ISSN 1440-771X

MONASH University

Department of Econometrics and Business Statistics

http://www.buseco.monash.edu.au/depts/ebs/pubs/wpapers/

Recursive and direct multi-step forecasting: the best of both worlds

Souhaib Ben Taieb and Rob J Hyndman

September 2012

Working Paper 19/12

Recursive and direct multi-step forecasting: the best of both worlds

Souhaib Ben Taieb
Machine Learning Group, Department of Computer Science,
Université Libre de Bruxelles,
Brussels, Belgium.
Email: Souhaib.Ben.Taieb@ulb.ac.be

Rob J Hyndman
Department of Econometrics and Business Statistics,
Monash University,
Clayton VIC 3800,
Australia.
Email: Rob.Hyndman@monash.edu

2 September 2012

JEL classification: J11, C53, C14, C32

Recursive and direct multi-step forecasting: the best of both worlds

Abstract

We propose a new forecasting strategy, called rectify, that seeks to combine the best properties of both the recursive and direct forecasting strategies. The rationale behind the rectify strategy is to begin with biased recursive forecasts and adjust them so they are unbiased and have smaller error. We use linear and nonlinear simulated time series to investigate the performance of the rectify strategy and compare the results with those from the recursive and the direct strategies. We also carry out some experiments using real world time series from the M3 and the NN5 forecasting competitions. We find that the rectify strategy is always better than, or at least has comparable performance to, the best of the recursive and the direct strategies. This finding makes the rectify strategy very attractive as it avoids making a choice between the recursive and the direct strategies which can be a difficult task in real-world applications.

Keywords: Multi-step forecasting; forecasting strategies; recursive forecasting; direct forecasting; linear time series; nonlinear time series; M3 competition; NN5 competition.

1 Introduction

Traditionally, multi-step forecasting has been handled recursively, where a single time series model is estimated and each forecast is computed using previous forecasts. More recently, direct calculation of multi-step forecasting has been proposed, where a separate time series model for each forecasting horizon is estimated, and forecasts are computed only on the observed data. Choosing between these different strategies involves a trade-off between bias and estimation variance. Recursive forecasting is biased when the underlying model is nonlinear, but direct forecasting has higher variance because it uses fewer observations when estimating the model, especially for longer forecast horizons.

The literature on this topic often involves comparing the recursive and direct strategies, and discussing the conditions under which one or other is better. For example, Ing (2003) shows that in the linear case, the recursive MSE is greater than the direct MSE. Chevillon (2007) concludes that the direct strategy is most beneficial when the model is misspecified.

In this paper, we take a different approach and propose a new forecasting strategy that seeks to combine the best properties of both the recursive and direct strategies. The rationale behind the rectify strategy is to begin with biased recursive forecasts and adjust them so they are unbiased and have smaller error.

In the next section, we present both the recursive and the direct strategy together with a decomposition of their corresponding mean squared error. Section 3 presents the rectify strategy using the same terminology as for the recursive and the direct strategies to allow theoretical comparisons. Section 4 gives some details about the set-up of our experiments. Section 5 investigates the performance of the different strategies using linear and nonlinear simulated time series. Section 6 shows the performance of the different strategies with time series from two forecasting competitions, namely the M3 and the NN5 competition. Finally, we conclude our work in Section 7.

2 Forecasting strategies

Given a univariate time series $\{y_1, \ldots, y_T\}$ comprising T observations, we want to forecast the next H values of the time series.

We will assume the data come from a possibly nonlinear autoregressive process of the form

$$y_t = f(\boldsymbol{x}_{t-1}) + \varepsilon_t \text{ with } \boldsymbol{x}_t = [y_t, \ldots, y_{t-d+1}]',$$

where $\{\varepsilon_t\}$ is a stochastic iid error process with mean zero, variance σ^2, $\mathbb{E}[\varepsilon_t^3] = 0$ (for simplicity) and $\kappa = \mathbb{E}(\varepsilon^4) > 0$. The process is specified by a function f, embedding dimension d, and an error term ε_t.

In this article, we assume the goal of forecasting is to estimate the conditional mean $\mu_{t+h|t} = \mathrm{E}(y_{t+h} \mid \boldsymbol{x}_t)$, and we will evaluate different forecasting strategies by how well they approximate $\mu_{t+h|t}$.

When $h = 1$, we have the simple expression $\mu_{t+1|t} = f(\boldsymbol{x}_t)$. If f is linear, we can also write down some relatively simple expressions (Fan & Yao 2003, p.118):

$$\mu_{t+h|t} = \begin{cases} f([f^{(h-1)}(\boldsymbol{x}_t), \ldots, f^{(h-d)}(\boldsymbol{x}_t)]'), & \text{if } h > 0; \\ \boldsymbol{x}_t' \boldsymbol{w}_h, & \text{if } 1-d \leq h \leq 0. \end{cases} \tag{1}$$

where $\boldsymbol{w}_h$ has jth element equal to 1 if $j = 1-h$ and equal to 0 otherwise. Thus, when f is linear, the conditional mean forecasts can be computed recursively. More generally, for nonlinear f and $h > 1$, the calculation of $\mu_{t+h|t}$ has no simple form.

Each forecasting strategy involves estimating one or more models which are not necessarily of the same form as f and may not have the same embedding dimension as f. For one-step forecasts, the model will be denoted by $y_t = m(\boldsymbol{x}_{t-1}; \boldsymbol{\theta}) + e_t$ where $\boldsymbol{x}_t = [y_t, \ldots, y_{t-p+1}]'$. That is we estimate (or assume) the form of m, the parameters $\boldsymbol{\theta}$ and the dimension p from a set of training data. If m is of the same form as f (up to some estimable parameters), we write $m \asymp f$. Ideally, we would like $p = d$, $m \asymp f$ and the estimates of $\boldsymbol{\theta}$ close to the true parameters. But we are allowing for model mis-specification by not making these assumptions. For multi-step forecasts, some strategies will use the same model m as for one-step forecasts, while other strategies may involve additional or alternative models to be estimated.

If we let $\hat{m}^{(h)}(\boldsymbol{x}_t)$ denote the forecasts of a given strategy at horizon h and define $m^{(h)}(\boldsymbol{x}_t) = \mathbb{E}[\hat{m}^{(h)}(\boldsymbol{x}_t) \mid \boldsymbol{x}_t]$, then the mean squared error (MSE) at horizon h is given by

$$\mathrm{MSE}_h = \mathbb{E}\left[(y_{t+h} - \hat{m}^{(h)}(\boldsymbol{x}_t))^2\right]$$

$$= \underbrace{\mathbb{E}\left[(y_{t+h} - \mu_{t+h|t})^2\right]}_{\text{Noise}} + \underbrace{(\mu_{t+h|t} - m^{(h)}(\boldsymbol{x}_t))^2}_{\text{Bias } b_h^2} + \underbrace{\mathbb{E}\left[(\hat{m}^{(h)}(\boldsymbol{x}_t) - m^{(h)}(\boldsymbol{x}_t))^2\right]}_{\text{Variance}} \tag{2}$$

where all expectations are conditional on $\boldsymbol{x}_t$. The variance term on the right will converge to zero as the size of the training set increases, and so we will not consider it further.

To simplify the derivation of the remaining terms, we use similar arguments to Atiya et al. (1999) and consider only $h = 2$. An analogous approach can be used to derive the MSE for larger forecast horizons.

Now y_{t+2} is given by

$$y_{t+2} = f(y_{t+1}, \ldots, y_{t-d+2}) + \varepsilon_{t+2} = f(f(\boldsymbol{x}_t) + \varepsilon_{t+1}, \ldots, y_{t-d+2}) + \varepsilon_{t+2}.$$

Using Taylor series approximations up to second-order terms, we get

$$y_{t+2} \approx f(f(\boldsymbol{x}_t), \ldots, y_{t-d+2}) + \varepsilon_{t+1} f_{x_1} + \tfrac{1}{2}(\varepsilon_{t+1})^2 f_{x_1 x_1} + \varepsilon_{t+2},$$

where f_{x_1} is the derivative of f with respect to its first argument, $f_{x_1 x_1}$ is its second derivative with respect to its first argument, and so on.

The noise term depends only on the data generating process and is given by

$$\begin{aligned}
&\mathbb{E}\left[(y_{t+2} - \mu_{t+2|t})^2\right] \\
&\quad \approx \mathbb{E}\left[\left(f(f(\boldsymbol{x}_t), \ldots, y_{t-d+2}) + \varepsilon_{t+1} f_{x_1} + \tfrac{1}{2}\varepsilon_{t+1}^2 f_{x_1 x_1} + \varepsilon_{t+2} - f(f(\boldsymbol{x}_t), \ldots, y_{t-d+2}) - \tfrac{1}{2}\sigma^2 f_{x_1 x_1}\right)^2\right] \\
&\quad = \sigma^2(1 + f_{x_1}^2) + \tfrac{1}{4}(\kappa - \sigma^4) f_{x_1 x_1}^2.
\end{aligned}$$

Consequently, the MSE at horizon $h = 2$ is given by

$$\text{MSE}_2 \approx \sigma^2(1 + f_{x_1}^2) + \tfrac{1}{4}(\kappa - \sigma^4) f_{x_1 x_1}^2 + (\mu_{t+h|t} - m^{(h)}(\boldsymbol{x}_t))^2 \tag{3}$$

2.1 The recursive strategy

In the recursive strategy, we estimate the model

$$y_t = m(\boldsymbol{x}_{t-1}; \boldsymbol{\theta}) + e_t, \quad \text{where} \quad \boldsymbol{x}_t = [y_t, \ldots, y_{t-p+1}]' \tag{4}$$

and $\mathbb{E}[e_t] = 0$. In standard one-step optimization with squared errors, the objective function of the recursive strategy is $\mathbb{E}\Big[(y_{t+1} - m(\boldsymbol{x}_t;\boldsymbol{\theta}))^2 \mid \boldsymbol{x}_t\Big]$ and the parameters $\boldsymbol{\theta}$ are estimated by

$$\hat{\boldsymbol{\theta}} = \operatorname*{argmin}_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \sum_t (y_t - m(\boldsymbol{x}_{t-1};\boldsymbol{\theta}))^2, \tag{5}$$

where $\boldsymbol{\Theta}$ denotes the parameter space.

We compute forecasts recursively as in (1):

$$\hat{m}^{(h)}(\boldsymbol{x}_t) = \begin{cases} \hat{m}([\hat{m}^{(h-1)}(\boldsymbol{x}_t), \ldots, \hat{m}^{(h-p)}(\boldsymbol{x}_t)]'), & \text{if } h > 0; \\ \boldsymbol{x}_t'\boldsymbol{w}_h, & \text{if } 1 - p \leq h \leq 0; \end{cases}$$

where $\hat{m}(\boldsymbol{x})$ is a shorthand notation for $m(\boldsymbol{x};\hat{\boldsymbol{\theta}})$. These forecasts are also sometimes called "iterated multi-step" (IMS) forecasts (e.g., Chevillon & Hendry 2005, Chevillon 2007, Franses & Legerstee 2009).

The choice of $\hat{\boldsymbol{\theta}}$ given by (5) minimizes the mean squared error of the one-step forecasts, and so ensures (see, for example, Hastie et al. 2008, p.18) that $m^{(1)}(\boldsymbol{x}_t) = \mu_{t+1|t}$ provided $m \asymp f$ and $p = d$. Thus, the one-step forecasts are unbiased under these conditions. The same unbiasedness property does not hold for higher forecast horizons except in some special cases.

Again, we will consider only $h = 2$. The bias of the recursive strategy can be calculated as

$$\begin{aligned} b_2 = \mu_{t+2|t} - m^{(2)}(\boldsymbol{x}_t) &\approx f(f(\boldsymbol{x}_t), \ldots, y_{t-d+2}) + \tfrac{1}{2}\sigma^2 f_{x_1x_1} - m(m(\boldsymbol{x}_t), \ldots, y_{t-p+2}) \\ &\approx \Big[f(f(\boldsymbol{x}_t), \ldots, y_{t-d+2}) - m(m(\boldsymbol{x}_t), \ldots, y_{t-p+2})\Big] + \tfrac{1}{2}\sigma^2 f_{x_1x_1}. \end{aligned} \tag{6}$$

So even when $m \asymp f$ and $d = p$, the forecasts will be biased unless $f_{x_1x_1} = 0$. That is, recursive forecasts are unbiased if and only if f is linear, a linear model m is used, and when the embedding dimension is correctly determined; in all other situations, recursive forecasts will be biased for $h \geq 2$. In particular, the bias will be large whenever $|f_{x_1x_1}|$ is large; that is when f has high curvature.

