Robustness of Forecast Combination in Unstable Environment: A Monte Carlo Study of Advanced Algorithms

Yongchen Zhao[1]

Department of Economics, Towson University, Towson, MD, 21252, USA

Abstract

Based on a set of carefully designed Monte Carlo exercises, this paper documents the behavior and performance of several newly developed advanced forecast combination algorithms in unstable environments, where performance of candidate forecasts are cross-sectionally heterogeneous and dynamically evolving over time. Results from these exercises provide guidelines regarding the selection of forecast combination method based on the nature, frequency, and magnitude of instabilities in forecasts as well as the target variable. Following these guidelines, a simple forecast combination procedure is proposed and demonstrated through a real-time forecast combination exercise using the U.S. Survey of Professional Forecasters, where combined forecasts are shown to have superior performance that is not only statistically significant but also of practical importance.

JEL Classification: C53, C22, C15

Keywords: Forecast combination, Exponential re-weighting, Shrinkage, Estimation error, Performance stability, Real-Time Data

1. Introduction

Combining forecasts has long been a standard practice for researchers and policymakers. A large number of studies have clearly demonstrated that combining forecasts results in improved forecast accuracy. Many new and exciting forecast combination algorithms have been proposed recently. However, even today, practitioners often resort to the most basic method of combination, simple averaging, despite the availability of sophisticated forecast combination algorithms. One of the main reasons behind this preference is, arguably, the "forecast combination puzzle"[2]. While explanations of the puzzle have since been offered

Email address: yzhao@towson.edu (Yongchen Zhao)

[1]This research is generously supported by the SAS/IIF Grant to Promote Research on Forecasting. The author thanks Kajal Lahiri, Herman Stekler, Tara Sinclair, as well as seminar participants at George Washington University and conference participants at the 35th International Symposium on Forecasting for their helpful comments. The author is responsible for any and all errors and omissions in the paper.

[2]Clemen (1989) reviewed more than 200 studies on forecast combination, and observed that simple averaging is an exceedingly robust procedure that often outperforms more complicated weighting schemes. This notion of "forecast combination puzzle" was later popularized by Stock & Watson (2004). In a recent study,

(c.f., Smith & Wallis (2009)), simple averaging remains the most popular method in the field. The relatively mediocre performance of more sophisticated algorithms in practice is often attributed to structural instabilities in real world data. Several studies[3] have stressed the need for combination algorithms to be robust to different types of instabilities.

However, most of relevant existing studies focus on some specific algorithm's robustness in some specific type of instability[4]. These studies, many of which also propose new combination algorithms or strategies, invariably show the effectiveness of proposed algorithms under specific and favorable conditions. While making valuable contributions in their own right, limited by space and scope, they give little consideration to alternative algorithms or alternative types of instabilities. There has been no systematic study that looks into the robustness of several algorithms in realistic unstable environments and documents their characteristics across a wide array of scenarios.

This void is in fact quite understandable. Theoretical work on this issue is difficult if not impossible. Each combination algorithm works under a strict set of assumptions with proved optimality. Deviations from the assumptions often make the math intractable, especially when the deviations involve time-varying parameters, thick-tailed distributions, and non-stationarity. Studies using real world data is also limited. Macroeconomists rarely see experimental data. Using observational data, while one easily observes the performance of different combination algorithms, no clear clue of why the algorithms perform the way they are can be easily deduced. To systematically study the performance and robustness of forecast combination algorithms in unstable environments, the only feasible way is simulation. In fact, as early as 1989, Armstrong (1989) have explicitly cited "*realistic simulations*" as one of three broad directions for future research, along with meta-analysis and rule-based forecasting.

The main objective of this study is to provide some evidence regarding the robustness of various forecast combination algorithms in a number of situations with structural instabilities that are commonly seen but difficult to quantify in practice. Two families of combination algorithms are studied. The first is the set of aggregate forecast through exponential re-weighting (AFTER) algorithms, proposed in Yang (2004), Wei & Yang (2012), and most recently Cheng & Yang (2015). These algorithms accommodate the squared error loss, absolute error loss, and a synthetic loss, which is a flexible mixture of squared and absolute error losses. They are designed to adapt quickly to changes in candidate forecasters' performance and deliver improved accuracy while guard against outliers. They work both in terms of being robust to outliers in candidate forecasts, and in terms of producing fewer outliers in combined forecasts. The other is the algorithm proposed in Sancetta (2010). This algorithm complements the AFTER algorithms by offering more flexibility in the choice of loss functions and by relaxing the assumptions on candidate forecasts. It also includes a shrinkage

Lahiri et al. (2015) documented that when combining U.S. SPF forecasts using many of the methods also considered in this study, simple average remains difficult to beat.

[3]See, among others, Elliott & Timmermann (2005), Aiolfi & Timmermann (2006), Smith & Wallis (2009), Pesaran et al. (2013) Tian & Anderson (2014)

[4]Recent ones include Giraitis et al. (2013), Pesaran et al. (2013), Tian & Anderson (2014), and Chevillon (2016).

step (i.e., shrinking individual weights towards equal weight) that helps to hedge against structural instabilities. Both families of algorithms have been shown by their respective authors to perform well in simulated environments.

In generating simulated data sets that accommodate structural instabilities, the standard approach is to use models such as simple linear regression models (such as in Cheng & Yang (2015)) and ARIMA models (such as in Pesaran & Timmermann (2005), Sancetta (2010), and Chevillon (2016)). Candidate forecasts are usually misspecified versions of the true model. While this approach creates clear linkage between the data generating process and theoretical assumptions, it is often not immediately clear how well candidate forecasts perform, especially when the models include complex settings to induce instabilities. In this study, a different approach is used, where aggregate shocks and forecast errors are directly drawn from standard distributions. This way, the magnitude of unbiasedness of candidate forecasts, the variance of idiosyncratic forecast errors, and that of aggregate shocks and uncertainty are immediately clear from the data generating process. Furthermore, it is immediately clear what is the true optimum weight for each candidate forecaster.

This novel design of the data generating process allows the creation of a comprehensive set of scenarios with different but general types of structural instabilities, including: heterogeneous and time-varying relative forecast performance induced by one-time or multiple breaks in forecast bias or forecast error variance; gradually but constantly changing forecast performance; unexpected aggregate shocks to all forecasters; and forecaster specific outliers. Results from these simulation exercises suggest that estimation errors that arise from having to estimate the weights used in combination is significant. For several algorithms in the AFTER family, there may be cases where the inaccuracies arising from estimation error outweigh the performance gain from combining forecasts. In such cases, pre-filtering or grouping candidate forecasts with similar performance, whereby significantly reduces the number of candidate forecasts, may be beneficial. On the other hand, different combination algorithms excel in different types of instabilities: The method proposed by Sancetta (2010) is more robust to discrete changes in forecast error variance and the algorithm proposed in Cheng & Yang (2015) are more robust to aggregate shocks. Based on the simulation results, a simple forecast combination strategy is demonstrated to be useful in real-time combination of forecasts of four important macroeconomic variables reported in the U.S. Survey of Professional Forecasters.

While performance of different combination algorithms are reported relative to that of simple average, the intention of this study is not to run a horse race of different algorithms. Whether an algorithm delivers superior performance than the simple average benchmark is clearly a function of the parameters in simulation design. Therefore, when comparisons are made, the focus lies in the performance of an algorithm across different types of instabilities.

The rest of the paper is organized as follows: Section 2 introduces the combination methods used in this study. Section 3 describes the setup of the simulation exercises and presents simulation results. In Section 4, a forecast combination strategy based on the lessons learned from Section 3 is applied to combining U.S. SPF forecasts. Concluding remarks are in Section 5.

2. Combination Methods

In this study, five recently developed advanced forecast combination algorithms are examined: s-AFTER algorithm from Yang (2004), L_1-AFTER algorithm from Wei & Yang (2012), h-AFTER algorithm from Wei & Yang (2012), L_{210}-AFTER algorithm from Cheng & Yang (2015), and the algorithm proposed in Sancetta (2010)[5]. In addition, six widely-used combination methods are also examined: mean, median, trimmed mean, Winsorized mean, recent best, and BG (Bates & Granger, 1969). Each of the methods and algorithms used here could potentially deliver superior performance than simple average, when used properly in a suitable environment.

Consider the task of combining a potentially large number of candidate forecasts of the same target variable in real time: At each time period $t + 1$, after the latest release (y_t) of the target variable y becomes available, a set of weights $\omega_{j,t+h}$ is calculated so as to combine the forecasts of y_{t+h} from candidates $j = 1, 2, 3 \ldots, n$, where h is the forecast horizon. The combined forecast is denoted $\hat{y}_{t+h}$. Without loss of generality, in this study, $h = 1$. Let $e_{j,t} \equiv y_t - \hat{y}_{j,t}$ be the most recent (period t) forecast error, and $\hat{\sigma}^2_{j,t}$ be the estimated variance for candidate forecast series j and time t. In addition, let t_o be the first period in the sample.

The AFTER algorithms differ in terms of what loss function is used as the key ingredient in combination formulas. Proper selection of loss functions for this purpose helps to create a more robust algorithm. The AFTER family examined in this study contains four algorithms each with a unique loss function. The s-AFTER uses squared error loss (or L_2 loss). The L_1-AFTER uses absolute error loss (or L_1 loss). The h-AFTER uses Huber loss. And the L_{210}-AFTER uses a synthetic loss function – a mixture of L_2, L_1, and L_0 loss (discussed below). When applying the s-AFTER algorithm to combining U.S. SPF forecasts, Lahiri et al. (2015) observed that often, performance of the algorithm is inversely affected by just a few large errors or outliers around turning points in the target variable, rather than many small errors scattered throughout the sample period. The latest addition to the AFTER family, L_{210}-AFTER, is designed to specifically address this issue. By providing a direct penalty for forecast outliers through the use of the L_0 loss, L_{210}-AFTER is more robust to outliers in candidate forecasts and it produces fewer outliers itself.

Weights of s-AFTER can be written recursively as

$$\hat{\omega}^{s\text{-}AFTER}_{j,t+1} = \frac{\hat{\omega}^{s\text{-}AFTER}_{j,t}\hat{\sigma}^{-1}_{j,t}\exp\left(-\frac{e^2_{j,t}}{2\hat{\sigma}^2_{j,t}}\right)}{\sum_{j=1}^{n}\left[\hat{\omega}^{s\text{-}AFTER}_{j,t}\hat{\sigma}^{-1}_{j,t}\exp\left(-\frac{e^2_{j,t}}{2\hat{\sigma}^2_{j,t}}\right)\right]} \text{ for } t \geq t_o + 1 \tag{1}$$

where $\hat{\omega}^{s\text{-}AFTER}_{j,t_o} = \frac{1}{n}\ \forall j$, i.e., equal weights are used in the very first period.

[5] Below, the first four may collectively be referred to as the AFTER algorithms or the AFTERs. Sancetta's algorithm is referred to as the SAN algorithm.

