International Journal of Forecasting 34 (2018) 748–761

Contents lists available at ScienceDirect

International Journal of Forecasting

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

Improving time series forecasting: An approach combining bootstrap aggregation, clusters and exponential smoothing

Tiago Mendes Dantas, Fernando Luiz Cyrino Oliveira *

Industrial Engineering Department, Pontifical Catholic University of Rio de Janeiro, Brazil

ARTICLE INFO

Keywords:
Bagging methods
Clustering time series
Exponential smoothing
Partitioning around medoids
Variance reduction

ABSTRACT

Some recent papers have demonstrated that combining bagging (bootstrap aggregating) with exponential smoothing methods can produce highly accurate forecasts and improve the forecast accuracy relative to traditional methods. We therefore propose a new approach that combines the bagging, exponential smoothing and clustering methods. The existing methods use bagging to generate and aggregate groups of forecasts in order to reduce the variance. However, none of them consider the effect of covariance among the group of forecasts, even though it could have a dramatic impact on the variance of the group, and therefore on the forecast accuracy. The proposed approach, referred to here as Bagged.Cluster.ETS, aims to reduce the covariance effect by using partitioning around medoids (PAM) to produce clusters of similar forecasts, then selecting several forecasts from each cluster to create a group with a reduced variance. This approach was tested on various different time series sets from the M3 and CIF 2016 competitions. The empirical results have shown a substantial reduction in the forecast error, considering sMAPE and MASE.

1. Introduction

Since the pioneering works by Barnard (1963) and (Bates & Granger, 1969), the idea of combining forecasts in order to improve the forecast accuracy has been explored widely, see (Timmermann, 2006). As was stated by Elliott (2011), the main reason for combining is to use the relative variances and covariances to produce a weighted average of forecasts that minimizes the mean squared error of the forecast group.

Following this line of thought, (Bergmeir, Hyndman, & Benítez, 2016; Cordeiro & Neves, 2009) sought to improve the forecast accuracy by proposing a new way of generating forecasts using a very popular machine learning technique called bagging (bootstrap aggregating), proposed by Breiman (1996), in combination with exponential smoothing methods. The main idea was to use bagging to generate an ensemble of forecasts that is combined into a single output. As Bergmeir et al. (2016) pointed out, (Cordeiro & Neves, 2009) obtained some good results on quarterly and monthly data, but the overall results were not very promising. Thus, they proposed a new approach using a Box–Cox transformation, proposed by Box and Cox (1964), a seasonal-trend based on loess (STL) decomposition, proposed by Cleveland, Cleveland, and Terpenning (1990), and a moving block bootstrap, proposed by Lahiri (2013). The approach was tested on time series from the M3-competition, see (Makridakis & Hibon, 2000), and the results were very promising for the monthly data, but not for the quarterly and yearly data. However, the authors failed to take into consideration the fact that an ensemble generated using a bootstrap might produce very correlated forecasts, which will affect the forecast error, since the mean squared forecast error (MSFE) consists of a sum of variances and squared biases.

Inspired by these ideas, we propose an approach, referred to here as Bagged.Cluster.ETS, that uses not only bagging and exponential smoothing, but also clustering methods, in order to reduce the correlation among the ensemble. The empirical tests produce promising results,

* Corresponding author.
E-mail address: cyrino@puc-rio.br (F.L. Cyrino Oliveira).

https://doi.org/10.1016/j.ijforecast.2018.05.006

showing that the approach can generate more precise forecasts than other time series forecasting methods.

The rest of the paper is structured as follows: Section 2 provides a review of the use of bagging with time series, and explains why bagging tends to work. Section 3 presents the methodology and the specific details of the proposed approach. Section 4 presents a simulation analysis from the bias and variance perspective. Section 5 conducts experiments using data from forecasting competitions in order to investigate the number of clusters and the forecast accuracy; it also provides ex-ante forecasts using the Bagged.Cluster.ETS approach and compares its results with those of other methods. Section 6 concludes the paper and provides further insights for future research.

2. Theoretical background

The bagging method was proposed by Breiman (1996) as a clever way of improving the forecast accuracy by using multiple versions of a predictor. These versions are created using bootstrapping to resample the learning set. Although many papers have been published in the field of machine learning, only a few have used bagging to improve time series forecasting. We next provide a chronological review of relevant works that combine bagging and time series forecasting.

Inoue & Kilian (2004) presented one of the first attempts to use bagging in a time series context. Using an econometric approach, they demonstrated that bagging could lead to more accurate forecasts. Lee and Yang (2006) used bagging to model binary and quantile time series data (e.g., time series of the sign of a financial return). Their result shows a large reduction in the prediction error, but no improvement with large samples. Inoue and Kilian (2008) proposed three variants of the bagging algorithm to investigate whether including indicators of real economic activity in a U.S. consumer price inflation forecasting model could lead to lower prediction mean squared forecast error (MSFE) estimates. They demonstrated that bagging can reduce the MSFE, although they argued that the method is not the only one that is capable of doing so. Cordeiro and Neves (2009) proposed a way of combining bagging and exponential smoothing methods, and tested it using series from the M3 competition. However, the overall results were not very promising, although they had some success for quarterly and monthly data. Hillebrand and Medeiros (2010) used bagging with a log-linear model and a nonlinear specification with logistic transitions to improve the forecast accuracy for the realized volatility. They showed that the bagging log-linear model provides larger improvements in forecast accuracy. Rapach and Strauss (2010) combined bagging with a dynamic linear regression model for forecasting U.S. employment growth. They compared it with several methods of combining 30 autoregressive distributed lags, each of which has one potential predictor, and showed that the use of bagging often reduces the MSFE. Wang, Xiao, and Zhou (2012) proposed a multi-ensemble hybrid system for producing forecasts for chaotic time series using bagging, support vector machines (SVM) and artificial neural networks (ANN). They showed that the approach using bagging is capable of generating more accurate results than other ensemble methods and single-model SVM or ANN. Zontul, Aydin, Doan, Sener, and Kaynar (2013) successfully combined bagging with an algorithm called REPTree to produce forecasts of wind speed in Kirklareli (Turkey). However, the lack of other regions forms a drawback of their paper. Jin, Su, and Ullah (2014) proposed a revised version of bagging to investigate the dependency in time series data, and also demonstrate that bagging can increase the robustness of financial time series forecasts even in the presence of misspecified models. Bergmeir et al. (2016) proposed an approach for combining bagging with exponential smoothing methods and performed an extensive evaluation by making forecasts for the M3 competition data set (645 yearly, 756 quarterly and 1428 monthly time series). They demonstrated that their approach is extremely accurate, especially for monthly time series. Dantas, Cyrino Oliveira, and Repolho (2017) then applied Bergmeir et al.'s proposal in the context of air transportation demand time series, and the results outperformed the benchmarks methods. More recently, (Petropoulos, Hyndman, & Bergmeir, 2018) made a significant contribution by exploring the sources of uncertainty (model, data and parameter) in the bagging procedures applied for time series forecasting.

This work aims to present an innovative way of making forecasts, using Bergmeir et al.'s (2016) core ideas as a starting point, but going further by trying to address important aspects that have been left unattended by previous authors. The result is an approach with a higher forecast accuracy.

2.1. Why bagging tends to work

Inoue & Kilian (2004) studied the properties of the mean squared forecast error (MSFE). The MSFE can be decomposed into three terms: the variance of the real values, the squared bias of the forecasts and their variance.

$$\begin{aligned}
\text{MSFE} &= E[(y_{t+1|t} - \hat{y}_{t+1|t})^2] \\
&= E[(y_{t+1|t} - E(y_{t+1|t}))^2] + [E(y_{t+1|t}) - E(\hat{y}_{t+1|t})]^2 \\
&\quad + E[(\hat{y}_{t+1|t} - E[\hat{y}_{t+1|t}])^2] \\
&= E[(y_{t+1|t} - E(y_{t+1|t}))^2] + [E(y_{t+1|t}) - E(\hat{y}_{t+1|t})]^2 \\
&\quad + Var(\hat{y}_{t+1|t}) \\
&= E[(y_{t+1|t} - E(y_{t+1|t}))^2] + bias(\hat{y}_{t+1|t})^2 + Var(\hat{y}_{t+1|t}) \\
&= Var(y_{t+1|t}) + bias(\hat{y}_{t+1|t})^2 + Var(\hat{y}_{t+1|t})
\end{aligned} \tag{1}$$

Note that the first term, $Var(y_{t+1|t})$, cannot be controlled, and the sum of the last two is precisely the mean squared error (MSE) of the predictor. Good forecasting methods tend to have low biases and low variances, and consequently, low MSFEs.