Using expression (3), we see that the recursive strategy has a mean squared error at horizon $h = 2$ equal to

$$\begin{aligned} &\text{MSE}_2^{\text{recursive}} \\ &\quad\approx \sigma^2(1 + f_{x_1}^2) + \tfrac{1}{4}(\kappa - \sigma^4)f_{x_1x_1}^2 + \Big(\Big[f(f(\boldsymbol{x}_t), \ldots, y_{t-d+2}) - m(m(\boldsymbol{x}_t), \ldots, y_{t-p+2})\Big] + \tfrac{1}{2}\sigma^2 f_{x_1x_1}\Big)^2. \end{aligned}$$

When $m \asymp f$ and $d = p$, the MSE simplifies to

$$\text{MSE}_2^{\text{recursive}} \approx \sigma^2(1 + f_{x_1}^2) + \tfrac{1}{4}\kappa f_{x_1 x_1}^2.$$

But when the model is misspecified, either in the embedding dimension, or in the functional form of m, the MSE can be much larger.

A variation on the recursive forecasting strategy is to use a different set of parameters for each forecasting horizon:

$$\hat{\boldsymbol{\theta}}_h = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}} \sum_t \left[y_t - m^{(h)}(\boldsymbol{x}_{t-h}; \boldsymbol{\theta})\right]^2.$$

A further variation selects the parameters to minimize the forecast errors over the first H forecast horizons:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}\in\Theta}{\operatorname{argmin}} \sum_{h=1}^{H} \sum_t \left[y_t - m^{(h)}(\boldsymbol{x}_{t-h}; \boldsymbol{\theta})\right]^2.$$

In both of these variants, forecasts are still obtained recursively from the one-step model. The only difference with standard recursive forecasting is that the parameters are optimized differently to allow more accurate multi-step forecasts. Examples of machine learning models using these variations of the recursive strategy are recurrent neural networks (e.g., Williams & Zipser 1989, Werbos 1990) and local models with the nearest trajectories criterion (McNames 1998).

An advantage of using the recursive strategy is that only one model is required, saving significant computational time, especially when a large number of time series and forecast horizons are involved. The strategy also ensures that the fitted model m matches the assumed data generating process f as closely as possible. On the other hand, the recursive forecasts are not equal to the conditional mean, even when the model is exactly equivalent to the data generating process.

2.2 The direct strategy

With the direct strategy, different forecasting models are used for each forecast horizon:

$$y_t = m_h(y_{t-h}, \dots, y_{t-h-p_h}; \boldsymbol{\theta}_h) + e_{t,h}, \tag{7}$$

where $h = 1,\ldots,H$. For each model, the parameters $\boldsymbol{\theta}_h$ are estimated as follows

$$\hat{\boldsymbol{\theta}}_h = \underset{\boldsymbol{\theta}_h \in \boldsymbol{\Theta}_h}{\operatorname{argmin}} \sum_t [y_t - m_h(\boldsymbol{x}_{t-h}; \boldsymbol{\theta}_h)]^2. \tag{8}$$

Then forecasts are obtained for each horizon from the corresponding model, $\hat{m}^{(h)}(\boldsymbol{x}_t) = m_h(\boldsymbol{x}_t; \hat{\boldsymbol{\theta}}_h)$. This is sometimes also known as "direct multi-step" (DMS) forecasting (e.g., Chevillon & Hendry 2005, Chevillon 2007, Franses & Legerstee 2009).

Because multiple models are used, this approach involves a heavier computational load than recursive forecasting. Also, we no longer match the model used for forecasting with the assumed model; the various models are estimated independently and can in practice be quite different from each other.

Because of the squared errors in (8), $m^{(h)}(\boldsymbol{x}_t) = m_h(\boldsymbol{x}_t)$, and the bias of the direct strategy at horizon $h = 2$ is given by

$$b_2 = \mu_{t+2|t} - m^{(2)}(\boldsymbol{x}_t) \approx f(f(\boldsymbol{x}_t),\ldots,y_{t-d+2}) + \tfrac{1}{2}\sigma^2 f_{x_1x_1} - m_2(y_t,\ldots,y_{t-p_2+1}).$$

Thus the strategy leads to unbiased forecasts when $m_2(y_t,\ldots,y_{t-p+1}) \asymp f(f(\boldsymbol{x}_t),\ldots,y_{t-d+2}) + \frac{1}{2}\sigma^2 f_{x_1x_1}$. These conditions will be satisfied whenever m_2 is a sufficiently flexible model. Similar arguments can be used for other forecast horizons.

As for the recursive strategy, we can get the MSE for the direct strategy with

$$\mathrm{MSE}_2^{\text{direct}} \approx \sigma^2(1 + f_{x_1}^2) + \tfrac{1}{4}(\kappa - \sigma^4) f_{x_1x_1}^2 + \left[f(f(\boldsymbol{x}_t),\ldots,y_{t-d+2}) + \tfrac{1}{2}\sigma^2 f_{x_1x_1} - m_2(y_t,\ldots,y_{t-p_2+1})\right]^2.$$

When the strategy is unbiased, the MSE simplifies to

$$\mathrm{MSE}_2^{\text{direct}} \approx \sigma^2(1 + f_{x_1}^2) + \tfrac{1}{4}(\kappa - \sigma^4) f_{x_1x_1}^2.$$

Consequently, under ideal conditions when $m \asymp f$ and $p = d$ for the recursive strategy, and the direct strategy is unbiased, we find that the recursive strategy has larger MSE than the direct strategy:

$$\mathrm{MSE}_2^{\text{recursive}} - \mathrm{MSE}_2^{\text{direct}} \approx \tfrac{1}{4}\sigma^4 f_{x_1x_1}^2.$$

Ing (2003) shows that even in the linear case (when $f_{x_1x_1} = 0$), when m_h and m are assumed linear and $\{y_t\}$ is assumed stationary, the recursive MSE is greater than the direct MSE, with the difference being of order $O(T^{-3/2})$ where T is the length of the time series.

Chevillon & Hendry (2005) provide a useful review of some of the literature comparing recursive and direct forecasting strategies, and explore in detail the differences between the strategies for VAR models applied to stationary and difference-stationary time series. They show that for these models, when f, m and m_h are all assumed multivariate linear, and under various mis-specification conditions, then the recursive MSE is greater than the direct MSE.

In a later paper, Chevillon (2007) provides a further review and unifies some of the results in the literature. He concludes that the direct strategy is most beneficial when the model is misspecified (i.e., that m and f are not asymptotically equivalent), in particular when the data contains misspecified unit roots, neglected residual autocorrelation and omitted location shifts.

Putting aside the computational disadvantage of using the direct strategy, it also has a problem in generating forecasts from potentially very different models at different forecast horizons. Because every model is selected independently at each horizon, it is possible for consecutive forecasts to be based on different conditioning information and different model forms. This can lead to irregularities in the forecast function. These irregularities are manifest as a contribution to the forecast variance. The problem is exacerbated when each of the m_h models is allowed to be nonlinear and nonparametrically estimated (Chen et al. 2004).

3 The RECTIFY strategy

In this section we propose a new forecasting strategy that borrows from the strengths of the recursive and direct strategies. We call this new strategy "RECTIFY" because it begins with recursive forecasts and adjusts (or rectifies) them so they are unbiased and have smaller MSE.

We begin with a simple linear base model, and produce forecasts from it using the recursive strategy. These are known to be biased, even when the true data generating process is correctly specified by the base model. Then we correct these forecasts by modelling the forecast errors using a direct strategy. The resulting forecasts will be unbiased, provided the models used in the direct strategy are sufficiently flexible.

The advantage of this two-stage process is that it links all the direct forecast models together with the same unifying base model, thus reducing the irregularities that can arise with independent models, and so reducing the forecast variance. Of course, it is still possible for the direct models to be different from each other, but these differences are likely to be much smaller when modelling the errors from the recursive strategy than when modelling the time series directly.

We shall denote the simple linear base model by $y_t = z(\boldsymbol{x}_{t-1};\boldsymbol{\theta}) + e_t$, from which recursive forecasts are calculated. In the second stage, we adjust the forecasts from the base model by applying direct forecasting models to the errors from the recursive base forecasts; that is, we fit the models

$$y_t - z^{(h)}(y_{t-h},\dots,y_{t-h-p};\hat{\boldsymbol{\theta}}) = r_h(y_{t-h},\dots,y_{t-h-p_h};\boldsymbol{\gamma}_h) + e_{t,h} \qquad (9)$$

where $h = 1,\dots,H$.

The parameters for all models are estimated by least squares. Then forecasts are obtained for each horizon by combining the base model and the rectification models: $\hat{m}^{(h)}(\boldsymbol{x}_t) = \hat{z}^{(h)}(\boldsymbol{x}_t)+\hat{r}_h(\boldsymbol{x}_t)$. Let $m^{(h)}(\boldsymbol{x}_t) = \mathbb{E}[\hat{m}^{(h)}(\boldsymbol{x}_t) \mid \boldsymbol{x}_t]$. Then the bias of the rectify strategy at horizon $h = 2$ is given by

$$\begin{aligned} b_2 &= \mu_{t+2|t} - m^{(2)}(\boldsymbol{x}_t) \\ &\approx f(f(\boldsymbol{x}_t),\dots,y_{t-d+2}) + \tfrac{1}{2}\sigma^2 f_{x_1x_1} - \Big[z(z(\boldsymbol{x}_t),\dots,y_{t-p+2}) + r_2(y_t,\dots,y_{t-p_2+1})\Big] \\ &= \Big[f(f(\boldsymbol{x}_t),\dots,y_{t-d+2}) - z(z(\boldsymbol{x}_t),\dots,y_{t-p+2}) + \tfrac{1}{2}\sigma^2 f_{x_1x_1}\Big] - r_2(y_t,\dots,y_{t-p_2+1}). \end{aligned}$$

Thus the strategy leads to unbiased forecasts when $r_2(y_t,\dots,y_{t-p_2+1}) \asymp f(f(\boldsymbol{x}_t),\dots,y_{t-d+2}) - z(z(\boldsymbol{x}_t),\dots,y_{t-p+2}) + \frac{1}{2}\sigma^2 f_{x_1x_1}$. In other words, when the rectification models are sufficiently flexible to estimate the conditional mean of the residuals from the base model.

Bias in the base model is corrected with the rectification model, so the value of the base model is not in getting low bias but low variance. Consequently a relatively simple parametric base model works best. In particular, we do not need $z \asymp f$. In our applications, we use a linear autoregressive model where the order is selected using the AIC. While this model will be biased for nonlinear time series processes, it will allow most of the signal in f to be modelled, and will give relatively small variances due to being linear. Another advantage of making the base model linear is that it provides an estimate of the underlying process in areas where the data is sparse.

The rectification models must be flexible in order to handle the bias produced by the base model. In our applications we use nearest neighbour estimates (kNN).

We can get the MSE for the rectify strategy with

$$\begin{aligned}\text{MSE}_2^{\text{rectify}} &\approx \sigma^2(1+f_{x_1}^2)+\tfrac{1}{4}(\kappa-\sigma^4)f_{x_1x_1}^2\\ &\quad+\Big[[f(f(\boldsymbol{x}_t),\ldots,y_{t-d+2})-z(z(\boldsymbol{x}_t),\ldots,y_{t-p+2})+\tfrac{1}{2}\sigma^2 f_{x_1x_1}]-r_2(y_t,\ldots,y_{t-p_2+1})\Big]^2.\end{aligned}$$

When the rectify strategy is unbiased, it has an asymptotic MSE equal to the direct strategy and less than the recursive strategy:

$$\text{MSE}_2^{\text{recursive}}-\text{MSE}_2^{\text{rectify}}=\text{MSE}_2^{\text{recursive}}-\text{MSE}_2^{\text{direct}}\approx\tfrac{1}{4}\sigma^4 f_{x_1x_1}^2.$$

However, this overlooks the variance advantage of the rectify strategy. While the asymptotic variances of the rectify and direct strategies are both zero (and hence not part of these MSE results), the rectify strategy should have smaller finite-sample variance due to the unifying effects of the base model. We will demonstrate this property empirically in the next section.

The rectify strategy proposed here is similar to the method proposed by Judd & Small (2000), although they do not consider the statistical properties (such as bias and MSE) of the method. One contribution we are making in this paper is an explanation of why the method of Judd & Small (2000) works.

4 Experimental set-up

4.1 Regression methods and model selection

Each forecasting strategy requires a different number of regression tasks. In our experiments, we considered two autoregression methods: a linear model and a kNN algorithm.

The linear models are estimated using conditional least squares with the order p selected using the AIC.

