Expert Systems With Applications 237 (2024) 121636

Contents lists available at ScienceDirect

Expert Systems With Applications

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

An algorithm for automatic selection and combination of forecast models

Carlos García-Aroca, Mª Asunción Martínez-Mayoral, Javier Morales-Socuéllamos, José Vicente Segura-Heras *

I.U. Operations Research Center, University Miguel Hernandez of Elche, Avda. Ferrocarril s/n, 03202 Elche, Spain

ARTICLE INFO

Keywords:
Forecasting
Time series methods
Forecasting combination
Principal component analysis
Intelligent Manufacturing

ABSTRACT

In this paper, we present an algorithm designed to automatically merge predictions from a collection of individual prediction methods coded in R. The algorithm employs varying weights and decision rules to ascertain the optimal amalgamation of these methods, with the aim of forecasting historical time series data while minimizing human intervention. The algorithm serves as an automated component within the artificial intelligence toolkit.

The proposed algorithm (al), denoted as "alPCA" is founded on principal component analysis (PCA), hence the acronym. Commencing with 52 configurations of 11 distinct methods available in R, we calculate several loss functions: specifically, scaled Mean Absolute Percentage Error (sMAPE) and Mean Absolute Scaled Error (MASE) for both fitting (Training Phase) and prediction (Validation Phase), along with Root Mean Squared Error (RMSE) and Overall Weighted Average (OWA) solely for prediction (Validation Phase). We then employ PCA to reduce the error matrix derived from this data to one or two dimensions. Subsequently, the methods are ranked based on their proximity to the highest score. A probability distribution is fitted to this proximity metric, and utilizing the percentiles of these values, the optimal methods for combination are selected. We propose three categories of weights derived from the PCA scores, encompassing the fitting sMAPE (Training Phase) and the prediction sMAPE (Validation Phase), to facilitate the amalgamation process.

This approach is applied to seven distinct univariate time series across diverse domains, including automobile sales, electricity production, and CO2 levels. Additionally, a set of 100 random monthly series from the M4 competition is included in the analysis. To assess the predictive precision of our algorithm, we compare its performance against three widely utilized combined prediction algorithms available in R. We evaluate the outcomes in Test Phase (unseen data) using four distinct loss functions and conduct a sensitivity analysis to gauge the algorithm's robustness and efficacy across various specifications.

1. Introduction

Forecasting plays a crucial role as an artificial intelligence (AI) tool in various domains and industries. By leveraging AI techniques, forecasting models can analyze historical data, identify patterns and trends, and generate predictions for future events or outcomes. AI-powered forecasting can be integrated with decision support systems, providing valuable insights for strategic planning, resource allocation, inventory management, risk assessment, and other business processes. This integration enhances the effectiveness of decision-making by incorporating data-driven predictions.

In contemporary times, an array of statistical forecasting methods is at our disposal. These encompass straightforward approaches like Naïve, simple moving average, and exponential smoothing, as well as more intricate techniques such as ARIMA (integrated autoregressive moving average), TBATS (Trigonometric seasonality, Box-Cox transform, ARMA errors, Trend, and Seasonal components), and STLM (seasonal and trend decomposition using Loess). Recent years have witnessed a heightened focus on the assimilation of Artificial Intelligence techniques (Haykin, 2009), exemplified by models such as NNAR (neural network autoregression) and GRNN (generalized regression neural networks).

Given the wide spectrum of individual methods available, some researchers have delved into combining them to yield enhanced forecasts. Forecast combination, also referred to as forecast pooling, entails amalgamating two or more individual forecasts from a panel to generate a singular pooled forecast. While Bates & Granger (1969) are often credited as pioneers of this technique, earlier work by Barnard (1963) suggested employing a simple arithmetic mean of two forecasts due to its

* Corresponding author.
E-mail addresses: carlos.garcia62@goumh.umh.es (C. García-Aroca), asun.mayoral@umh.es (M. Asunción Martínez-Mayoral), j.morales@umh.es (J. Morales-Socuéllamos), jvsh@umh.es (J.V. Segura-Heras).

https://doi.org/10.1016/j.eswa.2023.121636
Received 26 June 2023; Received in revised form 13 September 2023; Accepted 13 September 2023
Available online 18 September 2023

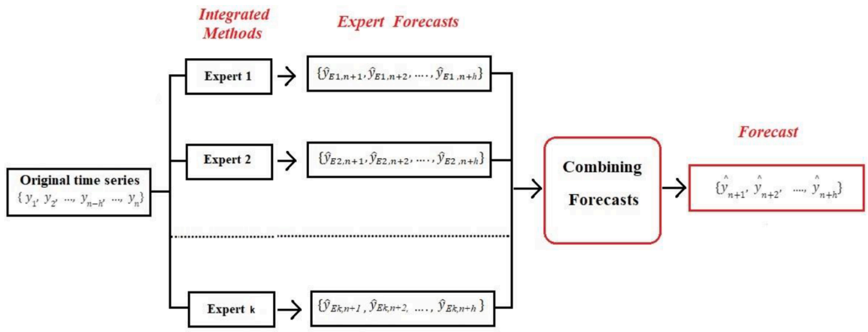

Fig. 1. Combining forecasting from integrated methods (or experts).

lower mean square error. The rationale behind this fusion of forecasts stems from the distinct approaches and characteristics each forecasting method employs, enabling the amalgamation to leverage this diversity (Armstrong, 2001a,b). As Breiman (2001) points out, averaging models can lead to variance reduction and are pivotal components in aggregation processes alongside expert selection (Brown et al., 2005).

Experimental results demonstrate that combinations based on a wide diversity of prediction methods achieve forecasts with less error (Thomson et al., 2019; Lichtendahl & Winkler, 2020). For instance, the combination of ARIMA with other methods, as used by Wang et al. (2020) to predict CO2 emissions by 2030 in China, India, and the United States. Fig. 1 illustrates a simple combination of methods or experts, where "h" signifies the number of periods constituting the prediction horizon.

The quality of the individual forecasts being combined and the estimation of combination weights assigned to each forecast are crucial factors (Timmermann, 2006; Cang & Yu, 2014). Combination schemes have evolved from simple methods without estimation to sophisticated approaches involving time-varying weights, nonlinear combinations, correlations among components, o cross-learning. These schemes encompass combining point forecasts and probabilistic forecasts (Li et al., 2020; Montero-Manso et al., 2020).

The development of robust offline or online aggregation algorithms has yielded promising prediction results even in complex environments. Ensemble methods, expert aggregation, and forecast combination have proven highly effective in time series forecasting across various real-world domains, including industrial production, sales and demand, oil prices, energy, pollution levels, health or car sales (Stock & Watson, 2006; Gurnani et al., 2017; Manescu & Van Robays, 2014; Gaillard et al., 2016; Auder et al., 2016; Perone, 2022; Fortsch et al., 2021).

In the R language (R Core Team, 2020), there are three packages that combine a wide set of forecasting methods (experts) and, according to the reviewed literature, significantly improve the predictions provided by these experts. These packages are Opera (Online Prediction by ExpeRt Aggregation), developed by Gaillard and Goude (2016), ForecastComb (Weiss et al., 2018), and ForecastHybrid (Shaub & Ellis, 2020).

The ForecastComb package (version 1.3.1) allows the use of up to fifteen forecasting methods, including simple methods, regression-based methods, and eigenvector-based methods. It also includes helpful tools to handle issues in the combination process, such as missing values or multicollinearity. Regression-based combinations are employed to optimize forecasting when certain component forecasts (experts) outperform the rest, while eigenvector-based combinations are used when all forecasts are at the same level (Hsiao & Wan, 2014). Several information criteria, including AIC (Akaike's information criterion), BIC (Bayesian information criterion or Schwarz information criterion), corrected AIC (Hurvich & Tsai, 1989), and Hannan Quinn information criterion (Hannan & Quinn, 1979), are available in the complete subset regression method. The package provides users with phases for data preparation, model estimation, and result interpretation through summaries and tracing functions. The component methods and models to be combined in ForecastComb include ARIMA, ETS (Exponential Triple Smoothing), NNETAR (Feed-forward neural networks with a single hidden layer and lagged inputs) and Dynamic Optimized Theta Model (Fiorucci, 2016).