Weights of L_1-AFTER can be written recursively as

$$\hat{\omega}_{j,t+1}^{L_1\text{-}AFTER} = \frac{\hat{\omega}_{j,t}^{L_1\text{-}AFTER}\hat{d}_{j,t}^{-1}\exp\left(-\frac{|e_{j,t}|}{\hat{d}_{j,t}}\right)}{\sum_{j=1}^{n}\left[\hat{\omega}_{j,t}^{L_1\text{-}AFTER}\hat{d}_{j,t}^{-1}\exp\left(-\frac{|e_{j,t}|}{\hat{d}_{j,t}}\right)\right]} \text{ for } t \geq t_o + 1 \tag{2}$$

where $\hat{\omega}_{j,t_o}^{L_1\text{-}AFTER} = \frac{1}{n}\ \forall j$, i.e., equal weights are used in the very first period. As suggested in Yang (2004), $\hat{d}_{j,t}$ is simply substituted by $\hat{\sigma}_{j,t}$.

Weights of h-AFTER can be written recursively as

$$\hat{\omega}_{j,t+1}^{h\text{-}AFTER} = \frac{\hat{\omega}_{j,t}^{h\text{-}AFTER}\hat{\sigma}_{j,t}^{-1}\exp\left(-h_{j,t}\right)}{\sum_{j=1}^{n}\left[\hat{\omega}_{j,t}^{h\text{-}AFTER}\hat{\sigma}_{j,t}^{-1}\exp\left(-h_{j,t}\right)\right]} \text{ for } t \geq t_o + 1 \tag{3}$$

where $\hat{\omega}_{j,t_o}^{h\text{-}AFTER} = \frac{1}{n}\ \forall j$, i.e., equal weights are used in the very first period; $h_{j,t} = \varphi_s(\frac{e_{j,t}}{\sqrt{2}\sigma_{j,t}})$.

$\varphi_s(\cdot)$ is the loss function defined as

$$\varphi_s(x) = \begin{cases} x^2 & \text{if } -1 \leq x \leq s \\ 2sx - s^2 & \text{if } x > s \\ -2x - 1 & \text{otherwise} \end{cases} \tag{4}$$

where the parameter $s > 0$ controls the shape of the loss, with the loss function being symmetric when $s = 1$ and asymmetric otherwise. Figure 1, a reproduction of Figure 1 in Wei & Yang (2012), compares the absolute error loss, Huber loss with $s = 1.5$ and squared error loss.

Weights of L_{210}-AFTER can be written recursively as

$$\hat{\omega}_{j,t+1}^{L_{210}\text{-}AFTER} = \frac{\hat{\omega}_{j,t}^{L_{210}\text{-}AFTER}\hat{\delta}_{j,t}^{-1/2}\exp\left(-L_{210}(e_{j,t})/(2\hat{\delta}_{j,t})\right)}{\sum_{j=1}^{n}\left[\hat{\omega}_{j,t}^{L_{210}\text{-}AFTER}\hat{\delta}_{j,t}^{-1/2}\exp\left(-L_{210}(e_{j,t})/(2\hat{\delta}_{j,t})\right)\right]} \text{ for } t \geq t_o + 1 \tag{5}$$

where $\hat{\omega}_{j,t_o}^{L_{210}\text{-}AFTER} = \frac{1}{n}\ \forall j$, i.e., equal weights are used in the very first period. As discussed in Cheng & Yang (2015), the version of the algorithm implemented here includes automated data-driven estimation of the scale-parameter[6], i.e., $\hat{\delta}_{j,t} = (1/t)\sum_{l=1}^{t} L_{210}(y_l - \hat{y}_{j,l})$.

$L_{210}(\cdot)$, the synthetic loss function designed for outlier-protective combination, is defined as

$$L_{210}(x) = |x| + \alpha_1\frac{x^2}{m} + \alpha_2 m\tilde{L}_0(x|\gamma_1 m, \gamma_2 m, r_1 r_2) \tag{6}$$

[6]See Remark 3 to Theorem 2 in Cheng & Yang (2015).

where $\alpha_1, \alpha_2, \gamma_1, \gamma_2, r_1, r_2$, and m are parameters of the loss function[7]. Figure 2, a reproduction of Figure 1 in Cheng and Yang(2015), shows the $\tilde{L}_0(\cdot)$ loss function, which is designed to provide direct penalty to outliers. The loss function is defined as

$$\tilde{L}_0(x) = \begin{cases} 1, & \text{if } x \geq \gamma_1 \text{ or } x \leq \gamma_2 \\ 1 - \frac{(x-\gamma_1)^2}{\gamma_1^2(1-r_1)^2} & \text{if } r_1\gamma_1 \leq x \leq \gamma_1 \\ 1 - \frac{(x-\gamma_2)^2}{\gamma_2^2(1-r_2)^2} & \text{if } \gamma_2 \leq x \leq r_2\gamma_2 \\ 0 & \text{if } \gamma_2 r_2 \leq x \leq \gamma_1 r_1 \end{cases} \tag{7}$$

The SAN algorithm is implemented according to Algorithm 1 in Sancetta (2010). In addition to allowing flexible loss functions to be used in deriving the weights, the algorithm features a shrinkage step, where estimated weights for candidate forecasters are shrinked towards equal weight. This feature in theory leads to suboptimal performance in stable environment, where either one candidate forecaster is clearly better than the rest, or when the optimal combination weights change only slowly. But this feature should help in unstable environments, where the optimal combination weights abruptly changes or when there are forecast outliers. In addition, shrinkage, even when not delivering improved accuracy, often serves well in reducing the variability of the algorithm and outliers in combined forecasts (see Sancetta (2007) for additional empirical evidence).

The core step in the SAN algorithm is to compute the $t+1$ weight before shrinkage $\omega_{j,t+1}^{SAN'}$ for each forecaster, based on her previous-period (t) weight $\omega_{j,t}^{SAN}$ and current-period loss $l_t(\omega_t^{SAN})$. Let $\nabla l_t(\omega_t^{SAN})$ be the gradient of the loss function with respect to previous period weight ω_t^{SAN}, and $\nabla_j l_t(\omega_t^{SAN})$ be its jth element. The current-period weight is calculated as:

$$\omega_{j,t+1}^{SAN'} = \frac{\omega_{j,t}^{SAN} \exp\left[-\eta t^{-\alpha} \nabla_j l_t(\omega_{j,t}^{SAN})\right]}{\sum_{j=1}^{n} \left\{\omega_{j,t}^{SAN} \exp\left[-\eta t^{-\alpha} \nabla_j l_t(\omega_{j,t}^{SAN})\right]\right\}} \tag{8}$$

where $\eta > 0$ is the learning rate, and $\alpha \in (0, 1/2]$ is a parameter that controls the speed of learning. In the shrinkage step that gives the current-period weight used for combination $\omega_{j,t+1}^{SAN}$, all the $\omega_{j,t+1}^{SAN'}$s that are lower than a predetermined small threshold γ/n is replaced by the threshold weight γ/n, and the remaining weights are scaled such that all weights add up to 1. The threshold γ/n is controlled by the parameter $\gamma \in [0, 1]$, given the number of candidate forecasts to be combined n.

In addition to the AFTERs and SAN algorithms, several standard forecast combination methods are implemented: Simple average (SA) assigns the same weight to every candidate forecasts, i.e., $\omega_{j,t} = 1/n, \forall j$. Median (ME) produces combined forecast as the median of all candidate forecasts. Bates and Granger's method (BG) is introduced in Bates & Granger

[7]Please refer to Cheng & Yang (2015) for detailed discussions and examples of their interpretation and selection. In general, different parameter choices are needed for different applications. The optimum selection of parameters is beyond the scope of this study.

(1969), where the weights are assigned as $\omega_{j,t+1} = \hat{\sigma}_{j,t}^{-2} / \sum_{i=1}^{n} \hat{\sigma}_{i,t}^{-2}$. Trimmed mean(TM) produces combined forecast as the mean of candidate forecasts after discarding the largest and the smallest k forecasts. k can be determined in one of two ways. (1) One may explicitly specify k. (2) One may specify the percentage of candidate forecasts to be trimmed, p, where then k is calculated as $k = \max\{||n \cdot \frac{p}{100}||, 1\}$ where $||x||$ gives the nearest integer to x. Similar to trimmed mean, Winsorized mean (WM) produces combined forecast as the mean of candidate forecasts after replacing the largest k forecasts with the largest forecast in the remaining n-$2k$ forecasts and replacing the smallest k forecasts with the smallest forecast in the remaining n - $2k$ forecasts. k is specified or calculated the same way as in trimmed mean. Finally, as a naive benchmark, recent best (RB) identifies the best forecast from the most recent period and uses the forecast made by the same forecaster as combined forecast. Specifically, weights are assigned as $\omega_{j,t+1} = 1$ if $e_{j,t} = \min\{e_{1,t}, e_{2,t}, \ldots, e_{n,t}\}$ and $\omega_{j,t+1} = 0$ otherwise.

For all the above methods where the calculation of $\hat{\sigma}_{j,t}^2$ is required, a window size parameter w is specified such that $\hat{\sigma}_{j,t}^2 = w^{-1} \sum_{\tau=1}^{w} e_{j,t-\tau+1}^2$ This parameter $w \in \mathbb{N}$ controls the length of the "memory" of combination algorithms. It puts a limit on how far back an algorithm goes when calculating the performance of a candidate forecaster. Since the focus of this study is on the behavior and performance of different combination algorithms in unstable environment, the need of limiting the "memory" of algorithms naturally arises. Note that this limit is imposed only on the calculation of $\hat{\sigma}_{j,t}^2$. For algorithms that recursively update each forecaster's weight, information from earlier time periods that are outside of the window w is carried over. As discussed below, this may explain the apparent lack of agility of certain algorithms in cases where there are breaks in performance of candidate forecasters.

3. Simulation Setup and Results

3.1. Simulation setup

Consider the task of combining multiple forecasts $\hat{y}_{jt}$ of y_t with $j = 1, 2, \ldots, n$ and $t = 1, 2, \ldots, T$. j is used to index forecasters and t is the usual time index. Whenever applicable, estimation of weights is based on a rolling window of w periods, i.e., weights used to combine forecasts of a_t is based on information from t - w to t - 1. Unless otherwise specified, for each exercise, consider $n \in \{5, 30\}$, $T = 300$, and $w \in \{24, 60\}$. The first 60 periods are used as training sample and are excluded when evaluating the performance of combined forecasts. In addition, let

$$y_t = s_t + a_t, \ a_t \sim \mathcal{N}(0, 1) \tag{9}$$

$$\hat{y}_{jt} = s_t + b_{jt} + \varepsilon_{jt}, \ \varepsilon_{it} \sim \mathcal{N}(0, \sigma_{it}^2) \tag{10}$$

where b_{jt} is an individual specific and potentially time-varying bias term, ε_{jt} is idiosyncratic component of forecast error, and σ_{jt}^2 is an individual specific and potentially time-varying variance of ε_{jt}. s_t is the structural or predictable part of y_t. a_t represents unpredictable aggregate shock. In this setup, the forecast error is

$$e_{jt} = y_t - \hat{y}_{jt} = a_t - b_{jt} - \varepsilon_{jt}, \tag{11}$$

which is to say that forecast error consists of aggregate shock, forecast bias, and forecaster-specific error.