The kNN model is a nonlinear and nonparametric model where the prediction for a given data point $\boldsymbol{x}_q$ is obtained by averaging the target outputs $y_{[i]}$ of the k nearest neighbors points of the given point $\boldsymbol{x}_q$ (Atkeson et al. 1997). We used a weighted kNN model by taking a weighted average rather than a simple average. The weights are a function of the Euclidean distance between the query point and the neighboring point (we used the biweight function of Atkeson et al. 1997). The key parameter k has to be selected with care as it is controlling the bias/variance tradeoff. A large k will lead to a smoother fit and therefore a lower variance (at the expense of a higher bias), and vice versa for a small k.

To select the best value of the parameter k, we adopted a holdout approach in which the first part (70%) of each time series is used for training and the second part (30%) is used for validation. More precisely, the first part is used to search for the neighbors and the second part to calculate a validation error.

4.2 The forecasting strategies

In our experiments, we will compare four forecasting strategies:

- REC: the recursive strategy using the kNN algorithm;
- DIRECT: the direct strategy using the kNN algorithm for each of the H forecast horizons;
- RECTIFY: the rectify strategy;

To show the impact of the base model, we will consider two implementations:

- LINARP-KNN: the linear model as base model and the kNN algorithm for the H rectification models;
- KNN-KNN: the kNN algorithm for both the base model and the rectification models.

5 Simulation experiments

We carry out a Monte Carlo study to investigate the performance of the rectify strategy compared to the other forecasting strategies from the perspective of bias and variance. We use simulated data with a controlled noise component to effectively measure bias and variance effects.

5.1 Data generating processes

Two different autoregressive data generating processes are considered in the simulation study.

First we use a linear AR(6) process given by

$$y_t = 1.32y_{t-1} - 0.52y_{t-2} - 0.16y_{t-3} + 0.18y_{t-4} - 0.26y_{t-5} + 0.19y_{t-6} + \varepsilon_t,$$

where $\varepsilon_t \sim \text{NID}(0,1)$. This process exhibits cyclic behaviour and was selected by fitting an AR(6) model to the famous annual sunspot series. Because it is a linear process, the variance of ε_t

simply scales the resulting series. Consequently, we set the error variance to one without loss of generality.

We also consider a nonlinear STAR process given by

$$y_t = 0.3y_{t-1} + 0.6y_{t-2} + (0.1 - 0.9y_{t-1} + 0.8y_{t-2})\left[1 + e^{(-10y_{t-1})}\right]^{-1} + \varepsilon_t$$

where $\varepsilon_t \sim \text{NID}(0, \sigma^2)$. We considered two values for the error variance of the STAR model noise $\sigma^2 = [0.05^2, 0.1^2]$.

Several other simulation studies used this STAR process for the purposes of model selection, model evaluation as well as model comparison. See Teräsvirta & Anderson (1992), Tong & Lim (1980) and Tong (1995) for some examples as well as related theoretical background and applications.

Figure 1 shows samples with $T = 200$ observations of both DGPs.

5.2 Bias and variance estimation

In expression (2), we have seen the decomposition of the MSE for a given strategy at horizon h. In this section, we will show how to estimate the different parts of the decomposition, namely the noise, the squared bias and the variance.

For a given DGP, we generate L independent time series $D^i = \{y_1, \ldots, y_T\}$ each composed of T observations using different random seeds for the error term, where $i \in \{1, \ldots, L\}$. These generated time series represent samples of the DGP.

To measure the bias and variance, we use an independent time series from the same DGP for testing purposes. We represent this independent testing time series by a set of R input/output pairs $\{(\boldsymbol{x}_j, \boldsymbol{y}_j)\}_{j=1}^{R}$ where $\boldsymbol{x}_j$ is a sequence of d consecutive observations and the vector $\boldsymbol{y}_j$ comprises the next H consecutive observations.

If we let $\hat{m}_{D^i}^{(h)}(\boldsymbol{x}_j)$ be the forecast of a given strategy for the input $\boldsymbol{x}_j$ at horizon h using dataset D^i and $\boldsymbol{y}_j(h)$ be the hth element of the vector $\boldsymbol{y}_j$, then the MSE can be calculated as

$$\begin{aligned}\text{MSE}_h &= \frac{1}{LR}\sum_{i=1}^{L}\sum_{j=1}^{R}\left(\boldsymbol{y}_j(h) - \hat{m}_{D^i}^{(h)}(\boldsymbol{x}_j)\right)^2 \\ &= \text{Noise}_h + \text{Bias}_h^2 + \text{Variance}_h.\end{aligned}$$

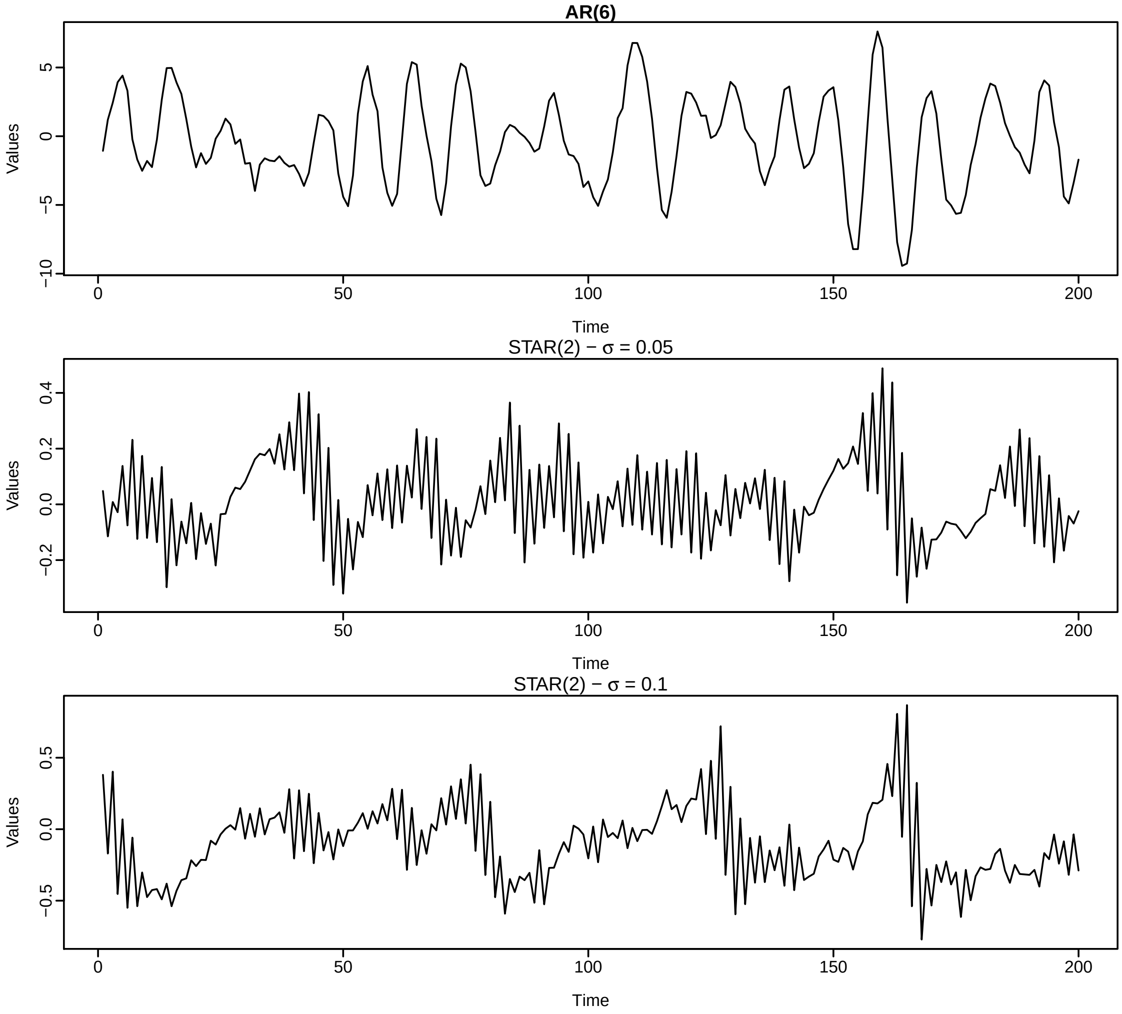

Figure 1

The three components can be estimated as follows

$$\text{Noise}_h = \frac{1}{R}\sum_{j=1}^{R}\Big(\boldsymbol{y}_j(h) - \mathbb{E}[\boldsymbol{y}_j(h) \mid \boldsymbol{x}_j]\Big)^2$$

$$\text{Bias}_h^2 = \frac{1}{R}\sum_{j=1}^{R}\Big(\mathbb{E}[\boldsymbol{y}_j(h) \mid \boldsymbol{x}_j] - \bar{m}(\boldsymbol{x}_j)\Big)^2$$

$$\text{Variance}_h = \frac{1}{LR}\sum_{i=1}^{L}\sum_{j=1}^{R}\Big(\hat{m}_{D^i}^{(h)}(\boldsymbol{x}_j) - \bar{m}(\boldsymbol{x}_j)\Big)^2$$

where $\bar{m}(\boldsymbol{x}_j) = \frac{1}{L}\sum_{i=1}^{L}\hat{m}_{D^i}^{(h)}(\boldsymbol{x}_j)$ is an estimate of $m^{(h)}(\boldsymbol{x}_j)$.

5.3 Simulation details

The first three hundred simulated values are discarded for each simulated series to stabilize the time series, as suggested by Law & Kelton (2000).

To show the importance of the size of the time series T for each strategy, we will compare different sizes, namely $T \in \{50, 100, 200, 400\}$.

We considered a maximum forecasting horizon $H = 10$ for the AR(6) DGP and $H = 20$ for the STAR(2) DGP.

For the bias and variance estimation, we generate $L = 500$ sample time series from the DGP. A large time series is independently generated and $R = 2000$ input/output pairs are extracted from it for testing purposes as described in Section 5.2.

For the AR(6) DGP, the conditional mean $\mathbb{E}[\boldsymbol{y}_j(h) \mid \boldsymbol{x}_j]$ at input $\boldsymbol{x}_j$ and horizons $h \in \{1, \ldots, H\}$ has been calculated analytically. For the STAR(2) DGP, we used simulations to compute $\frac{1}{S}\sum_{s=1}^{S} y_s^{(h)}(\boldsymbol{x}_j)$ where $y_s^{(h)}(\boldsymbol{x}_j)$ is one realization of the future of the input $\boldsymbol{x}_j$. We chose a large value for the parameter S.

The range of values for the parameters $p, p_1, \ldots, p_H$ (see equations (4), (7) and (9)) required by all the regression tasks is $\{2, 3, 4, 5, 6, 7\}$ and the set $\{2, 3, 4, 5\}$ for the base model of the rectify strategy. For the STAR(2) DGP, we used the set $\{2, 3, 4, 5\}$.

5.4 Results and discussion

Figure 2 gives for the AR(6) DGP and different values of T, the MSE together with the corresponding squared bias and variance. The same information is given in Figures 3 and 4 for the STAR(2) DGP with $\sigma = 0.05$ and $\sigma = 0.1$, respectively.

For both DGPs, we can see in Figures 2, 3 and 4 that the recursive strategy has virtually the same performance as the direct strategy in the first few horizons while it performs poorly for longer horizons. For the AR(6) DGP, this is mainly due to the nonlinear model which increases the forecast variance. For the STAR(2) DGP, both the nonlinear model and the nonlinear time series make the forecasts of the recursive strategy worse. Indeed, as we have seen in expression (6), if the time series is nonlinear and $f_{x_1 x_1} > 0$, there is a positive bias. Also, if the model is nonlinear then an error made at one forecast horizon will be propagated to the subsequent forecast horizons, thus increasing the bias of the recursive strategy as the horizon increases.

The same figures show also that the direct forecasts converge to the mean forecasts as the size of the time series T increases, which is not the case for the recursive strategy. This is consistent with the theoretical analysis of Section 2 where we have seen that the recursive strategy is biased when the time series is noisy and f has a non-zero second derivative.

Concerning the two implementations of the rectify strategy, we can see that KNN-KNN has significantly reduced the bias of the recursive strategy for both DGPs. However, this was at a price of an increase in variance which consequently made the forecasts worse. The increase in variance is particularly noticeable with small-sample time series and vanishes with large samples. In consequence, using a nonlinear base model seems not to be a good idea when implementing the rectify strategy.

If we look at the second implementation of the rectify strategy (LINARP-KNN in blue), we can see that it performs better compared to KNN-KNN for both DGPs. Instead of using a nonlinear base model, LINARP-KNN uses a linear AR(p) model which reduces the variance of the final forecasts.