The ForecastHybrid package (version 5.0.19) utilizes forecasts generated from experts such as ARIMA, ETS, theta method, NNETAR, STLM, TBATS, and SNAIVE (naive seasonal). These forecasts can be combined with equal weights, weights based on sample errors (Bates & Granger,1969), or validated crossweights. The package also supports cross-validation of time series data with user-supplied models and forecast functions to assess the accuracy of the model used. Hajirahimi & Khashei (2019) analyzed over 150 hybrid structures and models and compared their performance to that of non-hybrid models in the domains of time series modeling and forecasting. Many of these models included ARIMA and neural networks as components. Suhartono et al. (2017) developed a hybrid model that predicts time series containing trend, seasonal, and calendar variation patterns. This model combines ARIMA with Artificial Neural Networks and was designed for prediction in economics and business domains. Xiao et al. (2012) developed a hybrid model that combines ARIMA with Elman Artificial Neural Network (ANN) for container throughput forecasting at Tianjin Port. Ahn et al. (2022) constructed a Hybrid Model for forecasting indoor CO2 concentration by integrating a Mass balance equation model and a Bayesian Neural Network (BNN). Vavliakis et al. (2021) proposed a hybrid model suitable for modeling linear and non-linear sales trends in e-commerce by combining an ARIMA model with an LSTM (Long Short-Term Memory) neural network.

Opera (Online Prediction by ExpeRt Aggregation) provides several algorithms for robust prediction of time series using advice from experts, either online or provided by the user in offline mode. Opera offers three functions: mixture, for constructing the target algorithm; predict, for making predictions using the algorithm; and oracle, for evaluating the performance of experts and comparing it with the performance of the combination algorithm. In recent years, ensemble methods and expert aggregation, as demonstrated in Opera, have proven to be highly effective in the context of online time series forecasting. They have been applied to diverse real-world sectors, including pollution forecasting (Baudin, 2016), finance (Amat et al., 2018), and energy (Nowotarski & Weron, 2018). To obtain the results presented in this study, we used both offline and online options in Opera and incorporated the same experts as ForecastHybrid in offline mode, ensuring comparability of the results.

The remainder of this paper is organized as follows. The subsequent section outlines the study's objectives. In the Material and Methods section, we describe the time series used for validation, list the individual prediction methods employed, and elaborate on our proposed algorithm designed to achieve effective prediction combinations. In Section 4, we present the outcomes of applying our algorithm to unseen data from the mentioned series, comparing these with the results obtained from ForecastComb, ForecastHybrid, and Opera packages. The paper concludes with final remarks.

2. Objectives

The main objective of this study is to design and develop a Decision Support System (DSS) based on an algorithm that automatically combines individual forecasting methods available in R. The aim is to create a method that improves upon the results obtained from using these individual methods alone, as well as the results achieved by reference combined methods documented in the literature. The study focuses specifically on time series forecasting in economically impactful activities.

To accomplish this objective, the study has the following specific goals:

1. Reduce sensitivity to the choice of error measure when comparing results on the validation set.
2. Develop a program that automatically ranks methods and determines their weights to generate the final combination of forecasts.
3. Provide the user with various stopping rules to determine when to terminate the combination of forecasts.
4. Conduct a sensitivity analysis to assess the robustness of the method and its sensitivity to different specifications.
5. Compare the proposed method with other forecast combination methods using different relevant time series datasets found in the literature.
6. Create an automatic tool within the realm of artificial intelligence to optimize the prediction of historical time series data, regardless of the application area, with minimal human intervention.

By achieving these goals, the study aims to provide an effective and automated solution for improving time series forecasting accuracy in various domains.

3. Material and methods

3.1. Data: Key features

In this section, the data used in the study is described along with its key features. The aim is to introduce diversity in the type of time series data to be predicted in order to test the performance of the proposed algorithm under different patterns. The following datasets were considered:

1. Monthly total sales of new cars in the USA, as well as individual sales figures for four companies in the sector (Ford, General Motors, Honda, and Toyota). This dataset provides five alternative datasets to test the prediction methods. The data spans from January 1, 2005, to December 31, 2021, resulting in a panel data with 17 years and 204 observations for each of the five items. The data was obtained from the Federal Reserve of St. Louis and is available at https://www.goodcarbadcar.net/. The selection of these time series was based on the significance of car production in the economy and employment across various industries.
2. Monthly mean CO2 mole fraction (co2-mm-mlo) data, which represents CO2 emissions. The series is determined from daily averages and is corrected to the middle of the month when missing days are concentrated early or late in the month. The data spans from January 1, 2000, to December 31, 2019, resulting in a panel data with 20 years and 240 observations. The data was sourced from the US Government's Earth System Research Laboratory, Global Monitoring Division, and can be retrieved from the Trends in Atmospheric Carbon Dioxide website (https://datahub.io/core/co2-ppm#data). The choice of this series is justified by the influence of CO2 emissions on the environment and its impact on production and economic growth (Shpak et al., 2022).
3. Electricity sales to ultimate customers in the transportation sector, including availability from the public supply system. This dataset represents the generation and availability of electricity. The time series covers the period from January 1, 2005, to December 31, 2021, resulting in a panel data with 17 years and 204 observations. The data was obtained from the U.S. government and can be retrieved from the U.S. Energy Information Administration (https://www.eia.gov/totalenergy/data/monthly/). Given the economic importance of electricity generation, this dataset was included to evaluate the performance of the algorithm in this domain.
4. We selected a random sample of 100 monthly series from the 48,000 series collected in the M4 contest available in the R package M4comp2018 (Montero-Manso et al., 2018). This package contains 100,000 time series with different seasonal pattern (yearly, quarterly, monthly, …), the values of the series to perform the adjustments and the unseen values to assess the accuracy of the predictions. With this sample we expect to corroborate the improved forecast accuracy obtained with our algorithm.

The inclusion of these diverse datasets allows for testing the algorithm's performance under different scenarios and patterns. The chosen datasets have significant economic implications, such as their impact on production, employment, and the environment. By considering these datasets, the study addresses the importance of demand forecasting in industries and supply chain management, inventory control, and environmental considerations.

3.2. Methods

In the development of the alPCA algorithm proposed in this paper, the goal is to provide an automatic alternative to both individual forecasting methods and combination methods offered by packages like ForecastComb, Opera, and ForecastHybrid. These packages consist of individual forecasting methods referred to as "experts," which are equipped with functions to efficiently group the expert forecasts using combination methods such as weighted averages or regression models. The experts work individually on historical data sets ($y_1, y_2, \cdots, y_{n-h}, \cdots, y_n$) and provide successive predictions for each time point within a forecasting interval ($t_{n+1}, \ldots, t_{n+h}$). The algorithm, referred to as the forecaster, combines these predictions with historical data to determine the best forecast ($\hat{y}_{n+1}, \hat{y}_{n+2}, \cdots, \hat{y}_{n+h}$) for the interval.

To compare and evaluate the performance of the three packages, the study applied them under similar conditions, using the same series and forecasting methods as experts. Here are the main characteristics or usage recommendations of each method considered in the algorithm:

- Naïve-sNaïve: These are simple methods that do not consider seasonality and consider seasonality, respectively.
- ETS: This method is suitable for studying series with trend (damped or not) and seasonality. It assigns greater weight to more recent data and includes a parameter to dampen the trend to a flat line in the future.
- ARIMA: Recommended for non-stationary series, this method uses auto-correlations and moving averages of residual errors to forecast future values.
- THETA: It works on seasonally adjusted data and decomposes them into two lines. The first line estimates the long-term trend component by removing the curvature, while the second line approximates the short-term behavior by doubling the local curvatures.

- STL: This method is used to decompose seasonal time series with trends. It incorporates the Loess method to estimate non-linear relationships and handle occasional outliers.
- CROSTON: Included to handle time series representing intermittent demand, this method uses simple exponential smoothing to estimate the average demand size and the average interval between demands.
- PROPHET: Suitable for series with strong seasonal effects, missing data, changes in trend, and outliers. It uses an additive model to adjust non-linear trends for annual, weekly, and daily seasonality, including vacation effects.
- NNAR: This model approximates nonlinear functions. For non-seasonal series, the fitted model NNAR(p, k) is used, where k is the number of nodes. For seasonal data, the fitted model is NNAR(p, P, k).
- TBATS: This method applies a trigonometric transformation of seasonality and includes ARMA functionality and automatic Box-Cox transformation for error treatment. It can handle unequal variances and relational nonlinearity.
- GRNN: Suitable for time series with smaller-than-usual data sets, GRNN is a neural network-based technique that uses nonparametric regression and Gaussian functions for accurate estimation.
- MLP: The Multilayer Perceptrons (MLPs) are fully connected feed-forward neural networks commonly used for time series analysis. They are trained using error backpropagation. The alPCA algorithm incorporates the thief function, which applies the temporal hierarchical approach of Athanasopoulos et al (2017) for forecasting using MLPs.

