often working with high-dimensional data, sometimes with the number of observed predictors N exceeding the number of time series observations T. Moreover, $M \gg N$ is needed to allow for flexible forms of nonlinearity in the forecast equation. For example, if one approximates the unknown forecast function f by a dth order Taylor expansion, the mapping φ effectively transforms the $N \times 1$ vector x into the $M \times 1$ vector z containing powers and cross-products of its elements, with M proportional to N^d. Thus, M may become very large for realistic values of N and d. As the matrix $Z'Z$ has dimensions $M \times M$, this can cause computational difficulties. Moreover, as we shall discuss below, we are also interested in making M infinite for fixed values of N and T, thus rendering the computation in Eq. (1) infeasible.

An efficient method of alleviating this curse of dimensionality problem is provided by the so-called kernel trick. This method is essentially based on the idea that if the number of regressors M is much larger than the number of observations T, working with T-dimensional instead of M-dimensional objects can lead to notable computational savings. To appreciate the dimension reductions involved, consider the macroeconomic application that will be discussed in Section 4. In this application, we estimate models with $N = 132$ predictors on an estimation sample containing $T = 120$ observations. One of the models includes a constant, all observed predictors, their squares, and all pairwise cross-products, leading to a total of $M = (N + 1)(N + 2)/2 = 8911$ transformed predictor variables. The results described in the remainder of this section allow us to work with a 120×120 matrix instead of the 8911×8911 matrix $Z'Z$, which is a sizeable simplification. What is more, as we shall see in Section 2.4, the kernel trick can also be applied in cases with $M = \infty$, where standard ridge regression cannot be applied.

This dimension reduction can be achieved by relatively straightforward algebraic manipulations of the expression of the nonlinear ridge forecast equation $\hat{y}_* = z_*' \hat{\gamma}$. First, we rewrite the ridge regression estimator $\hat{\gamma} = (Z'Z + \lambda I_M)^{-1} Z'y$ as $Z'Z\hat{\gamma} + \lambda\hat{\gamma} = Z'y$, or

$$\hat{\gamma} = \frac{1}{\lambda}\left(Z'y - Z'Z\hat{\gamma}\right) = \frac{1}{\lambda} Z'\left(y - Z\hat{\gamma}\right).$$

Pre-multiplying $Z'Z\hat{\gamma} + \lambda\hat{\gamma} = Z'y$ by the matrix Z gives $ZZ'Z\hat{\gamma} + \lambda Z\hat{\gamma} = ZZ'y$, or

$$Z\hat{\gamma} = (ZZ' + \lambda I_T)^{-1} ZZ'y.$$

[1] Several alternative solutions have been proposed in the literature, most prominently the Lasso (Tibshirani, 1996) and other bridge estimators (Frank & Friedman, 1993). In the cases studied here, N will either be very large (placing these alternative estimators at a severe computational disadvantage) or even infinite (making them infeasible). For this reason, such alternative shrinkage techniques are not considered in the remainder of this paper.

Combining these two results, we find

$$\begin{aligned}
\hat{y}_* = z_*'\hat{\gamma} &= \frac{1}{\lambda} z_*' Z' \left(y - Z\hat{\gamma}\right) \\
&= \frac{1}{\lambda} z_*' Z' \left(y - \left(ZZ' + \lambda I_T\right)^{-1} ZZ'y\right) \\
&= \frac{1}{\lambda} z_*' Z' \left(ZZ' + \lambda I_T\right)^{-1} \left(ZZ' + \lambda I_T - ZZ'\right) y \\
&= z_*' Z' \left(ZZ' + \lambda I_T\right)^{-1} y.
\end{aligned}$$

If we define the $T \times T$ matrix $K = ZZ'$ and the $T \times 1$ vector $k_* = Zz_*$, this result can be written as

$$\hat{y}_* = k_*' \left(K + \lambda I_T\right)^{-1} y. \tag{2}$$

The forecast $\hat{y}_*$ in Eq. (2) is identical to the one in Eq. (1). The advantage of using Eq. (2) is that the inverse matrix in this equation has dimensions $T \times T$, so that the $M \times M$-dimensional computations in Eq. (1) are prevented.

To achieve computational savings over the straightforward application of a ridge regression, it is crucial that K and k_* be able to be computed in a relatively simple way. The (s, t)-th element of $K = ZZ'$ equals $z_s'z_t = \varphi\left(x_s\right)' \varphi\left(x_t\right)$, and similarly, the tth element of k_* equals $\varphi\left(x_t\right)' \varphi\left(x_*\right)$. This implies that the computational efficiency increases greatly if we choose a mapping φ for which the inner product $\kappa\left(a, b\right) = \varphi\left(a\right)' \varphi\left(b\right)$ can be computed quickly, that is, without computing $\varphi\left(a\right)$ and $\varphi\left(b\right)$ explicitly. In this context, κ is called the *kernel function* and K is the *kernel matrix*. This procedure for finding the optimal parameter vector $\hat{\gamma}$ in the "high" dimension M implicitly while working exclusively in the "low" dimension T is known as the *kernel trick*, and is due to Boser, Guyon, and Vapnik (1992).

As this discussion shows, KRR is no different from ordinary ridge regression for transformations of the regressors, apart from the inclusion of an algebraic trick for improving the computational efficiency. The key to a successful application of this kernel trick is to choose a mapping φ that leads to an easy-to-compute kernel function κ, while, obviously, the corresponding prediction function $\varphi\left(x_*\right)' \gamma$ should provide a good approximation of the unknown true nonlinear prediction function $f\left(x_*\right)$. Various such mappings are known, and an overview is given by Smola and Schölkopf (2004). Section 2.4 presents the mappings used in our study.

2.3. Kernel ridge regression for time series

In a time series context, we often prefer to include specific predictors in the forecast equation separately from the nonlinear mapping φ. In macroeconomic applications, these "preferred" predictors may include lags of the dependent variable, to account for serial correlation. In financial applications, such as the prediction of stock returns, these predictors may include valuation ratios such as the dividend yield or interest rate related variables; see for example Çakmaklıand van Dijk (2010) or Ludvigson and Ng (2007). In such cases, the generalized forecast equation takes the form $\hat{y}_* = w_*'\hat{\beta} + z_*'\hat{\gamma}$, where the $P \times 1$ vector w_* contains the variables to be treated linearly. As the number of these additional predictors is limited and the effects of these predictors are of particular interest, we do not penalize the parameters β, and restrict the ridge penalization to γ. We show in Appendix A.1 that the derivations that lead to Eq. (2) can be extended to include such linear unpenalized terms, resulting in the "extended" KRR forecast equation

$$\hat{y}_* = \begin{pmatrix} k_* \\ w_* \end{pmatrix}' \begin{pmatrix} K + \lambda I_T & W \\ W' & 0 \end{pmatrix}^{-1} \begin{pmatrix} y \\ 0 \end{pmatrix}, \tag{3}$$

where the $T \times P$ matrix W contains the historical observations on the variables to be treated linearly. This is the forecast equation that will be used in the empirical application in Section 4.

2.4. Some common kernel functions

A first and obvious example is the identity mapping $\varphi\left(a\right) = a$, for which $\kappa\left(a, b\right) = a'b$. With this choice of κ, the kernel forecast $\hat{y}_* = k_*'\left(K + \lambda I_T\right)^{-1} y = x_*'X'\left(XX' + \lambda I_T\right)^{-1} y$ equals the linear ridge forecast $\hat{y}_* = x_*'\left(X'X + \lambda I_N\right)^{-1} X'y$, as can be seen by taking $Z = X$ and $z_* = x_*$ in the derivations leading to Eq. (2).

Next, we consider a mapping such that $\varphi\left(a\right)$ contains a constant term, all variables $a_1, a_2, \ldots, a_N$, and all of their squares and cross products. The kernel function $\kappa\left(a, b\right)$ takes a particularly simple form if we multiply the linear and cross-product terms in $\varphi\left(a\right)$ by the constant $\sqrt{2}$. That is, if we choose

$$\varphi\left(a\right) = \Big(1, \sqrt{2}a_1, \sqrt{2}a_2, \ldots, \sqrt{2}a_N, a_1^2, a_2^2, \ldots, a_N^2, \sqrt{2}a_1a_2, \sqrt{2}a_1a_3, \ldots, \sqrt{2}a_{N-1}a_N\Big)',$$

the corresponding kernel function is

$$\begin{aligned}
\kappa\left(a, b\right) &= \varphi\left(a\right)' \varphi\left(b\right) \\
&= 1 + 2\left(a_1b_1 + a_2b_2 + \cdots + a_Nb_N\right) \\
&\quad + a_1^2b_1^2 + a_2^2b_2^2 + \cdots + a_N^2b_N^2 \\
&\quad + 2\left(a_1a_2b_1b_2 + a_1a_3b_1b_3 \right.\\
&\quad \left. + \cdots + a_{N-1}a_Nb_{N-1}b_N\right) \\
&= 1 + 2\left(a_1b_1 + a_2b_2 + \cdots + a_Nb_N\right) \\
&\quad + \left(a_1b_1 + a_2b_2 + \cdots + a_Nb_N\right)^2 \\
&= 1 + 2\,a'b + \left(a'b\right)^2 = \left(1 + a'b\right)^2.
\end{aligned}$$

This specification of the kernel function means that the computation of each of the $T\left(T + 1\right)/2$ distinct elements of the kernel matrix K requires $2N + 1$ additions and multiplications. In the absence of the indicated scaling, the vector of constant, first-order, and second-order terms contains a total of $M = \left(N + 1\right)\left(N + 2\right)/2$ elements. The computation of each element of the kernel matrix would then require $2M - 1 = N^2 + 3N + 1$ additions and multiplications. Thus, the proposed scaling reduces the computational complexity to being linear instead of quadratic in the number of observed predictors.

As was noted by Poggio (1975), this result can be generalized to the kernel function

$$\kappa\left(a, b\right) = \left(1 + a'b\right)^d \quad \text{for any integer } d \geq 1, \tag{4}$$

corresponding to a mapping for which $\varphi(a)$ consists of all polynomials in the elements of a of degree at most d. Observe that this class of so-called polynomial kernel functions encompasses not only the quadratic mapping, for $d = 2$, but also standard linear ridge regression, for $d = 1$. We shall refer to this kernel as "Poly(d)".

Because smart choices of φ enable us to avoid M-dimensional computations, the kernel methodology even allows the use of mappings for which M is infinite. A common way to do this, dating back to Broomhead and Lowe (1988), involves the use of the Gaussian kernel function

$$\kappa(a, b) = \exp\left(-\frac{1}{2}\|a-b\|^2\right), \qquad (5)$$

henceforth referred to as "Gauss". We show in Appendix A.2 that, for all degrees $d_1, d_2, \ldots, d_N \geq 0$, the corresponding mapping $\varphi(a)$ contains as elements the "dampened" polynomials

$$e^{-a'a/2}\prod_{n=1}^{N}\frac{a_n^{d_n}}{\sqrt{d_n!}}.$$

To control for the relative importance of the terms in $\varphi(x)$, we replace each observation x with $(1/\sigma)x$ before computing κ, for some positive scaling factor σ. Such scaling affects the weight placed on different polynomial degrees, as it amounts to dividing linear terms in $\varphi(x)$ by σ, second-order terms by σ^2, and so forth. Although we perform linear regression on $\varphi(x)$, such scaling is not without effect, as the regression coefficients in the forecast equation $\hat{y}_* = w_*'\hat{\beta} + \varphi(x_*)'\hat{\gamma}$ are all penalized equally by the ridge term in the criterion function $\|y - W\beta - Z\gamma\|^2 + \lambda\|\gamma\|^2$.

2.5. Selection of tuning parameters

The implementation of kernel ridge regression involves two tuning parameters, namely, the shrinkage parameter λ and the scaling parameter σ. In addition, our empirical application in Section 4 also involves the selection of lag lengths, which can also be seen as tuning parameters from a model selection perspective. This section addresses the issue of the selection of the values of these tuning parameters.