Figure 2 shows that LINARP-KNN has improved the forecasting performance over both the recursive and the direct strategies. The bias has been reduced over the forecasting horizons for both strategies. Also, LINARP-KNN has significantly reduced the variance for the recursive strategy. Compared to the direct strategy, the better performance is particularly noticeable at shorter horizons and we get similar performance at longer horizons.

Figures 3 and 4 gives the results for the STAR(2) DGP for two different levels of noise. Comparing these two figures, we can see that the gain with LINARP-KNN is greater with a higher level of noise. This can be explained by the fact that the recursive strategy will be more biased and also because the linear base model performs well in a high noise setting compared to nonlinear models.

We can also notice in Figure 4 that LINARP-KNN has a smaller MSE compared to other strategies at shorter horizons and converges to the forecasts of the direct strategy as the horizon increases. In addition, LINARP-KNN is better with small-sample time series and gets closer to the direct strategy as T increases.

The different results suggest that we can take advantage of the linear model in areas where the data is sparse and benefit from the small variance due the linearity.

To illustrate the MSE decomposition for the different strategies, Figure 5 shows for $T = 100$ and the AR(6) DGP, the noise, the squared bias and the variance stacked in each panel. The same information is given in Figures 6 and 7 for the STAR(2) DGP with $\sigma = 0.05$ and $\sigma = 0.1$, respectively.

Figure 5, 6 and 7 show that the main factors affecting the MSE for all horizons are the noise and the variance. For both DGPs, we can see that the recursive strategy has both higher bias (in grey) and higher variance (in yellow) at longer horizons compared to the direct strategy.

For the AR DGP, `KNN-KNN` has decreased the bias and increased the variance of the recursive strategy. However, for `LINARP-KNN`, we can clearly see the decrease in bias as well as in variance.

For the STAR DGP, we can see a higher decrease in terms of both bias and variance for `LINARP-KNN` in Figure 7 compared to Figure 6.

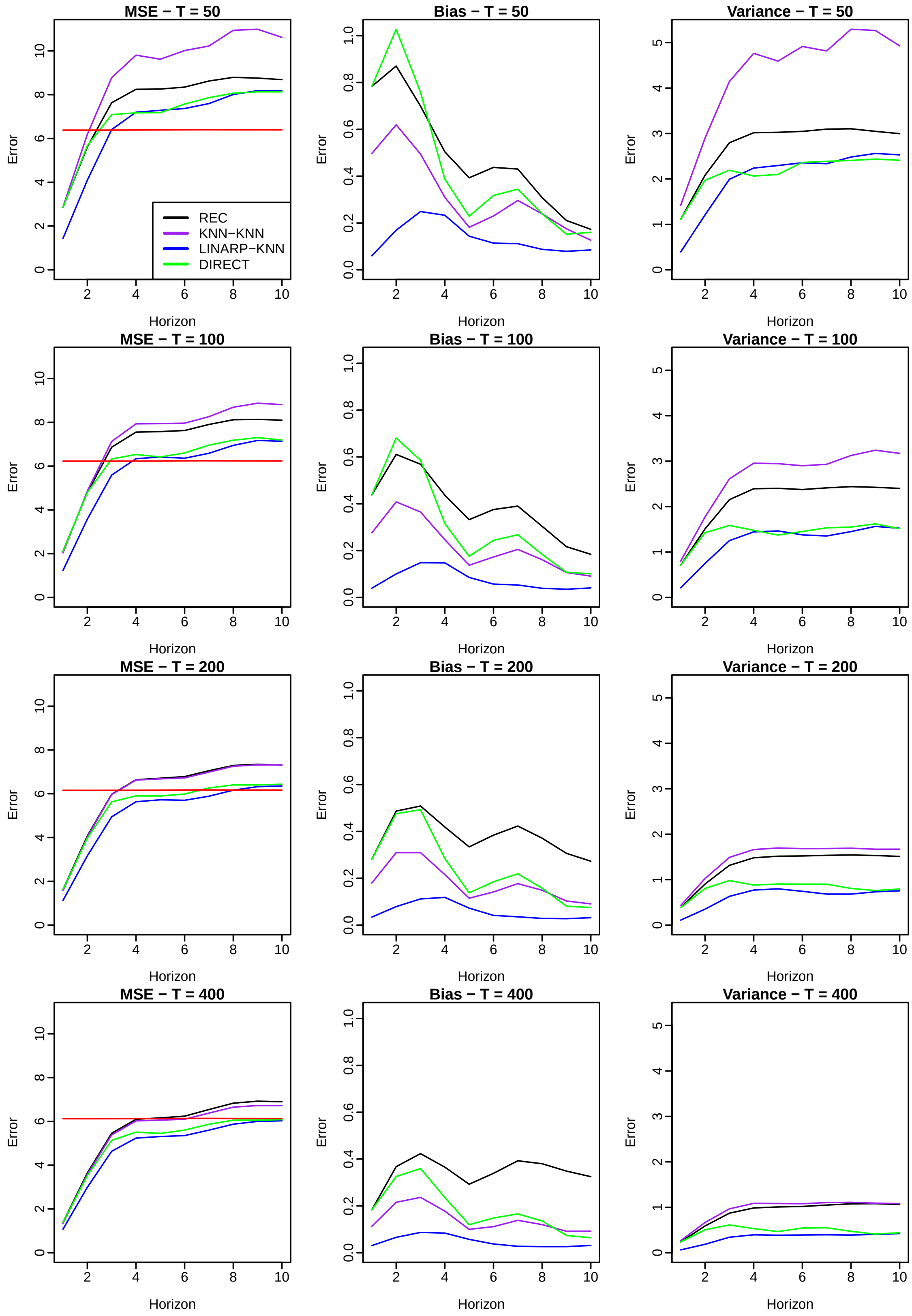

Figure 2: *AR(6) DGP and* $T = [50, 100, 200, 400]$. *MSE with corresponding squared bias and variance.*

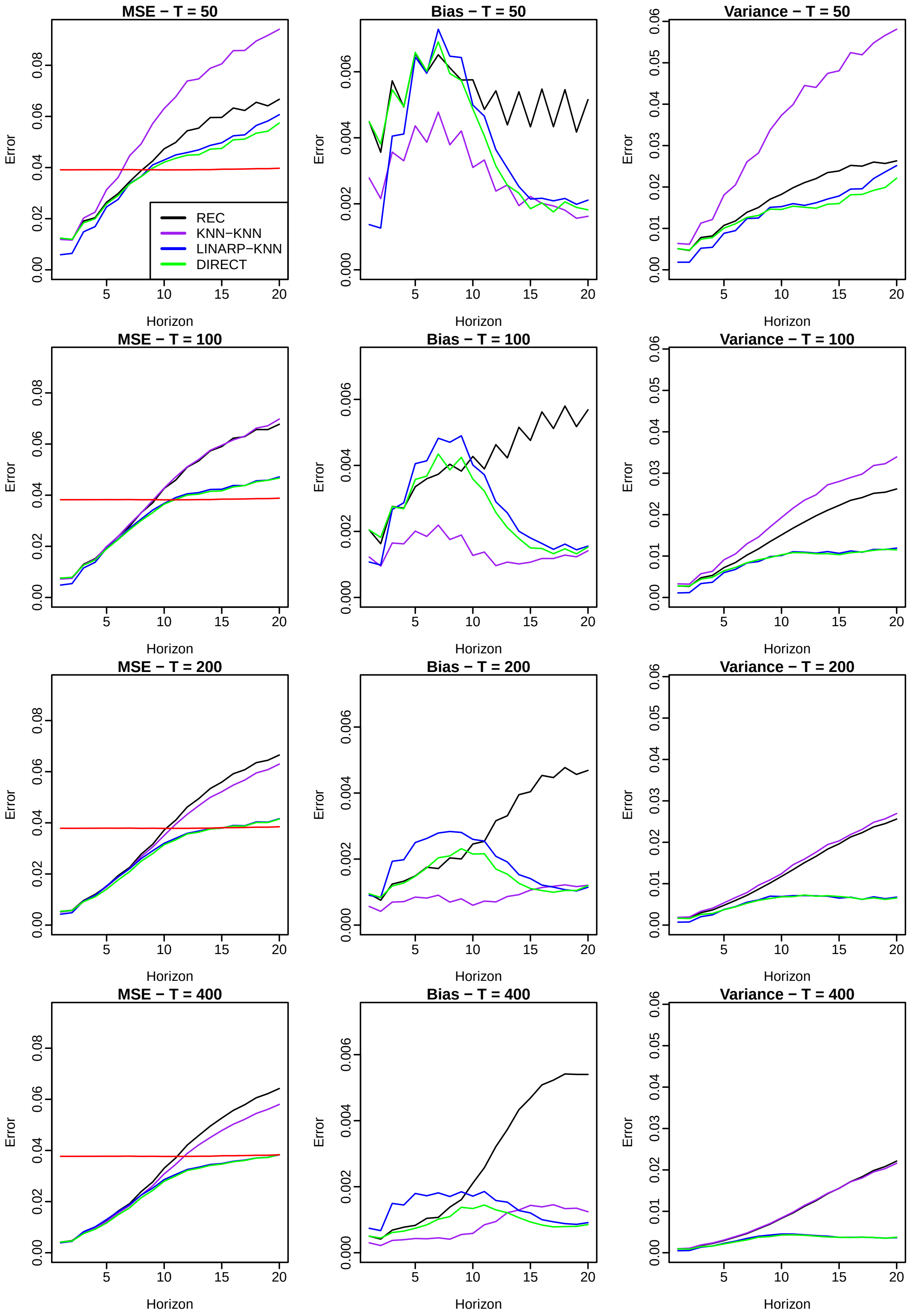

Figure 3: *STAR(2) DGP, σ = 0.05 and* $T = [50, 100, 200, 400]$. *MSE with corresponding squared bias and variance.*

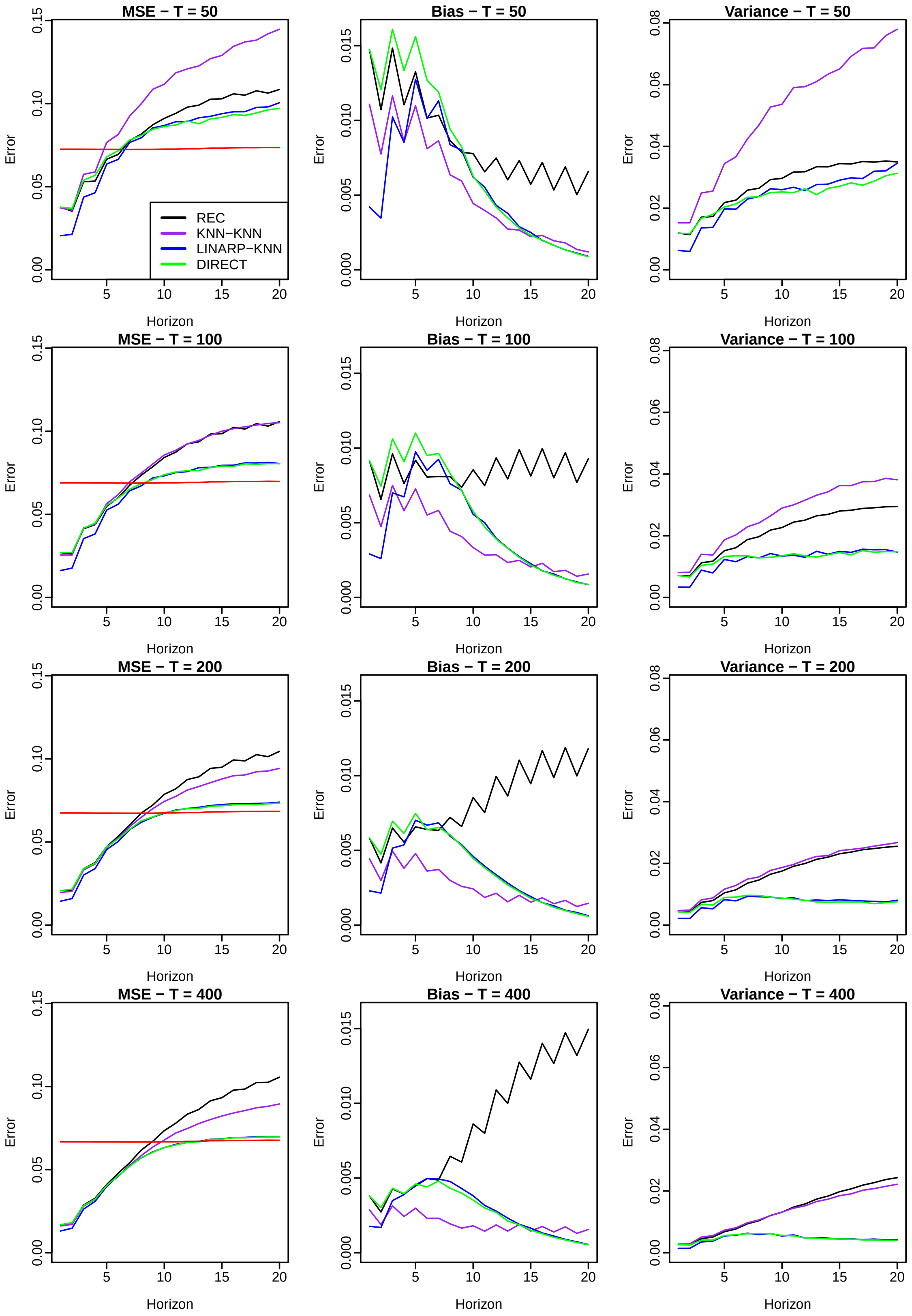

Figure 4: *STAR(2) DGP, $\sigma = 0.1$ and $T = [50, 100, 200, 400]$. MSE with corresponding squared bias and variance.*