The different predictors obtained by assigning parameters or arguments to each method are referred to as "configurations." These configurations quantify the possibilities in the configuration's column of the Table 1. The integration of a method into alPCA requires its implementation in R.

The process described involves a set of experts providing individual predictions based on historical data, which are then combined by the algorithm (forecaster) to improve the predictions. This process follows a protocol for sequential decisions by the forecaster, aiming to predict an unknown sequence of outcomes (Cesa-Bianchi & Lugosi, 2006). The forecaster's predictions are compared to the predictions of reference experts using a non-negative loss function.

The reason why these methods have been integrated is because this set of methods is able to deal with a wide range of series with different typologies, like series with high seasonality, with trend (short or long component, linear or non-linear), non-stationary, intermittent, with missing data, with effects of vacations, changes in trend and outliers and time series with a smaller than usual data set.

The selection of the integrated methods in the alPCA algorithm was driven by their ability to handle various types of time series with different characteristics. The chosen set of methods can effectively deal with series exhibiting diverse typologies, such as:

- High Seasonality: Some time series exhibit strong seasonal patterns, and the integrated methods can effectively capture and model this seasonality, allowing for accurate forecasting.
- Trend Components: Time series with trend components, whether short or long, linear or non-linear, can be handled by the integrated methods. These methods are capable of capturing and incorporating trend information into the forecasting process.
- Non-Stationary Series: Non-stationary series require specific modeling techniques to account for the changing properties over time. The integrated methods, such as ARIMA, are suitable for modeling and forecasting non-stationary series.
- Intermittent Demand: Time series representing intermittent demand, where periods of zero or low demand alternate with periods of high demand, are addressed by the integrated methods. The CROSTON method, for example, is designed to handle such intermittent demand patterns.
- Missing Data: Time series with missing data points pose challenges for forecasting. The integrated methods have mechanisms to handle missing data and provide accurate forecasts even in the presence of data gaps.

Table 1
Methods and configurations integrated in alPCA.

Method	Parameters/ criteria	Functions R	Configurations
Naïve - sNaïve	naive snaive	naive(), snaive()	2
ETS Exponential smoothing state space model	Mean Square Error: *mse* Average MSE over first nmse forecast horizons: *amse* Standard deviation of residuals: *sigma* Mean of absolute residuals: *mae* Log-likelihood: *lik* Combined with *damped = NULL/TRUE*	ets()	10
ARIMA automatic Hyndman-Khandakar algorithm	AIC BIC	auto.arima()	2
thetaModels	Nelder-Mead L-BFGS-B-SA	otm.arxiv(), dotm(), dstm(), otm(), stm(), stheta()	14
STL Seasonal Decomposition of Time Series by Loess	ets arima thetaf	stlf()	3
Croston Croston's method for intermittent demand forecasting	croston sba sbj funciones de pérdida combinadas: Mean Absolute Rate: *mar* Mean Squared Rate: *msr* Mean Absolute Error: *mae* Mean Squared Error: *mse*	crost()	12
Prophet	additive multiplicative	prophet()	2
NNAR Neural Network ARchitecture		nnetar()	1
TBATS Trigonometric seasonality Box-Cox transformation ARMA errors Trend Seasonal components	aic	tbats()	1
GRNN General Regression Neural Network	additive multiplicative	grnn_forecasting()	2
MLP Multilayer perceptron	thief median mean	mlp.thief()mlp()	3

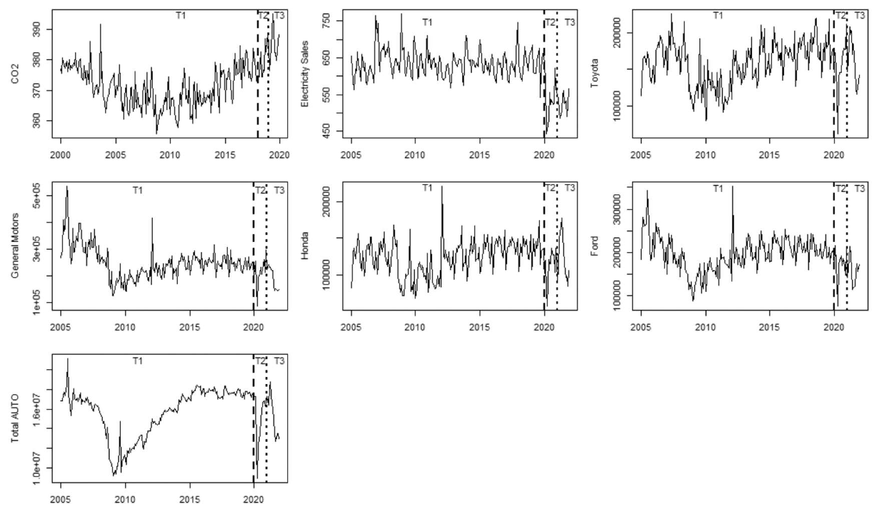

Fig. 2. Structure of the first 7 monthly time series for algorithm validation.

- Effects of Vacations: Time series influenced by vacation or holiday effects require models that can account for these specific patterns. The PROPHET method, for instance, incorporates annual, weekly, and daily seasonality, considering the effects of vacations.
- Changes in Trend and Outliers: Time series that experience changes in trend or contain outlier observations can be effectively modeled by the integrated methods. Techniques like STL (Seasonal and Trend decomposition using Loess) can identify and handle such anomalies.
- Smaller Than Usual Data Set: Time series with limited data points pose challenges for modeling and forecasting. The integrated methods, such as GRNN (Generalized Regression Neural Network), can handle smaller data sets and provide accurate predictions.

By integrating these diverse methods into the alPCA algorithm, a wide range of time series typologies can be effectively addressed, allowing for accurate and robust forecasting across different scenarios.

3.3. Predictive algorithm based on principal component analysis (alPCA)

The alPCA algorithm, which serves as the forecaster in this study, follows a similar structure to the one described earlier. It consists of eleven individual experts, each configured with different options available in R, resulting in a total of 52 different forecasts for each time series (refer to Table 1). The algorithm also adheres to the protocol introduced by Cesa-Bianchi & Lugosi (2006), similar to the benchmark packages. However, alPCA incorporates several innovative elements compared to these packages and individual methods.

The following are the step-by-step processes performed by the alPCA algorithm to achieve its objectives:

1. These phases are applied to each individual time series considered in the study (Fig. 2). Since the time series are monthly, the prediction horizon (h) is set to one year, equivalent to 12 months.
   - Training Phase (T_1): In this phase, the algorithm is trained and fitted using the available historical data. The training period includes the data from $y_1, y_2, \cdots, y_{n-h}$, , where n is the total number of observations in the time series.
   - Validation Phase (T_2): After training, the algorithm proceeds to the validation phase. In this phase, the algorithm generates predictions for the period from $y_{n-h+1}, \cdots, y_n$, which corresponds to the next h months after the training period. The prediction errors are calculated by comparing the predicted values with the actual values from the validation period.
   - Test Phase (T_3): In the test phase, the algorithm's performance is evaluated based on the predictions made for the future h months, starting from $y_{n+1}, y_{n+2}, \cdots, y_{n+h}$. These predictions are compared to the results obtained from the other prediction combination packages mentioned in the study. Data from this phase would be the unseen data

   alPCA automatically applies each of the 11 individual methods and its corresponding configurations (52 in all) that integrate the algorithm on the series fitting each configuration in the T_1. Predictions are generated for the T_2 period.