We determine the values of the tuning parameters by means of k-fold cross-validation, as this is a natural criterion for the purpose of out-of-sample forecasting. For given values of the tuning parameters, we estimate the model on the sample of size $\left(1-\frac{1}{k}\right)T$ that remains when a contiguous block of $\frac{1}{k}T$ observations is removed. We then use this model to "forecast" the values of y_t that were left out. This is repeated k times, leaving out each observation for $t = 1, 2, \ldots, T$ once. Performing this cross-validation exercise for each of a set of candidate values for the tuning parameters, we select the values that lead to the smallest mean squared prediction errors (MSPE) over these T forecasts. These values are then used to estimate the model on the full sample $t = 1, 2, \ldots, T$, from which we produce out-of-sample forecasts.

In the form stated above, this cross-validation procedure is computationally very expensive, as it requires the model to be estimated on k different samples for each possible setting of the tuning parameters. Cawley and Talbot (2008) propose a method that yields all of the leave-one-out (i.e., $k = T$) prediction errors that are obtained as a by-product of estimating Eq. (2) only once, that is, on the full sample. We derive a similar result, extended to allow for different values of k and for the additional linear terms in Eq. (3), in Appendix A.3.

In our simulation study and empirical application, we use five-fold cross-validation to select two lag lengths, the ridge parameter λ, and the scaling parameter σ simultaneously from pre-specified grids. For the lag lengths, we employ the grids specified by Stock and Watson (2002). For the KRR parameters, we construct grids based on an estimate of the signal-to-noise ratio (for λ) and a smoothness assumption (for σ). Specifically, we select $\sigma \in \left\{\frac{1}{2}\sigma_0, \sigma_0, 2\sigma_0, 4\sigma_0, 8\sigma_0\right\}$ and $\lambda \in \left\{\frac{1}{8}\lambda_0, \frac{1}{4}\lambda_0, \frac{1}{2}\lambda_0, \lambda_0, 2\lambda_0\right\}$, where

$$\sigma_0 = \sqrt{N/2} \quad \text{and}$$

$$\lambda_0 = \left(1 + N\sigma^{-2}\right)\left(1-\hat{R}^2\right)\Big/\hat{R}^2$$

for the Poly(1) kernel,

$$\sigma_0 = \sqrt{(N+2)/2} \quad \text{and}$$

$$\lambda_0 = \left(1 + 2N\sigma^{-2} + N(N+2)\sigma^{-4}\right)\left(1-\hat{R}^2\right)\Big/\hat{R}^2$$

for the Poly(2) kernel,

$$\sigma_0 = \sqrt{c_N}/\pi \quad \text{and}$$

$$\lambda_0 = \left(1-\hat{R}^2\right)\Big/\hat{R}^2 \quad \text{for the Gaussian kernel,}$$

where c_N is the 95th percentile of the χ^2 distribution with N degrees of freedom, and $\hat{R}^2$ denotes the R^2 value from an OLS regression of y on the first four principal components of X. A detailed motivation for using these grids is given by Exterkate (2013). We use a rolling window of fixed length for estimation, and reselect the values of the tuning parameters for each window.

As a technical note on cross-validation, serial correlation in time series data leads to dependence between the estimation and validation samples. This dependence implies that the standard cross-validation procedure may not be fully adequate; see Racine (2000) for an extensive discussion and a modification that overcomes the problem. Although the derivations in Appendix A.3 can be adapted to this modification easily, the resulting implementation is quite intensive computationally. In our applications, we find that the results from this modified procedure are not appreciably different from those obtained using simple k-fold cross-validation. Therefore, we only report the results obtained using the latter method. Similarly, using $k = T$ or $k = 10$ folds did not alter the results substantially, and therefore only five-fold cross-validation is employed in this paper; results for the other choices may be found in the electronic Appendix B to this article.

3. Monte Carlo simulation

We evaluate the potential of kernel ridge regression in a data-rich environment (that is, when many predictor variables are present) by assessing its forecasting performance for a set of factor models, through a Monte Carlo study. First, we consider a setting with two latent factors, f_{1t} and f_{2t}, which are taken to be uncorrelated standard normal variables. We begin by considering the case where $x_t = f_t$ are observed and can be used as predictor variables. Next, we turn to the more realistic scenario where $N = 100$ noisy linear combinations of these factors are generated by $x_{it} = \theta_{i1}f_{1t} + \theta_{i2}f_{2t} + \eta_{it}$, with the factor loadings θ_{ij} drawn from the standard normal distribution. The noise η_{it} is also normal with mean zero, while its variance is selected so as to control the fraction of the variance of each x_i variable that is explained by the factors, denoted by R_x^2. We take R_x^2 to be 0.4 or 0.8, which we label as "weak" and "strong" factor structures, respectively. The target variable y is constructed according to four different DGPs:

Linear: $$y_t = f_{1t} + f_{2t} + \varepsilon_t \tag{6}$$

Squared: $$y_t = f_{1t} + f_{2t} + 2\left(f_{1t}^2 + f_{2t}^2\right) + \varepsilon_t \tag{7}$$

Cross-product: $$y_t = f_{1t} + f_{2t} + 4f_{1t}f_{2t} + \varepsilon_t \tag{8}$$

Threshold: $$y_t = \begin{cases} -1.5 + 0.4f_{1t} + \varepsilon_t & \text{if } f_{2t} < 0, \\ 0.5 + 0.8f_{1t} + \varepsilon_t & \text{otherwise.} \end{cases} \tag{9}$$

Here, ε_t is normally distributed,[2] independent of the latent factors, with mean zero and a variance that is selected to control R_y^2, the fraction of the variance of y_t that is explained by the factors. For R_y^2, we also consider the values 0.4 and 0.8, which we refer to as "weak" and "strong" predictive structures, respectively. In what follows, we only report results for $R_x^2 = R_y^2$; thus, we omit the subscript and simply write R^2.[3]

In each Monte Carlo replication, we generate time series of $x_i, i = 1, \ldots, N$, and y, each consisting of $T + 1$ observations. The first T observations are used for estimation, and a forecast for y_{T+1} is made based on x_{T+1}. All variables are studentized to have mean zero and variance one in the estimation sample. We set $T = 120$, which corresponds to the length of each estimation window (ten years of monthly observations), in the empirical application in Section 4. We present results based on 5000 replications.

In principle, OLS regression using the individual predictors can be applied in this setting, as the number of regressors is smaller than the number of observations in the estimation sample. However, given the large amount of parameter estimation uncertainty when $N = 100$ and $T = 120$, it should not come as a surprise that this procedure leads to very poor forecasting performances. We therefore do not report the OLS results. Instead, we consider nine alternative prediction methods for comparison with KRR:

(i) the "mean" forecast, with $\hat{y}_{121} = (1/120)\sum_{t=1}^{120} y_t$;
(ii) the equally-weighted combination forecast (Comb EW), with $\hat{y}_{121} = (1/N)\sum_{i=1}^{N} \hat{y}_{i,121}$, where $\hat{y}_{i,121}$ is the forecast of y_{121} based on an OLS regression of y on the regressor x_i only[4];
(iii) the inverse-MSE-weighted combination forecast (Comb iMSE), with $\hat{y}_{121} = \sum_{i=1}^{N} \hat{\sigma}_i^{-2}\hat{y}_{i,121}$, where $\hat{\sigma}_i^2$ is the mean squared leave-one-out prediction error of OLS regression of y on x_i only;
(iv) the "Jackknife Model Averaging" forecast (Comb JMA), with $\hat{y}_{121} = \sum_{i=1}^{N} w_i\hat{y}_{i,121}$, where the weights w_i are selected to minimize the expected squared error using the leave-one-out cross-validation criterion introduced by Hansen and Racine (2012);
(v) principal component regression (PC), which amounts to OLS but with the regressors $\hat{f}_t$ being the first r principal components of the predictor variables x;
(vi) "PC-squared" (PC^2), as was suggested by Bai and Ng (2008), which corresponds to principal component regression but with the squares of $\hat{f}_t$ as additional regressors;
(vii) "Squared PC" (SPC), also proposed by Bai and Ng (2008), which is principal component regression but using the principal components of the original predictor variables x and their squares[5];
(viii) polynomial sieve regression on principal components (PC-Sieve), which is an extension of PC^2 to include higher powers of $\hat{f}_t$ and their cross-products as regressors; and
(ix) Nadaraya–Watson nonparametric regression on the principal components (PC-NW), with a local constant Gaussian product kernel and a standard plug-in bandwidth.

For KRR, the tuning parameters λ and σ are selected from the grids defined in Section 2.5 using five-fold cross-validation. For each of the principal-components-based methods except for PC-NW, we select the number of components by minimizing the Bayesian information criterion (BIC), with $1 \le r \le 10$. For PC-Sieve, we also use BIC to select the degree of the polynomial, with $1 \le d \le 5$. Our reason for minimizing the BIC instead of performing cross-validation for these methods is twofold. First, the use of BIC in principal components forecasting settings is common in the literature; see e.g. Bai and Ng (2008) and Stock and Watson (2002). Second, preliminary simulation evidence shows that the use of BIC leads to superior results

[2] We have also experimented with Student-t distributions for ε_t and η_t. As this did not lead to any appreciable differences in (relative) forecasting performances, the results are not reported here.

[3] The results for $R_x^2 \neq R_y^2$ are included in the electronic Appendix B to this article. In general, varying R_y^2 has a much larger effect on the results than varying R_x^2.

[4] The electronic Appendix B to this article also includes results for forecast combinations based on all possible two-regressor models. There are no substantial differences between those results and the results reported here.

[5] Bai and Ng (2008) also propose a quadratic variant (QPC), in which principal components are taken not only of the x_{it} and their squares (as in SPC), but also including their cross-products. They report high computational costs and a poor forecasting performance for QPC, and our preliminary analysis confirms these results. For this reason, QPC is not considered in our study.

Table 1
Relative mean squared prediction errors for the DGPs in Eqs. (6)–(9), with observable factors.

DGP	Linear		Squared		Cross-product		Threshold	
$R^2 =$	0.4	0.8	0.4	0.8	0.4	0.8	0.4	0.8
Benchmark methods								
Mean	1.03	1.02	1.02	1.05	1.05	1.08	1.03	1.03
Comb EW	0.72	*0.42*	0.99	*0.99*	1.02	*1.02*	0.82	0.60
Comb iMSE	*0.72*	0.42	*0.99*	0.99	*1.02*	1.02	0.81	0.59
Comb JMA	0.73	0.42	0.99	1.00	1.02	1.02	*0.81*	*0.59*
Principal-components-based methods								
PC	***0.62***	*0.20*	1.00	0.99	1.02	1.02	0.74	0.45
PC^2	0.62	0.21	***0.63***	0.28	***0.63***	*0.26*	*0.74*	*0.44*
SPC	0.83	0.63	0.84	0.68	1.06	1.08	0.90	0.76
PC-Sieve	0.62	0.21	0.65	*0.28*	0.64	0.26	0.75	0.45
PC-NW	0.79	0.55	0.92	0.81	0.92	0.77	0.85	0.68
Kernel ridge regression								
Poly(1)	*0.62*	***0.20***	1.00	0.99	1.02	1.02	0.74	0.45
Poly(2)	0.62	0.21	*0.65*	***0.27***	*0.65*	***0.26***	0.74	0.44
Gauss	0.63	0.21	0.72	0.33	0.78	0.42	***0.73***	***0.34***
Forecast combinations								
Linear	0.72	0.41	0.99	0.99	1.02	1.02	0.81	0.59
No KRR	0.69	0.35	0.81	0.63	0.85	0.70	0.79	0.53
All	*0.66*	*0.29*	*0.78*	*0.56*	*0.83*	*0.62*	*0.76*	*0.47*
Diebold–Mariano tests								
Nonlin.	**19.04**	**31.99**	**23.09**	**28.65**	**22.60**	**26.84**	**16.65**	**29.45**
Kernel	**18.65**	**34.47**	**19.56**	**24.39**	**20.92**	**24.53**	**19.34**	**38.96**

Notes: This table reports mean squared prediction errors (MSPEs) for the DGPs in Eqs. (6)–(9), averaged over 5000 forecasts, and relative to the variance of the series being predicted. It is assumed that $x_t = f_t$; that is, the factors are observed. These DGPs have no dynamic structure, so that x_{T+1} is used to forecast y_{T+1}. The combination forecasts are averages of the Mean, Comb, and PC forecasts ("Linear"), all benchmark and PC-based methods forecasts ("No KRR"), and all forecasts ("All"), respectively. The smallest relative MSPE for each DGP (column) within each group of methods (benchmarks, PC-based, kernel-based, or combinations) is printed in italics, with the overall smallest in boldface and italics. The last two rows report the t statistics of Diebold–Mariano tests for equal predictive ability. "Nonlin." compares "Linear" to "No KRR"; a positive statistic indicates that the latter performs better. Similarly, "Kernel" compares "No KRR" to "All". The statistic is printed in boldface if it is significant at the 5% level.

for principal-component methods, as compared to cross-validation. Finally, the BIC is not applicable to PC-NW, and cross-validation would be very costly computationally. As Nadaraya–Watson regression with many regressors tends to provide poor out-of-sample forecasts, we fix $r = 2$.