The setup introduced here is a very general one, consistent with a wide variety of forecasting scenarios. There is no assumption on the nature of the process of the target variable. s_t, the part of y_t that can be forecast, can be any process. Since s_t will not explicitly enter the expression of forecast error, no specific assumption or restriction need to be imposed on it. When $n = 5$, the situation is similar to one where forecasts being combined are from a small set of (not necessarily purely numerical) models. When $n = 30$, the situation is similar to combining survey forecasts. $w = 24$ can be associated with a highly unstable environment; $w = 60$ may be used in a more stable environment by forecasters in practice. These choices aim at creating an environment that is often found when combining macroeconomic forecasts on a monthly basis. Directly drawing different components of forecast error (as opposed to using potentially misspecified models to generate forecasts) makes differences in forecasting performance intuitively clear and precisely controllable. A fixed T comes without loss of generality too: When reporting the results below, when necessary, evaluation of combined forecasts is carried out without using the full T periods, mimicking settings with smaller Ts.

For each given set of parameters $\{n, T, w\}$, the simulation is carried out as follows:

1. Draw $\{b_{jt}, \sigma^2_{jt}\}\ \forall j$ according to the specification of the simulation exercise. Details on the set up of each exercise are presented below.
2. For each t, draw a_t and $\varepsilon_{jt}\ \forall j$ given σ^2_{jt}. The forecast error e_{jt} is calculated as y_t - $\hat{y}_{jt}$ = a_t - b_{jt} - ε_{jt}.
3. Apply combination algorithms discussed previously to generate combined forecasts. Record the MSE of the combined forecasts produced by each algorithm.
4. Repeat Step 2 to Step 3 200 times. For each algorithm, obtain the average MSE across these repetitions. These MSEs are conditioned on specific draws of b_{jt} and σ^2_{jt}.
5. Repeat Step 1 to Step 4 1000 times. For each algorithm, obtain the average MSE across these repetitions. These MSEs are what reported below, which are conditioned only on distributional assumptions about b_{jt} and σ^2_{jt}.

For each simulation exercise, relative MSEs are reported instead of MSEs, which vary from exercise to exercise. The relative MSE of an algorithm is its MSE from the last step divided by that of simple average. A relative MSE bigger than 1 means that the algorithm produces combined forecasts that have bigger MSE than the combined forecasts produced by simple average. For L_{210}-AFTER, parameter values are set to $a_1 = 1, a_2 = 1, g_1 = 2, g_2 =$ -2, $r_1 = 0.75, r_2 = 0.75, m = 2$, as suggested by the author. For Sancetta's algorithm, two sets of parameter values are considered. The first set (SAN1) has $\eta = 0.3$; the second set (SAN2) has $\eta = 0.7$. For both of them, $\alpha = 0.5$ and $\gamma = 0.5$.

While relative MSEs are reported and discussed, they do not provide any conclusion on whether a combination algorithm is "better than" or "worse than" simple average. As will become evident below, the choice of parameters of the simulation exercises is not made to facilitate such comparisons. In fact, in some exercises, simple average is the theoretically optimum method to use. A total of seven exercises are conducted. For each of the seven

exercises, four sets of parameter values are considered. This gives a total of 28 sets of simulation results. Each subsection below presents and discusses the results of one exercise.

3.2. Cost of estimation errors

In exercise 1, the focus is on the cost in terms of accuracy of combined forecasts that comes from having to estimate individual specific weights, while the optimum weights are equal weights. All forecasts are unbiased and homoskedastic, i.e., have the same variance for all time periods. Specifically, in this exercise, $b_{jt} = 0$ and $\sigma^2_{jt} = \sigma^2 \; \forall j, t$. While holding the variance of aggregate shocks fixed at 1, a set of variances that become progressively larger is considered here: $\sigma^2 \in \{0.2, 0.4, 0.6, \ldots, 20\}$.

Figure 3 plots the relative MSEs against σ^2, so that increase in the cost of estimation as forecast error variance increases can be clearly seen.[8] Relative MSEs of RB, s-AFTER, and L_{210}-AFTER increase at a much higher rate than that of BG, SAN1, and SAN2. This is even more so when there are 30 candidate forecasts to be combined, instead of 5. With this increase in the number of candidate forecasts, relative MSE of BG almost stays the same; relative MSE of SAN1 and SAN2 slightly decreases; relative MSE of RB and the two AFTERs becomes almost 3 times bigger. Comparing the results from using a shorter window of 24 periods with that using a longer window of 60 periods, no discernible difference can be seen for all but BG, where a longer window helps to achieve better performance.

These observations are consistent with theoretical results (see Smith & Wallis (2009) and references therein) on the forecast combination puzzle, that is, the need to estimate a potentially large number of weights may be so costly that despite the benefit of combining forecasts, at the end of the day, combined forecasts are less accurate than what simple average offers. What may also be learned here is that AFTER algorithms seem to perform better when the number of candidate forecasters is small. In other words, AFTER may be the choice when combining a small set of forecasters/models, but may not be the best option when combining forecasts from a large survey of many forecasters.

3.3. Biased forecasts in stable environment

Exercise 2 is conducted in a stable environment, where candidate forecasts are homoskedastic but biased, with both the amount of bias and the variance of the idiosyncratic component of the forecast error constant over time. Specifically, $b_{jt} = b_j \; \forall t$ and $\sigma^2_{jt} = \sigma^2 \; \forall j, t$. This is a situation with a significant amount of potential for performance based weighting algorithms to perform well – simply by identifying the forecaster with the smallest amount of bias. The simple average method is no longer optimum in this exercise. Four sets of parameter values are considered here. In exercise 2-1, the amount of bias that a forecaster may have is relatively small. And the variance of the idiosyncratic component of forecast error,

[8] In this and the following exercises, results from median (ME), trimmed and Winsorized mean (TM, WM), as well as h-AFTER and L_1-AFTER are omitted for brevity. These two AFTER algorithms perform similar to s-AFTER and L_{210}-AFTER. TM and WM give similar results. With small to moderate amount of trimming, they baheve similar to SA. With larger amount of trimming, they behave similar to ME. Omitted results are available from the author upon request. When reporting results, s-AFTER is labeled SAFTER, and L_{210}-AFTER is labeled L210A.

which is common to all forecasters, is also relatively small: $b_j \sim \mathcal{U}(0,1)$ and $\sigma^2 = 1$. In exercise 2-2, the distribution of the bias term is the same as in 2-1, but variance is much larger: $b_j \sim \mathcal{U}(0,1)$ and $\sigma^2 = 4$. In exercise 2-3, bias is larger and variance is relatively small: $b_j \sim \mathcal{U}(1,2)$ and $\sigma^2 = 1$. Exercise 2-4 is the one where both bias and variance are large: $b_j \sim \mathcal{U}(1,2)$ and $\sigma^2 = 4$.

Table 1 shows the results from this exercise. If one extreme is exercise 1, where all the forecasters have exactly the same performance and simple average is the optimum combination method, this is the opposite extreme, where the optimum weighting scheme is one that places all weights on the forecaster with smallest amount of bias. In fact, in this exercise, the performance based weighting algorithms perform well, as expected. Comparing exercise 2-1 with 2-2 or comparing 2-3 with 2-4, it can be seen that as forecast errors become more variable, it is more difficult to estimate forecasters' performance accurately, so that combined forecasts perform worse. It is particularly so for the AFTER methods, which, as shown in exercise 1, are more sensitive to increase in variance. On the other hand, comparing exercise 2-1 with 2-3 or comparing 2-2 with 2-4, it is obvious that combined forecasts are much more accurate when individual forecasts have larger bias. In particular, results from exercise 2-3 show that when combining forecasts with potentially large bias, aggressive weighting algorithms such as the AFTERs may deliver superior performance, despite having to combine a large number of forecasts.

While it is informative to examine the performance of combination algorithms over the entire evaluation sample, examination of how performance evolves reveals additional characteristics of the algorithms. Figure 4 shows the 12-period moving average of relative MSE of BG, SAFTER, L210A and SAN2 when combining 30 candidate forecasts in exercise 2-3. For BG, as more observations accumulate, performance of combined forecasts stay stable. However, this is not the case for the other three algorithms. For SAFTER and L210A, there is a clear upward trend in the MSE of combined forecasts, while there is a clear downward trend in that of SAN2. This observation suggests that one may benefit more from the AFTER type algorithms when applying them to shorter samples, or, by "resetting" the algorithm periodically, i.e., reverting the weights of individual forecasters to equal weight. Performance of all the algorithms eventually stabilizes given a long enough sample period in this simulated stable environment. But in reality, in unstable environment, there may never be long enough a sample in each regime for performance to stabilize. So, in the exercises below, structural instabilities are built into the simulation.

3.4. Biased forecasts with breaks in performance

In exercise 3, the setting in exercise 2 is reconsidered in the simplest kind of unstable environment – a one time mean shift, or more specifically, a one-time break in biases (i.e., biases are re-drawn) for all forecasters. Specifically, $b_{jt} = b_{j1}$ for $t \leq 180$ and $b_{jt} = b_{j2}$ for $t > 180$; $\sigma^2_{jt} = \sigma^2 \; \forall j, t$. The four sets of parameters are the same as in the previous exercise. In exercise 3-1, $b_{j1}, b_{j2} \sim \mathcal{U}(0,1)$ and $\sigma^2 = 1$. In exercise 3-2, $b_{j1}, b_{j2} \sim \mathcal{U}(0,1)$ and $\sigma^2 = 4$. In exercise 3-3, $b_{j1}, b_{j2} \sim \mathcal{U}(1,2)$ and $\sigma^2 = 1$. And in exercise 3-4, $b_{j1}, b_{j2} \sim \mathcal{U}(1,2)$ and $\sigma^2 = 4$.

Table 2 presents the results. As expected, even with the break present, combined forecasts are more accurate when the variances of individual forecasters are small and/or when the biases are large. Comparing the results in this table with the results in Table 1, one could easily see that even though all of the combination methods perform worse in presence of the break, performance deterioration is minimal. This observation, however, should not lead to the conclusion that the methods examined here are robust to breaks. Rather, it is the long pre- and post-break period that have masked much of the fluctuations in performance.

A closer look at how performance evolves provides additional insight: Figure 5 shows the 12-month moving average of relative MSE of BG, L210A, SAN1 and SAN2 when combining 30 candidate forecasts in exercise 3-3. Overall, one clearly sees the effect of the break, as the MSE of combined forecasts suddenly increases then gradually declines as more data become available post break. Comparing the cases with a longer estimation window (60 periods) versus a shorter estimation window (24 periods), it is somewhat surprising to note that, except in the case of BG, a shorter estimation window does not seem to shorten the time it takes for the performance of combined forecasts to stabilize post break. This may be a result of the design of the algorithms under examination, where in each period for each forecaster, the weight from previous period is updated according to most recent forecast performance. When a limited estimation window is imposed, even though only a limited amount of historical information is used in calculating recent performance, a forecaster's entire performance record is always embedded in his weight.