2. We obtained the values of the fitting error sMAPE and MASE in T_1, whereas sMAPE, MASE, RMSE and OWA are used in T_2. These error measures are defined below, indicating their name, nature, mathematical expression and the time interval over which they are measured. The four functions can be found in the R Metrics package.
   - Symmetric Mean Absolute Percentage Error (sMAPE)

$$sMAPE_{T_1} = \frac{200}{n-h} \sum_{t=1}^{n-h} \frac{|y_t - \hat{y}_t|}{|y_t + \hat{y}_t|} \quad [1]$$

$$sMAPE_{T_2} = \frac{200}{h} \sum_{t=n-h+1}^{n} \frac{|y_t - \hat{y}_t|}{|y_t + \hat{y}_t|} \quad [2]$$

   - Mean Absolute Scaled Error (MASE)

$$MASE_{T_1} = \frac{\sum_{t=1}^{n-h} |y_t - \hat{y}_t|}{\frac{n-h}{n-2h} \sum_{t=h+1}^{n-h} |y_t - y_{t-h}|} \quad [3]$$

$$MASE_{T_2} = \frac{\sum_{t=n-h+1}^{n} |y_t - \hat{y}_t|}{\frac{h}{h-1} \sum_{t=n-h+2}^{n} |y_t - y_{t-1}|} \quad [4]$$

   - Root Mean Square Error (RMSE)

$$RMSE_{T_2} = \sqrt{\frac{\sum_{t=1}^{h} (y_t - \hat{y}_t)^2}{h}} \quad [5]$$

- Overall Weighted Average (OWA), is a comparison metric that calculates the average of sMAPE and MASE in relation to the Naïve 2 model, as shown in the following expression:

$$OWA_{T_2} = \frac{1}{2}\left(\frac{MASE_{T_2}}{MASE_{snaive.T_2}} + \frac{sMAPE_{T_2}}{sMAPE_{snaive.T_2}}\right) \quad [6]$$

The error measures considered in this paper have been used in several predictives studies of similar areas to those covered in this article and using forecast combinations. Bakay & Ağbulut (2021) and Guermoui et al. (2020) applied RMSE in their respective researches. Symmetric mean absolute percentage error (sMAPE) loss function was used in the research of Castelo Branco & Werner, (2018) and Ensafi et al. (2022). RMSE and sMAPE both were used by Xiao et al. (2012) and Qu et al. (2022). sMAPE and MASE were used for Zhuang et al. (2022) in a combined forecasting method for intermittent demand using the automotive aftermarket data. sMAPE, mean absolute scaled error (MASE), and overall weighted average (OWA) were used to evaluate the performance of the forecasting methods by Montero-Manso et al. (2020), Cawood and Van Zyl (2022) and Hyndman et al. (2022), among others.

3. The algorithm performs a Principal Component Analysis to reduce the size of the error matrix, with six columns containing the six error measures above. We are left with one PC when it explains>80% of the variability, and with two PCs when the first one does not reach that percentage. This result, univariate CP_1 or bivariate CP_1 and CP_2, provides an error score for each of the proposed methods, that is calculated as a linear combination of the original error variables, with weights $p_1^i, \cdots, p_6^i$ (i = 1 for CP_1 and i = 2 for CP_2).

$$x_{CP_1} = p_1^1 sMAPE_{T_1} + p_2^1 MASE_{T_1} + p_3^1 RMSE_{T_2} + p_4^1 sMAPE_{T_2} + p_5^1 MASE_{T_2} + p_6^1 OWA_{T_2} \quad [7]$$

$$x_{CP_2} = p_1^2 sMAPE_{T_1} + p_2^2 MASE_{T_1} + p_3^2 RMSE_{T_2} + p_4^2 sMAPE_{T_2} + p_5^2 MASE_{T_2} + p_6^2 OWA_{T_2} \quad [8]$$

Since the fitting is performed with the training subseries (T_1), the largest in dimension, the fitting errors $sMAPE_{T_1}$, $MASE_{T_1}$ will have lower variability, and the prediction errors on the test subseries (T_2), $sMAPE_{T_2}$, $MASE_{T_2}, RMSE_{T_2}$ and OWA_{T_2} are expected to have higher variability since this sample has not been used in the fitting. This behaviour will have an impact on the error weights on the principal components, so there are expected:

a. similar signs for p_1^i, p_2^i, and similar signs for p_3^i, …, p_6^i, in any of the components; we therefore use the nomenclature.
b. $s_1^i = \text{sign}(p_1^i, p_2^i) = \text{sign}(p_1^i + p_2^i)$, $s_2^i = \text{sign}(p_3^i, \ldots, p_6^i) = \text{sign}(p_3^i + \ldots + p_6^i,)$ [9].
c. in component 1: higher magnitudes (in absolute value) for weights p_3^1, …, p_6^1 than for weights p_1^1, p_2^1 (because they contain more variability).
d. in component 2: magnitudes greater (in absolute value) for the weights p_1^2, p_2^2, than for the weights p_3^2, …, p_6^2, due to the orthogonal behaviour of the components.
e. the interpretation of the scores is extracted from the signs s_1^i and s_2^i and their magnitudes, as explained below.

4. Next, alPCA orders the fitting methods used, calculating for each of them the score CP_1, and also CP_2 when two components are required. The ordering of the methods will be given according to the Manhattan distance of a given method to the best one being this the one associated with the smallest error committed.

The best method, as well as the distance of each of the remaining methods to it, is identified through the sign of the weights of the error measures at T_1 and T_2:

a. when we consider only one component, CP_1, the best method will be given by the sign of the weights for the errors in T_2, i.e. by s_2^1:
   i. s_2^1 positive: identifies as the best method the one with the lowest score, that is, the one that provides the $\min(x_{CP_1})$, and the distance to the best will be given by $|x_{CP_1} - \min(x_{CP_1})|$.
   ii. s_2^1 negative: identifies as the best method the one with the highest score, that is, the one that gives the $\max(x_{CP_1})$, and the distance to the best will be given by $|\max(x_{CP_1}) - x_{CP_1}|$.
b. when we consider two components, CP_1 and CP_2, the best method in each of the components or dimensions will be defined according to the signs of the weights s_2^1 for CP_1 and s_1^2 for CP_2. The Manhattan distance to the best method will then be determined according to:
   i. $|x_{CP_1}\text{-}\min(x_{CP_1})| + |x_{CP_2}\text{-}\min(x_{CP_2})|$ if $s_2^1 > 0$ y $s_1^2 > 0$.
   ii. $|x_{CP_1}\text{-}\min(x_{CP_1})| + |\max(x_{CP_2})\text{-} x_{CP_2}|$ if $s_2^1 > 0$ y $s_1^2 < 0$.
   iii. $|\max(x_{CP_1})\text{-} x_{CP_1}| + |x_{CP_2}\text{-}\min(x_{CP_2})|$ if $s_2^1 < 0$ y $s_1^2 > 0$.
   iv. $|\max(x_{CP_1})\text{-} x_{CP_1}| + |\max(x_{CP_2})\text{-} x_{CP_2}|$ if $s_2^1 < 0$ y $s_1^2 < 0$.

5. A parametric cutoff point is assessed to determine how many methods should be combined. The goal is to include the methods that are closest to the best method based on their distance. This cutoff point is determined using a sequential inclusion rule, where the shortest distance to the best method is considered for each inclusion.

To determine the cutoff point, a parametric distribution is fitted to the distances between the methods and the best method. The Exponential, Gamma, and Inverse Gamma distributions are considered as alternatives, and the best distribution is selected using the Bayesian Information Criterion (BIC) criterion. The fitting process is performed using the univariateML package in R.

Once the distribution is fitted, theoretical percentiles are calculated to establish the cutoff points for method selection. The percentiles chosen are I={5, 10, 15, 20, 25, 50, 75, 80, 85, 90, 95, 100}. For example, the 5th percentile (I_5) will include only those methods whose distance to the best method is less than I_5, while the 100th percentile will include all methods in the combined estimator.

By using these cutoff points, the alPCA algorithm can determine the appropriate number of methods to include in the combination based on their proximity to the best method, providing a formal and less data-dependent approach compared to empirical percentiles.

6. The choice of how to combine the selected methods involves determining the weights to assign to each method. Three alternative weights, denoted as w_1, w_2, and w_3, are proposed. These weights are described in Table 2:
   a. For w_1, the absolute error scores × are used for each of the m methods in the CP_1 component. Each method is weighted based on the proportion of its error score compared to the sum of all errors. This means that methods with lower absolute error scores will receive higher weights.
   b. For w_2, the sMAPE fitted error for T_1 is used. T_1 represents the closest horizon to the data to be predicted at T_2. The weight assigned to each method is higher when the fitted error at T_1 is

Table 2
Weights used for combining the methods in the alPCA forecast.