As a first check, we consider the unrealistic case in which the factors f_{1t} and f_{2t} are observed, to ensure that we are dealing with only two predictors. Table 1 shows the resulting mean squared prediction errors (MSPEs), relative to the variance of the series being predicted. Note that if the data-generating processes were known, these relative MSPEs would be close to $1 - R^2$, or 0.6 and 0.2 in the "weak" and "strong" predictive structures considered here. Standard PC shows a good performance for the linear DGP, while PC^2 performs well for the squared and cross-product DGPs. Such results are as expected, because the forecast equations that are used in these methods correspond exactly to these DGPs. Interestingly, the kernel methods attain levels of accuracy similar to those of these "optimal" methods, with the obvious exception of the Poly(1) (that is, linear) kernel in the nonlinear DGPs (for which standard PC also fares badly). Gaussian KRR outperforms all of the other methods in the threshold DGP, for which all methods considered estimate misspecified models. Polynomial sieve regression performs similarly to polynomial KRR, whereas the Nadaraya–Watson regression performs rather poorly overall.

The Comb forecasts, which are based on a combination of two univariate linear forecasts, perform better than the Mean benchmark in all DGPs. As expected, this is particularly evident in the linear model, while the gains are marginal in the squared and cross-product DGPs. However, these forecasts are clearly outperformed by all KRR-based and most PC-based forecasts in all cases considered. Since only two forecasts are combined, no large differences between the three combination schemes emerge.

We now turn to a more realistic scenario, where f_{1t} and f_{2t} are unobserved and need to be estimated from the x_{it}. Table 2 shows the relative MSPEs; note that the Comb forecasts are now based on 120 univariate forecasts, rather than two. The findings that we presented based on Table 1 still largely hold, regardless of whether R^2 is high or low. Thus, we find that kernel methods can work well in standard factor model settings, even though these methods are not based on any factor model assumptions. We observe that KRR always performs well, even if the factor structure of the predictors is not very strong, as is often the case for empirical macroeconomic and financial data.

We also explore the potential for combining kernel-based forecasts with more traditional forecasts. Three types of combination forecasts are considered:

(i) "Linear", combining all of the forecasts from traditional linear methods;
(ii) "No KRR", combining all of the forecasts from non-kernel methods; and
(iii) "All", combining all forecasts.

These combination forecasts are defined by taking a simple arithmetic average of the individual forecasts;

Table 2
Relative mean squared prediction errors for the DGPs in Eqs. (6)–(9), with i.i.d. latent factors.

DGP	Linear		Squared		Cross-product		Threshold	
$R^2 =$	0.4	0.8	0.4	0.8	0.4	0.8	0.4	0.8
Benchmark methods								
Mean	1.02	1.03	1.00	1.02	1.02	1.04	1.00	1.00
Comb EW	0.87	0.50	0.99	0.97	1.01	0.99	0.90	0.64
Comb iMSE	0.87	0.41	0.99	0.97	*1.01*	*0.98*	0.90	0.60
Comb JMA	*0.77*	*0.26*	*0.99*	*0.96*	1.01	0.99	*0.83*	*0.48*
Principal-components-based methods								
PC	***0.63***	0.21	1.00	0.98	1.01	0.99	0.74	*0.44*
PC^2	0.64	0.21	***0.66***	0.27	0.85	0.64	0.75	0.44
SPC	0.63	0.22	0.69	0.27	0.77	0.28	***0.74***	0.45
PC-Sieve	0.63	***0.21***	0.68	*0.27*	***0.67***	***0.27***	0.74	0.44
PC-NW	0.79	0.55	0.92	0.80	0.92	0.75	0.83	0.65
Kernel ridge regression								
Poly(1)	*0.64*	*0.21*	0.99	0.97	1.01	0.99	0.75	0.45
Poly(2)	0.65	0.21	*0.71*	***0.26***	*0.72*	*0.27*	0.76	0.44
Gauss	0.65	0.22	0.79	0.34	0.86	0.43	*0.75*	***0.37***
Forecast combinations								
Linear	0.78	0.38	0.98	0.96	1.00	0.98	0.83	0.56
No KRR	0.70	0.30	0.80	0.56	0.86	0.62	0.78	0.48
All	*0.68*	*0.26*	*0.80*	*0.51*	*0.85*	*0.57*	*0.76*	*0.45*
Diebold–Mariano tests								
Nonlin.	**24.02**	**34.54**	**22.17**	**28.01**	**21.15**	**25.92**	**20.36**	**31.99**
Kernel	**16.41**	**30.86**	**4.57**	**21.48**	**9.37**	**28.30**	**12.76**	**31.30**

Notes: This table has the same structure as Table 1. The f_t are now treated as latent factors, and only $x_t = \Theta f_t + \eta_t$ are observed. These DGPs have no dynamic structure, so that x_{T+1} is used to forecast y_{T+1}.

preliminary experimentation with using the median or a weighted average based on inverse in-sample root mean squared errors did not provide substantially different results. Jackknife model averaging (Hansen & Racine, 2012) was not considered, since it is based on a leave-one-out procedure, which would be very computationally intensive for the principal-components-based methods. Combination techniques based on information criteria, such as Mallows model averaging (Hansen, 2007), cannot be applied here, as these methods assume that the number of parameters is finite, which is not the case for Gaussian KRR.

In the groups of rows labeled "Forecast combinations" in Tables 1 and 2, it can be seen that the linear combination forecast always performs worse than the combination of all forecasts excluding KRR, and that both of these are dominated by the combination of all forecasts. Thus, including traditional nonlinear methods enhances the forecast performance, and including KRR as well improves it even further. To quantify these gains, the final two rows of these tables report t-statistics from Diebold and Mariano (1995) tests comparing these forecasts. We observe that the gains are strongly significant in all 32 cases.

KRR was originally developed for cross-sectional data, and, as we have seen, its performance compares favorably to those of more traditional techniques in this context. Since we are interested in forecasting data with a temporal dependence structure, we now modify the data-generating processes to check the performance of KRR under such circumstances. We replace the independent normal factors f_{it} with the following autoregressive processes:

$$f_{it} = 0.7 f_{i,t-1} + \zeta_{it},$$

where the ζ_{it} are uncorrelated normal variables with mean zero and variance $1 - 0.7^2 = 0.51$. We continue to construct x_{it} and y_t as described in Eqs. (6)–(9) and the paragraph preceding these equations. In order to obtain a realistic forecast setting, we base our forecasts of y_{T+1} on x_T rather than on x_{T+1}. This is also the route that we take in the empirical application in Section 4 below.

The relative MSPEs are presented in Table 3. As the DGPs now have a dynamic component, there are three additional benchmark methods:

(i) the random-walk forecast (RW), $\hat{y}_{121} = y_{120}$;
(ii) an autoregressive forecast (AR), from a linear AR(1) model; and
(iii) the forecast from a self-exciting threshold autoregressive model (SETAR), given by

$$y_t = \begin{cases} \rho_{00} + \rho_{10} y_{t-1} + \varepsilon_t & \text{if } y_{t-1} < \tau, \\ \rho_{01} + \rho_{11} y_{t-1} + \varepsilon_t & \text{otherwise.} \end{cases}$$

Compared to the results that we obtained for the static factor models in Table 2, the main points still stand for the linear and threshold DGPs. In the quadratic and cross-product DGPs, the performance of KRR is similar to those of the best-performing benchmarks if $R^2 = 0.4$. If the factor structure is strong ($R^2 = 0.8$), the simple linear AR model provides slightly better forecasts than any of the other methods.

We conclude that kernel methods still work quite well in a factor context, even though they do not assume any kind of factor structure. Individual KRR-based forecasts perform similarly to or better than specialized principal-component-based forecast methods. This is especially true for nonlinear relationships, where both PC-based and KRR-based methods are estimating misspecified models. This feature may be expected to be present in empirical macroeconomic and financial data. Moreover, we find that there is the potential to combine the two types of forecasts, leading to significant gains in forecast accuracy.

Table 3
Relative mean squared prediction errors for the DGPs in Eqs. (6)–(9), with AR(1) latent factors.

DGP	Linear		Squared		Cross-product		Threshold	
$R^2 =$	0.4	0.8	0.4	0.8	0.4	0.8	0.4	0.8
Benchmark methods								
Mean	1.05	1.10	1.08	1.16	1.08	1.17	1.04	1.06
RW	1.54	0.97	1.73	1.39	1.73	1.38	1.65	1.20
AR	0.99	0.78	*1.05*	***1.01***	*1.05*	***1.01***	1.01	0.88
SETAR	1.05	0.81	1.12	1.09	1.11	1.10	1.07	0.94
Comb EW	0.98	0.85	1.07	1.14	1.07	1.15	0.99	0.89
Comb iMSE	0.98	0.83	1.07	1.14	1.07	1.15	0.99	0.88
Comb JMA	*0.94*	*0.74*	1.09	1.16	1.09	1.17	*0.97*	*0.81*
Principal-components-based methods								
PC	***0.88***	***0.71***	1.09	1.16	1.09	1.17	***0.93***	***0.79***
PC^2	0.91	0.72	1.06	1.06	1.07	1.11	0.96	0.81
SPC	0.89	0.74	1.06	*1.05*	1.08	1.10	0.94	0.82
PC-Sieve	0.88	0.71	1.07	1.11	1.08	1.10	0.93	0.80
PC-NW	0.94	0.86	*1.05*	1.10	*1.05*	*1.09*	0.96	0.90
Kernel ridge regression								
Poly(1)	0.90	*0.72*	1.08	1.15	1.08	1.15	0.94	0.80
Poly(2)	0.90	0.72	*1.04*	*1.04*	*1.04*	*1.04*	0.95	0.81
Gauss	*0.90*	0.72	1.05	1.04	1.06	1.07	*0.94*	*0.80*
Forecast combinations								
Linear	0.93	0.75	1.04	1.03	1.04	1.04	0.96	0.82
No KRR	0.91	0.73	***1.03***	*1.02*	***1.04***	*1.03*	0.95	0.81
All	*0.90*	*0.72*	1.03	1.02	1.04	1.04	*0.94*	*0.80*
Diebold–Mariano tests								
Nonlin.	**11.17**	**11.49**	**2.46**	**3.66**	1.81	0.99	**8.58**	**6.97**
Kernel	**8.35**	**10.50**	−0.25	**−2.41**	−0.01	**−2.57**	**6.48**	**8.26**

Notes: This table has the same structure as Table 1, except that the "Linear" combination forecast also includes the RW and AR forecasts. The f_t are assumed to be latent factors following AR(1) processes, so x_T and y_T are used to forecast y_{T+1}.