One other interesting observation made from Figure 5 is that the performance of combined forecasts produced by different algorithms takes a different amount of time to stabilize after the break. For BG, performance return to its pre-break level before the end of the evaluation sample. For L210A, it takes significantly longer, such that even after 120 periods, MSE of the combined forecasts remains much worse than its pre-break level. Comparing SAN1 with SAN2, where the latter has higher learning rate, one observes that SAN2 adapts much quicker after the break, but the overall performance is actually slightly worse – an indication of possibly excessive sensitivity to small performance changes during stable periods. This is often the trade-off practitioners face: A more aggressive algorithm may work better in unstable periods, but may be over sensitive during stable periods.

3.5. Heteroskedastic forecasts with breaks in performance

In the previous two exercises, candidate forecasts were assumed to be biased. From this exercise onwards, the assumption of unbiasedness will be maintained. In practice, this could either be because that candidate forecasts are indeed unbiased, or, it could be that candidate forecasts are pre-screend or pre-processed so that biased forecasts are either eliminated or adjusted. Of course, in reality, when the sample is sufficiently short or is suspected to be subjected to frequent structural breaks, it is difficult to assess whether candidate forecasts are unbiased. Most combination algorithms do not explicitly differentiate components of the forecast error. So, the assumption of unbiasedness, especially in short and unstable sample, may have little practical impact.

Exercise 4 considers forecasts that are unbiased but heteroskedastic, in that the variance of idiosyncratic component of forecast error is different from forecaster to forecaster. Forecasts with smaller variance are preferable and should receive higher weights. Two types

of instability are introduced. The first is a one-time break half way through the sample. The second is three breaks distributed throughout the evaluation sample. More specifically, $b_{jt} = 0 \ \forall j, t$, and $\sigma^2_{jt} = \sigma^2_{jr}$ if $\delta_{r\text{-}1} < t \leq \delta_r$, where $r \in \{1, 2, \ldots, R\}$ indexes regimes, $\delta_0 = 0, \delta_R = +\infty$. There are only two regimes ($R = 2, \delta_1 = 180$), i.e., one break, in exercise 4-1 and 4-2. In exercise 4-1, $\sigma^2_{jr} \sim \mathcal{U}(0.1, 2.5)$. In exercise 4-2, $\sigma^2_{jr} \sim \mathcal{U}(0.1, 6.5)$. There are four regimes ($R = 4$ and $\delta_1 = 90, \delta_2 = 150, \delta_3 = 210$), i.e., three breaks, in exercise 4-3 and 4-4. In exercise 4-3, $\sigma^2_{jr} \sim \mathcal{U}(0.1, 2.5)$. In exercise 4-4, $\sigma^2_{jr} \sim \mathcal{U}(0.1, 6.5)$.

Table 3 shows the results of this exercise. Comparing exercise 4-1 with 4-3 or comparing exercise 4-2 with 4-4, it seems that while the introduction of two additional breaks does lead to worse combined forecasts for all the methods, the effect of the two additional breaks are marginal. But it is interesting to observe the behavior of the algorithms in cases with different numbers of forecasters: Regardless of whether there are 5 or 30 forecasters, BG performs better when variances of individual forecasters are scattered over a wider range, as in exercise 4-2 and 4-4, where the two AFTER algorithms perform worse. However, the performance of SAN1 and SAN2 is different. As an example, comparing 4-1 with 4-2, SAN1 performs better in 4-2 when there are only 5 forecasters, but it performs worse in 4-2 when there are 30 forecasters. When there are more forecasters in the pool, the cost from having to estimate all the weights becomes higher. But at the same time, there may be a higher level of dispersion in forecasters' performance. Performance of combined forecasts depends on which effect dominates.

Figure 6 shows how performance of combined forecasts change over time when there are three breaks in the sample (exercise 4-4). Much like the results from the previous exercise, BG adapts quickly after the break, especially when the window size is small. SAFTER and L210A suffer from slow adaptation and long memory such that as more breaks hit, errors accumulate and combined forecasts become increasingly unreliable. In this simulation exercise, there are 60 periods between two breaks. In the real world, time between two breaks may be much shorter. This may present a challenge for the AFTER algorithms. As suggested earlier, "resetting" the algorithm from time to time may be an empirically viable strategy, especially when breaks can be determined ex post.

3.6. Dynamically heteroskedastic forecasts

Previous exercise considers discrete break(s) in forecast performance. In this exercise, continuously changing forecast performance is examined. In some cases, performance gradually improves over time. In other cases, performance gradually deteriorates over time. In this setup, it is unlikely for one single forecaster to stay the best for the entire sample period. Combination algorithms that quickly and accurately identify and adapt to the best forecaster may produce good combined forecasts.

All forecasts are unbiased. $b_{jt} = 0 \ \forall j, t$. For a forecaster, given the performance (i.e., variance of idiosyncratic component of forecast error) at the beginning and end of a sample period, performance changes gradually following a linear time trend:

$$\sigma^2_{jt} = \sigma^2_{j1} + \frac{\sigma^2_{jT} - \sigma^2_{j1}}{T - 1} \times (t - 1) \tag{12}$$

with $\sigma^2_{j,1} \sim \mathcal{U}(p_1, q_1)$ and $\sigma^2_{j,T} \sim \mathcal{U}(p_T, q_T)$, where p_1, q_1, p_T, and q_T are given parameters.

In exercise 5-1, random performance changes happen to all forecasters: $p_1 = p_T = 0.1$ and $q_1 = q_T = 6.5$. In exercise 5-2, with probability 0.5, $p_1 = p_T = 0.1, q_1 = q_T = 3.5$. with probability 0.5, $p_1 = p_T = 3.5, q_1 = q_T = 6.5$. In this setting, with equal probability, a forecaster is placed in one of two groups, a good group and a poor group. Performance changes in a way that forecasters do not move to the other group, but the relative performance of forecasters within a group may change. In other words, the worst forecaster in the group of good forecasters is always better than the best one from the group of poor forecasters. In exercise 5-3, with probability 0.5, $p_1 = 0.1, q_1 = 3.5, p_T = 3.5, q_T = 6.5$. with probability 0.5, $p_1 = 3.5, q_1 = 6.5, p_T = 0.5, q_T = 3.5$. This setting is identical to the setting of exercise 5-2, except that here everyone slowly moves to the other group, i.e., the group of good forecasters slowly become the group of poor forecasters, vice versa. Relative performance within a group may change as well. In exercise 5-4, with probability 0.5, $p_1 = 0.1, q_1 = 6.5$ and σ^2_{jT} is non-random and equals σ^2_{j1}. With probability 0.5, $p_1 = p_T = 0.1$ and $q_1 = q_T = 6.5$. In this setting, roughly half the forecasters' performance is stable, but the other half's performance changes.

Results of exercise 5 are presented in Table 4. While one may expect that the more widespread the breaks, the worse the performance of combination algorithms, it is not necessarily true. Comparing exercise 5-1 with 5-4, where in the former, everyone's performance changes and in the latter, only half the forecasters' performance changes, it can be seen that SAN1 and SAN2 produce slightly better forecasts in exercise 5-1, when there are 30 forecasters in the pool. A comparison between exercise 5-2 and 5-3 leads to similar observations. When forecasters move only within group, as in exercise 5-2, performance of combination algorithms are generally better. However, exception exists again for SAN1 and SAN2 when there are 30 forecasters. Comparing exercise 5-1 with exercise 4-2 and 4-4, where in exercise 4, forecasters' performance change discretely, it can be seen that performance of combination algorithms in 5-1 is generally worse than that in 4-2, but better than that in 4-4, especially for the AFTERs.

3.7. Heteroskedastic forecasts with unexpected aggregate shock

Unlike the previous two exercises that consider breaks in forecasters' performance, in this exercise, large and unexpected aggregate shocks are considered. Candidate forecasts continue to be unbiased. But the variance of idiosyncratic component of forecast error is different for different forecasters. There is no break in this variance. The optimum weighting scheme should place all weights on the forecaster with the lowest variance. The complication here is the occasional aggregate shocks that make accurate estimation of forecasters' performance difficult.

Specifically, let $b_{jt} = 0 \ \forall j, t$ and $\sigma^2_{jt} = \sigma^2_j \sim \mathcal{U}(0.1, 6.5)$. To accommodate aggregate shocks, a_t now follows a mixture distribution: With probability 1 - p, $a_t \sim \mathcal{N}(0, 1)$. With probability p, $a_t \sim \mathcal{U}(2.5, q)$. p and q are given parameters. p controls the frequency of aggregate shock and q controls its magnitude. In exercise 6-1, low probability of small shocks is considered, where $p = 0.05, q = 4.5$. Exercise 6-2 considers high probability of small shocks with $p = 0.2, q = 4.5$. Exercise 6-3 considers low probability of large shock, where

$p = 0.05, q = 6.5$. And finally, high probability of large shocks is considered in exercise 6-4 with $p = 0.2, q = 6.5$.

Table 5 shows the results. Aggregate shocks make it harder to differentiate good forecasters from poor ones. As a result, one would expect that as aggregate shocks come more frequently and/or at a higher magnitude, combined forecasts would perform poorer. This turns out to be the case for most of the algorithms examined here, where the best performance is observed in exercise 6-1, followed by 6-3, then 6-2, and finally 6-4. However, there is an important exception. The AFTER algorithms perform the opposite way, showing strong robustness to aggregate shocks. Take SAFTER as an example, it performs the best in exercise 6-4 in presence of large and frequent aggregate shocks, and performs the worst in exercise 6-1, where shocks are small and infrequent. Compared with BG, SAN1 and SAN2 also display some robustness to aggregate shocks, in that their performance do not vary much as the frequency and intensity of shocks increase. While this exercise is not recreating business cycles, the scenarios here closely mimic the case when forecasters fall behind business cycle turning points, i.e., fail to foresee a large change in the actual value. So the results here may also shed some light on the behavior of the algorithms when target variable is cyclical.

3.8. Heteroskedastic forecasts with outliers

In exercise 7, forecast outliers are considered. Each period, there is a small chance for a forecaster to produce an outlier, i.e., an unusually large forecast error. Except when influenced by the outliers, forecasts are all unbiased with $b_{jt} = 0 \ \forall j, t$ and have stable performance $\sigma_{jt}^2 = \sigma_j^2 \ \forall t$. With equal probability, a forecaster can be a good one or a poor one: with probability 0.5, $\sigma_j^2 \sim \mathcal{U}(0.1, 3.5)$, and with probability 0.5, $\sigma_j^2 \sim \mathcal{U}(3.5, 6.5)$. Forecast outliers are introduced by letting ε_{jt} follow a mixture distribution. With probability (1 - p_j), $\varepsilon_{jt} \sim \mathcal{N}(0, \sigma_{jt}^2)$. With probability p_j, $\varepsilon_{jt} \sim \mathcal{U}(2\sigma_{jt}, q\sigma_{jt})$. p_j controls the frequency of outliers and q controls their magnitude.