$w_1 = \frac{\lvert x_j\rvert}{\sum_{j=1}^m \lvert x_j\rvert}$	$w_2 = \frac{\frac{1}{sMAPE_{T_{1j}}}}{\sum_{j=1}^m \frac{1}{sMAPE_{T_{1j}}}}$	$w_3 = \frac{\frac{1}{sMAPE_{T_{2j}}}}{\sum_{j=1}^m \frac{1}{sMAPE_{T_{2j}}}}$

lower. The weights are constructed by standardizing the inverses of the sMAPE errors at T_1.

c. Similar to w_2, w_3 has the same structure and measurements, but it uses the sMAPE forecast error for T_2. T_2 represents the closest horizon to the data to be predicted at T_3. The weights are constructed based on the standardized inverses of the sMAPE forecast errors at T_2.

7. Applies each of the individual methods on the series, fitting each configuration in the $T_1 + T_2$.
8. For each percentile (cut-off point) and each proposed weight (w_i), forecasts of T_3 are obtained for the combination of methods selected.

Next, the algorithm described is written, in order to make it clearer to the reader.

Diagram 1: alPCA algorithm process.

Algorithm alPCA: forecasts combination algorithm based on Principal Component Analysis

procedure alPCA algorithm (S, Ys, N_s, h, j, i)
Let Y_s be a time series in S with length N_s. Divide Y_s into two subsets: $T_{s,1} = y_1, y_2, \ldots, y_{n\text{-}h}$, $T_{s,2} = y_{n\text{-}h+1}, \ldots, y_{Ns}$ with h = 12
Let M_j be one of the J considered forecasting configurations.
for all s∈S do
for all j∈J do
Obtains the fitting values in T_1 and predictions at T_2
Calculate sMAPE and MASE in T_1, and sMAPE, MASE, RMSE and OWA in T_2
end for
Builds $E_{J,6}$ and removes duplicate rows
Obtains a PCA score
Order the rows of $E_{J,6}$ according to their Manhattan distance to the best method.
Obtains I quantiles for fitted distribution to the distance to the best method.
Let $K_I \subset J$ be the final subset of selected configurations
for all i∈ I do
for all z∈K_i do
Determine the weights ($w^v_{s,z}$) to combine the z configurations for v = 1, 2, 3
Apply the M_z configuration to fit Y_s and generate the h-step-ahead forecast $\hat{y}^z_{s,t}$, for $t = N_s + 1, \ldots, N_s + h$
end for
end for
Compute the combined forecasts: $\hat{y}^v_{s,i,t} = \sum_{z\in K_i} w^v_{s,z}\hat{y}^z_{s,t}$ for v = 1, 2, 3
end for
end procedure

4. Experimental results

To compare the performance of the alPCA algorithm with the reference forecasting packages (ForecastComb, Opera, and ForecastHybrid), the ex-post errors on T_3 are considered (unseen data). By comparing the sMAPE, RMSE, and MAE values, we can draw conclusions about the relative performance of the different methods and determine which approach yields more accurate forecasts. The results obtained from applying the alPCA algorithm and the reference forecasting packages to the seven series can be analyzed based on these error metrics.

4.1. Weights analysis

We compare the performance of the combined forecasts when using weights w_1, w_2 and w_3. In Table 3 we show, for each series, the percentile that gives the best model for each weight, and also the corresponding errors sMAPE, RMSE and MAE on T_3.

In Table 3 we can observe that the lowest errors are generally obtained for the predictions constructed with the weights w_2 or w_3. For the case of the CO2 series, the lowest error coincides for w_1 and w_3 in the three error measures, and in the ELECTRICITY series the smallest value is obtained for the three errors in w_2.

In Table 3, we observe that the best error values for the CO2 series are obtained for percentile I_{50} in any of the three weights. In the rest of the series, the best values are obtained in two cases for percentile I_5 (w_2) and in two cases for I_{100} (w_2 and w_3). We could not conclude on a specific recommendation about which percentile to use.

Whatever the series, it seems clear that weights w_2 and w_3 provide the best solutions in terms of minimum error, whatever the error measure considered, but especially when using the sMAPE measure, probably because both of them are based on sMAPE. It is important to note that the w_2 weights are based on the sMAPE associated with the fit of each model at $T_1 + T_2$, while the w_3 weights are based on the sMAPE associated with the prediction at T_2. The w_1 weights are based on the CP scores.

4.2. Comparison with the other packages

In order to compare the results obtained with alPCA to the other methods, we have assessed all possible configurations for all prediction packages considered, included alPCA for proposed percentiles. Table 4 shows the best and worst models obtained within each package, in terms of the ex-post error, and for each of the error measures considered. This table provides information on (a) which is the best algorithm configuration for each of the error measures, (b) the robustness of the best to different error measures, and (c) also about the range of variation of the alternatives within each package, and so the opportunities for a naive user, to find the best prediction model from the different possible configurations.

In the BEST CONFIGURATION columns, the best model in terms of providing minimum error is emphasised. The alPCA algorithm gives the best performance in all series but in FORD, where the difference is just 0.68 in sMAPE, 6.17 in RMSE and 7.37 in MAE, with the best combination coming from the ForecastComb_EIG4 configuration. Weight w_2 is mostly the one that provides the best model in all considered series (but in CO2 and TOYOTA).

The range of variation of the error between the WORST

Table 3
Ex-post errors for the best alPCA model obtained from the different weights w_1, w_2 and w_3. Column I contains percentiles used for the best combination. Bold type identifies the lowest error for each series.

	alPCA_w1				***alPCA_w2***				***alPCA_w3***			
Series	***I***	**sMAPE**	**RMSE**	**MAE**	***I***	**sMAPE**	**RMSE**	**MAE**	***I***	**sMAPE**	**RMSE**	**MAE**
CO2	I_{50}	**1.19**	15.3	**4.6**	I_{50}	1.20	**15.2**	**4.6**	I_{50}	**1.19**	**15.2**	**4.6**
TOYOTA	I_5	16.95	31079.4	27128.5	I_5	**16.94**	31078.7	27126.5	I_5	**16.94**	**31077.7**	**27123.6**
HONDA	I_{75}	19.24	26186.6	23750.3	I_{75}	**16.43**	24853.6	**20702.7**	I_{75}	19.19	**24627.1**	20851.4
GM	I_{100}	21.23	47606.7	40066.7	I_{100}	**20.66**	**43254.9**	**38756.3**	I_5	21.65	54144.9	41187.9
FORD	I_5	18.15	36343.3	29190.1	I_5	**17.56**	37011.8	**28318.8**	I_5	17.99	**35897.0**	28841.0
Total AUTO	I_{25}	12.23	2077720.0	1882412.0	I_{25}	**7.28**	**1359490.1**	**1129581.4**	I_{25}	10.22	1757963.6	1569414.6
ELECTRICITY	I_{80}	3.49	22.0	18.4	I_{80}	**2.37**	**15.5**	**12.5**	I_{90}	3.72	22.7	19.8

Table 4
The best and worst model in terms of the ex-post error, obtained from alPCA and all the other packages for combined forecast.