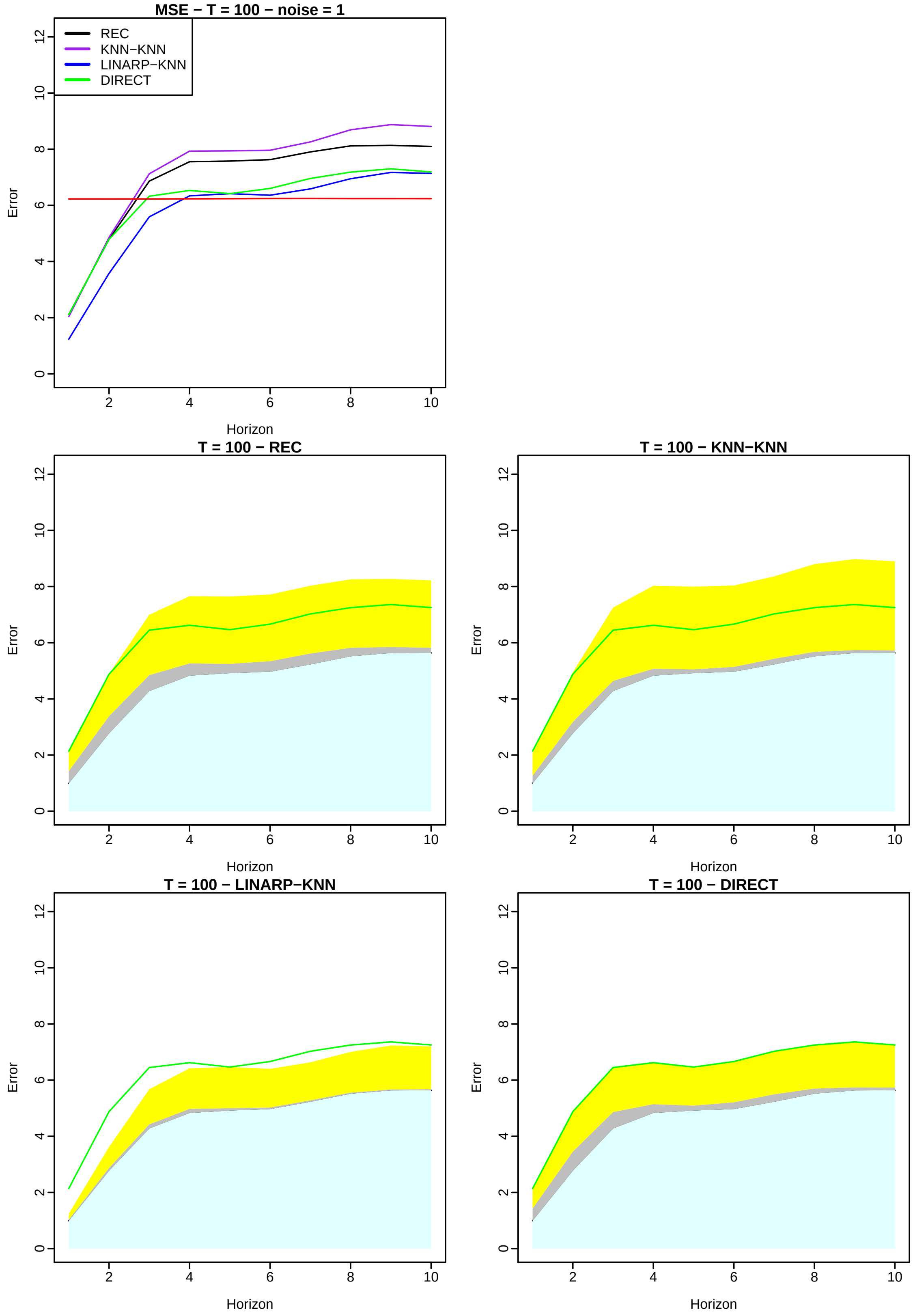

Figure 5: *AR(6) DGP. MSE of the different forecasting strategies (top left). Decomposed MSE in noise (cyan), squared bias (grey) and variance (yellow).*

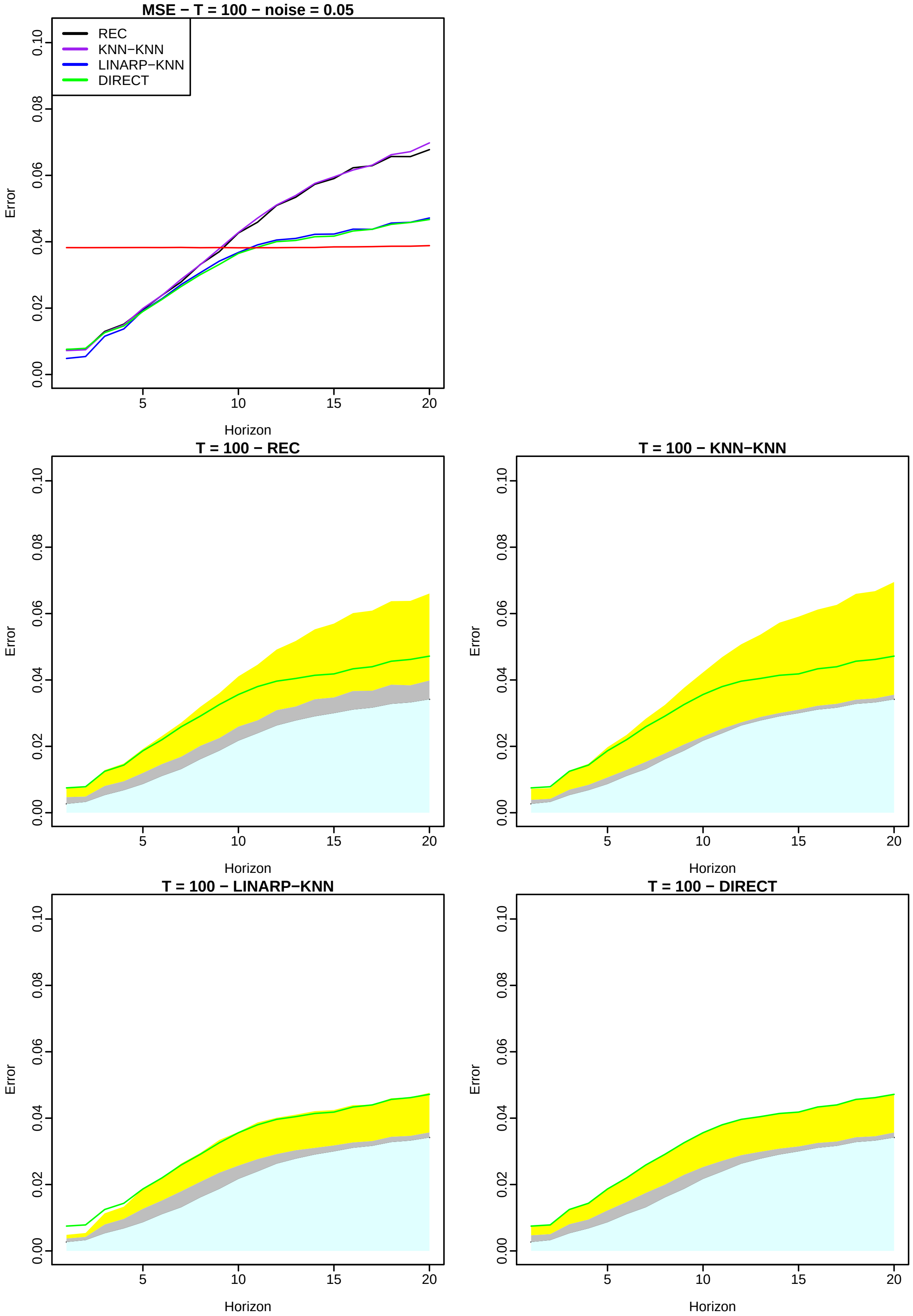

Figure 6: *STAR(2) DGP and $\sigma = 0.05$. MSE of the different forecasting strategies (top left). Decomposed MSE in noise (cyan), squared bias (grey) and variance (yellow).*

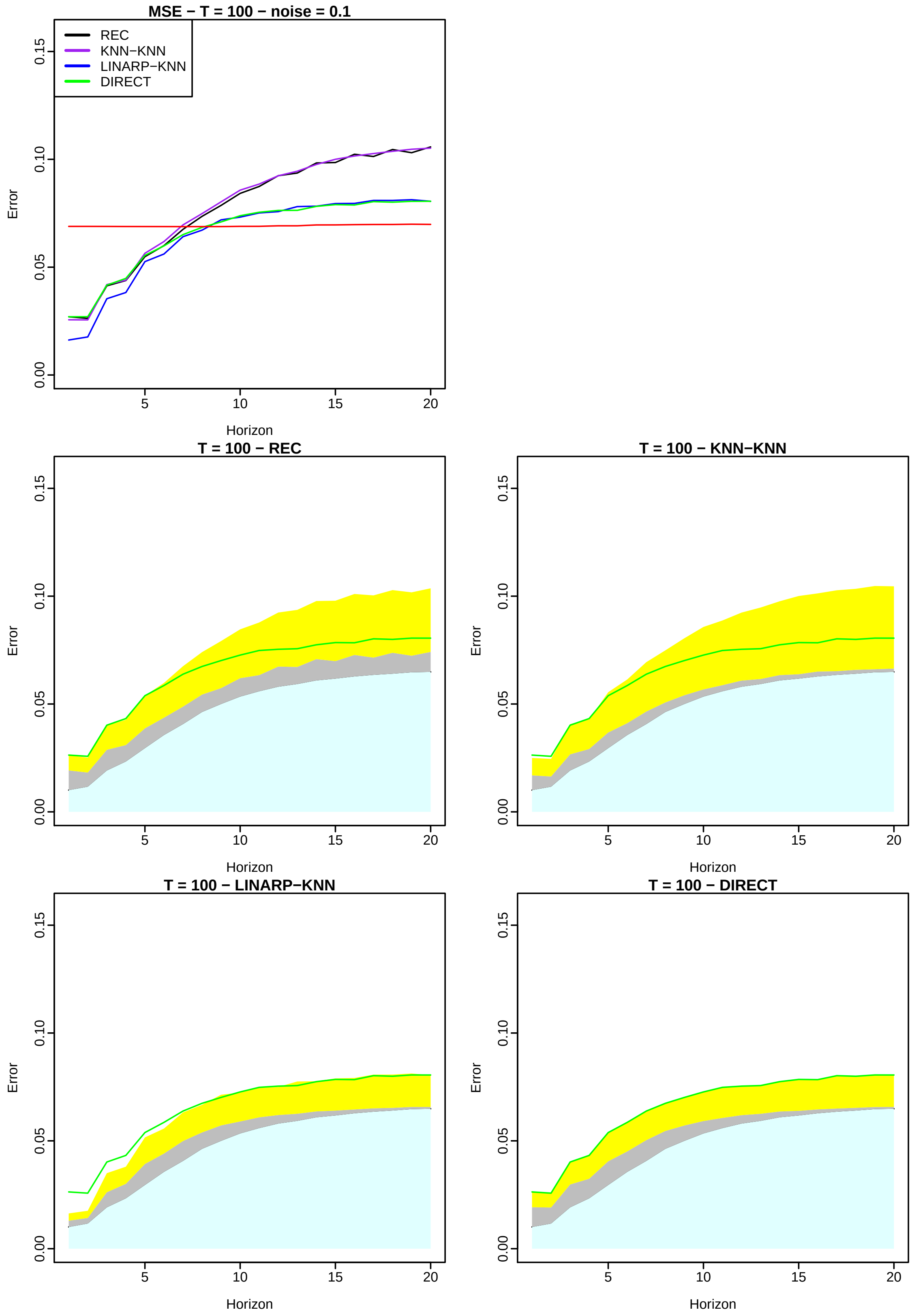

Figure 7: *STAR(2) DGP and $\sigma = 0.1$. MSE of the different forecasting strategies (top left). Decomposed MSE in noise (cyan), squared bias (grey) and variance (yellow).*

6 Real data applications

To shed some light on the performance of the rectify strategy with real-world time series, we carried out some experiments using time series from two forecasting competitions, namely the M3 and the NN5 competitions.