When performing bagging, the average forecast over the bootstrap samples can be written as:

$$\tilde{y}_{t+1|t} = \frac{1}{B}\sum_{i=1}^{B} \hat{y}^*_{(i)t+1|t}, \tag{2}$$

where $\hat{y}^*_{(i)t+1|t}$ indicates the forecast for instant $t+1$ at time t, using the bootstrapped version i, and B is the total number of bootstrap samples.

The intuition on bagging states that, when one is resampling the learning set, the group of forecasts generated by the resampled versions are expected to have similar biases but a reduced variance. This is expected due to the expressions:

$$\begin{aligned} bias(\tilde{y}_{t+1|t}) &= E[\frac{1}{B}\sum_{i=1}^{B}\hat{y}^{*}_{(i)t+1|t}] - E[y_{t+1|t}] \\ &= \frac{1}{B}\sum_{i=1}^{B} bias(\hat{y}^{*}_{(i)t+1|t}) \end{aligned} \quad (3)$$

$$\begin{aligned} Var(\tilde{y}_{t+1|t}) = \frac{1}{B^2}\sum_{i=1}^{B} Var(\hat{y}^{*}_{(i)t+1|t}) \\ + \frac{1}{B^2}\sum_{i\neq i'} Cov[\hat{y}^{*}_{(i)t+1|t}, \hat{y}^{*}_{(i')t+1|t}]. \end{aligned} \quad (4)$$

When considering the bias, note that unbiased bootstrapped versions lead to an unbiased ensemble. Thus, if one has a single, relatively unbiased forecast that is generated using the original series, bagging will not help much. Regarding the variance, note that if the forecasts produced using the bootstrapped versions are approximately equal and there is no correlation among them, the variance term is reduced to:

$$Var(\tilde{y}_{t+1|t}) \approx \frac{1}{B} Var(\hat{y}^{*}_{(1)t+1|t}). \quad (5)$$

Note that a reduction of B times in the variance will have a large impact on the MSFE.

Although it is relatively reasonable to expect the variances of the forecasts using the bootstrapped versions to be similar, the assumption that there is no correlation among them is misleading. All of the previous studies have successfully applied bagging and reduced the forecasting error by reducing the variance, but have not taken the covariance effect into consideration. This means that previous authors have worked on the first term of Eq. (4) but neglected the second one. This is a gap that we try to bridge with the proposed approach.

Finally, it is worth noting that, considering static data, the correlation among the ensemble, specifically with bagging trees, is an issue that was addressed by Breiman (2001) in his seminal paper that presented the random forest method. The author's idea was to generate trees that were less correlated, resulting in a lower prediction error.

3. Methodology

The proposed approach, Bagged.Cluster.ETS, combines the bagging, cluster and exponential smoothing methods. The only related approaches are Boot.Expos, proposed by Cordeiro and Neves (2009), and Bagged.BLD.MBB.ETS, proposed by Bergmeir et al. (2016). Both articles make an implicit attempt to reduce variance through the artificial generation of new samples by bootstrapping. However, neither provided any specific treatment for the covariance effect stated in Eq. (4), nor do they provide any way to avoid the selection of very biased forecasts. Reducing both the bias and the variance is the ultimate goal, but this is not an easy task due to the bias and variance trade-off which means that the variance increases when the bias decreases and vice versa; see (Geman, Bienenstock, & Doursat, 1992).

We attempt to prevent the selection of inaccurate forecasts through the use of a validation set. Thus, the suggestion here is to use the same amount of data as is to be forecast. If there are not enough data points available, then the validation set should be equal to the frequency of the time series.

The proposal is to generate a reasonably large number of bootstrapped versions (e.g., 1,000) and aggregate only a small portion (e.g., 100) of versions, those that lead to the best forecasts (e.g., lowest MAPEs, sMAPE or MASE) in the validation set. The number of series selected is aligned with the work of Bergmeir et al. (2016) and is justifiable in terms of convergence.

We ensure a less correlated ensemble by generating clusters from the bootstrapped version series. The main reason for adopting clusters relates to the fact that cluster procedures maximize the similarity within each group and minimize it between groups. In this sense, it is expected that picking series from different clusters will lead to less correlated ensembles, and therefore less correlated forecasts.

The number of clusters, k, can be defined by the user either beforehand or afterward, depending on the technique chosen. Liao (2005) provides a comprehensive overview of clustering time series methods and applications. Essentially, the choice of a clustering method depends heavily on what the user defines as similar, which will determine the correct similarity/dissimilarity distance. Montero and Vilar (2014) provide examples of several situations in which each distance is either appropriate or not. Since we are interested only in group profiles of series (a one-to-one mapping of each pair of series), the Euclidean distance produces fast and good results, and is the method adopted in this work. Due to its velocity and robustness to outliers, we use the partitioning around medoids (PAM) cluster algorithm; see (Kaufman & Rousseeuw, 2009) for details. The number of clusters, k, is defined by the user. An automatic way of doing this involves using the silhouette information, which measures how similar an object is to its cluster; see (Rousseeuw, 1987).

Another important issue is the number of time series to pick from each cluster. For that, we decided to select time series in each cluster in proportion to the total number of them in each cluster. Thus, the number of series picked in each cluster h, n_h, can be defined as

$$n_h = \frac{N_h}{N} * n, \quad (6)$$

where n_h is the number of time series selected in each cluster h, n is the number of series to be aggregated (in our case, 100), N_h is the total number of time series in cluster h, and N is the total number of bootstrapped versions.

The proposed approach can be divided in two parts. The first part uses the exact same idea for generating the bootstrap versions as (Bergmeir et al., 2016), and can be seen in Algorithm 1. The second part performs the proposed procedure, see Algorithm 2.

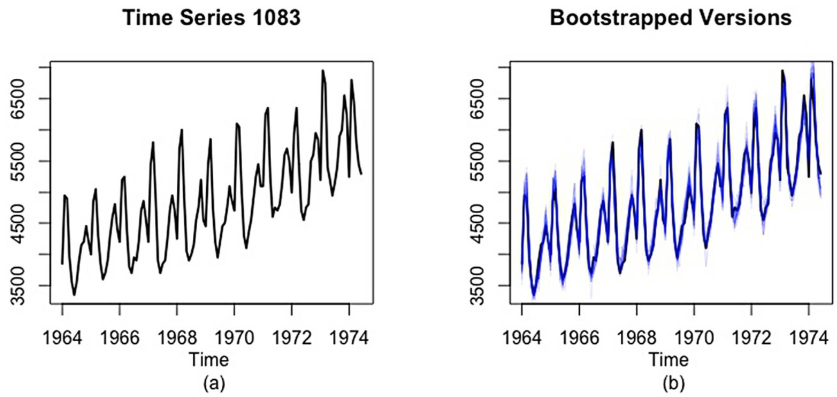

Fig. 1. Time series 1083 and bootstrapped versions.