Series	*BEST CONFIGURATION*				*WORST CONFIGURATION*			
	alPCA		Forecast packages		*alPCA*		Forecast packages	
	weight/ I	**sMAPE**	**configuration**	**sMAPE**	***weight/I***	**sMAPE**	**configuration**	**sMAPE**
CO2	**w_1 I_{50}**	**1.19**	Opera_zFTRLon	1.20	**w_2 I_5**	**1.29**	ForecastComb_BG	4.23
TOYOTA	**w_3 I_5**	**16.94**	Opera_zMLprodoff	20.01	w_2 I_{25}	20.02	ForecastComb_EIG4	24.82
HONDA	**w_2 I_{75}**	**16.43**	Opera_zEWAon	19.12	w_2 I_5	23.59	ForecastComb_LAD	24.47
GM	**w_2 I_{100}**	**20.66**	Opera_zEWAon	24.16	w_2 I_{75}	27.24	Opera_zEWAoff	27.29
FORD	w_2 I_5	17.56	ForecastComb_EIG4	16.88	w_3 I_{90}	18.72	ForecastComb_LAD	26.91
Total AUTO	**w_2 I_{25}**	**7.28**	ForecastComb_EIG2	12.53	w_3 I_{90}	12.98	ForecastComb_OLS	32.69
ELECTRICITY	**w_2 I_{80}**	**2.37**	Hybrid_Equal-MASE	3.09	w_2 I_5	5.48	ForecastComb_EIG4	19.15
	weight/I	**RMSE**	**configuration**	**RMSE**	***weight/I***	**RMSE**	**configuration**	**RMSE**
CO2	**w_1 I_5**	**14.9**	Opera_zFTRLon	15.3	**w_2 I_5**	**15.2**	ForecastComb_BG	17.9
TOYOTA	**w_3 I_5**	**31077.7**	Opera_zMLprodoff	38725.0	**w_2 I_{20}**	**38639.5**	ForecastComb_EIG4	48156.0
HONDA	**w_2 I_{80}**	**24853.6**	Opera_zEWAon	28338.7	w_2 I_5	32547.5	ForecastComb_LAD	33344.0
GM	**w_2 I_{100}**	**43254.9**	Opera_zEWAon	56780.2	w_2 I_{75}	69776.4	Opera_zEWAoff	68393.5
FORD	w_2 I_5	37011.8	ForecastComb_EIG4	34878.0	w_3 I_{90}	38050.8	ForecastComb_LAD	53031.4
Total AUTO	**w_2 I_{25}**	**1359490.1**	ForecastComb_EIG2	2247470.1	w_3 I_{90}	2316162.0	ForecastComb_OLS	6686460.9
ELECTRICITY	**w_2 I_{80}**	**15.5**	Hybrid_Equal-MASE	20.4	w_2 I_5	34.0	ForecastComb_EIG4	112.2
	weight/I	**MAE**	**configuration**	**MAE**	***weight/I***	**MAE**	**configuration**	**MAE**
CO2	**w_1 I_{50}**	**4.6**	**Opera_zFTRLon**	**4.6**	w_2 I_5	5.0	ForecastComb_BG	16.7
TOYOTA	**w_2 I_5**	**27126.5**	Opera_zMLprodoff	33048.8	**w_2 I_{20}**	**32917.1**	ForecastComb_EIG4	40358.4
HONDA	**w_2 I_{75}**	**20702.7**	Opera_zEWAon	23973.9	w_2 I_5	29158.2	ForecastComb_LAD	30035.6
GM	**w_2 I_{100}**	**40066.7**	Opera_zEWAon	46844.5	w_2 I_{75}	55074.8	Opera_zEWAoff	54733.8
FORD	w_2 I_5	28318.8	ForecastComb_EIG4	27185.9	w_3 I_{90}	30364.2	ForecastComb_LAD	46300.3
Total AUTO	**w_2 I_{25}**	**1,129,581**	ForecastComb_EIG2	1918607.8	w_3 I_{90}	2005441.1	ForecastComb_OLS	5846654.7
ELECTRICITY	**w_2 I_{80}**	**12.5**	Hybrid_Equal-MASE	16.6	w_2 I_5	29.1	ForecastComb_EIG4	111.3

Table 5a
1: Top ten best methods for the CO2, Electricity and totalAUTOS series with the lowest ex-post sMAPE.

Ranking	Series		Series		Series	
sMape	**CO2**	Method	**Electricity**	Method	**totalAUTOS**	Method
1	1.19	alPCA_$w_1.I_{50}$	2.37	alPCA_$w_2.I_{80}$	10.22	alPCA_$w_3.I_{25}$
2	1.20	alPCA_$w_2.I_{50}$	2.38	alPCA_$w_2.I_{85}$	11.23	alPCA_$w_2.I_{50}$
3	1.20	alPCA_$w_2.I_{75}$	2.65	alPCA_$w_2.I_{90}$	11.77	alPCA_$w_3.I_{50}$
4	1.20	alPCA_$w_2.I_{80}$	2.68	alPCA_$w_2.I_{100}$	12.11	alPCA_$w_3.I_{15}$
5	1.20	alPCA_$w_2.I_{85}$	2.68	alPCA_$w_2.I_{95}$	12.11	alPCA_$w_3.I_{20}$
6	1.20	alPCA_$w_2.I_{90}$	3.09	**Hybrid_equal-MASE**	12.28	alPCA_$w_3.I_5$
7	1.20	alPCA_$w_2.I_{95}$	3.15	**Hybrid_equal-MAE**	12.28	alPCA_$w_3.I_{10}$
8	1.20	alPCA_$w_2.I_{100}$	3.15	**Hybrid_isample-RMSE**	12.39	alPCA_$w_2.I_5$
9	1.20	**OP_zFTRLon**	3.17	**Hybrid_isample-MASE**	12.39	alPCA_$w_2.I_{10}$
10	1.21	**OP_zFTRLoff**	3.17	**Hybrid_isample-MAE**	12.47	alPCA_$w_2.I_{15}$

Table 5b
2: Top ten best methods for the cars series, with the lowest ex-post sMAPE.

Ranking	Series		Series		Series		Series	
sMape	**Toyota**	Method	**Honda**	Method	**GM**	Method	**Ford**	Method
1	16.94	alPCA_$w_3.I_{10}$	16.43	alPCA_$w_2.I_{75}$	20.66	alPCA_$w_2.I_{100}$	16.88	**FC_EIG4**
2	16.94	alPCA_$w_2.I_5$	16.59	alPCA_$w_2.I_{80}$	21.32	alPCA_$w_2.I_{95}$	17.56	alPCA_$w_2.I_5$
3	17.18	alPCA_$w_2.I_{90}$	16.59	alPCA_$w_2.I_{85}$	21.65	alPCA_$w_2.I_5$	17.56	alPCA_$w_2.I_{10}$
4	17.18	alPCA_$w_2.I_{95}$	16.59	alPCA_$w_2.I_{90}$	21.65	alPCA_$w_2.I_{10}$	17.56	alPCA_$w_2.I_{15}$
5	17.18	alPCA_$w_2.I_{100}$	16.59	alPCA_$w_2.I_{95}$	21.65	alPCA_$w_3.I_5$	17.61	**FC_EIG2**
6	18.94	alPCA_$w_2.I_{85}$	16.68	alPCA_$w_2.I_{100}$	21.65	alPCA_$w_3.I_{10}$	17.63	alPCA_$w_3.I_{100}$
7	19.13	alPCA_$w_3.I_{90}$	18.86	alPCA_$w_3.I_{100}$	23.93	alPCA_$w_3.I_{15}$	17.66	alPCA_$w_2.I_{20}$
8	19.13	alPCA_$w_3.I_{95}$	18.94	alPCA_$w_3.I_{80}$	23.93	alPCA_$w_3.I_{20}$	17.67	alPCA_$w_2.I_{75}$
9	19.13	alPCA_$w_3.I_{100}$	18.94	alPCA_$w_3.I_{85}$	23.93	alPCA_$w_3.I_{25}$	17.74	alPCA_$w_2.I_{80}$
10	19.49	alPCA_$w_3.I_{85}$	18.94	alPCA_$w_3.I_{90}$	24.16	**OP_zEWAon**	17.79	alPCA_$w_2.I_{85}$

CONFIGURATION and the BEST one, is considerably smaller for alPCA than for the alternative methods/configurations basically in all series but in HONDA and GM. This fact, in conjunction with the first conclusion in the paragraph above, implies that, even in the cases when a naive solution is looked for, the chances of finding a reasonable fit with the alPCA algorithm are greater than by using alternative methods/configurations.

Finally, in the WORST CONFIGURATION columns, the values emphasised identify those alPCA configurations where the worst alPCA configuration is even better than the best model provided by the package's alternative to alPCA. It is not common, but the fact that this situation occurs in some series (like CO2 and TOYOTA), increases the reliability of the alPCA algorithm.

4.3. Robustness

Table 5a and 5b show the performance of the top ten forecasters in terms of the ex-post sMAPE, for each one of the seven series: Table 5a for CO2 and ELECTRICITY series, and Table 5b for the others. Given the stability of the results with different error measures, we have decided just to present this ranking in terms of the sMAPE error and not the others.

We can appreciate that the alPCA algorithm remains mostly among the top 10 models for all series. Only 10 (emphasised in the Tables) of the 70 top-ten displayed results in the seven series come from methods other than alPCA. In fact, the top-ten for Toyota, Honda and totalAUTO series do all come from the alPCA algorithm. This fact reaffirms the robustness of the alPCA algorithm in providing the best prediction results, especially when using weights w_2 and w_3.

In reference to the influence of the percentile in its combination with the weight to obtain the best prediction, we observe that the lowest values of sMAPE and MASE coincide, at the percentile level, with the median, thus in Table 4 we see that in the CO2 series the best values of the error are given by the combination $w_1.I_{50}$. For the Electricity series, these lower values are obtained for the three error measures, with intervention of the I_{80} percentile with the w_2 weight. In our experience, the w_3 weight obtains the best results with the I_5 percentile.