6.1 Forecasting competition data

The M3 competition dataset consists of 3003 monthly, quarterly, and annual time series. The competition was organized by the *International Journal of Forecasting* (Makridakis & Hibon 2000), and has attracted a lot of attention. The time series of the M3 competition have a variety of features. Some have a seasonal component, some possess a trend, and some are just fluctuating around some level.

We have considered all the monthly time series in the M3 data that have more than 110 data points. The number of time series considered was $M = 800$ with a range of lengths between $T = 111$ and $T = 126$. For these monthly time series, the competition required forecasts for the next $H = 18$ months, using the given historical points. Figure 8 shows four time series from the set of 800 time series.

The NN5 competition dataset **?** comprises $M = 111$ daily time series each containing $T = 735$ observations. Each of these time series represents roughly two years of daily cash money withdrawal amounts at ATM machines at one of several cities in the UK. The competition was organized in order to compared and evaluate the performance of computational intelligence methods. For all these time series, the competition required forecasts of the next $H = 56$ days, using the given historical points. Figure 9 shows four time series from the NN5 data set.

6.2 Experiment details

The NN5 dataset includes some zero values that indicate no money withdrawal occurred and missing observations for which no value was recorded. We replaced these two types of gaps using the method proposed in Wichard (2010): the missing or zero observation y_t is replaced by the median of the set $\{y_{t-365}, y_{t-7}, y_{t+7}, y_{t+365}\}$ using only non-zero and non-missing values.

For both competitions, we deseasonalized the time series using the STL (Seasonal-Trend decomposition based on Loess smoothing) (Cleveland et al. 1990). Of course, the seasonality has been restored after forecasting. For the parameter controlling the loess window for seasonal

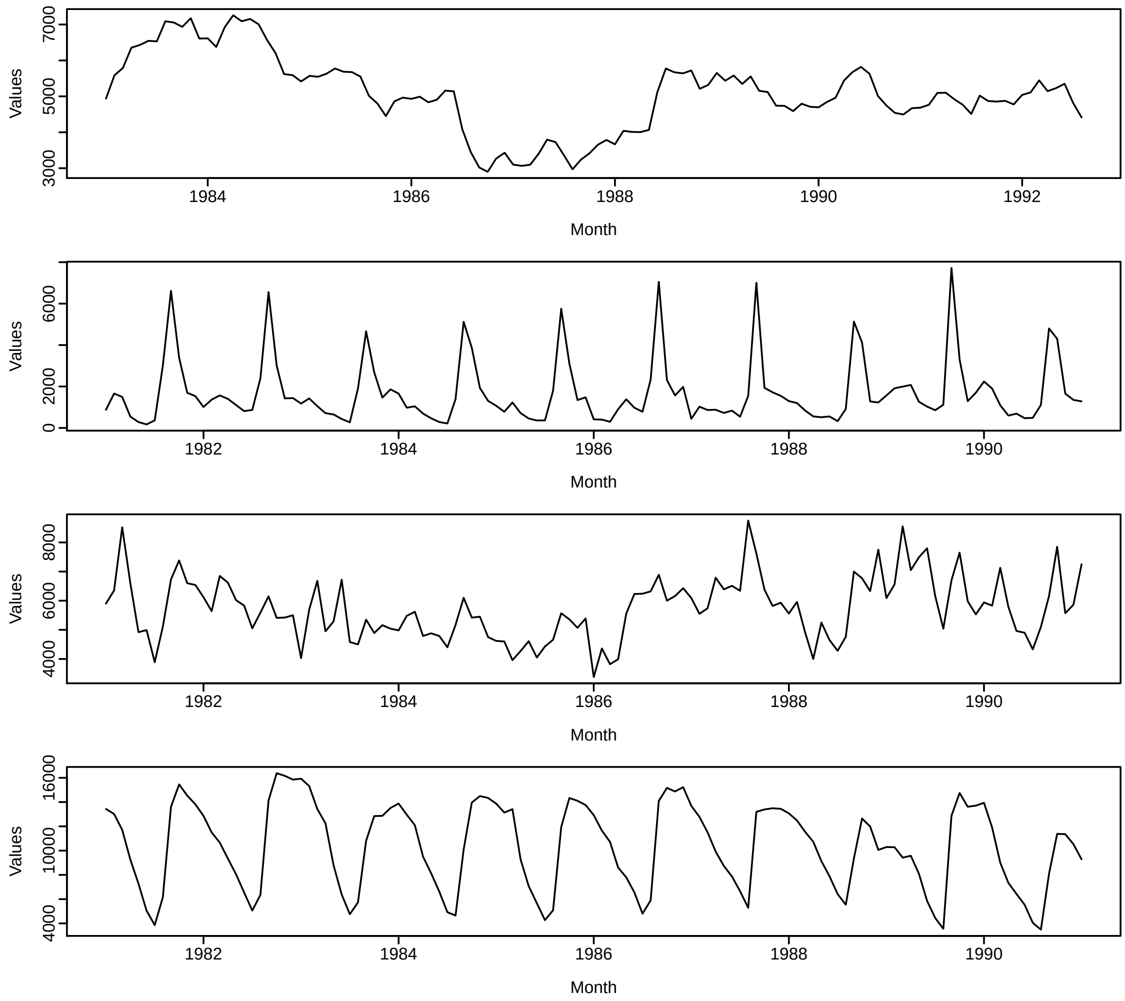

Figure 8: *Four time series from the M3 forecasting competition.*

extraction, we used the value $s = 50$ for the M3 competition and $s =$ periodic for the NN5 competition.

For the M3 competition, the value of each parameters $p, p_1, \ldots, p_H$ were selected from the set $\{2 : 10\}$ for all strategies and from the set $\{2 : 5\}$ for the base model of the rectify strategy. For the NN5 competition, we used the set $\{2 : 14\}$ for all strategies and the set $\{2 : 7\}$ for the base model of the rectify strategy.

We considered two forecast accuracy measures. The first was the symmetric mean absolute percentage error (SMAPE). Let

$$\mathrm{SMAPE}_h^m = \frac{2|\hat{y}_{t+h}^m - y_{t+h}^m|}{\hat{y}_t^m + y_t^m} * 100$$

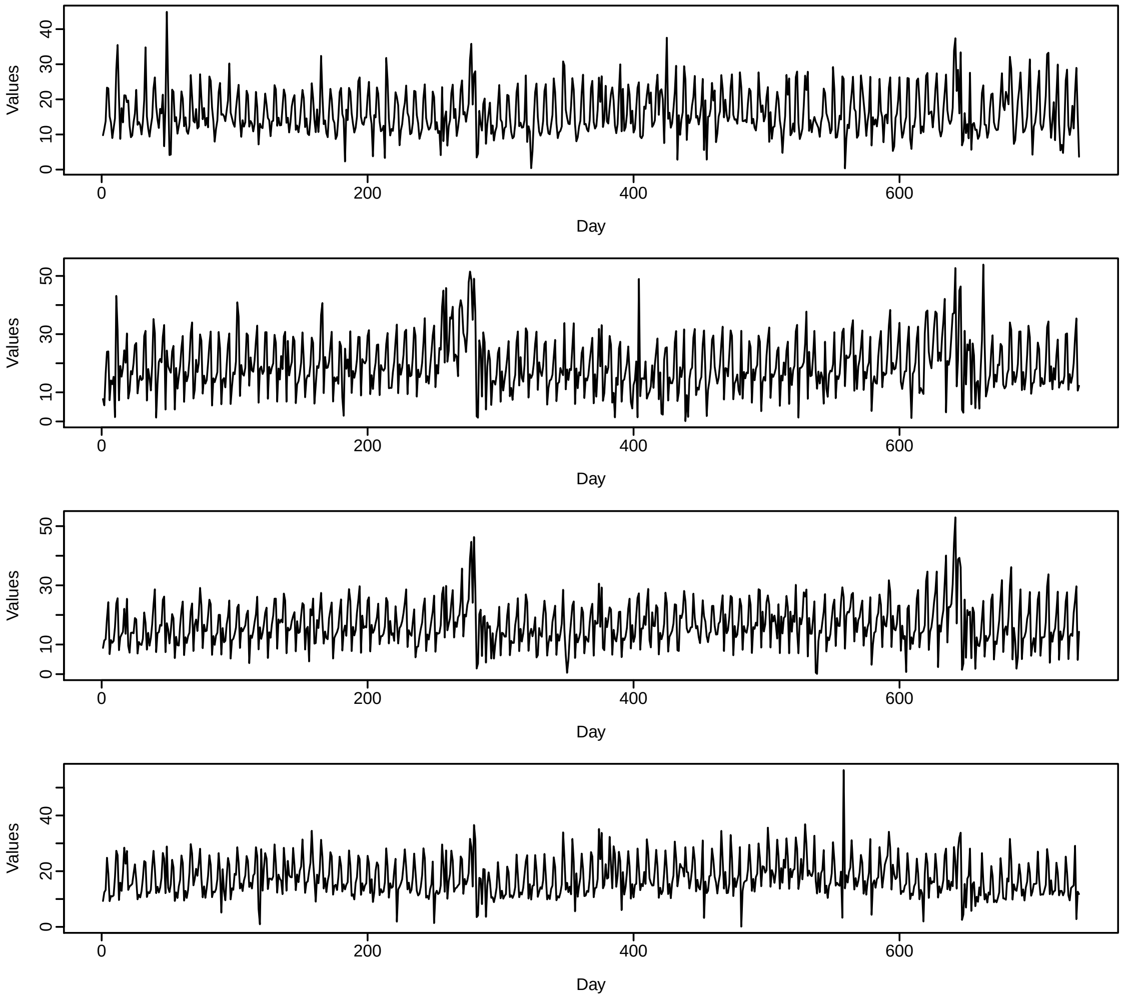

Figure 9: *Four time series from the NN5 forecasting competition.*

for the mth time series at horizon h where $m \in \{1,\ldots,M\}$ and $h \in \{1,\ldots,H\}$. Then SMAPE is the average of SMAPE_h^m across the M time series. Hyndman & Koehler (2006) discussed some problems with this measure, but as it was used by Makridakis & Hibon (2000), we use it to enable comparisons with the M3 competition.

The second accuracy measure was the mean absolute scaled error (MASE) introduced by Hyndman & Koehler (2006). Let

$$\text{MASE}_h^m = \frac{|\hat{y}_{t+h}^m - y_{t+h}^m|}{\frac{1}{T-s}\sum_{t=s+1}^{T}|y_t^m - y_{t-s}^m|}$$

where s is the seasonal lag. We used $s = 7$ for the NN5 time series and $s = 12$ for the M3 time series. Then MASE is the average of MASE_h^m across the M time series.