Algorithm 1 Generating bootstrapped series

```
procedure BOOTSTRAP(ts,num.boot)
    λ ← BoxCox.lambda(ts,min=0,max=1)
    ts.bc← BoxCox(ts,λ = 1)
    if ts is seasonal then
        [trend seasonal remainder] ← stl(ts.bc)
    else
        seasonal ← 0
        [trend,remainder] ← loess(ts.bc)
    end if
     recon.series[1] ← ts
     for i in 2 to num.boot do
         boot.sample[i] ← MBB(remainder)
         recon.series.bc[i] ← trend + seasonal +boot.sample[i]
         recon.series[i] ← InvBoxCox(recon.series.bc[i],λ)
     end for
     return recon.series
 end procedure
```

Algorithm 2 Proposed approach

```
procedure PROPOSED APPROACH(recon.series,k)
    if length(forecast.period)<prop*length(recon.series) then
        pseudo.recon.series ← recon.series without the last
length(forecast.period) observations
    else
        pseudo.recon.series ← recon.series without the last
frequency(recon.series) observations
    end if
    for i in 1 to num.boot do
        pseudo.model[i] ← ets(pseudo.recon.series[i])
        pseudo.forecast[i] ← forecast(pseudo.model[i])
         pseudo.forecast.sMAPE[i] ← sMAPE(pseudo.forecast[i])
     end for
     cluster.1, cluster.2, . . . , cluster.k ← PAM(recon.series)
     for h in 1 to k do
         ensemble ← select n_h recon.series from cluster.h with lowest
rank(pseudo.forecast.sMAPE)
     end for
     for j in 1 to 100 do
         model[j] ← ets(ensemble[j])
         ensemble.forecast[j] ← forecast(model[j])
     end for
     final.forecast ← median(ensemble.forecast)
     return final.forecast
 end procedure
```

It is important to note that the value of *prop* in Algorithm 2 ranges from zero to one, and it is up to the user to decide this proportion (the forecast period). In the study, this proportion was set to 0.67.

Time series number 1083 from the M3 competition was considered to demonstrate the approach. Fig. 1 shows the time series (black) and the bootstrapped versions (blue) generated using Algorithm 1. The bootstrapped versions introduce a desirable level of noise into the series, leading to different forecasts. Fig. 2 demonstrates the application of Algorithm 2, with panel (a) displaying the forecasts generated by the series selected from each cluster (series with the same color belong to the same cluster) and the actual values (shown in black) for the out-of-sample period, while panel (b) shows the final forecast using the median and the actual values.

The next section is devoted to an investigation of the performance of the proposed approach from the perspectives of the bias and the variance using simulated data.

4. Bias and variance analysis

A simulation study was carried out with the aim of comparing the performances of Bagged.BLD.MBB.ETS and the proposed approach from the bias and variance perspectives. To achieve this, four data generation processes (DGP) were considered using Monte Carlo simulation: an autoregressive (AR), a smooth transition autoregressive (STAR), an exponential smoothing state space model (ETS) and a seasonal autoregressive integrated moving average (SARIMA). These DGPs were chosen due to their popularity and level of complexity. The parameters and hyperparameters were chosen in such a way that it would be possible to make relatively accurate forecasts. Thus, the parameters for the AR were obtained by adjusting an AR(6) to the sunspot dataset (Tong, 1990) in a way similar to that described by Taieb and Atiya (2016). The STAR parameters were chosen following those reported by Taieb and Atiya (2016). According to the authors, these settings have been subject to many simulation studies for purposes of model selection, evaluation and comparisons. The ETS was estimated by applying an ETS to the time series of accidental deaths in the US from 1973 to 1978, which was used as example by Brockwell and Davis (2013) and is included in the R forecast package for demonstrating ETS models. The parameters for the SARIMA were obtained following the steps listed by Hyndman and Athanasopoulos (2014) for obtaining the best SARIMA for the time series of monthly corticosteroid drug sales in Australia from 1992 to 2008. The DGPs are:

Table 1
Squared bias and variance: AR.

Measures	Bagged.BLD.MBB.ETS	Proposed approach					
		$k=5$	$k=10$	$k=20$	$k=40$	$k=80$	Automatic
Squared bias	24.24961	**23.95768**	24.60746	25.50372	25.42617	24.08474	24.48792
Variance	1878.05877	1796.92845	1805.87991	**1800.33918**	1812.08169	1869.32690	1836.71178

Table 2
Squared bias and variance: STAR.

Measures	Bagged.BLD.MBB.ETS	Proposed approach					
		$k=5$	$k=10$	$k=20$	$k=40$	$k=80$	Automatic
Squared bias	**0.00188**	0.00214	0.00215	0.00190	0.00194	0.00201	0.00198
Variance	6.00350	**5.55466**	5.65972	5.58463	5.72152	5.79445	5.69631

Table 3
Squared bias and variance: ETS.

Measures	Bagged.BLD.MBB.ETS	Proposed approach					
		$k=5$	$k=10$	$k=20$	$k=40$	$k=80$	Automatic
Squared bias	**172.29797**	183.12050	183.40112	176.13825	181.46536	179.87989	173.58747
Variance	41154.93728	39434.44026	39456.12883	**39401.36690**	39572.46175	40531.28877	40429.45484

Table 4
Squared bias and variance: SARIMA.

Measures	Bagged.BLD.MBB.ETS	Proposed approach					
		$k=5$	$k=10$	$k=20$	$k=40$	$k=80$	Automatic
Squared bias	3.66877e−05	3.49816e−05	3.51077e−05	3.57547e−05	**3.4729e−05**	3.58039e−05	3.5621e−05
Variance	0.00387	**0.00357**	0.00358	0.00361	0.00361	0.00369	0.00368

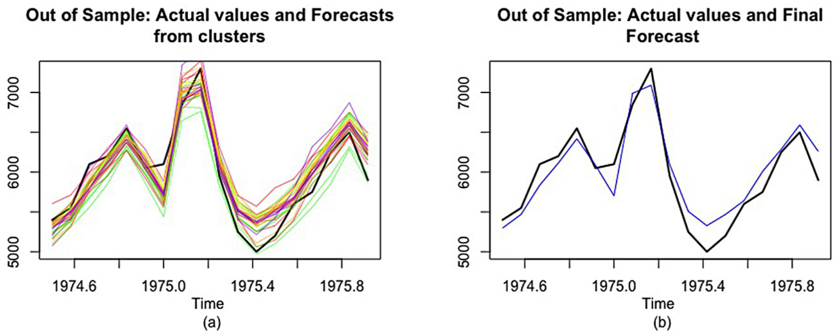

Fig. 2. Forecasts for time series 1083.

- Autoregressive: AR(6)

$$y_t = 100 + 1.2401y_{t-1} - 0.419y_{t-2} - 0.1797y_{t-3} + 0.1267y_{t-4} - 0.2259y_{t-5} + 0.1697y_{t-6} + \epsilon_t, \tag{7}$$

where ϵ_t is independently and identically distributed (i.i.d.), $N(0, 17.28)$.

- Smooth transition autoregressive: STAR

$$y_t = 500 + 0.3y_{t-1} + 0.6y_{t-1} + (0.1 - 0.9y_{t-1} + 0.8y_{t-2})[1 + e^{-10y_{t-1}}]^{-1} + \epsilon_t, \tag{8}$$

where ϵ_t is independently and identically distributed (i.i.d.), $N(0, 1)$.

- Exponential smoothing state space model: ETS (A,N,A)

$$\begin{aligned} y_t &= l_{t-1} + s_{t-m} + \epsilon_t \\ l_t &= l_{t-1} + 0.5891\epsilon_t \\ s_t &= s_{t-m} + 0.001\epsilon_t, \end{aligned} \tag{9}$$

where ϵ_t is independently and identically distributed (i.i.d.), $N(0, 264.75)$.

- Seasonal autoregressive integrated moving average: SARIMA$(3, 0, 1)(0, 1, 2)_{12}$

$$\begin{aligned} y_t = y_{t-12} &+ 0.1603(y_{t-1} - y_{t-13}) \\ &-0.5481(y_{t-2} - y_{t-14}) \\ &-0.5678(y_{t-3} - y_{t-15}) \\ &+\epsilon_t - 0.5222\epsilon_{t-12} - 0.1768\epsilon_{t-24} \\ &+0.3827\epsilon_{t-1} - 0.1998459\epsilon_{t-13} \\ &-0.06766136\epsilon_{t-25}, \end{aligned} \tag{10}$$

where ϵ_t is independently and identically distributed (i.i.d.), $N(0, 0.06)$.