Although our focus is on forecasting in an environment with many predictors that exhibit a factor structure, it may also be instructive to investigate the performance of KRR in more traditional nonlinear autoregressive data-generating processes. We therefore generate data from the threshold autoregression

$$y_t = \begin{cases} -1.5 + 0.4y_{t-1} + \varepsilon_t & \text{if } c_{t-1} < 0, \\ 0.5 + 0.8y_{t-1} + \varepsilon_t & \text{otherwise}, \end{cases} \qquad (10)$$

with ε_t being i.i.d. standard normal. We consider the following four options for the transition variable c_{t-1}:

(i) "Self-exciting", where $c_{t-1} = y_{t-1}$;
(ii) "Observed", where c_{t-1} is an observed variable following an AR(1) process with coefficient 0.7;
(iii) "Weak factor", where c_{t-1} is a latent factor following an AR(1) process with coefficient 0.7, while $N = 100$ noisy linear combinations $x_{it} = \theta_i c_t + \eta_{it}$ are observed, with the variances controlled to have $R^2 = 0.4$ in this equation; and
(iv) "Strong factor", similar to the previous case, except that $R^2 = 0.8$.

The relative MSPEs for these models, again based on 5000 replications with $T+1 = 121$ observations, are listed in Table 4. In the self-exciting DGP, y_T is the only predictor for y_{T+1}, so no combination or principal-components-based forecasts are reported. The results reported for PC-Sieve and PC-NW were obtained using sieve and Nadaraya–Watson regressions on this single predictor. KRR, especially using the Gaussian kernel, outperforms all of the other methods in this case. The same result is found in the DGP with an observed transition variable.

Kernel methods remain competitive in the DGPs with factor structures. If the factor structure is weak, all KRR variants outperform all PC variants, being dominated only by univariate techniques. The Gaussian KRR is marginally outperformed by polynomial PC-based methods, most notably sieve regression, if the factor structure is strong. Finally, the results of the Diebold–Mariano tests indicate that including kernel-based forecasts improves combination forecasts strongly in this set of DGPs too.

4. Macroeconomic forecasting

4.1. Data and forecast models

We evaluate the forecast performance of kernel ridge regression in an empirical application involving a large panel of US macroeconomic and financial variables. The data set consists of monthly observations on 132 variables, including various measures of production, consumption, income, sales, employment, monetary aggregates, prices, interest rates, and exchange rates. All series are transformed to stationarity by taking logarithms and/or differences, as was described by Stock and Watson (2005). We have updated their data set, which starts in January 1959 and ends in December 2003, to cover the period up to January 2010. The cross-sectional dimension varies somewhat over time, due to data availability: some series start later than January 1959, while a few other variables were discontinued before the end of our sample period. No more than five variables have missing observations for each month under consideration.

We focus on forecasting four key measures of real economic activity: industrial production, personal income

Table 4
Relative mean squared prediction errors for the threshold autoregressive DGPs in Eq. (10).

DGP	Self-exciting	Observed	Weak factor	Strong factor
Benchmark methods				
Mean	1.47	1.10	1.08	1.08
RW	0.65	0.54	0.50	0.50
AR	*0.56*	0.48	***0.46***	0.46
SETAR	0.59	0.53	0.50	0.50
Comb EW	–	0.47	0.88	0.77
Comb iMSE	–	0.45	0.87	0.76
Comb JMA	–	*0.44*	0.48	*0.45*
Principal-components-based methods				
PC	–	0.39	*0.71*	0.43
PC^2	–	*0.38*	0.75	0.44
SPC	–	0.50	0.76	0.68
PC-Sieve	*0.62*	0.39	0.72	***0.43***
PC-NW	0.75	0.65	0.87	0.69
Kernel ridge regression				
Poly(1)	0.57	0.38	0.58	0.48
Poly(2)	0.57	0.37	0.58	0.48
Gauss	**0.55**	**0.37**	*0.55*	*0.45*
Forecast combinations				
Linear	0.63	0.44	0.55	0.48
No KRR	0.59	0.42	0.56	0.47
All	*0.57*	*0.40*	*0.54*	*0.45*
Diebold–Mariano tests				
Nonlin.	**10.19**	**15.15**	**−4.46**	**14.19**
Kernel	**12.61**	**25.26**	**12.83**	**15.21**

Notes: This table has the same structure as Table 1, except that the "Linear" combination forecast also includes the RW and AR forecasts. y_T and x_T are used to forecast y_{T+1}, except in the self-exciting DGP, where only y_T is available.

less transfer payments (henceforth "personal income"), manufacturing and trade sales, and employment on non-agricultural payrolls ("employment"), as per Stock and Watson (2002). For each of these variables, we produce out-of-sample forecasts for the annualized h-month percentage growth rate

$$y_{t+h}^{h} = \frac{1200}{h} \ln\left(\frac{v_{t+h}}{v_t}\right),$$

where v_t is the untransformed observation on the level of each variable in month t. We will write y_{t+1}^{1} as y_{t+1} to simplify the notation. We consider growth rate forecasts for $h = 1, 3, 6$, and 12 months, and we follow Stock and Watson (2002) in modelling the h-month growth rate directly, rather than making iterated one-month-ahead forecasts.

Kernel ridge regression is compared against several alternative forecasting approaches. As benchmarks, we include the same methods as in the Monte Carlo experiments: a "mean" forecast (the average growth over the estimation window), the "no-change" or random-walk (RW) forecast, an autoregressive (AR) forecast (using lagged values of the one-month growth rate as predictors), a self-exciting threshold autoregressive (SETAR) forecast (with transition variable y_t^{12}, the growth rate over the last year, as per Teräsvirta et al., 2005), and three combination (Comb) forecasts (weighted averages of 132 individual forecasts, each obtained by augmenting an AR(2) model with one additional explanatory variable).[6]

In addition, as the primary competitor for kernel methods, we consider the diffusion index (DI) approach of Stock and Watson (2002), who document its good performance for forecasting the four macroeconomic variables considered here. The DI methodology extends the standard principal component regression to a dynamic setting by including autoregressive lags in the forecast equation as well as lags of the principal components. Specifically, at time t, using p autoregressive lags and q lags of r factors, this "extended" principal-components method produces the forecast

$$\hat{y}_{t+h|t}^{h} = w_t'\hat{\beta} + \hat{f}_t'\hat{\gamma},$$

where $w_t = \left(1, y_t, y_{t-1} \ldots, y_{t-(p-1)}\right)'$ and $\hat{f}_t = \left(\hat{f}_{1,t}, \hat{f}_{2,t}, \ldots, \hat{f}_{r,t}, \hat{f}_{1,t-1}, \ldots, \hat{f}_{r,t-(q-1)}\right)'$. The lags of the dependent variable in w_t are one-month growth rates, irrespective of the forecast horizon h, because using h-month growth rates for $h > 1$ would lead to highly correlated regressors. The factors $\hat{f}$ are principal components extracted from all 132 predictor variables, and $\hat{\beta}$ and $\hat{\gamma}$ are OLS estimates.

In addition to standard principal components (PC), we also consider its extensions PC^2, SPC, and PC-Sieve, discussed in Section 3. In each case, the lag lengths p and q, the number of factors r, and the degree of the PC-Sieve polynomial d are selected by minimizing the Bayesian information criterion (BIC). This criterion is used instead of cross-validation for two reasons: we want our results to be comparable to those of Bai and Ng (2008) and Stock

[6] Results for combination forecasts based on AR(2)-plus-two-regressor models, as per Elliott, Gargano, and Timmermann (2013), are reported in the electronic Appendix B to this article; they are only marginally different to the results shown here. Additional benchmarks that we considered include smooth-transition autoregressions (STAR), as per Teräsvirta et al. (2005), and factor-augmented STAR models. These methods exhibit worse forecasting performances than the other benchmarks, and we do not report the results.

Table 5
Frequency of application of the insanity filter for the macroeconomic series.

Forecast method	Industrial production				Personal income				Manuf. and trade sales				Employment			
$h =$	1	3	6	12	1	3	6	12	1	3	6	12	1	3	6	12
Benchmark methods																
Mean	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
RW	0.2	0.2	–	–	0.4	–	–	–	–	–	–	–	0.2	–	–	–
AR	–	–	–	–	0.2	–	–	–	–	–	–	–	–	–	–	–
SETAR	0.4	0.2	0.2	–	–	–	–	–	0.6	–	–	–	–	–	–	–
Comb EW	–	–	–	–	0.2	–	–	–	–	–	–	–	–	–	–	–
Comb iMSE	–	–	–	–	0.2	–	–	–	–	–	–	–	–	–	–	–
Comb JMA	–	–	–	–	0.2	–	–	–	–	–	–	–	–	–	–	–
Principal-components-based methods																
PC	–	–	–	–	0.2	–	–	–	–	–	–	–	–	–	–	–
PC^2	0.2	0.4	0.4	–	0.2	–	0.2	–	–	–	–	–	–	0.2	0.4	0.7
SPC	0.2	0.6	1.5	0.7	0.2	0.4	0.2	0.2	–	0.4	1.7	0.7	–	0.8	0.6	–
PC-Sieve	0.8	0.2	–	–	0.2	–	0.2	–	–	–	–	–	–	0.2	–	–
PC-NW	–	–	–	–	0.2	–	–	–	–	–	–	–	–	–	–	–
Kernel ridge regression																
Poly(1)	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Poly(2)	0.4	0.2	–	–	–	–	–	–	–	–	0.4	–	–	–	–	–
Gauss	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–

Notes: This table lists the percentages of forecasts to which an insanity filter was applied, for each forecast method and each target series. If the filter was never applied, this is indicated by a dash (–).

and Watson (2002), and preliminary experimentation with the PC methods has revealed that using the BIC leads to superior results. Following Stock and Watson (2002), we allow $0 \leq p \leq 6$ (where $p = 0$ means that $w_t = 1$), $1 \leq q \leq 3$, and $1 \leq r \leq 4$. Thus, the simplest model that can be selected uses no information on current or lagged values of the dependent variable, and information from the other predictors in the current month only, summarized by a single factor. Also in line with Stock and Watson (2002), we do not perform an exhaustive search across all possible combinations of the first four principal components and lag structures. Instead, we assume that factors are included sequentially in order of importance, while the number of lags is assumed to be the same for all factors included. As a final competing nonlinear data-rich forecasting method, we also include PC-NW, which we define by augmenting a linear AR(2) model with a nonparamametric regression function, with the first two principal components $\hat{f}_{1,t}$ and $\hat{f}_{2,t}$ as regressors.

For KRR, the corresponding forecast equation is

$$\hat{y}^h_{t+h|t} = w_t'\hat{\beta} + \varphi\left(\left(x_t', x_{t-1}', \ldots, x_{t-(q-1)}'\right)'\right)'\hat{\gamma},$$

in the notation of Section 2.2, where w_t is as above and x_t contains all 132 predictors at time t. The parameter vectors $\hat{\beta}$ and $\hat{\gamma}$ are estimated by KRR, resulting in Eq. (3). In particular, note that β (which contains the intercept and the autoregressive coefficients) is not subject to a ridge penalty, in order to avoid a biased estimation of this short vector of relatively important parameters. The lag lengths p and q are selected by five-fold cross-validation, as are the KRR parameters λ and σ.

All of the models are estimated on rolling windows with a fixed length of 120 months, such that the first forecast is produced for the growth rate during the first h months of 1970. The tuning parameter values are re-selected for each window, and the regression coefficients are re-estimated. That is, all of the tuning parameters $(p, q, r, d, \lambda, \sigma)$ may vary over time and across target variables, horizons, and methods.

The nonlinear principal-component-based models produce erratic forecasts on some occasions, and these have an undue impact on the reported mean squared prediction errors. To mitigate the impacts of such forecasts, we follow Swanson and White (1995) by using an insanity filter to "substitute ignorance for craziness". We calculate the mean and standard deviation of the target variable over the estimation window. If a forecast is more than five standard deviations away from the mean, it is replaced by the mean.

4.2. Results

We begin by assessing the stability of the various forecast methods. In Table 5, we record how often the "insanity filter" was invoked to replace an unreasonably extreme forecast with the sample mean. We note that this filter was needed relatively frequently for traditional nonlinear models, especially SPC, whereas it was hardly ever needed for kernel-based forecasts. It appears that the ridge term is a sufficient safeguard against such extreme forecasts. In what follows, we report only the filtered results.