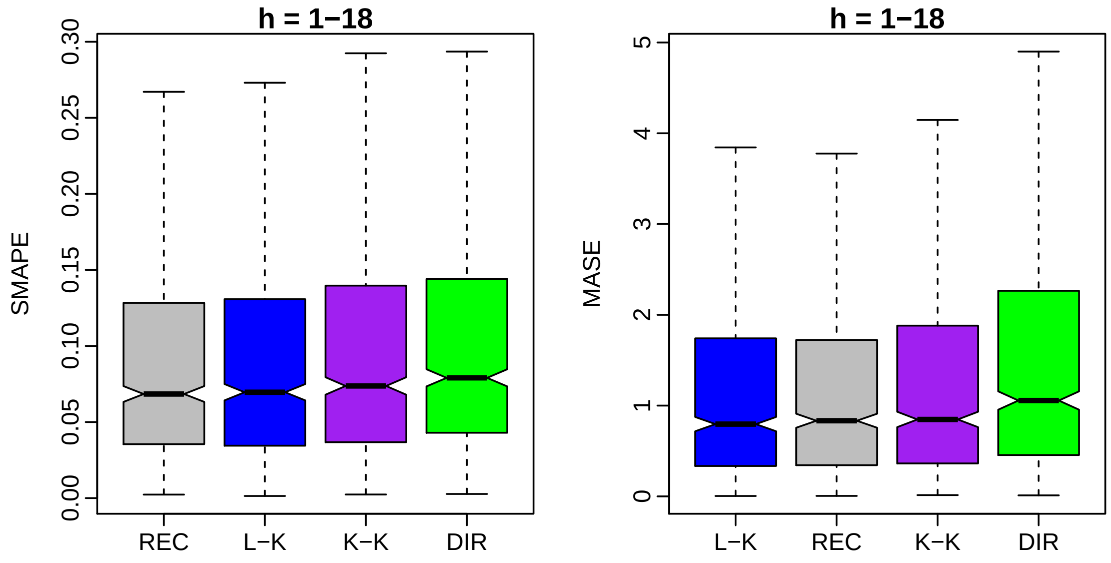

Figure 10: *Boxplots of the SMAPE (left) and MASE (right) measures averaged over the $H = 18$ consecutive horizons and obtained across the 800 time series for the different strategies.*

6.3 Results and discussion

Figures 10 and 11 present a graphical depiction of the performance of each strategy on both the M3 and the NN5 Competitions. Each figure has two plots for both SMAPE and MASE measures and each plot gives notched boxplots ranked according to the median and summarizing the distribution of the SMAPE/MASE measures averaged over the forecasting horizon H for each strategy. Outliers were omitted to better facilitate the graphical comparison between the various forecasting strategies.

For both the M3 and the NN5 competitions, Figures 10 and 11 show that the rectify strategy is almost always better or at least equivalent to the best of both the recursive and the direct strategy.

In fact, for the M3 competition, we can see on Figure 10 that the recursive strategy is significantly better than the direct strategy both in term of SMAPE and MASE measures. We can also see that the rectify strategy is better or equivalent to the recursive strategy.

For the NN5 competition, the direct strategy performs better than the recursive strategy and as before, the rectify strategy is better or equivalent to the direct strategy.

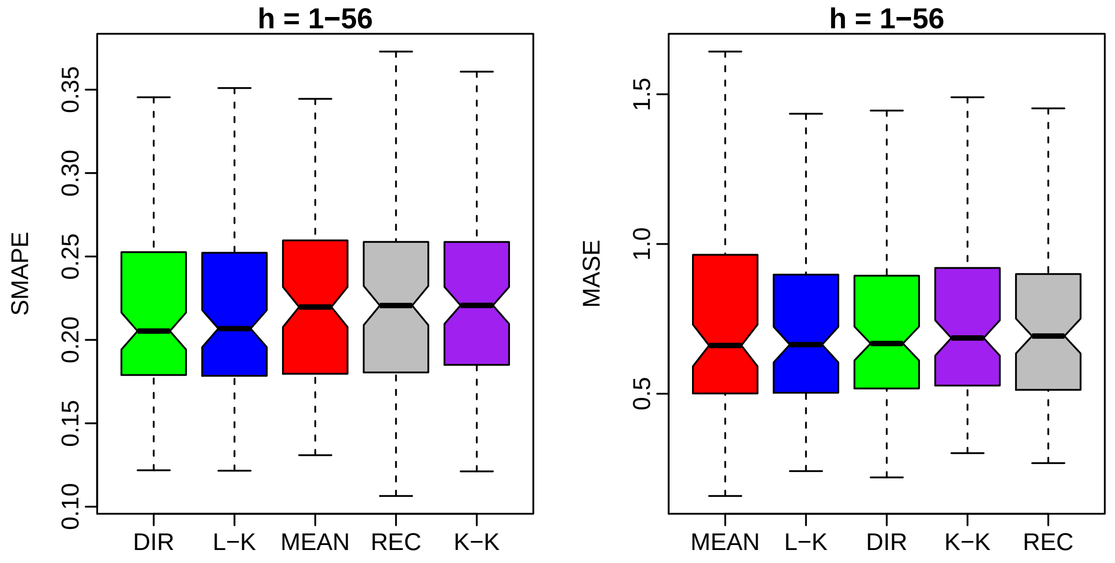

Figure 11: *Boxplots of the SMAPE (left) and MASE (right) measures averaged over the $H = 56$ consecutive horizons and obtained across the* 111 *time series for the different strategies.*

This suggests that using the rectify strategy in all situations is a reasonable approach, and avoids making a choice between the recursive and the direct strategies which can be a difficult task in real-world applications.

To allow a more detailed comparison between the different strategies, Figures 12–15 present the same result as in Figures 10 and 11 but at a more disaggregated level. Figures 12 and 13 give the SMAPE/MASE measures for each horizon h while Figures 14 and 15 averaged it over 7 consecutive horizons instead of $H = 56$ as in Figure 11.

If we compare the performance of the different forecasting strategies in term of lowest median, upper and lower quartiles of the distribution of errors, the following conclusions can be drawn.

For the M3 forecasting competition, Figure 12 and 13 suggest that direct is the least accurate strategy consistently over the entire horizon both in term of SMAPE and MASE. This might be explained by the size of the time series (between 111 and 126) and the large forecasting horizon $H = 18$ which can reduce the dataset by 15% for high horizons.

The LINARP-KNN strategy performs particularly well in short horizons and competes closely with KNN-KNN and REC according to MASE and SMAPE respectively.

For the NN5 forecasting competition, Figures 14 and 15 suggest that the direct strategy provides better forecasts compared to the recursive strategy, probably because of the greater nonlinearity present in the NN5 data compared to the M3 data.

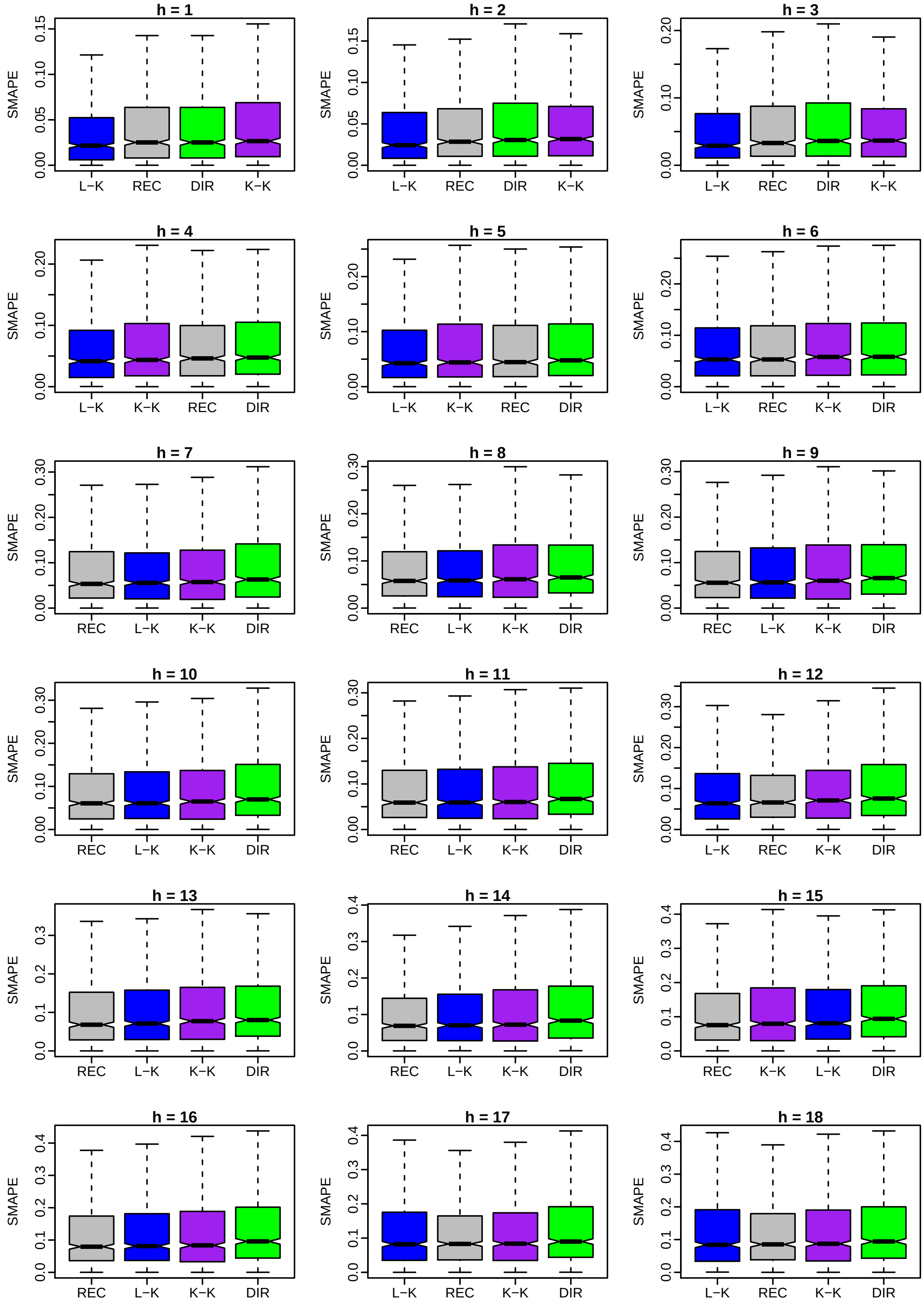

Figure 12: *Boxplots of the SMAPE measures for horizon $h = 1$ to $h = 18$ obtained across the 800 time series for the different strategies.*

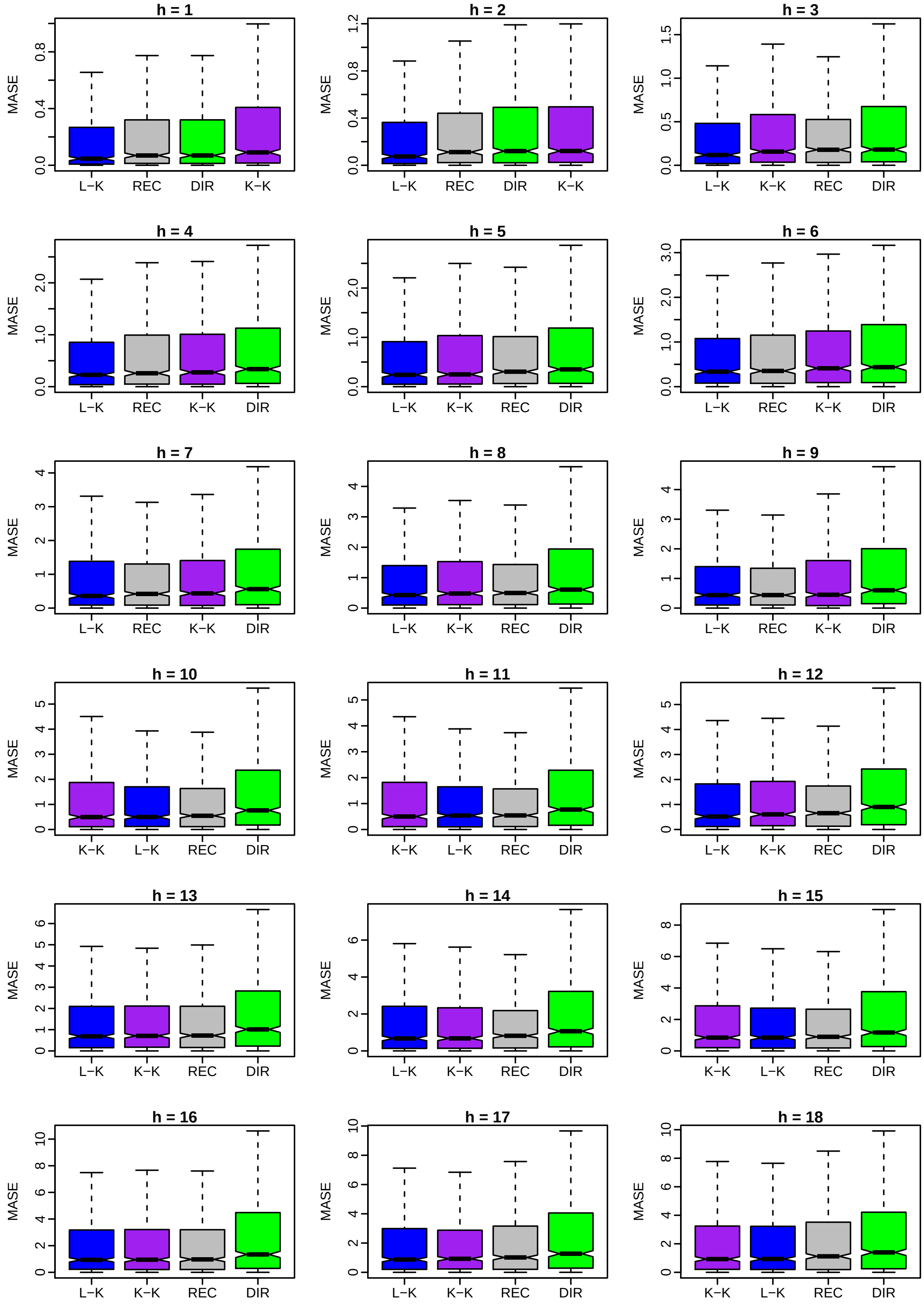

Figure 13: *Boxplots of the MASE measures for horizon $h = 1$ to $h = 18$ obtained across the 800 time series for the different strategies.*