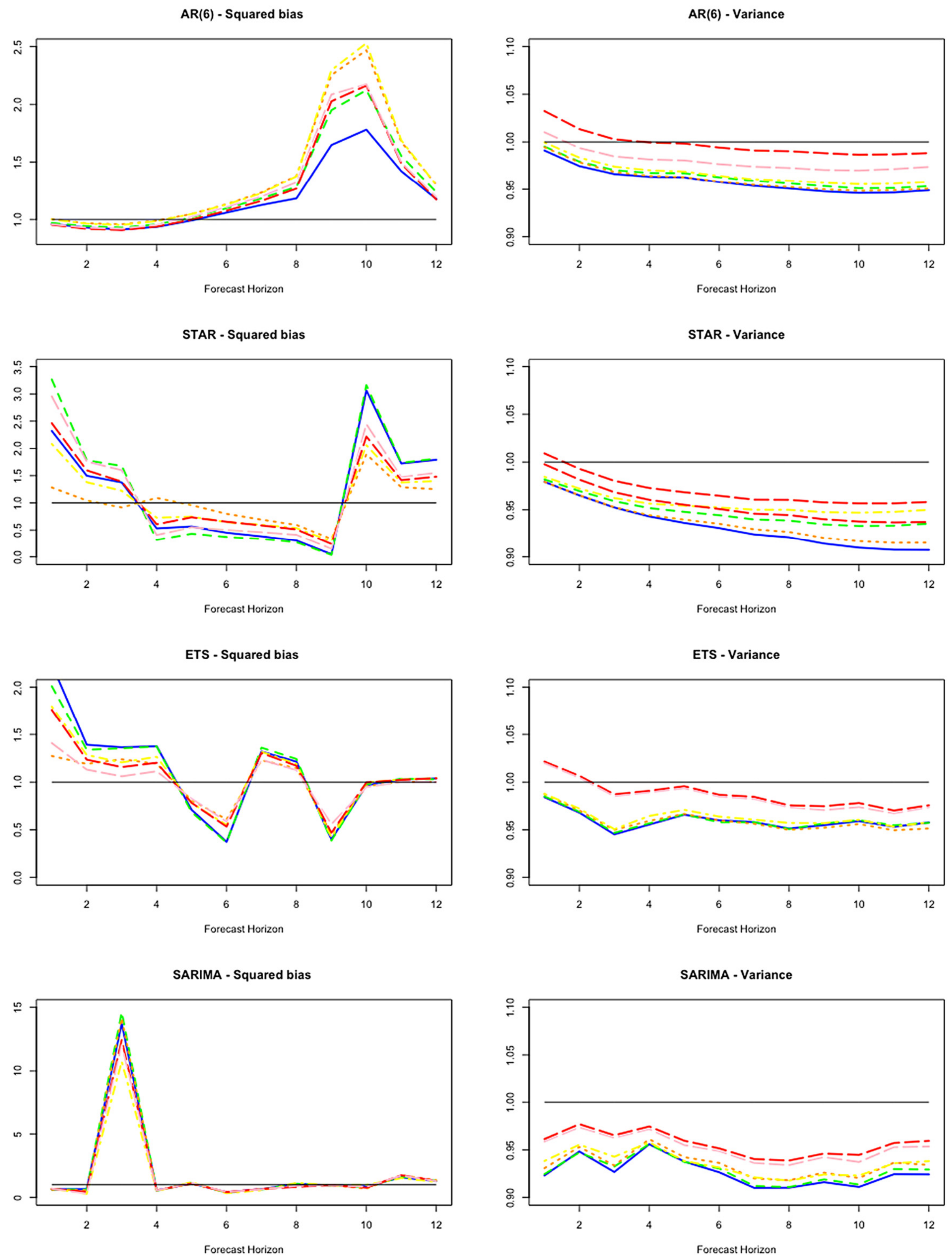

Fig. 3. Bias and variance from the various DGPs. Bagged.BLD.MBB.ETS (benchmark) is the black line. The proposed approach with 5, 10, 20, 40 and 80 clusters and the automatic selection are shown in blue, green, orange, yellow, red and pink, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

A total of 1,000 time series of length 100 were generated for each DGP. The first 88 observations were used as the training set and the last 12 as the test set on which the bias and variance were calculated. The Bagged.BLD.MBB.ETS and the proposed approach with 5, 10, 20, 40, 80 clusters and an automatic selection of clusters using the silhouette

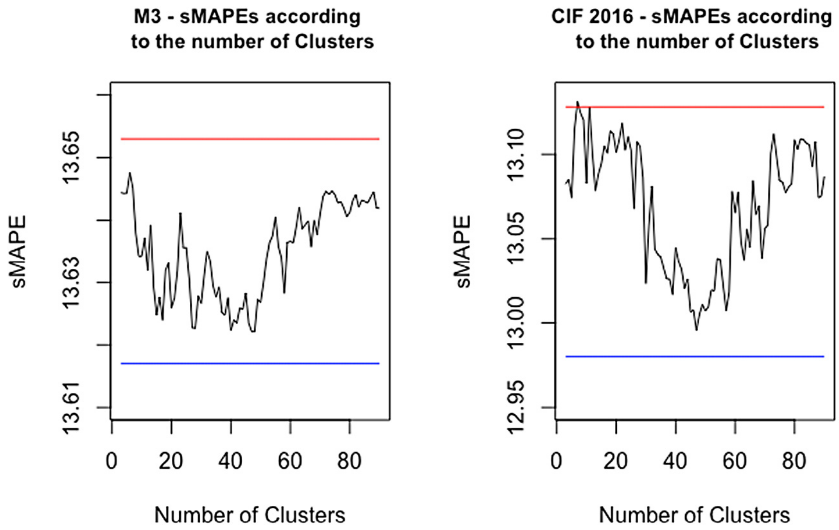

Fig. 4. sMAPE values based on the number of clusters in the M3 and CIF 2016 competitions. The black, blue and red lines represent the proposed approach, the proposed approach with the automatic selection of clusters and Bagged.BLD.MBB.ETS, respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

information are adjusted to each time series, thus enabling us to calculate the biases and variances of the forecasts.

The bias and variance ratios between the proposed approach and Bagged.BLD.MBB.ETS were also calculated and can be seen in Fig. 3, where the black lines represent the Bagged.BLD.MBB.ETS and can be considered as the benchmark. All other lines were generated using the proposed approach, varying only the number of clusters, with the blue, green, orange, yellow, red and pink lines being the approach with 5, 10, 20, 40, 80 clusters and the automatic selection, respectively. Values smaller (greater) than 1 indicate that the proposed approach has higher (lower) values for the bias or variance.

Considering the squared bias, the proposed approach seems to oscillate between values above and below 1 for all DGPs, indicating no consistent behavior (see Fig. 3). The proposed approach obtained smaller values for the mean of the squared biases than Bagged.BLD.MBB.ETS for AR with 5 and 80 clusters and SARIMA considering all variations of clusters, but the bias was higher in all other cases. On the other hand, the variance was reduced for all DGPs, see Tables 1–4. It is important to note that the variance reduction is significantly higher than the increase in bias in all cases, which is a desirable effect that leads to better forecasts.

The simulation indicates that the proposed approach sometimes increases the bias, but generally reduces the variance. In the simulation, this trade-off weights more on the variance side due to its magnitude. The ideal number of clusters seems to vary depending on the DGP, but appears to favor up to 40 clusters.

The next section takes a closer look at the effects of the number of clusters on the forecast accuracy using data from forecasting competitions, then produces forecasts using the proposed approach and compares the results ex ante.

5. Experiments on forecasting competition data

We use databases from the M3 and CIF 2016 competitions to both investigate the effect of the number of clusters on the forecast accuracy and produce forecasts which we compare with those from other methods. The first database is very comprehensive, with 1428 monthly, 756 quarterly and 645 yearly time series from several fields (e.g. macroeconomics, demographics and industry, among others), and is one of the main references for evaluating and comparing time series forecasting methods; see (Makridakis & Hibon, 2000). The CIF 2016 database contains 72 monthly time series, of which 24 are real time series from the banking domain and 48 are generated artificially; see (Štěpnička & Burda, 2017).