In exercise 7-1, $p_j = 0.05 \ \forall j$, so that everyone may produce outliers. In exercise 7-2, $p_j = 0.05$ if $\sigma_j^2 \sim \mathcal{U}(0.1, 3.5)$, otherwise, $p_j = 0$ so that only those good forecasters may produce outliers. In exercise 7-3, the opposite is considered, where only poor forecasters may produce outliers: $p_j = 0.05$ if $\sigma_j^2 \sim \mathcal{U}(3.5, 6.5)$, otherwise, $p_j = 0$. In the last setting, exercise 7-4, a random set of 20% of all forecasters may produce outliers, regardless of whether they are good forecasters or poor forecasters. So, with probability 0.2, $p_j = 0.05$, and with probability 0.8, $p_j = 0$.

Results from this exercise is presented in Table 6. First, consider the comparison between exercise 7-1 and exercise 7-4. Since in exercise 7-4, only 20% of forecasters may produce outliers, it is expected that combination algorithms perform better in this case. This turns out to be so for all except BG, which performs slightly better in exercise 7-1 when there are 30 forecasters, especially when window size is bigger. For the AFTERs, SAN1, and SAN2, performance in exercise 7-1 is only slightly worse than that in 7-4, regardless of the size of estimation window. Comparing exercise 7-2 with 7-3, as expected, the result is clearly that the combination algorithms examined here take a bigger performance hit when good forecasters produce outliers. Another meaningful comparison is exercise 7-1 versus

exercise 5-2, where the former considers forecast outliers, and the latter considers within group performance instabilities. These two scenarios may easily become indistinguishable in practice in small samples. This comparison shows that, conditioned on the set frequency and magnitude of outliers, having outliers seems to affect the performance of combined forecasts less than having dynamically changing performance, especially for BG and the AFTER algorithms.

4. Combining U.S. SPF Forecasts

The simulation exercises in the previous section revealed several characteristics of the combination algorithms. These findings should help researchers and policy makers to better construct forecast combination procedures when faced with real world problems. In this section, the problem of combining forecasts reported in the U.S. Survey of Professional Forecasters (SPF) is considered as an example of how the findings in the previous section may be of practical use.

Before presenting the combination procedure to be used with the SPF data, a brief discussion of the structure of the data and some pre-processing of the data is in order. The SPF is a well-respected quarterly survey that collects forecasts made by professional forecasters on a wide range of important macroeconomic variables, four of which are used here: real GDP growth rate (RGDP), CPI inflation rate (CPI), GDP deflator inflation rate (PGDP), and unemployment rate (UNEMP). For each of the four variables, current quarter forecasts to three-quarter-ahead forecasts ($h = 0, 1, 2, 3$) are considered[9]. Partly due to the change in survey administrator[10], there is a large amount of missing values in the survey. As shown in Lahiri et al. (2015), implementing combination algorithms on unbalanced panel produces results that are not comparable. Therefore, instead of combining using the full sample, combination is performed using two subsamples separately. The first subsample spans 1968:IV to 1990:IV, and the second starts from 2000:I and ends at 2014:IV[11].

In addition, forecasters with an excessively large amount of missing forecasts are excluded from the combination and the remaining missing forecasts are imputed[12]. The exclusion restriction is that forecasters must have at least 45 non-missing forecasts in subsample 1 or at least 36 in subsample 2. This leaves around 15 forecasters in the first subsample and about 30 forecasters in the second subsample, depending on specific variable and horizon. Missing values are then imputed for each subsample period before combination is carried out. Specifically, a missing forecast f_{jt} is imputed as $f_{jt} = \bar{f}_t + \hat{\beta}_j[\sum_{s=1}^{4}(f_{jt-s} - \bar{f}_{t-s})]$, where

[9] Longer horizon forecasts are not considered here because several studies have found that they are often worse than simple benchmark forecasts. Combining these forecasts is of limited practical use.

[10] The survey was initially conducted by the American Statistical Association (ASA) and the National Bureau of Economic Research (NBER). Starting from 1990, the survey was taken over by the Federal Reserve Bank of Philadelphia.

[11] CPI forecasts before 1981 are not available. Therefore, only subsample 2 of CPI forecasts are considered here. For the other three variables, both subsamples are considered.

[12] No evidence is found to suggest that performance of forecasters are systematically related to their level of participation, as reported in Capistrán & Timmermann (2009), Genre et al. (2013), as well as Lahiri et al. (2015).

$\hat{\beta}_j$ is OLS estimate. This is to say that a missing forecast is imputed as the adjusted average forecast for that time period, where the average is taken over non-missing forecasts for that period made by other forecasters. The adjustment is forecaster specific, and is based on the forecaster's usual amount of deviation from the average forecast in the past year, $\sum_{s=1}^{4}(f_{jt-s} - \bar{f}_{t-s})$. This imputation method is based on Genre et al. (2013) and is used with good results in Lahiri et al. (2015).

After dividing the entire data set into two subsamples, removing forecasters with too many missing forecasts, and imputing the remaining missing forecasts, balanced panels of forecasts are obtained. These forecasts are combined in real time using the algorithms studied in the simulation exercises in the previous section. Combined forecasts are then evaluated against the first vintage/release of the actual values. To prevent the possibility of excessive data mining, where multiple sets of parameter values of each combination algorithms are tried and the set that produces the best combined forecasts is chosen, no parameter selection is attempted here. All the algorithms are implemented using the exact same set of parameters as used in the simulation exercises.

One of the main observations made in the previous section is that the cost of estimating many individual weights increases quickly as the number of forecasters increase and may eventually cancel out any benefit from combing forecasts. The natural solution to this problem is to combine fewer series of forecasts where each series contains combined forecasts from a group of forecasters. This is to say that forecasters in the panel are to be grouped first, based on their past forecasting performance. Then, a group forecast is set to the average of all the individual forecasts in the group. As discussed in Aiolfi & Timmermann (2006), performance of group forecasts is more persistent, especially in the top and bottom group. As exercise 5 from the previous section shows, this is a case in which performance based combination algorithms may produce superior forecasts. In implementing this combination strategy in real time, each period, forecasters are placed in roughly equal-sized groups based on the latest estimate of their MSEs. In subsample 1, there are 5 groups. In subsample 2, there are 3 groups. So 5 to 6 forecasters are in each group. In the very first period in each subsample, simple average is used to combine forecasts so that no grouping is necessary. A window of 20 quarters is imposed when grouping forecasters as well as when implementing the combination algorithms[13].

For each variable, horizon, subsample, and combination algorithm, relative MSE is calculated as the MSE of the combined forecasts produced by the algorithm divided by the MSE of the combined forecasts produced by simply average. So, a relative MSE smaller than 1 means that the combination algorithm performs better than simple average, which is used as a benchmark. Table 7 reports these relative MSEs. Whenever a relative MSE is less than 1, the table cell is shaded. Whenever a relative MSE is significantly less than 1, the number is reported in bold. Statistical significance is from one-sided modified Diebold-Mariano test (Harvey et al., 1997) at 10% level.

As Table 7 shows, the combination algorithms considered in this study often outperform the simple average benchmark, especially when combining PGDP in subsample 1, UNEMP

[13] Alternative window sizes and combining individual forecasts directly without grouping are also attempted. Results are omitted from the paper but are available from the author.

in subsample 2, as well higher horizon forecasts of PGDP, RGDP and UNEMP in both subsamples. In addition, significant performance gains are observed when combining current-quarter CPI forecasts. In several cases, such as current-quarter forecasts of CPI subsample 2 and PGDP subsample 1, performance gains are substantial, up to 30% reduction in MSE compared with simple average benchmark. In cases where the combination algorithms do not deliver better performance than simple average, the loss is very limited: In the majority of such cases, MSE of combined forecasts is not more than 5% to 10% higher than that of simple average combined forecasts.

These results contrast what reported in Lahiri et al. (2015), where in most cases, performance based combination algorithms deliver very modest improvement. Here, after imposing a short estimation window and grouping individual forecasts before combining, statistically significant and practically meaningful performance gains are achieved, without extensive search of the optimum parameter values for each combination algorithm. It should be expected that in a real life forecast combination scenario, when parameter values of combination algorithms can be more appropriately selected, one may obtain even better results.

5. Concluding Remarks

In this study, performance of several recently developed sophisticated forecast combination algorithms in unstable environment is examined in a series of simulation exercises. The first exercise reveals the cost of estimating weights for individual forecasters as the number of forecasters increases. The second exercise creates a stable environment in which individual forecasts are biased but homoskedastic. The third exercise introduces a one time break in bias. The fourth exercise turns to multiple breaks in the performance of heteroskedastic but unbiased forecasts. Next, forecast performance is allowed to dynamically evolve, such that performance may gradually become better (or worse). The first of the two remaining exercises considers the role of aggregate shocks and the second considers the effect of forecaster-specific outliers. Each of these exercises are carried out using four different settings, allowing comprehensive and in-depth analysis of the performance of different combination algorithms.

The simulation exercises lead to several observations. First of all, there is a significant amount of cost associated with estimating weights for individual forecasters. The higher the number of forecasters or the more variable the performance of individual forecasters, the more difficult it is to obtain significant performance gains when combining forecasts using performance-based algorithms. The second observation is that, the length of estimation window have little effect on the overall performance over a long sample period. However, for certain algorithms, such as Sancetta's algorithm (Sancetta, 2010), having a shorter estimation window helps to reduce the effect of breaks on forecast performance. Another observation is that higher degree of heteroskedasticity helps the algorithms to differentiate good forecasters from poor ones, but at the same time makes estimating weights more difficult. This may be good for less aggressive algorithms such as BG, but may present a challenge for algorithms that are more sensitive to small changes in forecasters' performance, such as the AFTERs. Yet another important observation is that it takes a number of periods for the performance of a combination algorithm to stabilize after the first period or after a break. For algorithms

that takes a long time to stabilize in environments with frequent breaks, performance may never reach the optimum (stable) level.

These observations lead to a simple but effective combination strategy which is demonstrated in combining U.S. SPF forecasts of real GDP growth, CPI and GDP defaltor inflation rate, and unemployment rate. The strategy involves two improvements in two stages. In the first stage, apart from filtering out infrequent forecasters and imputing missing forecasts, forecasters are grouped into a few groups based on their past performance. In each group, individual forecasts are averaged to obtain a group forecast. In the second stage, the small number of series of group forecasts are combined, with a short estimation window. This strategy reduces cost of estimating individual weights by reducing the number of forecast series to be combined. It induces different and persistent forecast performance, which makes it easier for combination algorithms to identify good forecasts. In addition, it limits the amount of historical information used in estimating performance through the imposition of a short window, allowing the algorithms to more quickly adapt to instabilities in the data. While not implemented here, additional strategies for selecting parameters of combination algorithms to induce good performance may be of use in real world applications of forecast combination algorithms.