Table 6 shows the MSPEs for the period 1970–2010 for the seven benchmark methods, five PC-based methods and three kernel methods, as well as for the three forecast combination methods that were introduced in Section 3. Several conclusions can be drawn from these results.

First, we observe that KRR provides more accurate forecasts than any of the seven benchmarks (mean, random walk, autoregression, SETAR, and the three combination forecasts) for almost all target variables and all forecast horizons, with larger gains for longer horizons. This holds in 15 out of 16 cases for the Gaussian and Poly(2) kernels, and in 14 out of 16 cases for the linear kernel. In many cases, the improvements in predictive accuracy are substantial, even compared to the JMA-weighted combination

Table 6
Relative mean squared prediction errors for the macroeconomic series.

Forecast method	Industrial production				Personal income				Manuf. and trade sales				Employment			
$h =$	1	3	6	12	1	3	6	12	1	3	6	12	1	3	6	12
Benchmark methods																
Mean	1.01	1.04	1.05	1.07	1.02	1.05	1.09	1.16	1.01	1.02	1.04	1.08	0.98	0.96	0.96	0.97
RW	1.18	1.09	1.36	1.61	1.20	1.36	1.14	1.34	2.18	1.49	1.45	1.50	1.58	0.95	0.99	1.19
AR	0.92	0.87	1.01	1.02	1.04	1.04	1.09	1.14	1.01	1.01	1.09	1.08	0.96	0.85	0.89	0.95
SETAR	1.01	1.27	1.17	1.06	*0.94*	1.00	1.17	1.13	1.03	1.02	1.05	1.08	0.98	0.89	0.90	0.98
Comb EW	0.85	0.81	0.90	0.91	0.97	0.94	0.93	0.98	0.98	0.96	0.98	0.98	0.89	0.75	0.76	0.81
Comb iMSE	0.85	0.81	*0.89*	0.87	0.97	0.94	0.92	0.95	0.98	0.96	0.96	0.94	0.89	0.74	0.74	0.78
Comb JMA	***0.75***	*0.74*	0.92	*0.71*	0.95	*0.83*	*0.81*	*0.83*	*0.94*	*0.94*	*0.91*	*0.71*	*0.81*	*0.63*	*0.66*	*0.74*
Principal-components-based methods																
PC	0.80	*0.73*	*0.76*	0.68	*0.89*	***0.78***	*0.86*	*0.87*	0.88	***0.80***	0.80	*0.68*	0.76	***0.56***	***0.49***	*0.54*
PC^2	*0.78*	0.79	0.85	0.75	0.90	0.85	0.91	0.95	0.91	0.83	***0.77***	0.77	***0.75***	0.60	0.53	0.57
SPC	0.85	0.86	0.80	0.97	0.96	0.90	0.95	1.11	0.97	1.03	0.90	1.04	0.78	0.69	0.61	0.80
PC-Sieve	0.94	0.77	0.82	*0.67*	0.90	0.79	0.92	0.98	***0.88***	0.81	0.79	0.68	0.76	0.57	0.53	0.56
PC-NW	0.90	0.82	0.86	0.83	0.94	0.95	0.90	0.94	0.95	0.93	0.92	0.91	0.86	0.70	0.68	0.73
Kernel ridge regression																
Poly(1)	*0.77*	***0.69***	0.82	0.72	0.89	0.83	0.80	0.80	*0.88*	0.86	0.84	***0.65***	*0.80*	0.58	0.55	0.50
Poly(2)	0.79	0.70	0.80	0.68	0.89	*0.83*	0.81	0.77	0.90	0.86	0.81	0.65	0.80	0.59	0.55	***0.48***
Gauss	0.79	0.71	***0.74***	***0.64***	***0.87***	0.83	***0.76***	***0.74***	0.89	*0.84*	*0.80*	0.66	0.80	*0.57*	*0.52*	0.51
Forecast combinations																
Linear	0.80	0.75	0.83	0.80	0.92	0.86	0.82	0.87	0.96	0.89	0.87	0.82	0.86	0.66	0.65	0.74
No KRR	0.79	0.74	0.77	0.75	0.90	0.82	0.80	0.86	0.92	0.86	0.82	0.80	0.81	0.63	0.60	0.68
All	*0.77*	*0.71*	*0.74*	*0.69*	*0.88*	*0.81*	*0.77*	*0.80*	*0.89*	*0.83*	*0.79*	*0.73*	*0.80*	*0.60*	*0.56*	*0.61*
Diebold–Mariano tests																
Nonlin.	0.81	0.90	**2.24**	1.81	1.82	**2.25**	1.00	0.86	**3.63**	**2.72**	**3.54**	1.09	**5.00**	**2.38**	**2.52**	**3.16**
Kernel	**3.46**	**2.66**	1.46	1.49	**2.80**	**2.11**	**2.11**	**2.28**	**3.89**	**2.27**	**2.03**	**1.98**	**3.69**	**3.95**	**2.86**	1.93

Notes: This table reports mean squared prediction errors (MSPEs) for four macroeconomic series, over the period 1970–2010, relative to the variance of the series being predicted. The forecast combinations are averages of the Mean, RW, AR, Comb, and PC forecasts ("Linear"), all benchmark and principal-components-based forecasts ("No KRR"), and all forecasts ("All"), respectively. The smallest relative MSPE for each series (column) within each group of methods (benchmarks, PC-based, KRR, or combinations) is printed in italics, with the overall smallest in boldface italics. The last two rows report t-statistics from Diebold–Mariano tests of equal predictive ability. "Nonlin." compares "Linear" to "No KRR"; a positive statistic indicates that the latter performs better. Similarly, "Kernel" compares "No KRR" to "All". The statistic is printed in boldface if it is significant at the 5% level.

forecast, which seems the best of the seven benchmarks. For example, for 12-month growth rate forecasts, kernel ridge regression based on the Gaussian kernel achieves a 31% reduction in MSPE for employment, and around 10% for the other three variables (relative to Comb JMA).

Second, if we compare the forecasts based on KRR and the linear PC-based approach, we find somewhat mixed results, but the performance of the kernel methods is generally comparable or better. Kernel ridge forecasts are superior for industrial production and personal income for all four and three out of the four horizons considered, respectively. The improvements in relative MSPE range from 2% for personal income at the one-month horizon to 14% for the same target variable at the longest horizon of one year. For manufacturing and trade sales, kernels perform somewhat worse than linear principal components at short horizons, but somewhat better at the longest horizon. Finally, for employment, the PC-based forecasts are about 5% more accurate than kernel-based forecasts, except at the twelve-month horizon, where Poly(2) KRR achieves a 12% improvement. The KRR-based employment forecasts are at least as accurate as those for the other series; in this case, the difference is driven by the very good performance of principal component regression. That is, our results suggest that kernel ridge regression performs better than principal component regression unless the latter performs very well.

Third, the KRR approach outperforms the nonlinear variants of the principal component regression framework, except for the sales and employment series at shorter horizons. In fact, the linear PC specification also renders substantially more accurate forecasts than these four extensions in many cases. Apparently, the PC^2, SPC, and PC-NW methods cannot cope successfully with the possibly nonlinear relationships between the target variables and the predictors in this application.[7] PC-Sieve is a closer competitor to the kernel methodology, but it still generally fails to outperform the simple linear PC specification.

Fourth, the forecast combinations exhibit the same pattern as in the simulation study. Adding traditional nonlinear methods always helps with forecasting (compare "No KRR" to "Linear"), and adding kernel methods helps even more (compare "All" to "No KRR"), for every series, at every horizon. The Diebold–Mariano t-statistics in the last two rows of Table 6 indicate that these improvements are statistically significant in the majority of cases: 9 out of 16 for adding traditional nonlinear methods, and 13 out of 16 for additionally adding KRR to the set of forecasting models.

Fifth, among the kernel-based methods, the Gaussian kernel generally performs best, achieving lower MSPEs than either of the polynomial kernels in most cases. Although Poly(1) (that is, linear ridge regression) performs better in a few cases, Gaussian KRR shows satisfactory results in all situations.

[7] Bai and Ng (2008) report a somewhat better forecast performance if SPC is applied to a selected subset of the predictors, rather than to the full predictor set. However, even with this modification, SPC has difficulty outperforming simpler linear methods in their application.

Table 7
Estimated coefficients $\hat{\alpha}$ from the forecast-combining regression in Eq. (11).

Forecast method	Industrial production				Personal income			
	$h=1$	$h=3$	$h=6$	$h=12$	$h=1$	$h=3$	$h=6$	$h=12$
Principal-components-based methods								
PC^2	0.61 (0.33)	0.13** (0.16)	0.09** (0.22)	0.12** (0.19)	0.44*,**(0.22)	0.18** (0.15)	0.38*,**(0.19)	0.23** (0.21)
SPC	0.29*,**(0.11)	0.23** (0.23)	0.44*,**(0.16)	0.12** (0.10)	0.02** (0.13)	0.21** (0.19)	0.34** (0.28)	0.15** (0.13)
PC-Sieve	−0.24** (0.19)	−0.15** (0.27)	−0.21** (0.36)	0.67 (0.59)	−0.37*,**(0.18)	−0.28** (0.49)	0.29*,**(0.12)	0.01** (0.19)
PC-NW	0.11** (0.20)	0.28** (0.16)	0.34*,**(0.16)	0.22** (0.16)	0.17** (0.38)	−0.03** (0.19)	0.40*,**(0.13)	0.36*,**(0.12)
Kernel ridge regression								
Poly(1)	0.57*,**(0.11)	0.62*,**(0.12)	0.33*,**(0.13)	0.42*,**(0.19)	0.56*,**(0.18)	0.22** (0.27)	0.70* (0.22)	0.62*,**(0.19)
Poly(2)	0.52*,**(0.14)	0.59*,**(0.14)	0.37*,**(0.18)	0.50*,**(0.21)	0.54*,**(0.15)	0.35** (0.21)	0.61*,**(0.16)	0.81* (0.22)
Gauss	0.54*,**(0.14)	0.56*,**(0.18)	0.58*,**(0.15)	0.62* (0.21)	0.65*,**(0.17)	0.28** (0.22)	0.75* (0.21)	0.88* (0.21)
Forecast method	Manufacturing and trade sales				Employment			
	$h=1$	$h=3$	$h=6$	$h=12$	$h=1$	$h=3$	$h=6$	$h=12$
Principal-components-based methods								
PC^2	−0.04** (0.15)	0.27** (0.17)	0.61*,**(0.19)	−0.06** (0.27)	0.67* (0.18)	0.15** (0.21)	0.26** (0.15)	0.38*,**(0.15)
SPC	−0.04** (0.15)	−0.08** (0.16)	0.31** (0.19)	−0.04** (0.09)	0.36*,**(0.12)	0.09** (0.10)	0.12** (0.12)	−0.17** (0.17)
PC-Sieve	–	0.36*,**(0.03)	0.53*,**(0.14)	0.32 (1.65)	–	0.29** (0.27)	−0.37** (0.21)	0.29** (0.23)
PC-NW	0.08** (0.21)	−0.06** (0.22)	0.20** (0.22)	−0.11** (0.19)	−0.11** (0.13)	−0.11** (0.15)	−0.03** (0.16)	−0.06** (0.18)
Kernel ridge regression								
Poly(1)	0.48*,**(0.12)	0.29*,**(0.13)	0.38*,**(0.14)	0.58* (0.22)	0.32*,**(0.11)	0.37*,**(0.13)	0.20** (0.13)	0.63* (0.25)
Poly(2)	0.41*,**(0.15)	0.25** (0.20)	0.47*,**(0.12)	0.59* (0.23)	0.30*,**(0.12)	0.33*,**(0.16)	0.24** (0.14)	0.76* (0.25)
Gauss	0.40*,**(0.17)	0.30** (0.19)	0.50*,**(0.13)	0.57* (0.26)	0.25** (0.13)	0.39*,**(0.16)	0.34*,**(0.16)	0.62* (0.22)

Notes: This table reports $\hat{\alpha}$, the weight placed on the candidate forecast in the forecast combining regression in Eq. (11). HAC standard errors follow in parentheses. The PC and PC-Sieve forecasts of the one-month growth rates of manufacturing and trade sales and employment are equal in every estimation window; hence, no $\hat{\alpha}$ can be computed in these two cases.