5.1. Effects on the number of clusters

The effect of the number of clusters in the Bagged.Cluster.ETS was investigated using the monthly data from M3 and CIF 2016. The approach involved generating 1,000 bootstrapped versions using Algorithm 1 and selecting only 100 to be averaged from the clusters using Algorithm 2. The number of clusters considered varied from 3 to 90.

Fig. 4 shows the behavior of sMAPE using the proposed approach when the number of clusters varies from 3 to 90 (lines in black), as well as the sMAPE values for Bagged.BLD.MBB.ETS (lines in red) and for the proposed approach with automatic selection (lines in blue). The results show that the ideal number of clusters in both the M3 and CIF 2016 competitions is between 40 and 50.

Since each cluster has to have at least one time series selected, the results show that creating only a few clusters does not allow the procedure to diversify the series selected, as the majority of the series are part of the same cluster. However, creating a large number of clusters forces the procedure to select ones that are not too distant,

Table 5
Comparison of the methodologies: M3 monthly data.

Methods	Rank sMAPE	Mean sMAPE	Median sMAPE	Rank MASE	Mean MASE	Median MASE
Proposed approach	**11.553**	**13.617**	**8.738**	**11.558**	**0.835**	**0.685**
Bagged.BLD.MBB.ETS	11.709	13.653	8.848	11.737	0.837	0.691
THETA	11.983	13.892	8.925	11.930	0.858	0.706
ForecastPro	12.000	13.898	8.809	12.023	0.848	0.702
COMB S-H-D	13.028	14.466	9.374	13.095	0.896	0.736
ETS	13.056	14.135	9.073	13.074	0.865	0.716
ForcX	13.260	14.466	9.212	13.314	0.894	0.741
HOLT	13.288	15.795	9.281	13.243	0.909	0.730
WINTER	13.582	15.926	9.305	13.575	1.165	0.735
RBF	13.808	14.760	9.209	13.840	0.910	0.762
DAMPEN	14.006	14.576	9.441	14.088	0.908	0.750
AAM1	14.009	15.670	9.675	13.841	0.905	0.769
AutoBox2	14.151	15.731	9.282	14.209	1.082	0.758
B-J auto	14.220	14.796	9.320	14.239	0.914	0.749
AutoBox1	14.250	15.811	9.268	14.268	0.924	0.748
SMARTFCS	14.374	15.007	9.517	14.283	0.919	0.749
AAM2	14.388	15.938	9.621	14.184	0.923	0.779
Flors-Pearc2	14.414	15.186	9.614	14.474	0.950	0.790
Auto-ANN	14.483	15.031	9.616	14.505	0.928	0.778
PP-Autocast	14.699	15.328	9.897	14.783	0.994	0.759
ARARMA	14.743	15.826	9.800	14.774	0.907	0.777
AutoBox3	14.800	16.590	9.397	14.697	0.962	0.775
Flors-Pearc1	15.126	15.986	9.959	15.159	1.008	0.797
THETAsm	15.177	15.380	9.650	15.176	0.950	0.771
ROBUST-Trend	15.372	18.931	9.733	15.293	1.039	0.830
SINGLE	15.834	15.300	10.028	15.919	0.974	0.810
NAIVE2	16.687	16.891	10.115	16.721	1.037	0.838

since the clusters are not that different from each other. Therefore, it is not surprising that having an equilibrium between the number of clusters and the number of time series (in this case, the number of clusters is between 40% and 50% of the number of series) leads to smaller forecast errors.

5.2. Performance of the proposed approach on competition data (ex-ante analysis)

In both the M3 and CIF 2016 datasets, the time series were divided into training and test sets, where the latter was used only for evaluating the forecasting performances (out of sample). The proposed method was compared with the original M3 and CIF 2016 competition methods, as well as with the method proposed by Bergmeir et al. (2016).

The methods were evaluated using the symmetric mean absolute percentage error (sMAPE) and the mean absolute scaled error (MASE). The former was used to classify the methods in both the M3 and CIF 2016 competitions, and has many advantages due to both its scale-free nature, which allows the results of time series with different scales to be compared, and the symmetric penalties that it gives to negative and positive values. On the other hand, the latter metric has many desirable properties, such as better interpretation relative to sMAPE; see (Hyndman & Koehler, 2006). The results were summarized in six columns: mean of the sMAPE ranks (Rank sMAPE) of each series, mean of the sMAPEs (Mean sMAPE), median of the sMAPEs (Median sMAPE), mean of the MASE ranks (Rank MASE) of each series, mean of the MASEs (Mean MASE) and median of the MASEs (Median MASE). The inclusion of the median in the analysis is justifiable due to the asymmetric distribution of

Table 6
Friedman rank-sum test: M3 monthly data.

Hypothesis	Adjusted p-value
Proposed approach	–
Bagged.BLD.MBB.ETS	0.599
THETA	0.147
ForecastPro	0.132
COMB S-H-D	6.791E−7
ETS	4.159E−7
ForcX	8.991E−9
HOLT	5.125E−9
WINTER	8.304E−12
RBF	3.114E−14
DAMPEN	1.449E−16
AAM1	1.325E−16
AutoBox2	2.205E−18
B-J auto	2.744E−19
AutoBox1	1.064E−19
SMARTFCS	2.108E−21
AAM2	1.339E−21
Flors-Pearc2	5.817E−22
Auto-ANN	5.838E−23
PP-Autocast	3.198E−26
ARARMA	6.636E−27
AutoBox3	8.026E−28
THETAsm	3.062E−34
Flors-Pearc1	2.498E−33
ROBUST-Trend	7.661E−38
SINGLE	4.242E−47
NAIVE2	6.212E−67

Table 7
Comparison of the methodologies: M3 quarterly data.

Methods	Rank sMAPE	Mean sMAPE	Median sMAPE	Rank MASE	Mean MASE	Median MASE
THETA	**11.817**	**8.956**	5.369	**11.821**	**1.087**	**0.774**
COMB S-H-D	12.620	9.216	5.315	12.614	1.105	0.817
ROBUST-Trend	12.915	9.789	**5.000**	12.944	1.152	0.823
DAMPEN	13.109	9.361	5.586	13.101	1.126	0.839
PP-Autocast	13.272	9.395	5.256	13.278	1.128	0.825
ForcX	13.349	9.537	5.620	13.339	1.155	0.810
Bagged.BLD.MBB.ETS	13.464	9.803	5.810	13.464	1.163	0.855
B-J auto	13.655	10.260	5.685	13.655	1.188	0.880
ETS	13.717	9.605	5.761	13.687	1.186	0.872
ForecastPro	13.729	9.815	5.837	13.763	1.204	0.853
Proposed approach	13.742	9.891	5.817	13.684	1.171	0.862
HOLT	13.771	10.938	5.711	13.731	1.225	0.861
RBF	13.796	9.565	5.665	13.757	1.173	0.847
AutoBox2	13.871	10.004	5.595	13.906	1.185	0.85
WINTER	13.895	10.840	5.710	13.871	1.217	0.874
Flors-Pearc1	13.988	9.954	5.612	14.007	1.184	0.844
ARARMA	14.005	10.186	6.108	13.975	1.185	0.860
Auto-ANN	14.416	10.199	6.282	14.444	1.241	0.923
THETAsm	14.705	9.821	5.647	14.683	1.211	0.942
AAM1	14.798	10.165	6.365	14.852	1.240	0.944
SMARTFCS	14.813	10.153	5.708	14.855	1.226	0.858
Flors-Pearc2	14.832	10.431	6.220	14.913	1.255	0.925
AutoBox3	14.931	11.192	6.150	14.882	1.272	0.921
AAM2	14.966	10.260	6.443	15.017	1.256	0.956
SINGLE	15.203	9.717	6.184	15.151	1.229	0.980
AutoBox1	15.257	10.961	6.145	15.278	1.331	0.957
NAIVE2	15.362	9.951	6.184	15.328	1.238	0.985

Table 8
Comparison of the methodologies: M3 yearly data.