References

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

Armstrong, J. S. (1989). Combining forecasts: The end of the beginning or the beginning of the end? *Special Section: Time Series Monitoring*, *5*, 585–588.

Bates, J. M., & Granger, C. W. J. (1969). The Combination of Forecasts. *Operations Research Quarterly*, *20*, 451–468.

Capistrán, C., & Timmermann, A. (2009). Forecast Combination With Entry and Exit of Experts. *Journal of Business & Economic Statistics*, *27*, 428–440.

Cheng, G., & Yang, Y. (2015). Forecast combination with outlier protection. *International Journal of Forecasting*, *31*, 223–237.

Chevillon, G. (2016). Multistep forecasting in the presence of location shifts. *International Journal of Forecasting*, *32*, 121–137.

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

Elliott, G., & Timmermann, A. (2005). Optimal forecast combination under regime switching*. *International Economic Review*, *46*, 1081–1102.

Genre, V., Kenny, G., Meyler, A., & Timmermann, A. (2013). Combining expert forecasts: Can anything beat the simple average? *International Journal of Forecasting*, *29*, 108–121.

Giraitis, L., Kapetanios, G., & Price, S. (2013). Adaptive forecasting in the presence of recent and ongoing structural change. *Journal of Econometrics*, *177*, 153–170.

Harvey, D., Leybourne, S., & Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, *13*, 281–291.

Lahiri, K., Peng, H., & Zhao, Y. (2015). On-line Learning and Forecast Combination in Unbalanced Panels. *Econometric Reviews*, .

Pesaran, M. H., Pick, A., & Pranovich, M. (2013). Optimal forecasts in the presence of structural breaks. *Journal of Econometrics*, *177*, 134–152.

Pesaran, M. H., & Timmermann, A. (2005). Small sample properties of forecasts from autoregressive models under structural breaks. *Journal of Econometrics*, *129*, 183–217.

Sancetta, A. (2007). Online forecast combinations of distributions: Worst case bounds. *Journal of Econometrics*, *141*, 621–651.

Sancetta, A. (2010). Recursive forecast combination for dependent heterogeneous data. *Econometric Theory*, *26*, 598–631.

Smith, J., & Wallis, K. F. (2009). A Simple Explanation of the Forecast Combination Puzzle. *Oxford Bulletin of Economics and Statistics*, *71*, 331–355.

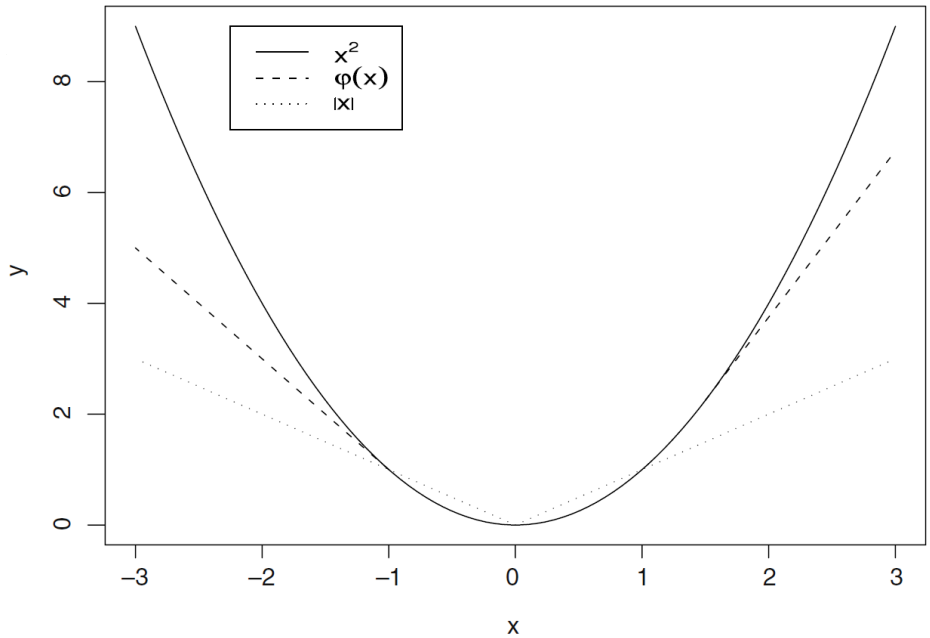
Stock, J. H., & Watson, M. W. (2004). Combination forecasts of output growth in a seven-country data set. *Journal of Forecasting*, *23*, 405–430.

Tian, J., & Anderson, H. M. (2014). Forecast combinations under structural break uncertainty. *International Journal of Forecasting*, *30*, 161–175.

Wei, X., & Yang, Y. (2012). Robust forecast combination. *Journal of Econometrics*, *166*, 224–236.

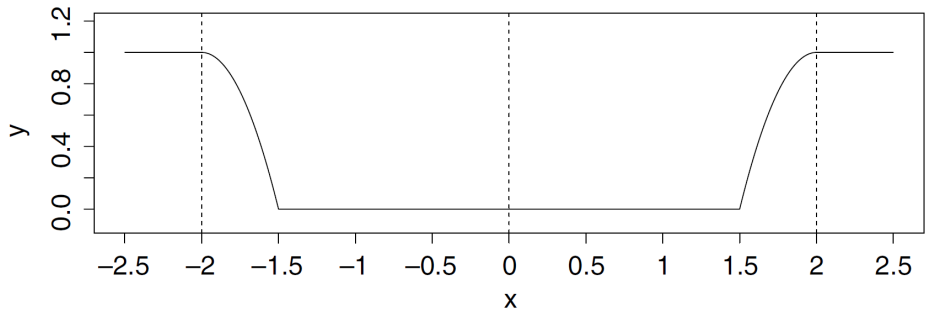
Yang, Y. (2004). Combining forecasting procedures: some theoretical results. *Econometric Theory*, *20*, 176–222.

Figure 1: Comparing Loss Functions of s-AFTER, L_1-AFTER, and h-AFTER



This figure compares the absolute error loss (L_1-AFTER), Huber loss with $s = 1.5$ (h-AFTER) and squared error loss (s-AFTER). This is a reproduction of Figure 1 in Wei and Yang (2012).

Figure 2: $\tilde{L}_0(\cdot)$ Loss Function



This figure, a reproduction of Figure 1 in Cheng and Yang(2015), shows the $\tilde{L}_0(\cdot)$ loss function.