* Indicates rejection of the hypothesis $\alpha = 0$.

** Indicates rejection of $\alpha = 1$, at the 5% significance level.

It appears that linear principal component regression is by far the strongest competitor of kernel ridge regression. Following Stock and Watson (2002), we provide a further evaluation of our results by estimating the forecast combining regression

$$y^h_{t+h} = \alpha\, \hat{y}^h_{t+h|t} + (1-\alpha)\, \hat{y}^{h,\mathrm{PC}}_{t+h|t} + u^h_{t+h}, \tag{11}$$

where y^h_{t+h} is the realized growth rate over the h-month period ending in month $t+h$, $\hat{y}^h_{t+h|t}$ is a candidate forecast made at time t, and $\hat{y}^{h,\mathrm{PC}}_{t+h|t}$ is the corresponding linear PC-based forecast.

Estimates of α are shown in Table 7, with heteroscedasticity and autocorrelation consistent (HAC) standard errors in parentheses. The null hypothesis that the PC forecast receives zero weight ($\alpha = 1$) is rejected in 95 out of the 110 tests, which means that none of the alternative methods encompass linear principal component regression. The majority of these fifteen nonrejections are for KRR variants at the twelve-month horizon. The other null hypothesis of interest is $\alpha = 0$, which would imply that PC encompasses the alternative method in question. This hypothesis is almost always rejected for KRR forecasts (in 40 out of 48 tests), but not for the other methods (in only 15 out of 62 tests). Thus, we conclude that KRR encompasses PC for the longest horizon, and the KRR and PC forecasts are complementary for the other three horizons; but the other nonlinear forecast methods generally fail to improve on linear principal component regression. This result implies that the favorable results that we found for the "No KRR" combination forecast hinge on the fact that we aggregated all of these PC-based methods. None of the individual nonlinear PC-based methods consistently adds to linear PC, whereas KRR does.

Finally, we examine the stability of the performances of KRR and PC over time. For this purpose, Fig. 1 shows time series plots of rolling MSPEs for the best-performing method from each of the four groups: benchmark, PC-based, kernel, and forecast combination methods. The value plotted for date t is the MSPE computed over the ten-year subsample ending with the forecast for date t, that is, $\hat{y}^h_{t|t-h}$. We show only the series for $h = 12$; the results for the other horizons are qualitatively similar. This figure confirms that, when the KRR forecasts are less accurate than the PC-based forecasts, this is because the PC forecasts are very accurate, not because the KRR forecasts are inaccurate. Another interesting feature seen in Fig. 1 is the fact that, although the Great Financial Crisis reduces the accuracy of all forecasts from 2008 onwards, it affects the kernel-based forecasts least. The combination forecast almost always performs very well, despite the fact that our combination scheme is a simple unweighted average.[8]

Fig. 2 shows the same time series of MSPEs, rescaled by the rolling variance of the series being predicted. These rolling variances are also included in the graphs. These two figures lead to the following two conclusions. First, predictability improves in an absolute sense during less volatile times, in the sense that the MSPEs in Fig. 1 typically decline when the rolling variance of the series being predicted in Fig. 2 goes down. Second, forecasting becomes

[8] Weighting the forecasts by the inverse mean squared prediction errors instead did not alter the results substantially.

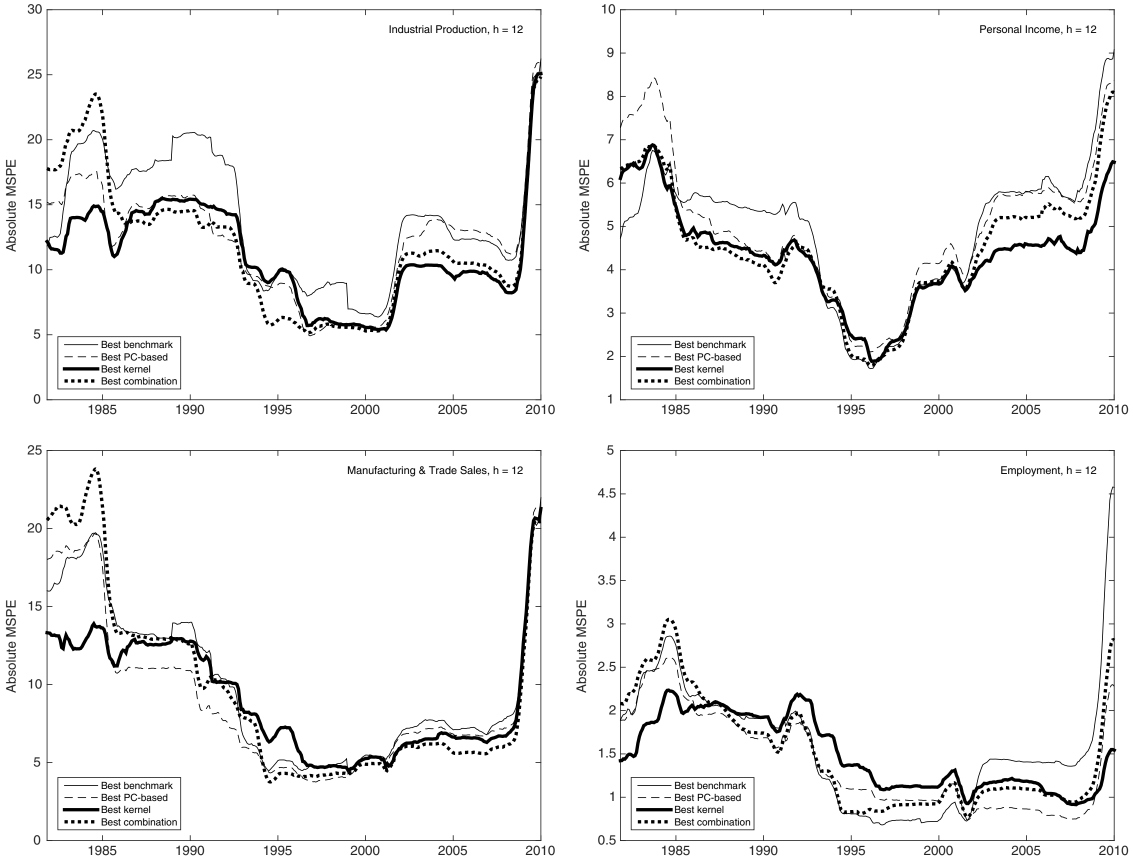

Fig. 1. Ten-year rolling-window mean squared prediction errors for four macroeconomic series, for a forecast horizon of $h = 12$ months, for the best-performing benchmark, PC-based, kernel, and combination methods.

relatively more difficult during less volatile periods, in the sense that the relative MSPEs seem to be inversely related to the rolling variance of the series being predicted; see Fig. 2. These results corroborate the findings of Stock and Watson (2007) for US inflation. Concerning the second point, it is interesting to note that the fluctuations in the relative MSPE are generally more pronounced for KRR than for PC-based methods. This suggests that kernel-based forecasts are most valuable during periods of turmoil.

5. Conclusion

We have introduced kernel ridge regression as a framework for accommodating nonlinear predictive relationships in a data-rich environment. We have extended the existing kernel methodology to enable it to be used in the time series contexts that are typical of macroeconomic and financial applications. These extensions involve the incorporation of unpenalized linear terms into the forecast equation and the use of a computationally efficient cross-validation procedure for model selection. Our simulation study suggests that this method can deal with the type of data that arise frequently in economic analysis, namely, data with a factor structure.

The empirical application to forecasting four key US macroeconomic variables — production, income, sales, and employment — shows that kernel-based methods can provide more accurate forecasts than well-established linear and nonlinear, autoregressive and principal-components-based methods. Kernel ridge regression exhibits a relatively consistently good predictive performance, especially at longer horizons, even during the crisis period in 2008–09. Of the kernel methods, the Gaussian kernel is found to produce the most reliable forecasts. This finding implies that it is not just the ridge term that contributes to the predictive accuracy, with accounting for nonlinearity leading to additional improvements in many cases. As the use of the Gaussian kernel does not require the forecaster to specify the form of nonlinearity in advance, this method is a powerful tool. Another advantage relative to other nonlinear techniques is the fact that kernel ridge regression is much less prone to producing occasional “insane” forecasts.

Finally, we provide statistical evidence that kernel-based forecasts contain information that is missed by principal-components-based forecasts. Other nonlinear forecast methods, including threshold autoregressions, polynomial sieves, nonparametric regressions, and nonlinear principal component regressions, do not have this property. This suggests that forecast combinations have potential, and indeed, we find that combination forecasts that include kernel methods significantly outperform those that exclude KRR. We conclude that the kernel

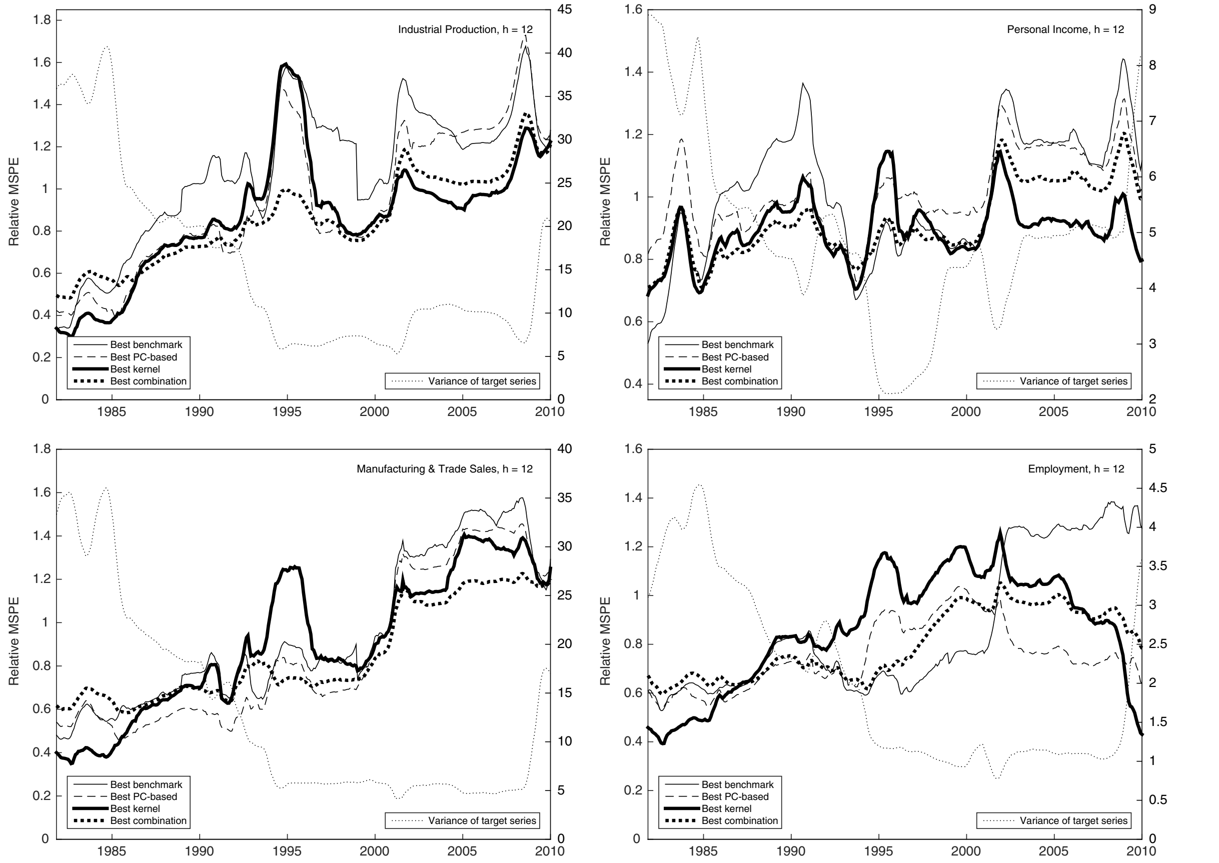

Fig. 2. Ten-year rolling-window MSPEs for four macroeconomic series, for the best-performing benchmark, PC-based, kernel, and combination methods, relative to the variance of the series being predicted.

methodology is a valuable addition to the macroeconomic forecaster's toolkit.