Methods	Rank sMAPE	Mean sMAPE	Median sMAPE	Rank MASE	Mean MASE	Median MASE
ForcX	**11.596**	16.480	11.337	**11.567**	2.769	1.809
RBF	11.929	**16.424**	10.738	11.947	2.720	1.902
AutoBox2	11.953	16.593	11.309	11.970	2.754	1.835
Flors-Pearc1	12.044	17.205	**10.724**	12.054	2.938	1.914
THETA	12.068	16.974	11.252	12.112	2.806	1.971
ForecastPro	12.238	17.271	11.049	12.253	3.026	1.886
ROBUST-Trend	12.302	17.033	11.298	12.347	**2.625**	1.887
PP-Autocast	12.366	17.128	10.825	12.360	3.016	1.919
Bagged.BLD.MBB.ETS	12.402	17.397	11.200	12.422	2.891	2.000
DAMPEN	12.426	17.36	10.948	12.416	3.032	1.911
COMB S-H-D	12.499	17.072	11.682	12.454	2.876	1.950
ETS	12.535	17.114	11.535	12.576	2.893	2.011
Proposed approach	12.727	17.560	11.417	12.715	2.931	1.978
SMARTFCS	12.901	17.706	11.834	12.922	2.996	2.095
HOLT	13.160	20.021	11.766	13.174	3.182	2.079
WINTER	13.160	20.021	11.766	13.174	3.182	2.079
Flors-Pearc2	13.556	17.843	12.548	13.584	3.016	2.189
B-J auto	13.572	17.726	11.699	13.578	3.165	1.918
ARARMA	13.595	18.356	11.353	13.688	3.481	1.933
Auto-ANN	13.891	18.565	13.079	13.865	3.058	2.112
AutoBox3	14.091	20.877	12.891	14.078	3.177	2.232
THETAsm	14.116	17.922	12.215	14.036	3.006	2.179
AutoBox1	14.395	21.588	12.747	14.401	3.679	2.256
NAIVE2	14.712	17.880	12.369	14.629	3.172	2.267
SINGLE	14.766	17.817	12.445	14.674	3.171	2.262

the sMAPEs and MASEs. Following (Bergmeir et al., 2016), the first column was used to sort the results.

We used the Friedman rank-sum test with the post-hoc procedure of Hochberg and Rom to test whether the

Table 9
Friedman rank-sum test: M3 quarterly and yearly data.

Quarterly		Yearly	
Hypothesis	Adjusted p-value	Hypothesis	Adjusted p-value
THETA	–	ForcX	–
COMB S-H-D	0.049	RBF	0.417
ROBUST-Trend	0.007	AutoBox2	0.384
DAMPEN	0.002	Flors-Pearc1	0.274
PP-Autocast	3.674E−4	THETA	0.249
ForcX	1.754E−4	ForecastPro	0.117
Bagged.BLD.MBB.ETS	5.486E−5	ROBUST-Trend	0.085
B-J auto	6.729E−6	PP-Autocast	0.060
ETS	3.276E−6	Bagged.BLD.MBB.ETS	0.049
ForecastPro	2.842E−6	DAMPEN	0.043
Proposed approach	2.425E−6	COMB S-H-D	0.027
HOLT	1.719E−6	ETS	0.022
RBF	1.252E−6	Proposed approach	0.006
AutoBox2	4.900E−7	SMARTFCS	0.001
WINTER	3.609E−7	WINTER	1.350E−4
Flors-Pearc1	1.055E−7	HOLT	1.350E−4
ARARMA	8.365E−8	Flors-Pearc2	1.738E−6
Auto-ANN	1.951E−10	B-J auto	1.425E−6
THETAsm	1.515E−12	ARARMA	1.070E−6
AAM1	2.880E−13	Auto-ANN	2.134E−8
SMARTFCS	2.156E−13	AutoBox3	1.151E−9
Flors-Pearc2	1.535E−13	THETAsm	7.782E−10
AutoBox3	2.401E−14	AutoBox1	8.589E−12
AAM2	1.245E−14	NAIVE2	2.916E−14
SINGLE	1.105E−16	SINGLE	1.039E−14
AutoBox1	3.583E−17		
NAIVE2	3.897E−18		

Table 10
Percentage of time series in which the forecasts from the proposed method had a smaller variance than those from Bagged.BLD.MBB.ETS.

h	Monthly	Quarterly	Yearly
1	50.98	49.47	45.43
2	53.15	50.40	48.06
3	53.15	49.47	47.60
4	53.99	47.22	47.75
5	53.78	46.96	48.68
6	55.25	48.81	48.53
7	56.93	48.54	–
8	58.26	48.15	–
9	55.39	–	–
10	53.99	–	–
11	53.01	–	–
12	51.54	–	–
13	52.59	–	–
14	52.94	–	–
15	53.99	–	–
16	53.29	–	–
17	53.99	–	–
18	54.97	–	–

proposed approach was statistically different from the other methods considered in this study; see (García, Fernández, Luengo, & Herrera, 2010) for details of the procedure and its implementation. Thus, the differences between the sMAPEs of the best method, according to Rank sMAPE, and each method considered were tested for statistical significance.

The experiment was conducted using R and the forecast package (version 8.0). The results for Bagged.BLD.MBB.ETS were obtained using the baggedETS function, which improved the results for most of the cases relative to those presented by Bergmeir et al. (2016).

5.2.1. M3: monthly results

The performances of the various methods on the M3 competition series demonstrates the superiority of our approach over the other 25 benchmarks for monthly time series, considering all metrics; see Table 5. The overall p-value of the Friedman rank-sum test, 3.15×10^{-10}, also shows that there are statistically significant differences among the results. Considering the proposed approach as the control method, the adjusted p-values from the post-hoc procedure shows that there are statistically significant differences between the results from the approach and all other methods at $\alpha = 5\%$, with the exception of Bagged.BLD.MBB.ETS, THETA and Forecast Pro; see Table 6.

5.2.2. M3: quarterly and yearly results

Similarly to the results of Bergmeir et al. (2016), the forecasts for the quarterly and yearly time series indicate a severe decline in the performance of the proposed method, with the THETA method obtaining the best results for quarterly data on all metrics except for the Median MASE; see Table 7. Considering yearly data, the ForcX method obtained the best results of all methods in the study according to Rank sMAPE, Rank MASE and Median MASE. On the other metrics, RBF obtained the best results according to the Mean sMAPE, Flors-Pearc1 got the most accurate results according to the Median sMAPE, and ROBUST-Trend obtained the best results according to the Mean MASE; see Table 8. The proposed approach performed poorly at both frequencies, producing worse results than Bagged.BLD.MBB.ETS and even than ETS.

The overall p-values of the Friedman rank-sum test for quarterly and yearly data, 9.54×10^{-11} and 9.70×10^{-11}, respectively, show that there are statistically significant differences among the methods at both frequencies. Of these

Table 11
CIF 2016: comparison of the methodologies on the artificial series.

Methods	Rank sMAPE	Mean sMAPE	Median sMAPE	Rank MASE	Mean MASE	Median MASE
Proposed approach	**6.896**	**6.308**	5.078	**6.792**	**0.964**	0.559
Bagged.BLD.MBB.ETS	7.146	6.319	**5.000**	6.938	0.968	**0.552**
Ensemble of LSTMs and ETS	8.604	6.706	5.373	8.604	0.979	0.640
ETS	8.771	6.615	5.340	8.646	1.003	0.573
FRBE	9.208	7.024	5.375	9.312	1.064	0.630
LSTM deseasonalized	9.312	6.710	5.235	9.250	0.986	0.662
Boot.EXPOS	9.729	6.904	5.496	9.688	1.054	0.682
MLP	9.979	6.761	5.368	10.104	1.021	0.657
ARIMA	10.958	7.349	5.492	10.875	1.088	0.677
HEM	10.979	7.322	5.129	11.062	1.085	0.668
REST	11.688	7.342	6.259	11.792	1.076	0.717
PB-GRNN	11.917	7.754	5.591	11.917	1.140	0.711
PB-RF	11.917	7.754	5.591	11.917	1.140	0.711
PB-MLP	12.229	7.718	5.649	12.333	1.133	0.692
AVG	13.042	8.208	6.642	13.000	1.236	0.85
LSTM	13.479	7.795	6.619	13.458	1.124	0.835
MTSFA	14.292	9.464	6.414	14.312	1.333	0.775
Fuzzy c-regression m	14.625	9.588	7.274	14.667	1.430	1.219
FCDNN	15.646	8.587	7.475	15.625	1.259	0.833
Random walk	17.062	10.69	8.855	17.125	1.621	1.195
THETA	17.750	10.834	8.790	17.917	1.604	1.396
TSFIS	17.833	10.697	9.489	17.792	1.625	1.279
HFM	19.021	14.564	9.607	18.979	3.675	1.293
MSAKAF	19.229	14.634	12.840	19.208	2.003	1.568
CORN	23.688	19.327	18.867	23.688	2.758	2.349

Table 12
CIF 2016: comparison of the methodologies on the real series.