Figure 3: Exercise 1 Results: Cost of Estimation

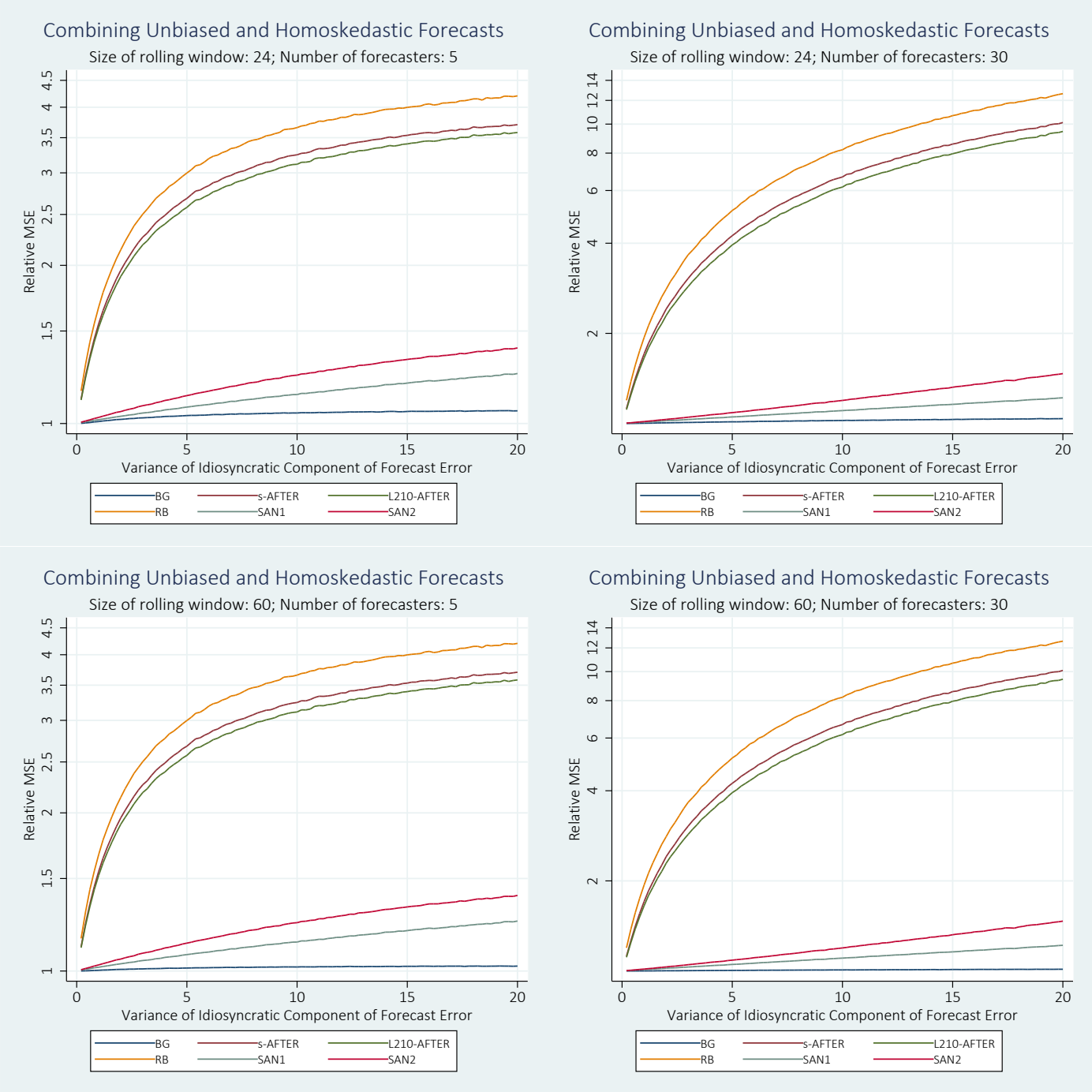


Figure 4: 12-Month Moving Average of Relative MSE: Exercise 2-3

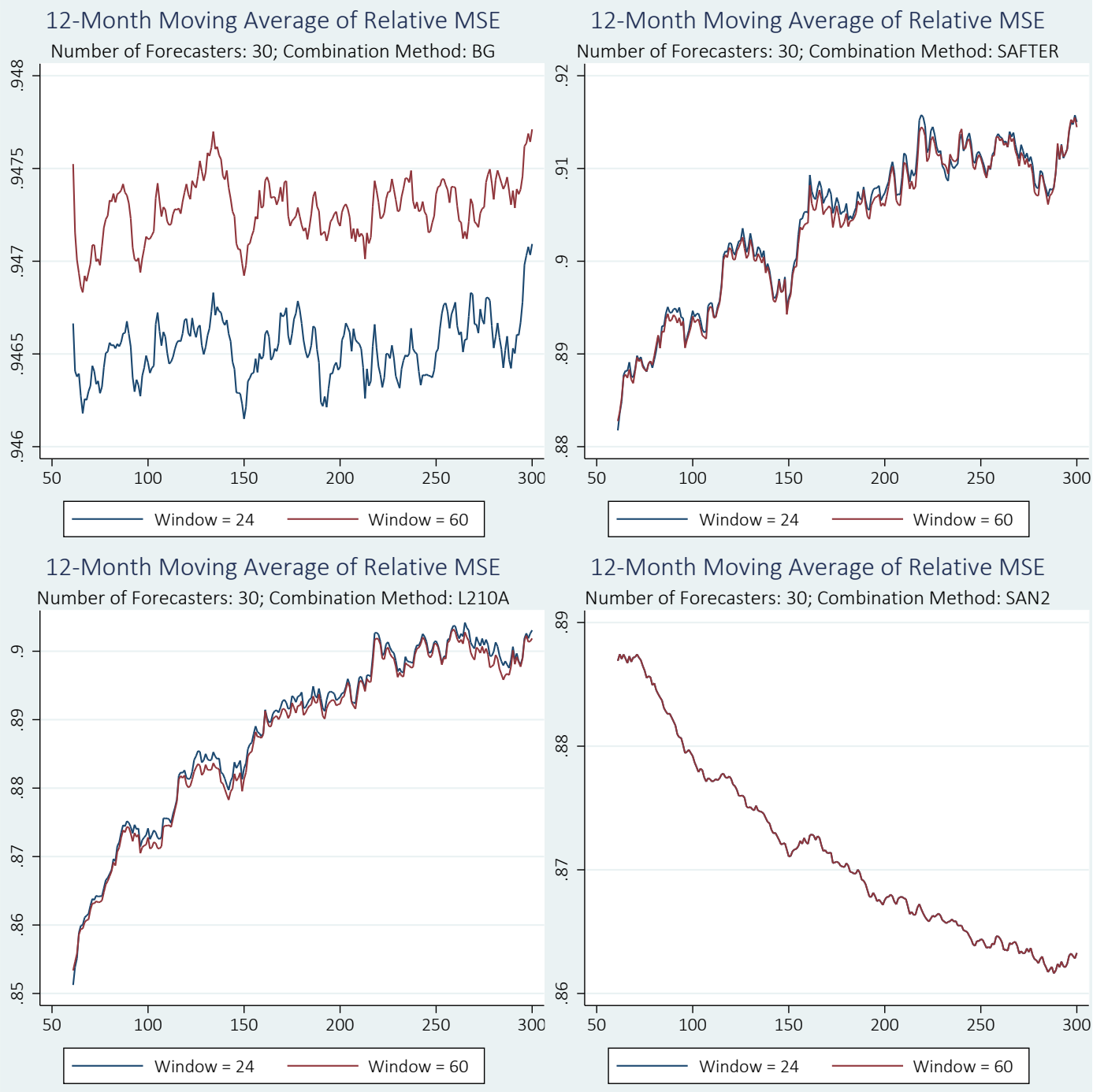


Figure 5: 12-Month Moving Average of Relative MSE: Exercise 3-3

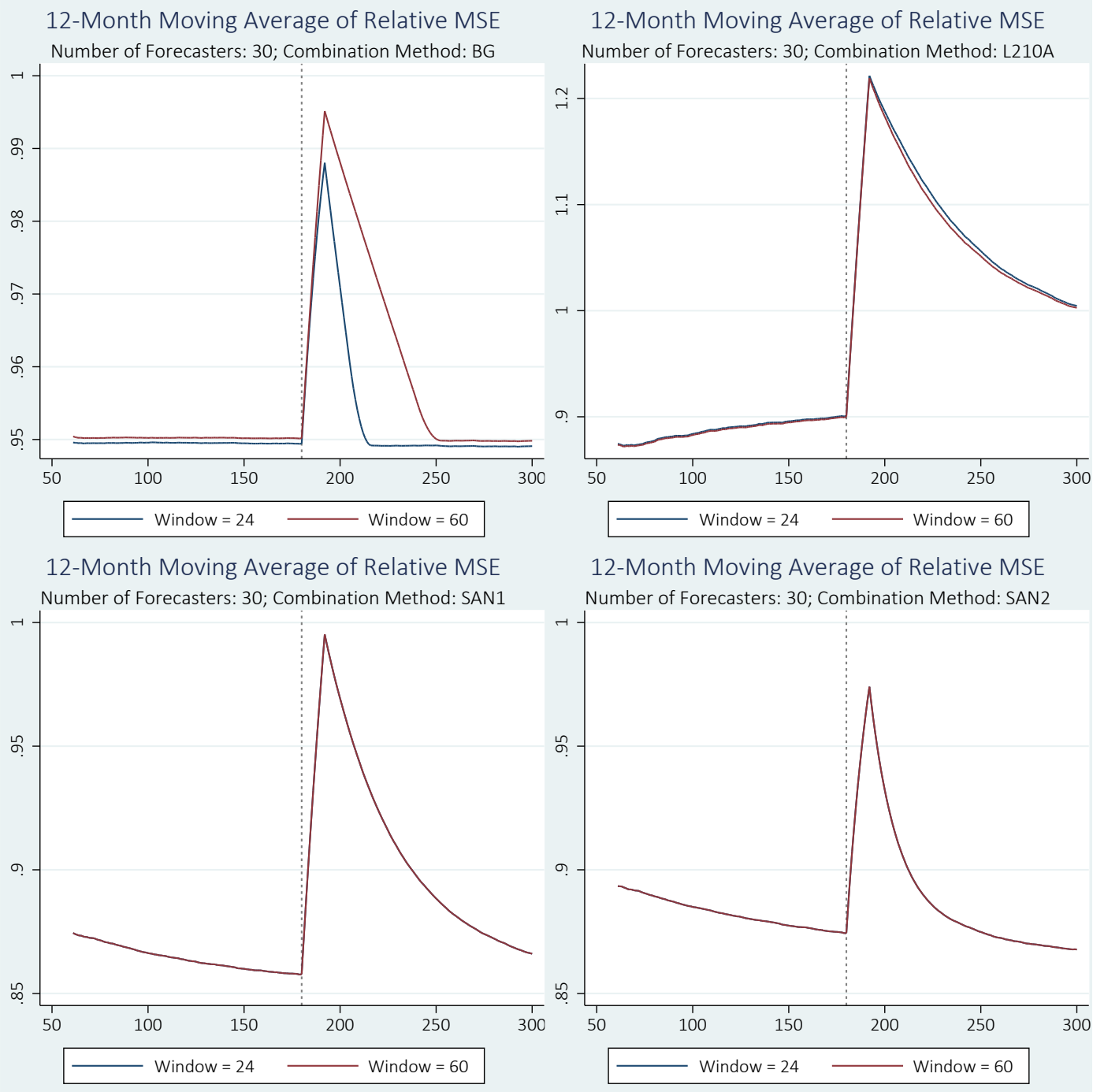


Figure 6: 12-Month Moving Average of Relative MSE: Exercise 4-4

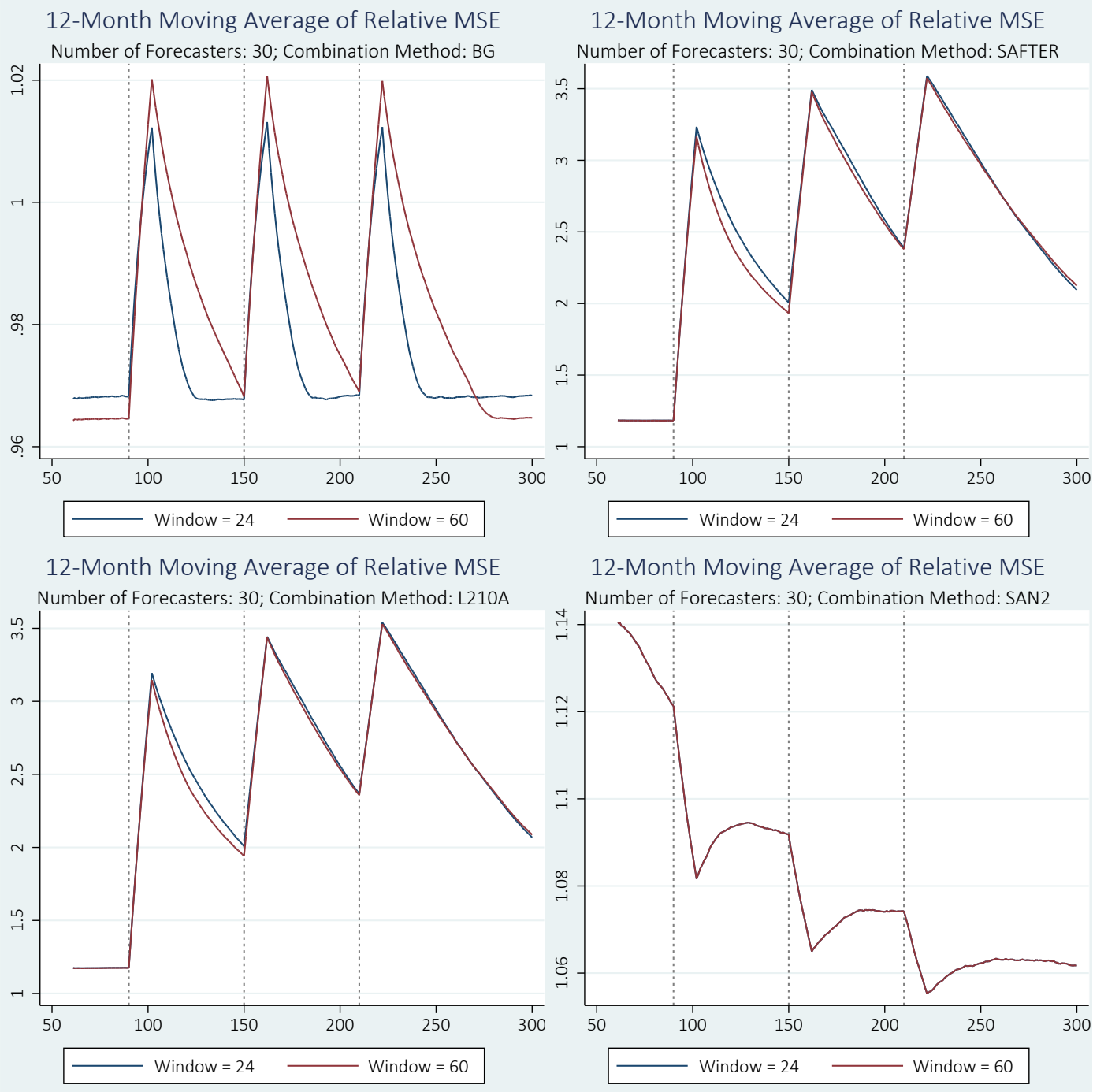


Table 1: Exercise 2 Results: Biased Forecasts in Stable Environment

Window Size and Algorithm	Exercise and Number of Forecasters E21 5	E21 30	E22 5	E22 30	E23 5	E23 30	E24 5	E24 30
24								
BG	0.991	0.977	1.022	0.998	0.964	0.949	0.994	0.974
L210A	1.348	1.402	2.212	2.931	0.968	0.898	1.504	1.592
RB	1.574	1.788	2.579	3.844	1.192	1.229	1.787	2.141
SAFTER	1.366	1.440	2.288	3.124	0.974	0.913	1.535	1.663
SAN1	0.978	0.930	1.049	1.000	0.922	0.858	1.010	0.940
SAN2	0.993	0.944	1.093	1.034	0.938	0.875	1.042	0.976
60								
BG	0.986	0.976	1.005	0.993	0.963	0.949	0.987	0.972
L210A	1.347	1.400	2.209	2.923	0.967	0.897	1.503	1.590
RB	1.574	1.788	2.579	3.844	1.192	1.229	1.787	2.141
SAFTER	1.365	1.438	2.286	3.119	0.974	0.912	1.534	1.661
SAN1	0.978	0.930	1.049	1.000	0.922	0.858	1.010	0.940
SAN2	0.993	0.944	1.093	1.034	0.938	0.875	1.042	0.976