Acknowledgments

We thank conference participants at the International Conferences on Computational and Financial Econometrics (Limassol, October 2009, and London, December 2010), at the Eurostat Colloquium on Modern Tools for Business Cycle Analysis (Luxembourg, September 2010), at the International Conference on High-Dimensional Econometric Modelling at Cass Business School (London, December 2010), at the Applied Time Series Econometrics Workshop at the Federal Reserve Bank of St. Louis (St. Louis, April 2011), at the Netherlands Econometric Study Group (Rotterdam, June 2011), at the International Symposium on Forecasting (Prague, June 2011), at the Info-Metrics Workshop on Information Theory and Shrinkage Estimation (Washington DC, November 2011), and at the European Central Bank Workshop on Forecasting Techniques (Frankfurt, May 2012), as well as the Associate Editor and the anonymous referees, for useful comments and suggestions.

The first author acknowledges support from CREATES, Center for Research in Econometric Analysis of Time Series, funded by the Danish National Research Foundation (DNRF78), and additional financial support from the Danish Council for Independent Research (Grant #12-125914).

Appendix A. Technical results

This appendix contains the derivations of three results stated in Section 2. In Appendix A.1, we derive the forecast equation (Eq. (3)) for kernel ridge regression with additional unpenalized linear terms. In Appendix A.2, we obtain the mapping that corresponds to the Gaussian kernel function. Finally, in Appendix A.3. we describe a computationally efficient cross-validation method for selecting tuning parameters in KRR.

A.1. Kernel ridge regression with unpenalized linear terms

We showed in Section 2.2 that minimizing the penalized least-squares criterion $\|y - Z\gamma\|^2 + \lambda \|\gamma\|^2$ leads to the forecast $\hat{y}_* = k_*' (K + \lambda I_T)^{-1} y$, as given in Eq. (2). In this appendix, we modify this forecast equation to allow for unpenalized linear terms as in the generalized forecast equation $\hat{y}_* = w_*'\hat{\beta} + z_*'\hat{\gamma}$, where the $P \times 1$ vector w_* contains the variables to be treated linearly. In this case, we seek to minimize

$$\|y - W\beta - Z\gamma\|^2 + \lambda \|\gamma\|^2 \tag{A.1}$$

over the $P \times 1$ vector β and the $M \times 1$ vector γ. For a given $\hat{\beta}$, we can proceed, as in Section 2.2, to find

$$\hat{\gamma} = Z' \left(K + \lambda I_T\right)^{-1} \left(y - W\hat{\beta}\right). \tag{A.2}$$

On the other hand, for a given $\hat{\gamma}$, minimizing the criterion in Eq. (A.1) is equivalent to an ordinary least squares regression, which gives

$$\hat{\beta} = \left(W'W\right)^{-1} W' \left(y - Z\hat{\gamma}\right). \tag{A.3}$$

If we pre-multiply both sides of Eq. (A.3) by $W'W$, substitute the expression for $\hat{\gamma}$ from Eq. (A.2) into Eq. (A.3), and recall that $K = ZZ'$, we get

$$\begin{aligned} W'W\hat{\beta} &= W' \left(y - K\left(K + \lambda I_T\right)^{-1} \left(y - W\hat{\beta}\right)\right) \\ &= W'y - W'K\left(K + \lambda I_T\right)^{-1} y \\ &\quad + W'K\left(K + \lambda I_T\right)^{-1} W\hat{\beta}. \end{aligned}$$

Collecting the terms involving $\hat{\beta}$ on the left-hand side of this equation, and rearranging, we obtain

$$\begin{aligned} & W' \left(I_T - K\left(K + \lambda I_T\right)^{-1}\right) W\hat{\beta} \\ &\quad = W' \left(I_T - K\left(K + \lambda I_T\right)^{-1}\right) y. \end{aligned}$$

Using the fact that $I_T - K\left(K + \lambda I_T\right)^{-1} = \left(K + \lambda I_T - K\right)\left(K + \lambda I_T\right)^{-1} = \lambda\left(K + \lambda I_T\right)^{-1}$ then leads to

$$\hat{\beta} = \left(W'\left(K + \lambda I_T\right)^{-1} W\right)^{-1} W'\left(K + \lambda I_T\right)^{-1} y.$$

If we then substitute this result and Eq. (A.2) into the forecast equation $\hat{y}_* = z_*'\hat{\gamma} + w_*'\hat{\beta}$, and recall that $k_* = Zz_*$, we find

$$\begin{aligned} \hat{y}_* &= k_*'\left(K + \lambda I_T\right)^{-1} \left(I_T - W\left(W'\left(K + \lambda I_T\right)^{-1} W\right)^{-1}\right. \\ &\qquad \left. \times W'\left(K + \lambda I_T\right)^{-1}\right) y \\ &\qquad + w_*'\left(W'\left(K + \lambda I_T\right)^{-1} W\right)^{-1} W'\left(K + \lambda I_T\right)^{-1} y. \end{aligned} \tag{A.4}$$

To obtain a more manageable equation, note that by the partitioned matrix inversion formula,

$$\begin{aligned} & \begin{pmatrix} K + \lambda I_T & W \\ W' & 0 \end{pmatrix}^{-1} \\ &= \begin{pmatrix} \left(K + \lambda I_T\right)^{-1} \left(I_T - WSW'\left(K + \lambda I_T\right)^{-1}\right) & \left(K + \lambda I_T\right)^{-1} WS \\ SW'\left(K + \lambda I_T\right)^{-1} & -S \end{pmatrix}, \end{aligned} \tag{A.5}$$

where $S = \left(W'\left(K + \lambda I_T\right)^{-1} W\right)^{-1}$. It follows from this result that Eq. (A.4) is equivalent to the forecast equation (Eq. (3)) in Section 2.2:

$$\hat{y}_* = \begin{pmatrix} k_* \\ w_* \end{pmatrix}' \begin{pmatrix} K + \lambda I_T & W \\ W' & 0 \end{pmatrix}^{-1} \begin{pmatrix} y \\ 0 \end{pmatrix}.$$

A.2. Expansion of the Gaussian kernel

In this appendix, we derive the mapping φ that corresponds to the Gaussian kernel function. As was stated in Eq. (5) in Section 2.4, this kernel function is defined as $\kappa\left(a, b\right) = \exp\left(\|a - b\|^2 / 2\right)$. If we write $-(1/2)\|a - b\|^2 = -a'a/2 - b'b/2 + a'b$ and expand the Taylor series for $\exp\left(a'b\right)$, we obtain

$$\kappa\left(a, b\right) = e^{-a'a/2}\, e^{-b'b/2} \sum_{m=0}^{\infty} \frac{1}{m!} \left(a'b\right)^m. \tag{A.6}$$

We proceed by expanding $\left(a'b\right)^m$ as a multinomial series:

$$\begin{aligned} \left(a'b\right)^m &= \left(\sum_{n=1}^{N} a_n b_n\right)^m \\ &= \sum\sum \cdots \sum_{\left\{\sum_{n=1}^{N} d_n = m,\ \text{all } d_n \geq 0\right\}} \left(\frac{m!}{\prod_{n=1}^{N} d_n!} \prod_{n=1}^{N} \left(a_n b_n\right)^{d_n}\right). \end{aligned}$$

Substituting this result into Eq. (A.6), we find

$$\begin{aligned} \kappa\left(a, b\right) &= e^{-a'a/2}\, e^{-b'b/2} \\ &\quad \times \sum_{m=0}^{\infty} \left(\frac{1}{m!} \sum\sum \cdots \sum_{\left\{\sum_{n=1}^{N} d_n = m,\ \text{all } d_n \geq 0\right\}} \left(\frac{m!}{\prod_{n=1}^{N} d_n!} \prod_{n=1}^{N} \left(a_n b_n\right)^{d_n}\right)\right) \\ &= e^{-a'a/2}\, e^{-b'b/2} \sum_{d_1=0}^{\infty} \sum_{d_2=0}^{\infty} \cdots \sum_{d_N=0}^{\infty} \left(\prod_{n=1}^{N} \frac{\left(a_n b_n\right)^{d_n}}{d_n!}\right). \end{aligned}$$

Finally, we split the product into two factors that depend only on a and only on b, respectively:

$$\begin{aligned} \kappa\left(a, b\right) &= \sum_{d_1=0}^{\infty} \sum_{d_2=0}^{\infty} \cdots \sum_{d_N=0}^{\infty} \left(e^{-a'a/2} \prod_{n=1}^{N} \frac{a_n^{d_n}}{\sqrt{d_n!}}\right) \\ &\qquad \times \left(e^{-b'b/2} \prod_{n=1}^{N} \frac{b_n^{d_n}}{\sqrt{d_n!}}\right). \end{aligned} \tag{A.7}$$

As this expression shows, $\kappa\left(a, b\right) = \varphi\left(a\right)' \varphi\left(b\right)$, where, as was claimed in Section 2.4, $\varphi\left(a\right)$ contains as elements, for each combination of degrees $d_1, d_2, \ldots, d_N \geq 0$,

$$e^{-a'a/2} \prod_{n=1}^{N} \frac{a_n^{d_n}}{\sqrt{d_n!}}.$$

A.3. Computationally efficient cross-validation

In this appendix, we describe a computationally efficient method of performing cross-validation, which we employ to select the tuning parameters in KRR. Our derivation extends the results of Cawley and Talbot (2008) in two ways: it allows for the unpenalized linear terms in the forecast equation (Eq. (3)), and our results are applicable to general k-fold cross-validation rather than just leave-one-out cross-validation. The result of Appendix A.1 can be summarized as follows: kernel ridge regression leads to the forecast

$$\hat{y}_* = \begin{pmatrix} k_* \\ w_* \end{pmatrix}' \begin{pmatrix} \hat{\alpha} \\ \hat{\beta} \end{pmatrix} \quad \text{with}$$

$$\begin{pmatrix} K+\lambda I_T & W \\ W' & 0 \end{pmatrix} \begin{pmatrix} \hat{\alpha} \\ \hat{\beta} \end{pmatrix} = \begin{pmatrix} y \\ 0 \end{pmatrix}. \tag{A.8}$$

The first step in cross-validation is to estimate the model on a subset of the available data; say, the first T_{est} observations, leaving out $T - T_{est} = T_{val}$ observations for validating the model. As $K = ZZ'$, and each row of Z depends only on the corresponding row of X, the only elements in K that depend on the first T_{est} observation are those in the first T_{est} rows and those in the first T_{est} columns. We therefore partition K, and likewise W, $\hat{\alpha}$, and y, as follows:

$$K = \begin{pmatrix} K_{est,est} & K_{est,val} \\ K'_{est,val} & K_{val,val} \end{pmatrix}, \qquad W = \begin{pmatrix} W_{est} \\ W_{val} \end{pmatrix},$$

$$\hat{\alpha} = \begin{pmatrix} \hat{\alpha}_{est} \\ \hat{\alpha}_{val} \end{pmatrix} \quad \text{and} \quad y = \begin{pmatrix} y_{est} \\ y_{val} \end{pmatrix}.$$

From Eq. (A.8), we then have

$$\begin{pmatrix} K_{est,est} + \lambda I_{T_{est}} & K_{est,val} & W_{est} \\ K'_{est,val} & K_{val,val} + \lambda I_{T_{val}} & W_{val} \\ W'_{est} & W'_{val} & 0 \end{pmatrix} \begin{pmatrix} \hat{\alpha}_{est} \\ \hat{\alpha}_{val} \\ \hat{\beta} \end{pmatrix} = \begin{pmatrix} y_{est} \\ y_{val} \\ 0 \end{pmatrix},$$

or equivalently, separating the equations involving y_{val} from the others,

$$\left(K_{val,val} + \lambda I_{T_{val}} \right) \hat{\alpha}_{val} + \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix}' \begin{pmatrix} \hat{\alpha}_{est} \\ \hat{\beta} \end{pmatrix} = y_{val}, \tag{A.9}$$

$$\begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix} \hat{\alpha}_{val} + \begin{pmatrix} K_{est,est} + \lambda I_{T_{est}} & W_{est} \\ W'_{est} & 0 \end{pmatrix} \begin{pmatrix} \hat{\alpha}_{est} \\ \hat{\beta} \end{pmatrix} = \begin{pmatrix} y_{est} \\ 0 \end{pmatrix}. \tag{A.10}$$