Methods	Rank sMAPE	Mean sMAPE	Median sMAPE	Rank MASE	Mean MASE	Median MASE
Ensemble of LSTMs and ETS	**8.292**	**19.090**	**14.649**	**8.167**	**0.424**	**0.321**
LSTM deseasonalized	9.125	18.178	15.735	8.958	0.494	0.327
MLP	10.583	22.882	23.514	10.583	0.491	0.343
Fuzzy c-regression m	10.750	22.010	20.041	10.750	0.521	0.353
REST	10.917	22.654	19.696	11.208	0.541	0.409
TSFIS	11.062	23.928	20.880	11.021	0.555	0.449
AVG	11.458	22.746	19.343	11.542	0.520	0.377
Random walk	11.792	22.379	17.451	11.917	0.526	0.419
ETS	11.833	22.409	18.732	11.708	0.513	0.391
Proposed approach	12.000	26.325	20.141	11.938	0.550	0.376
HEM	12.042	24.466	20.777	12.083	0.532	0.371
FRBE	12.250	24.667	18.616	12.167	0.531	0.369
LSTM	12.958	24.414	16.683	12.708	0.593	0.344
THETA	13.125	22.599	20.776	13.208	0.546	0.320
Bagged.BLD.MBB.ETS	13.250	26.748	20.117	13.146	0.552	0.388
MSAKAF	13.625	31.891	22.721	13.708	0.709	0.569
ARIMA	13.771	28.988	21.679	13.729	0.584	0.441
PB-GRNN	13.875	27.992	24.830	14.042	0.744	0.527
PB-RF	13.875	27.992	24.830	14.042	0.744	0.527
MTSFA	14.167	30.608	27.100	13.958	0.715	0.472
PB-MLP	14.458	29.392	25.906	14.042	0.711	0.469
Boot.EXPOS	15.125	31.942	21.633	15.042	0.674	0.458
HFM	16.500	38.055	24.736	16.917	2.463	0.543
FCDNN	17.125	32.698	26.650	17.417	0.908	0.620
CORN	21.042	47.624	34.024	21.000	1.207	1.100

differences, selecting the THETA method as the control, there are statistically significant differences between it and all other methods. Considering yearly time series and using the winning method, ForcX, as the control method, the adjusted p-value indicates significant differences between ForcX and all methods but RBF, AutoBox2, Flors-Pearc1, THETA, ForecastPro, Robust-Trend and PP-Autocast; see Table 9.

Table 13
CIF 2016: comparison of the methodologies on all series.

Methods	Rank sMAPE	Mean sMAPE	Median sMAPE	Rank MASE	Mean MASE	Median MASE
Ensemble of LSTMs an ETS	**8.500**	**10.834**	6.598	**8.458**	**0.794**	0.559
Proposed approach	8.597	12.980	6.048	8.507	0.826	0.545
Bagged.BLD.MBB.ETS	9.181	13.128	**5.978**	9.007	0.829	**0.537**
LSTM deseasonalized	9.250	10.532	7.017	9.153	0.822	0.597
ETS	9.792	11.880	6.666	9.667	0.840	0.532
MLP	10.181	12.135	6.923	10.264	0.845	0.545
FRBE	10.222	12.905	6.769	10.264	0.886	0.566
HEM	11.333	13.037	7.317	11.403	0.900	0.590
REST	11.431	12.446	7.574	11.597	0.898	0.591
Boot.EXPOS	11.528	15.250	6.923	11.472	0.928	0.614
ARIMA	11.896	14.562	7.027	11.826	0.920	0.562
AVG	12.514	13.054	8.020	12.514	0.997	0.676
PB-GRNN	12.569	14.500	7.856	12.625	1.008	0.653
PB-RF	12.569	14.500	7.856	12.625	1.008	0.653
PB-MLP	12.972	14.943	8.052	12.903	0.992	0.681
LSTM	13.306	13.334	8.202	13.208	0.947	0.684
Fuzzy c-regression m	13.333	13.729	10.036	13.361	1.127	0.722
MTSFA	14.250	16.512	9.692	14.194	1.127	0.707
Random walk	15.306	14.586	9.141	15.389	1.256	0.833
TSFIS	15.576	15.107	10.183	15.535	1.269	0.911
FCDNN	16.139	16.624	8.713	16.222	1.142	0.818
THETA	16.208	14.756	11.012	16.347	1.251	0.744
MSAKAF	17.361	20.386	14.239	17.375	1.572	1.314
HFM	18.181	22.394	11.890	18.292	3.271	1.144
CORN	22.806	28.760	19.858	22.792	2.241	1.826

Table 14
Friedman rank-sum test: CIF 2016.

Hypothesis	Adjusted p-value
Ensemble of LSTMs and ETS	–
Proposed approach	0.937
Bagged.BLD.MBB.ETS	0.579
LSTM deseasonalized	0.541
ETS	0.292
MLP	0.171
FRBE	0.160
HEM	0.021
REST	0.017
Boot.EXPOS	0.014
ARIMA	0.006
AVG	0.001
PB-RF	9.080E−4
PB-GRNN	9.080E−4
PB-MLP	2.664E−4
LSTM	8.941E−5
Fuzzy c-regression m	8.137E−5
MTSFA	2.764E−6
Random walk	2.887E−8
TSFIS	7.977E−9
FCDNN	4.739E−10
THETA	3.297E−10
MSAKAF	5.051E−13
HFM	2.974E−15
CORN	1.982E−31

5.3. Discussion

The variances of the groups of bootstrapped forecasts using the proposed approach and Bagged.BLD.MBB.ETS were calculated for each period over the forecasting horizon (18 months for monthly data, eight quarters for quarterly data and six years for yearly data). The results show that, as intended, the forecasts from the proposed approach have lower variances for the entire forecasting horizon for the majority of the monthly time series. However, such is not the case for the quarterly and yearly data relative to Bagged.BLD.MBB.ETS. See Table 10.

The increase in the variance for the quarterly and yearly cases helps to explain the poor forecasting performances. One point that needs to be highlighted is that the quarterly and yearly time series are significantly shorter than the monthly series; that is, while the median length for the monthly time series is 115, those for the quarterly and yearly cases are 44 and 19, respectively.

5.4. CIF 2016

The results obtained on the M3 data indicate that the proposed approach was able to generate promising results using monthly time series. The CIF 2016 competition is a recent data set containing 72 monthly time series, making it perfect for trying out the method.

The proposed approach was able to obtain the best results for the 48 artificially generated series considering all metrics except Median sMAPE and Median MASE (see Table 11), but the same is not true for the 24 real time series in the dataset, where the ensemble of LSTMs and ETS obtained the best results considering all metrics. However, the comparison between the proposed approach and Bagged.BLD.MBB.ETS was able to generate better results considering all metrics except for Median sMAPE and Median MASE; see Table 12.

Although the top performer considering all series in the competition was the ensemble of LSTMs and ETS, the proposed approach was able to produce better results than Bagged.BLD.MBB.ETS considering Rank sMAPE, Mean sMAPE, Rank MASE and Mean MASE, see Table 13.