We also observe that the greatest distance between error measures is always in the totalAUTOS series for the combination $w_2.I_{25}$ in alPCA versus ForecastComb_EIG2, so if we measure with sMAPE we obtain a distance of 5.3 between the results, if we do it with RMSE the distance is 888.0 and with MASE we obtain 789026.8.

We can see that the best error for the ELECTRICITY dataset (2.37) is obtained by alPCA with w_2 weight, comparing this result with the other three packages, we note that the worst value (19.15) is provided by ForecastComb, with a distance of 16.79 points between them.

alPCA with w_3 weight obtains its lowest error (16.94) for TOYOTA, comparing this result with the other three packages, the highest error (26.59) is provided by ForecastComb, with a distance of 8.64 points between it and the value obtained by alPCA. On HONDA the combination constructed by alPCA with w_2 weight obtains the lowest error (16.43) compared to that obtained by the other three packages, we note that the highest prediction error (24.47) corresponds to ForecastComb. Distance between them is 8.04.

alPCA with w_2 weight produces the best error (20.66) working with GM series. Comparing this result with the other three packages, we see that the worst value (27.24) is provided by ForecastComb, with a distance of 6.58 between it and the value obtained by alPCA.

For totalAUTOS series, the best error (7.28) is obtained applying the w_2 weight for alPCA. Comparing this result with the other three packages, we see that the worst value (32.69) is provided by ForecastComb, with a distance of 25.41 points between them.

Table 6
Number of models that make up the best predictor combination of alPCA.

Serie	Number of models	weights	percentile	sMAPE
CO2	14	w_3	50	1.19
Toyota	3	w_3	10	16.94
Honda	23	w_2	75	16.43
Gmotors	41	w_2	100	20.66
Ford	6	w_2	5	17.56
TotalAUTOS	10	w_2	25	7.28
Electricity	27	w_2	80	2.37

Table 7
Models in the combination that provides the best prediction in all studied series.

Series	Configurations
CO2	Naïve, sNaïve, auto.arima(aic) Theta: otm.arxiv, dotm(L-BFGS-B), dstm(Nelder-Mead),otm(L-BFGS-B), stm(L-BFGS-B), stfl(ets,arima)Croston(mae), NNETAR, GRNN (additive,multiplicative)
Electricity	ETS(lik,amse,mae,sigma), auto.arima(aic,bic) Theta: otm.arxiv, dotm(Nelder-Mead, SANN, L-BFGS-B), otm(SANN, Nelder-Mead), stm(SANN), stheta, stlf(ets,arima). Croston (sba_mae_mse,sbj_mar_mae_mse)Prophet (additive,multiplicative)NNETAR, TBATS, MLP: mlp.thief, mlp (median)
Toyota	Croston: sba(mar,msr), sbj(msr)
Honda	Naive, sNaive ETS (lik, amse, sigma, mae, lik, mse)Theta: otm.arxiv, dstm(SANN), stm(SANN), stlf(ets,arima,thetaf) ,Croston: mae,mse, sba(mse), sbj (mar,mse)NNETAR, TBATS, MLP (median, mean)
Gmotors	Combine all except: ETS(lik, mae), Theta: otm(L-BFGS-B)
Ford	Croston: sba(mae, mse), sbj(mae, mse), MLP: mlp.thief, mlp(median)
totalAUTOS	Croston: sba(mar, msr, mae,mse), sbj(mar, msr, mae,mse) MLP(median, mean)

4.4. Combined methods

In Table 6 the number of models included in the best alPCA configuration (in terms of minimum sMAPE) are displayed. We can see that there is not a modal value for the number of models that provide the best alPCA combination, nor does it appreciate a specific trend related to the weights w_2 or w_3.

Table 7 shows the name of the models in the best alPCA configuration for all the studied series. The methods that, in these cases, have a higher participation in the best combinations are: Croston, Theta Model, Exponential smoothing state space model (ETS) and the General Interface for Single Layer Neural Network (MLP) for multilayer perceptron.

4.5. Sample monthly series of the M4 competition

In Fig. 3 we can see that in almost half of the 100 series the best ex-post error is obtained with the w1 weights, based on PC scores. We could propose, in view of these results, the use of 3 percentiles: 5, 50 and 95, so that if the user wants a solution with few methods, he should choose I_5, and if he wants one with many methods I_{95}, being I_{50} the intermediate solution.

Our proposal obtains better results, in terms of ex-post sMAPE error, in 57 of the 100 series, with the Hybrid package being the worst performer overall. It is true that, on average, it would need 3 min to obtain the results of a series when the ForecastComb package would only need 9.37 s. It is important to note that the ForecastComb package was only able to run smoothly with 6 of the 19 possible configurations. This explains the low computation time. The other two packages, Hybrid and Opera, were able to run with all their configurations, 9 and 14, respectively. When analyzing the 95% confidence interval for the mean ex-post sMAPE with these 100 series, we observe that our proposal achieves better results overall, although it is true that it overlaps with the results obtained with the Opera package (Table 8).

5. Conclusions

The alPCA algorithm proposed in this study offers a new and efficient approach to forecast combination, aiming to improve forecast accuracy. By combining the forecasts of 11 methods with their various configurations, all of them available in R, the algorithm utilizes principal

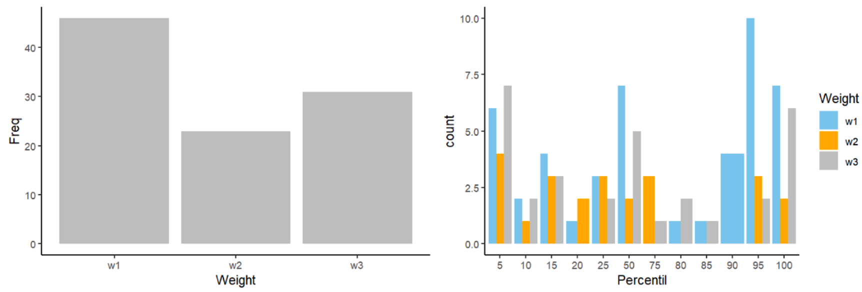

Fig. 3. Smallest ex-post SMape error obtained with each weight on the sample series.

Table 8
Best ex-post forecast method for each sample series.

Method	Best	95 %CI (sMAPE)	Time (in mean) (s.)*
alPCA	57	3.93 – 6.41	185.95
Hybrid	5	14.63 – 29.75	120.49
Opera	19	4.55 – 7.22	15.46
ForecastComb	19	7.49 – 11.57	9.37

* Intel(R) Core(TM) i5-10400 CPU @ 2.90 GHz.RAM: 12 GB.

components to summarize the information contained in multiple error measures, enabling the establishment of a ranking criterion for weighing the contribution of each model in the overall estimation.

To determine the number of models to include in the final combination, the study proposed a parametric fitting and percentile approach, which exhibited low variability in the results and was less sensitive to the choice of percentile.

With our proposal the user can choose the type of weighting to use based on the PC score, the adjustment sMAPE or the prediction sMAPE. The choice of the percentile allows you to decide on the number of configurations of the 11 methods to include in the combination.

Although the analysis of the first 7 series led us to choose the w2 or w3 weights, the results of the M4 sample would lead us to choose w1 with three possible percentiles, depending on the number of configurations the user wants in the final solution. However, the results obtained with the other weights and/or percentiles would not be too far off.

We understand that it can be confusing for a user to decide on one of the configurations to analyze a series (9 with Hybrid, 14 with Opera and 16 with ForecastComb). In the examples we have observed that not selecting an appropriate configuration can lead to major ex-post errors. However, with our proposal the ex-post errors have a lower variability so that the ex-post errors will not be too far from the best.

A time of 3 min, on average, to obtain the predictions of a series can be understood to be high, compared to the other packages, however the design of the algorithm would allow to easily apply parallel programming techniques that would reduce it considerably.

Overall, the alPCA algorithm provides a promising alternative to existing forecast combination methods, offering a robust and user-friendly approach with minimal dependence on user choices. The algorithm consistently selects top-performing models for the given series, considering different weight and percentile configurations. Although further testing on additional series is desirable, the authors intend to develop the alPCA algorithm into an R-package in the coming months, making it readily available for use.

The results affirm the value of combined forecasting methods in reducing prediction errors compared to individual models. The ongoing research in this area aims to refine and enhance the algorithms used for combining existing models, with the ultimate goal of providing users with an automated DSS procedure for obtaining the best forecasts.