The forecast of y_{val} based on a model estimated on the other observations clearly equals

$$\tilde{y}_{val} = \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix}' \begin{pmatrix} K_{est,est} + \lambda I_{T_{est}} & W_{est} \\ W'_{est} & 0 \end{pmatrix}^{-1} \begin{pmatrix} y_{est} \\ 0 \end{pmatrix}, \tag{A.11}$$

and, using Eqs. (A.9) and (A.10), we may write

$$\begin{aligned} \tilde{y}_{val} &= \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix}' \begin{pmatrix} \hat{\alpha}_{est} \\ \hat{\beta} \end{pmatrix} + \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix}' \\ &\quad \times \begin{pmatrix} K_{est,est} + \lambda I_{T_{est}} & W_{est} \\ W'_{est} & 0 \end{pmatrix}^{-1} \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix} \hat{\alpha}_{val} \\ &= y_{val} - \left(K_{val,val} + \lambda I_{T_{val}} \right) \hat{\alpha}_{val} + \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix}' \\ &\quad \times \begin{pmatrix} K_{est,est} + \lambda I_{T_{est}} & W_{est} \\ W'_{est} & 0 \end{pmatrix}^{-1} \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix} \hat{\alpha}_{val} \\ &= y_{val} - \left(K_{val,val} + \lambda I_{T_{val}} - \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix}' \right. \\ &\quad \left. \times \begin{pmatrix} K_{est,est} + \lambda I_{T_{est}} & W_{est} \\ W'_{est} & 0 \end{pmatrix}^{-1} \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix} \right) \hat{\alpha}_{val}. \end{aligned}$$

Now, $K_{val,val} + \lambda I_{T_{val}} - \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix}' \begin{pmatrix} K_{est,est} + \lambda I_{T_{est}} & W_{est} \\ W'_{est} & 0 \end{pmatrix}^{-1} \begin{pmatrix} K_{est,val} \\ W'_{val} \end{pmatrix}$ is the inverse of the (val, val) block of $\begin{pmatrix} K_{est,est} + \lambda I_{T_{est}} & K_{est,val} & W_{est} \\ K'_{est,val} & K_{val,val} + \lambda I_{T_{val}} & W_{val} \\ W'_{est} & W'_{val} & 0 \end{pmatrix}^{-1} = \begin{pmatrix} K+\lambda I_T & W \\ W' & 0 \end{pmatrix}^{-1}$, as can be seen by using the partitioned matrix inversion formula. Therefore, the vector of cross-validation errors for this choice of estimation and validation samples is equal to

$$\left[\text{block } (val, val) \text{ of } \begin{pmatrix} K+\lambda I_T & W \\ W' & 0 \end{pmatrix}^{-1} \right]^{-1} \times \left[\text{subvector } (val) \text{ of } \hat{\alpha} \right]. \tag{A.12}$$

Observe that both $\hat{\alpha}$ and this inverse are needed to compute the forecast $\hat{y}_*$. Thus, in the process of making the out-of-sample prediction, we can find all cross-validation errors without performing any additional computations, aside from solving a T_{val}-dimensional system of linear equations to obtain the cross-validation errors using Eq. (A.12) for each validation sample. In terms of computational efficiency, this is a marked improvement over a naïve implementation of cross-validation using Eq. (A.11), where the system to be solved is of dimension $T_{est} + P$, followed by multiplication of the result by a $T_{val} \times (T_{est} + P)$ matrix. In the applications in this paper, we use five-fold cross-validation in samples of size $T = 120$, which implies that $T_{est} = 96$ and $T_{val} = 24$. The value of P varies from 0 to 7.

As a final note, we mention that the matrix inverse in Eq. (A.12) can also be computed efficiently. As $K + \lambda I_T$ is symmetric and positive definite, its inverse can be computed from its Cholesky decomposition. The inverse of the full matrix can then be calculated using Eq. (A.5) in Appendix A.1.

Appendix B. Supplementary data

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.ijforecast.2015.11.017.

References

Aiolfi, M., & Favero, C. A. (2005). Model uncertainty, thick modelling and the predictability of stock returns. *Journal of Forecasting*, *24*, 233–254.

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

Bańbura, M., Giannone, D., & Reichlin, L. (2010). Large Bayesian vector autoregressions. *Journal of Applied Econometrics*, *25*, 71–92.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In D. Haussler (Ed.), *Proceedings of the annual conference on computational learning theory* (pp. 144–152). Pittsburgh, Pennsylvania: ACM Press.

Broomhead, D. S., & Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, *2*, 321–355.

Çakmaklı, C., & van Dijk, D. (2010). Getting the most out of macroeconomic information for predicting stock returns and volatility. Tinbergen Institute Discussion Paper 2010-115/4.

Carriero, A., Kapetanios, G., & Marcellino, M. (2011). Forecasting large datasets with Bayesian reduced rank multivariate models. *Journal of Applied Econometrics*, *26*, 735–761.

Cawley, G. C., & Talbot, N. L. C. (2008). Efficient approximate leave-one-out cross-validation for kernel logistic regression. *Machine Learning*, *71*, 243–264.

Chen, X. (2007). Large-sample sieve estimation of semi-nonparametric models. In J. J. Heckman, & E. E. Leamer (Eds.), *Handbook of econometrics* (pp. 5549–5632). Amsterdam: Elsevier.

De Mol, C., Giannone, D., & Reichlin, L. (2008). Forecasting using a large number of predictors: Is Bayesian shrinkage a valid alternative to principal components? *Journal of Econometrics*, *146*, 318–328.

Diebold, F. X., & Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business and Economic Statistics*, *13*, 134–144.

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

Exterkate, P. (2013). Model selection in kernel ridge regression. *Computational Statistics and Data Analysis*, *68*, 1–16.

Faust, J., & Wright, J. H. (2009). Comparing Greenbook and reduced form forecasts using a large realtime dataset. *Journal of Business and Economic Statistics*, *27*, 468–479.

Frank, I. E., & Friedman, J. H. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, *35*, 109–148.

Giovannetti, B. C. (2013). Nonlinear forecasting using factor-augmented models. *Journal of Forecasting*, *32*, 32–40.

Groen, J. J. J., & Kapetanios, G. (2008). Revisiting useful approaches to data-rich macroeconomic forecasting. In *Federal Reserve Bank of New York Staff Report 327*.

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

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

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

Kim, H. H., & Swanson, N. R. (2014). Forecasting financial and macroeconomic variables using data reduction methods: New empirical evidence. *Journal of Econometrics*, *178*, 352–367.

Kock, A. B., & Teräsvirta, T. (2011). Forecasting with non-linear models. In M. P. Clements, & D. F. Hendry (Eds.), *Oxford handbook of economic forecasting* (pp. 61–87). Oxford University Press.

Ludvigson, S. C., & Ng, S. (2007). The empirical risk-return relation: A factor analysis approach. *Journal of Financial Economics*, *83*, 171–222.

Ludvigson, S. C., & Ng, S. (2009). Macro factors in bond risk premia. *Review of Financial Studies*, *22*, 5027–5067.

Medeiros, M. C., Teräsvirta, T., & Rech, G. (2006). Building neural network models for time series: A statistical approach. *Journal of Forecasting*, *25*, 49–75.

Müller, K.-R., Smola, A., Rätsch, G., Schölkopf, B., Kohlmorgen, J., & Vapnik, V. (1997). Predicting time series with support vector machines. In W. Gerstner, A. Germond, M. Hasler, & J.-D. Nicoud (Eds.), *ICANN'97, Artificial neural networks* (pp. 999–1004). Berlin: Springer.

Pagan, A. R., & Ullah, A. (1999). *Nonparametric econometrics*. Cambridge University Press.

Poggio, T. (1975). On optimal nonlinear associative recall. *Biological Cybernetics*, *19*, 201–209.

Racine, J. (2000). Consistent cross-validatory model-selection for dependent data: hv-block cross-validation. *Journal of Econometrics*, *99*, 39–61.

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

Schölkopf, B., Smola, A., & Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, *10*, 1299–1319.

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

Stock, J. H., & Watson, M. W. (1999). A comparison of linear and nonlinear univariate models for forecasting macroeconomic time series. In R. F. Engle, & H. White (Eds.), *Cointegration, causality and forecasting. A festschrift in honour of Clive W.J. Granger* (pp. 1–44). Oxford University Press.

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

Stock, J.H., & Watson, M.W. (2005). Implications of dynamic factor models for VAR analysis. NBER Working Paper No. 11467.

Stock, J. H., & Watson, M. W. (2006). Forecasting with many predictors. In G. Elliot, C. W. J. Granger, & A. Timmermann (Eds.), *Handbook of economic forecasting* (pp. 515–554). Amsterdam: Elsevier.

Stock, J. H., & Watson, M. W. (2007). Why has US inflation become harder to forecast? *Journal of Money, Credit and Banking*, *39*, 3–33.

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

Swanson, N. R., & White, H. (1995). A model-selection approach to assessing the information in the term structure using linear models and artificial neural networks. *Journal of Business and Economic Statistics*, *13*, 265–275.

Teräsvirta, T. (2006). Forecasting economic variables with nonlinear models. In G. Elliot, C. W. J. Granger, & A. Timmermann (Eds.), *Handbook of economic forecasting* (pp. 413–458). Amsterdam: Elsevier.

Teräsvirta, T., van Dijk, D., & Medeiros, M. C. (2005). Linear models, smooth transition autoregressions, and neural networks for forecasting macroeconomic time series: A re-examination. *International Journal of Forecasting*, *21*, 755–774.

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

Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York: Springer.

White, H. (2006). Approximate nonlinear forecasting methods. In G. Elliot, C. W. J. Granger, & A. Timmermann (Eds.), *Handbook of economic forecasting* (pp. 459–514). Amsterdam: Elsevier.

Wright, J. H. (2009). Forecasting US inflation by Bayesian model averaging. *Journal of Forecasting*, *28*, 131–144.

Peter Exterkate is Lecturer in Econometrics at the School of Economics, University of Sydney, as well as International Research Fellow at CREATES, the Center for Research in Econometric Analysis of Time Series, Aarhus University. He is interested in pursuing nonlinear forecasting techniques using large data sets, in regard to both the interesting theoretical questions and the many different empirical applications it gives rise to. His applications are mainly in financial econometrics and macroeconomics.

Patrick J.F. Groenen is Professor in Statistics and Institute Director at the Econometric Institute, Erasmus University Rotterdam. His research interests lie in data modeling, multivariate analysis, visualization and optimization. He has written papers in journals such as *Quantitative Finance*, *Journal of Empirical Finance*, *Journal of Marketing Research*, *Psychometrika*, *Computational Statistics and Data Analysis*, *Journal of Classification*, and *Multivariate Behavioural Research*. He is coauthor of a textbook on multidimensional scaling published in the Statistics Series by Springer.

Christiaan Heij is Assistant Professor in Econometrics and Statistics at the Econometric Institute, Erasmus University Rotterdam. His current research interests lie in the areas of applied econometrics, statistics, and forecasting. He has published in journals in systems and control, econometrics, and statistics. He has (co-)authored four books, among which is the textbook "Econometric methods with applications in business and economics", published in 2004 by Oxford University Press.

Dick van Dijk is Professor in Financial Econometrics at the Econometric Institute, Erasmus University Rotterdam. His research interests include volatility modeling and forecasting, high-frequency data, business cycle analysis, and nonlinear time series analysis. He has published in the *Journal of Applied Econometrics*, *Journal of Business and Economic Statistics*, *Journal of Econometrics*, *Journal of Empirical Finance*, and *Review of Economics and Statistics*, among others.