The overall p-value of the Friedman rank-sum test, 1.62×10^{-10}, indicates statistically significant differences among the results. The use of the ensemble of LSTMs and ETS as the control method indicates statistically significant differences among all results except for those generated by the proposed approach, Bagged.BLD.MBB.ETS, LSTM deseasonalized, ETS, MLP and FRBE. See Table 14.

The results showed that the top three performances among all the contestants were obtained by methods that combine forecasts, namely the ensemble of LSTMs and ETS, the proposed approach and the Bagged.BLD.MBB.ETS. These results confirm the reasoning explained in Section 2.1 and are related intrinsically to Eq. (4).

6. Conclusion

This paper proposes an innovative way of producing forecasts that combines the bagging, exponential smoothing and cluster methods. Its main contribution lies in the way in which the proposed approach looks at the previously neglected effect of covariance on the combination of bagging and exponential smoothing, trying to minimize it by generating clusters and selecting series from them. Doing this allows the proposed approach to reduce the forecast error.

The overall comparison on the 1428 monthly time series from the M3 competition and the 72 series from CIF 2016 showed that the proposed approach is a tough competitor, with its forecasts consistently being more accurate than those of all 25 other benchmarks in the first competition and 23 in the second, including Bagged.BLD.MBB.ETS, proposed by Bergmeir et al. (2016), in both cases.

The Bagged.Cluster.ETS and Bagged.BLD.MBB.ETS approaches are similar, but we believe that the specific attempt to reduce the covariance effect through the use of clusters is the reason for the improvement in the results. However, it is worth mentioning that this advantage can become a drawback if the time series is too short, since the algorithm for forming clusters can be affected by the length of the series.

It is important to highlight that the use of the silhouette method was necessary for automating the selection of the number of clusters for each time series, due to the large number of series considered in the study. However, a recommended approach for selecting the ideal number of clusters would be through the use of cross validation.

Although the results from the statistical tests did not show statistically significant differences between the proposed approach and some of the methods, the results from the other analyses are important findings and are still valid, as (Armstrong, 2007; Kostenko & Hyndman, 2008) pointed out. However, the existence of new competitions with more series, such as the M4 competition that is in progress with 100,000 series, could be helpful in finding statistical significance in the results.

Finally, as a methodological extension of this work, it is our intention to study other weighting schemes for selected series, as well as other decomposition and forecasting methods.

Acknowledgments

We would like to thank Dr. Christoph Bergmeir for providing code for Bagged.BLD.MBB.ETS and detailed explanations of every aspect of it. We would also like to thank the CIF 2016 organizers for providing quick answers to all our questions. Finally, we would like to thank both reviewers for their insightful comments. This work was supported by CAPES; CNPq [304843/2016-4, 443595/2014-3] and FAPERJ [E-26/202.806/2015].

References

Armstrong, J. S. (2007). Significance tests harm progress in forecasting. *International Journal of Forecasting*, *23*(2), 321–327.

Barnard, G. A. (1963). New methods of quality control. *Journal of the Royal Statistical Society, Series A (General)*, *126*(2), 255–258.

Bates, J. M., & Granger, C. W. J. (1969). The combination of forecasts. *The Journal of the Operational Research Society*, *20*(4), 451–468.

Bergmeir, C., Hyndman, R. J., & Benítez, J. M. (2016). Bagging exponential smoothing methods using STL decomposition and Box–Cox transformation. *International Journal of Forecasting*, *32*(2), 303–312.

Box, G. E., & Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, 211–252.

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

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32.

Brockwell, P. J., & Davis, R. A. (2013). *Time series: theory and methods*. Springer Science & Business Media.

Cleveland, R. B., Cleveland, W. S., & Terpenning, I. (1990). STL: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, *6*(1), 3–73.

Cordeiro, C., & Neves, M. M. (2009). Forecasting time series with Boot. EXPOS procedure. *Revstat*, *7*(2), 135–149.

Dantas, T. M., Cyrino Oliveira, F. L., & Repolho, H. M. V. (2017). Air transportation demand forecast through bagging holt winters methods. *Journal of Air Transport Management*, *59*, 116–123.

Elliott, G. (2011). *Averaging and the optimal combination of forecasts*. University of California, San Diego.

García, S., Fernández, A., Luengo, J., & Herrera, F. (2010). Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences*, *180*(10), 2044–2064.

Geman, S., Bienenstock, E., & Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, *4*(1), 1–58.

Hillebrand, E., & Medeiros, M. C. (2010). The benefits of bagging for forecast models of realized volatility. *Econometric Reviews*, *29*(5–6), 571–593.

Hyndman, R. J., & Athanasopoulos, G. (2014). *Forecasting: principles and practice*. OTexts, http://OTexts.org/fpp.

Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, *22*(4), 679–688.

Inoue, A., & Kilian, L. (2004). *Bagging time series models. Discussion paper*. London: Centre for Economic Policy Research.

Inoue, A., & Kilian, L. (2008). How useful is bagging in forecasting economic time series? A case study of US consumer price inflation. *Journal of the American Statistical Association*, *103*(482), 511–522.

Jin, S., Su, L., & Ullah, A. (2014). Robustify financial time series forecasting with bagging. *Econometric Reviews*, *33*(5–6), 575–605.

Kaufman, L., & Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons.

Kostenko, A. V., & Hyndman, R. J. (2008). *Forecasting without significance test. Manuscript*. Australia: Monash University.

Lahiri, S. N. (2013). *Resampling methods for dependent data*. Springer Science & Business Media.

Lee, T. H., & Yang, Y. (2006). Bagging binary and quantile predictors for time series. *Journal of Econometrics*, *135*(1), 465–497.

Liao, T. W. (2005). Clustering of time series data—a survey. *Pattern Recognition*, *38*(11), 1857–1874.

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

Montero, P., & Vilar, J. (2014). TSclust: An R package for time series clustering. *Journal of Statistical Software*, *62*(1), 1–43.

Petropoulos, F., Hyndman, R. J., & Bergmeir, C. (2018). Exploring the sources of uncertainty: Why does bagging for time series forecasting work? *European Journal of Operational Research*, *268*, 545–554.

Rapach, D. E., & Strauss, J. K. (2010). Bagging or combining (or both)? An analysis based on forecasting US employment growth. *Econometric Reviews*, *29*(5–6), 511–533.

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, *20*, 53–65.

Štěpnička, M., & Burda, M. (2017). On the results and observations of the time series forecasting competition CIF 2016. In *2017 IEEE international conference on fuzzy systems* (pp. 1–6). IEEE.

Taieb, S. B., & Atiya, A. F. (2016). A bias and variance analysis for multistep-ahead time series forecasting. *IEEE Transactions on Neural Networks and Learning Systems*, *27*(1), 62–76.

Timmermann, A. (2006). Forecast combinations. In G. Elliott, C. W. J. Granger, & A. Timmermann (Eds.), *Handbook of economic forecasting* (Vol. 1) (pp. 135–196). Elsevier.

Tong, H. (1990). *Non-linear time series: a dynamical system approach*. Oxford University Press.

Wang, Y., Xiao, M., & Zhou, Y. (2012). A hybrid ensemble approximation method for chaotic time series forecast. *Journal of Information and Computational Science*, *9*(18), 5849–5856.

Zontul, M., Aydin, F., Doan, G., Sener, S., & Kaynar, O. (2013). Wind speed forecasting using REPTree and bagging methods in Kirklareli-Turkey. *Journal of Theoretical and Applied Information Technology*, *56*, 17–29.

Tiago Mendes Dantas is a Statistician at the *Brazilian Institute of Geography and Statistics* (IBGE). He holds a Bachelors degree in Statistics, a master's degree in Electrical Engineering with focus on time series forecasting methods and is a Ph.D. candidate at PUC-Rio (Pontifical Catholic University of Rio de Janeiro). His research interests include time series forecasting, machine learning and statistical analysis.

Fernando Luiz Cyrino Oliveira is an Associate Professor (Senior Lecturer) in the Industrial Engineering Department of PUC-Rio (Pontifical Catholic University of Rio de Janeiro). His research interests include time series forecasting, integrated business forecasting processes, operations management and simulation methods, especially with applications in Energy and Health Care. He is member of the International Institute of Forecasters since 2010.