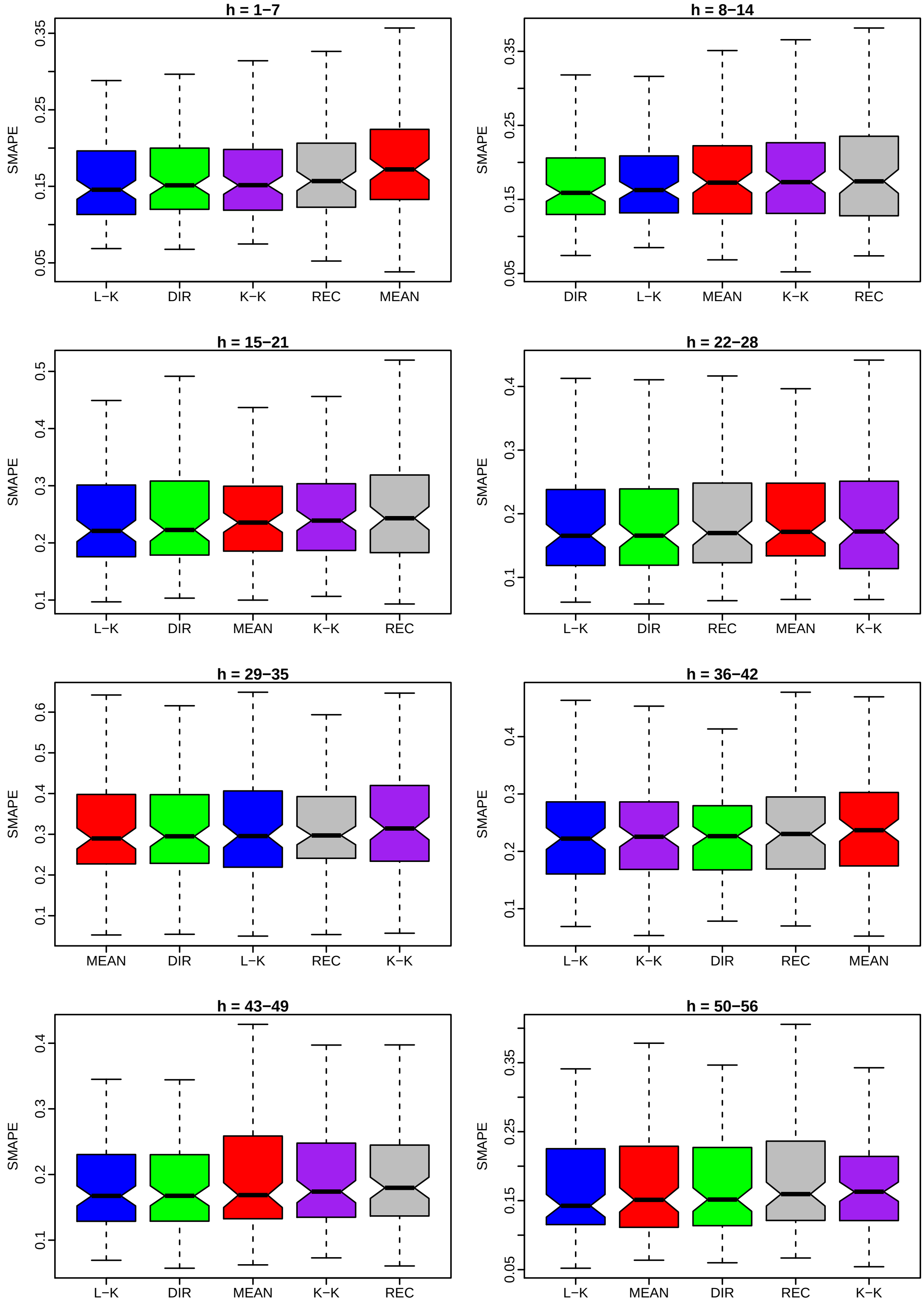

Figure 14: *Boxplots of the SMAPE measures averaged over 7 consecutive horizons and obtained across the 111 time series for the different strategies.*

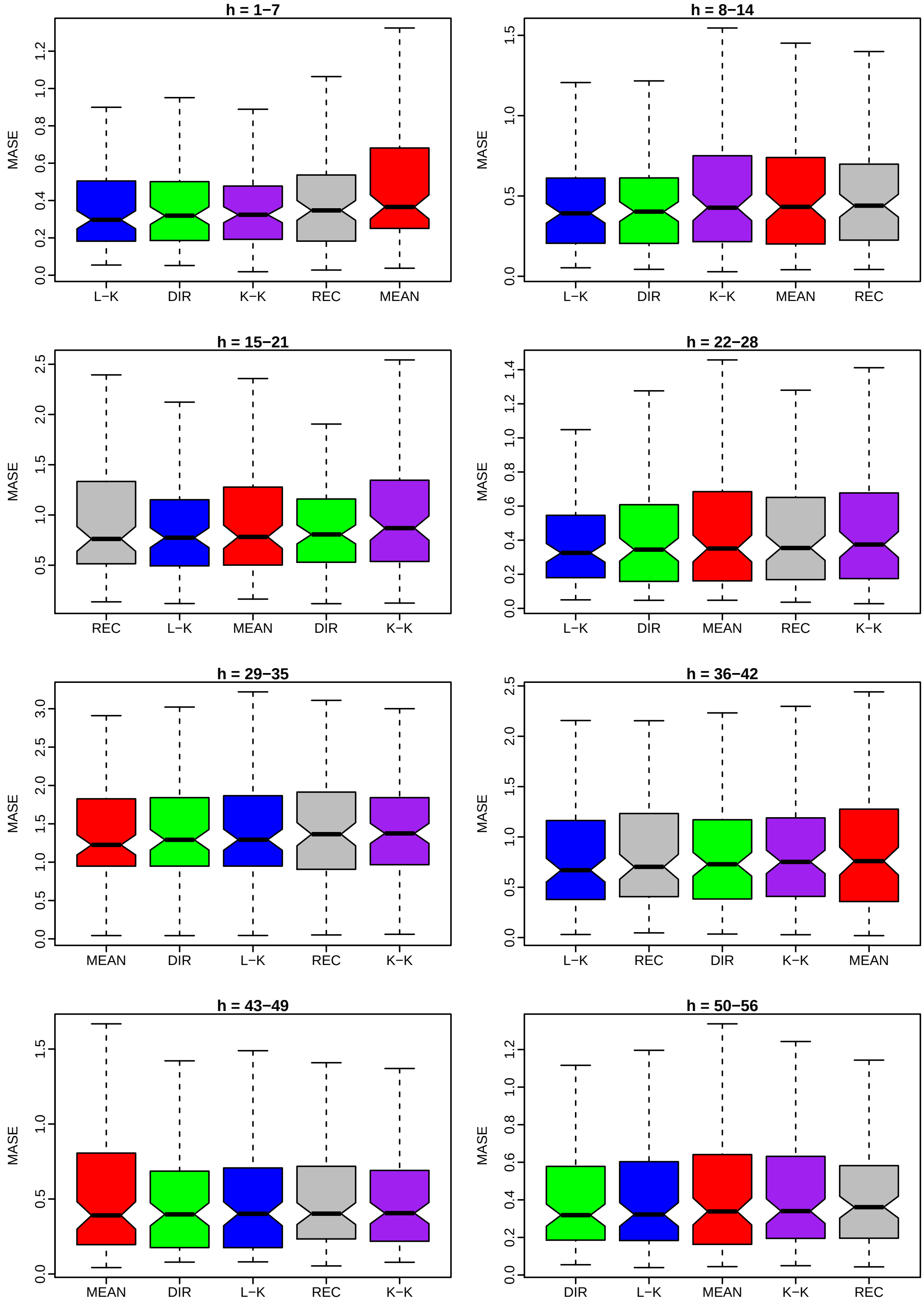

Figure 15: *Boxplots of the MASE measures averaged over 7 consecutive horizons and obtained across the 111 time series for the different strategies.*

7 Conclusion

The recursive strategy can produce biased forecasts and the direct strategy can have high variance forecasts with short time series or long forecasting horizons. The rectify strategy has been proposed to take advantage of the strengths of both the recursive and direct strategies.

Using simulations, we have found with nonlinear time series that the rectify strategy reduces the bias of the recursive strategy as well as the variance if a linear base model is used. A higher level of noise and a shorter time series make the performance of the rectify strategy even better, relative to other strategies. With linear time series, we also found that the rectify strategy has performed particularly well by decreasing both the bias and the variance.

Using real time series, we found that the rectify strategy is always better than, or at least has comparable performance to, the best of the recursive and the direct strategy. This finding is interesting as it avoids the difficult task of choosing between the recursive and the direct strategy and at the same time it shows that the rectify strategy is better or competitive with both the recursive and the direct strategy.

All these findings make the rectify strategy very attractive for multi-step forecasting tasks.

References

Atiya, A. F., El-shoura, S. M., Shaheen, S. I. & El-sherif, M. S. (1999), 'A comparison between neural-network forecasting techniques–case study: river flow forecasting.', *IEEE Transactions on Neural Networks* **10**(2), 402–409.

Atkeson, C. G., Moore, A. W. & Schaal, S. (1997), 'Locally weighted learning', *Artificial Intelligence Review* **11**(1-5), 11–73.

Chen, R., Yang, L. & Hafner, C. (2004), 'Nonparametric multistep-ahead prediction in time series analysis', *Journal of the Royal Statistical Society, Series B* **66**(3), 669–686.

Chevillon, G. (2007), 'Direct multi-step estimation and forecasting', *Journal of Economic Surveys* **21**(4), 746–785.

Chevillon, G. & Hendry, D. (2005), 'Non-parametric direct multi-step estimation for forecasting economic processes', *International Journal of Forecasting* **21**(2), 201–218.

Cleveland, R. (1990), 'STL : A seasonal-trend decomposition procedure based on loess', *Journal of Official ...* .

Fan, J. & Yao, Q. (2003), *Nonlinear time series: nonparametric and parametric methods*, Springer, New York.

Franses, P. H. & Legerstee, R. (2009), 'A unifying view on multi-step forecasting using an autoregression', *Journal of Economic Surveys* **24**(3), 389–401.

Hastie, T. J., Tibshirani, R. & Friedman, J. H. (2008), *The elements of statistical learning*, 2nd edn, Springer-Verlag, New York.

Ing, C.-K. (2003), 'Multistep Prediction in Autoregressive Processes', *Econometric Theory* **19**(02), 254–279.

Judd, K. & Small, M. (2000), 'Towards long-term prediction', *Physica D* **136**, 31–44.

Law, A. M. & Kelton, W. D. (2000), *Simulation Modelling and Analysis*, 3rd edn, McGraw-Hill.

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

McNames, J. (1998), A nearest trajectory strategy for time series prediction, *in* 'Proceedings of the International Workshop on Advanced Black-Box Techniques for Nonlinear Modeling', Citeseer, pp. 112–128.

Teräsvirta, T. & Anderson, H. M. (1992), 'Characterizing nonlinearties in business cycles using smooth transition autoregressive models', *Journal of Applied Econometrics* **7**, 119–136.

Tong, H. (1995), 'A personal overview of non-linear time series analysis from a chaos perspective', *Scandinavian Journal of Statistics* **22**, 399–445.

Tong, H. & Lim, K. S. (1980), 'Threshold autoregression, limit cycles and cyclical data', *Journal of the Royal Statistical Society, Series B* **42**(3), 245–292.

Werbos, P. (1990), 'Backpropagation through time: What it does and how to do it', *Proceedings of the IEEE* **78**(10), 1550–1560.

Wichard, J. D. (2010), 'Forecasting the NN5 time series with hybrid models', *International Journal of Forecasting* .

Williams, R. J. & Zipser, D. (1989), 'A Learning Algorithm for Continually Running Fully Recurrent Neural Networks', *Neural computation* **1**(2), 270–280.