Table 2: Exercise 3 Results: Biased Forecasts with Breaks in Performance

Window Size and Algorithm	Exercise and Number of Forecasters E31 5	E31 30	E32 5	E32 30	E33 5	E33 30	E34 5	E34 30
24								
BG	0.992	0.978	1.022	0.999	0.966	0.952	0.995	0.976
L210A	1.398	1.457	2.233	2.975	1.059	0.986	1.567	1.682
RB	1.573	1.788	2.578	3.842	1.192	1.230	1.786	2.141
SAFTER	1.417	1.494	2.310	3.170	1.065	1.002	1.600	1.754
SAN1	0.987	0.954	1.050	1.007	0.934	0.885	1.012	0.949
SAN2	0.995	0.954	1.093	1.036	0.942	0.886	1.042	0.979
60								
BG	0.988	0.979	1.006	0.994	0.968	0.956	0.989	0.976
L210A	1.397	1.453	2.231	2.967	1.058	0.984	1.566	1.679
RB	1.573	1.788	2.578	3.842	1.192	1.230	1.786	2.141
SAFTER	1.415	1.492	2.309	3.165	1.064	0.999	1.599	1.751
SAN1	0.987	0.954	1.050	1.007	0.934	0.885	1.012	0.949
SAN2	0.995	0.954	1.093	1.036	0.942	0.886	1.042	0.979

Table 3: Exercise 4 Results: Heteroskedastic Forecasts with Breaks in Performance

Window Size and Algorithm	Exercise and Number of Forecasters E41 5	E41 30	E42 5	E42 30	E43 5	E43 30	E44 5	E44 30
24								
BG	0.971	0.993	0.912	0.971	0.976	0.994	0.930	0.976
L210A	1.406	1.373	1.766	1.891	1.548	1.620	2.068	2.532
RB	1.762	2.175	2.348	3.542	1.768	2.179	2.361	3.552
SAFTER	1.412	1.377	1.776	1.884	1.561	1.638	2.088	2.557
SAN1	0.983	1.017	0.942	1.038	0.998	1.015	0.968	1.035
SAN2	0.993	1.030	0.986	1.086	1.003	1.026	0.999	1.080
60								
BG	0.968	0.992	0.911	0.971	0.981	0.995	0.950	0.984
L210A	1.403	1.365	1.761	1.852	1.546	1.615	2.060	2.502
RB	1.762	2.175	2.348	3.542	1.768	2.179	2.361	3.552
SAFTER	1.409	1.367	1.769	1.836	1.558	1.631	2.079	2.520
SAN1	0.983	1.017	0.942	1.038	0.998	1.015	0.968	1.035
SAN2	0.993	1.030	0.986	1.086	1.003	1.026	0.999	1.080

Table 4: Exercise 5 Results: Dynamically Heteroskedastic Forecasts

Window Size and Algorithm	Exercise and Number of Forecasters E51 5	E51 30	E52 5	E52 30	E53 5	E53 30	E54 5	E54 30
24								
BG	0.957	0.985	1.006	1.002	1.011	1.004	0.931	0.977
L210A	1.722	1.704	2.053	2.557	2.108	2.691	1.510	1.441
RB	2.453	3.665	2.557	3.841	2.588	3.910	2.393	3.603
SAFTER	1.738	1.718	2.098	2.661	2.146	2.779	1.522	1.450
SAN1	0.977	1.036	1.031	1.036	1.035	1.036	0.951	1.038
SAN2	1.027	1.074	1.075	1.064	1.078	1.062	1.003	1.082
60								
BG	0.944	0.981	0.990	0.997	0.995	0.998	0.919	0.973
L210A	1.721	1.700	2.051	2.553	2.106	2.686	1.509	1.439
RB	2.453	3.665	2.557	3.841	2.588	3.910	2.393	3.603
SAFTER	1.736	1.715	2.096	2.658	2.144	2.776	1.521	1.448
SAN1	0.977	1.036	1.031	1.036	1.035	1.036	0.951	1.038
SAN2	1.027	1.074	1.075	1.064	1.078	1.062	1.003	1.082

Table 5: Exercise 6 Results: Heteroskedastic Forecasts with Unexpected Aggregate Shock

Window Size and Algorithm	Exercise and Number of Forecasters E61 5	E61 30	E62 5	E62 30	E63 5	E63 30	E64 5	E64 30
24								
BG	0.938	0.982	0.974	0.994	0.952	0.987	0.986	0.997
L210A	1.217	1.109	1.121	1.052	1.180	1.085	1.082	1.033
RB	2.028	2.715	1.609	1.910	1.857	2.357	1.420	1.602
SAFTER	1.225	1.113	1.128	1.059	1.187	1.090	1.088	1.041
SAN1	0.956	1.040	0.995	1.033	0.970	1.037	1.007	1.028
SAN2	1.008	1.078	1.034	1.058	1.014	1.068	1.032	1.045
60								
BG	0.929	0.979	0.969	0.992	0.945	0.985	0.983	0.996
L210A	1.217	1.108	1.121	1.051	1.180	1.084	1.082	1.033
RB	2.028	2.715	1.609	1.910	1.857	2.357	1.420	1.602
SAFTER	1.225	1.113	1.127	1.059	1.187	1.090	1.088	1.040
SAN1	0.956	1.040	0.995	1.033	0.970	1.037	1.007	1.028
SAN2	1.008	1.078	1.034	1.058	1.014	1.068	1.032	1.045

Table 6: Exercise 7 Results: Heteroskedastic Forecasts with Outliers

Window Size and Algorithm	Exercise and Number of Forecasters E71 5	E71 30	E72 5	E72 30	E73 5	E73 30	E74 5	E74 30
24								
BG	0.891	0.921	0.953	0.982	0.801	0.898	0.890	0.958
L210A	1.413	1.186	1.617	1.368	1.095	1.068	1.308	1.180
RB	2.777	4.901	2.622	4.190	2.623	4.562	2.502	3.966
SAFTER	1.429	1.194	1.634	1.378	1.103	1.071	1.315	1.183
SAN1	0.930	1.052	0.971	1.048	0.849	1.037	0.915	1.045
SAN2	1.024	1.173	1.033	1.099	0.950	1.176	0.985	1.114
60								
BG	0.856	0.908	0.930	0.974	0.776	0.890	0.872	0.953
L210A	1.413	1.185	1.616	1.368	1.095	1.067	1.307	1.179
RB	2.777	4.901	2.622	4.190	2.623	4.562	2.502	3.966
SAFTER	1.430	1.194	1.634	1.378	1.103	1.071	1.315	1.183
SAN1	0.930	1.052	0.971	1.048	0.849	1.037	0.915	1.045
SAN2	1.024	1.173	1.033	1.099	0.950	1.176	0.985	1.114

Table 7: Combining SPF Forecasts: Relative MSEs and DM Test Results

Variable	Subsample 1: 1968:IV to 1990:IV				Subsample 2: 2000:I to 2014:IV			
and Method	h=0	h=1	h=2	h=3	h=0	h=1	h=2	h=3
	CPI				CPI			
BG					**0.947**	1.002	1.002	1.004
L210A					**0.734**	1.011	1.026	1.015
RB	CPI data available				**0.797**	1.046	1.012	1.073
SAFTER	for subsample 2 only				**0.697**	1.028	1.028	1.017
SAN1					**0.870**	1.016	1.007	1.015
SAN2					**0.829**	1.032	1.014	1.024
	PGDP				PGDP			
BG	**0.734**	**0.994**	**0.987**	**0.989**	1.012	1.004	1.003	**0.998**
L210A	**0.715**	**0.977**	**0.922**	**0.939**	1.066	1.031	1.020	1.010
RB	**0.831**	1.024	**0.917**	**0.896**	1.067	1.011	1.122	1.069
SAFTER	**0.717**	**0.963**	**0.911**	**0.930**	1.053	1.035	1.032	1.007
SAN1	**0.825**	**0.980**	**0.946**	**0.954**	1.009	1.004	1.005	**0.996**
SAN2	**0.869**	**0.989**	**0.950**	**0.951**	1.017	1.009	1.011	**0.995**
	RGDP				RGDP			
BG	1.004	1.007	**0.996**	**0.994**	1.001	1.005	**0.999**	**0.991**
L210A	1.042	1.044	1.001	1.016	1.015	1.021	1.010	**0.956**
RB	**0.947**	**0.986**	1.001	1.040	1.075	1.062	1.037	**0.938**
SAFTER	1.071	1.048	**0.998**	1.012	1.016	1.037	1.011	**0.945**
SAN1	1.023	1.012	1.003	**0.989**	1.004	1.013	1.003	**0.965**
SAN2	1.012	1.011	**0.995**	**0.983**	1.003	1.032	1.013	**0.967**
	UNEMP				UNEMP			
BG	1.002	1.002	**0.982**	**0.984**	**0.960**	**0.984**	**0.973**	**0.975**
L210A	**0.996**	1.001	**0.902**	**0.912**	**0.886**	**0.932**	**0.829**	**0.830**
RB	1.082	1.013	**0.946**	**0.923**	1.094	**0.867**	**0.828**	**0.770**
SAFTER	**0.992**	1.004	**0.898**	**0.909**	**0.893**	**0.933**	**0.822**	**0.820**
SAN1	**1.000**	**1.000**	**0.977**	**0.963**	**0.999**	**0.997**	**0.985**	**0.965**
SAN2	**0.999**	**0.999**	**0.959**	**0.947**	**0.997**	**0.992**	**0.967**	**0.934**