Hyndman and Koehler (2006), Wintenberger (2017).

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data are public and accessible

Acknowledgement

José V. Segura-Heras acknowledges the financial support from the Generalitat Valenciana under project PROMETEO/2021/063.

References

Ahn, K. U., Kim, D. W., Cho, K., Cho, D., Cho, H. M., & Chae, C. U. (2022). Hybrid Model for Forecasting Indoor CO2 Concentration. *Buildings, 2022*(12), 1540. https://doi.org/10.3390/buildings12101540

Amat, C., Michalski, T., & Stoltz, G. (2018). Fundamentals and exchange rate forecastability with simple machine learning methods. *Journal of International Money and Finance, 88*, 1–24. https://doi.org/10.1016/j.jimonfin.2018.06.003

Armstrong, J. S. (2001a). *Combining forecasts.* Wharton Faculty Research: University of Pennsylvania. Retrieved from https://repository.upenn.edu/marketing_papers/34.

Armstrong, J. S. (2001b). *Principles of Forecasting: A Handbook for Researchers and Practitioners.* Kluwer Academic Publishers.

Athanasopoulos, G., Hyndman, R. J., Kourentzes, N., & Petropoulos, F. (2017). Forecasting with temporal hierarchies, European Journal of Operational Research, 262(1), 2017. *ISSN, 60–74*, 0377–2217. https://doi.org/10.1016/j.ejor.2017.02.046

Auder, B., Bobbia, M., Poggi, J.-M., & Portier, B. (2016). Sequential aggregation of heterogeneous experts for pm10 forecasting. *Atmospheric Pollution Research, 7*(6), 1101–1109. https://doi.org/10.1016/j.apr.2016.06.013

Bakay, M.S. & Ağbulut, Ü.(2021). Electricity production based forecasting of greenhouse gas emissions in Turkey with deep learning, support vector machine and artificial neural network algorithms. *Journal of Cleaner Production*, 285 (2021), Article 125324. 10.1016/j.jclepro.2020.125324.

Barnard, G. A. (1963). New methods of quality control. *Journal of the Royal Statistical Society: Series A (Gener.), 126*(2), 255–258. https://doi.org/10.2307/2982365

Bates, J. M., & Granger, C. W. (1969). The Combination of Forecasts. *Journal of the Operational Research Society, 20*(4), 451–468. https://doi.org/10.2307/3008764

Baudin, P. (2016). Sequential prediction with ensemble aggregation: application to meteorological prediction with uncertainties.Theses,Universitè Paris 11.

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

Brown, G., Wyatt, J., Harris, R., & Yao, X. (2005). Diversity creation methods: A survey and categorisation. *Information Fusion, 6*(1), 5–20. https://doi.org/10.1016/j.inffus.2004.04.004

Cang, S., & Yu, H. (2014). A combination selection algorithm on forecasting. *European Journal of Operational Research, 234*(1), 127–139. https://doi.org/10.1016/j.ejor.2013.08.045

Castelo Branco, A., & Werner, L. (2018). *A comparative study on combinations of forecasts and their individual forecasts by means of simulated series*. Brasil: Universidade Estadual de Maringá.

Cawood, P., & Van Zyl, T. (2022). Evaluating State-of-the-Art, Forecasting Ensembles and Meta-Learning Strategies for Model Fusion. *Forecasting, 2022*(4), 732–751. https://doi.org/10.3390/forecast4030040

Cesa-Bianchi, N., & Lugosi, G. (2006). *Prediction, Learning, and Game*. Cambridge University Press.

Ensafi, Y., Amin, S. H., Zhang, G., & Shah, B. (2022). Time-series forecasting of seasonal items sales using machine learning – A comparative analysis. *International Journal of Information Management Data Insights, 2*(1), 2022. https://doi.org/10.1016/j.jjimei.2022.100058

Fiorucci, J.A. (2016). Forecasting Time Series by Theta Models. R Documentation.

Fortsch, S.M., Choi, J.H. & Khapalova, E.A. (2021). Competition can help predict sales. *Journal of Forecasting*. First published: 19 August 2021. Wiley Online Library. 10.1002/for.2818.

Gaillard, P., Goude, Y., & Nedellec, R. (2016). Additive models and robust aggregation for gefcom2014 probabilistic electric load and electricity price forecasting. *International Journal of forecasting, 32*(3), 1038–1050. https://doi.org/10.1016/j.ijforecast.2015.12.001

Gaillard, P. & Goude, Y. (2016). OPERA: Online prediction by expert aggregation. Retrieved from https://cran.r-project.org/web/packages/opera.

Guermoui,M. Melgani, F., Gairaa, K., Mekhalfi, M.L. (2020). A comprehensive review of hybrid models for solar radiation forecasting. *Journal of Cleaner Production*, Volume 258, 2020, 120357, ISSN 0959-6526, 10.1016/j.jclepro.2020.120357.

Gurnani M., Korke Y., Shah P., Udmale S., Sambhe V., & Bhirud S. (2017). Forecasting of sales by using fusion of machine learning techniques. *International Conference on Data Management, Analytics and Innovation* (ICDMAI).

Hajirahimi, Z., & Khashei, M. (2019). Hybrid structures in time series modelling and forecasting: A review. *Engineering Applications of Artificial Intelligence, 86*(2019), 83–106. https://doi.org/10.1016/j.engappai.2019.08.018

Hannan, E.J. & Quinn, B.G.(1979). The determination of the order of an autoregression. *Journal of the Royal Statistical Society*. Series B (Methodological), pages 190–195, 1979. Retrieved from http://www.jstor.org/stable/2985032.

Haykin, S. (2009). *Neural Networks and Learning Machines*. New York: Prentice Hall/Pearson.

Hsiao, C., & Wan, S. K. (2014). (2014) Is there an optimal forecast combination? *Journal of Econometrics, 178*(2), 294–309. https://doi.org/10.1016/j.jeconom.2013.11.003

Hurvich, C.M. & Tsai, C.L. (1989). Regression and time series model selection in small samples. *Biometrika*, 76(2):297–307, 1989. 10.2307/2336663.

Hyndman, R.J., Athanasopoulos, G, Bergmeir, C., Caceres, G., Chhay, L., O'Hara-Wild, M., Petropoulos, F., Razbash, S. Wang, E. & Yasmeen, F. (2022). Forecasting Functions for Time Series and Linear Models. Package "Forecast". Retrieved from https://cran.r-project.org/web/packages/forecast/forecast.pdf.

Hyndman, R.J. & Koehler, A.B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting*, Volume 22, Issue 4, 2006, Pages 679-688. 10.1016/j.ijforecast.2006.03.001.

Li, X., Kang, Y., & Li, F. (2020). Forecasting with time series imaging. *Expert Systems with Applications, 160*(113680), Article 113680. https://doi.org/10.1016/j.eswa.2020.113680

Lichtendahl, K.C. & Winkler R.L. (2020). Why do some combinations perform better than others? *International Journal of Forecasting*, Volume 36, Issue 1, 2020, Pages 142-149. 10.1016/j.ijforecast.2019.03.027.

Manescu, C. & Van Robays, I. (2014). Forecasting the brent oil price addressing time-variation in forecast performance. Working paper series NO 1735 / September 2014, European Central Bank.

Montero-Manso, P.; Netto, C.; Talagala, C. M4comp2018: Data from the M4-Competition; R Package Version 0.2.0. 2018. Available online: https://github.com/carlanetto/M4comp2018/releases/download/0.2.0/M4comp2018_0.2.0.tar.gz (accessed on 2 January 2022).

Montero-Manso, P., Athanasopoulos, G., Hyndman, R. J., & Talagala, T. S. (2020). FFORMA: Feature-based forecast model averaging. *International Journal of Forecasting, 36*(1), 86–92. https://doi.org/10.1016/j.ijforecast.2019.02.011

Nowotarski, J., & Weron, R. (2018). Recent advances in electricity price forecasting: A review of probabilistic forecasting. *Renewable and Sustainable Energy Reviews, 81*, 1548–1568. https://doi.org/10.1016/j.rser.2017.05.234

Perone G. (2022). Comparison of ARIMA, ETS, NNAR, TBATS and hybrid models to forecast the second wave of COVID-19 hospitalizations in Italy. *The European Journal of Health Economics* Aug;23(6): 917-940. 10.1007/s10198-021-01347-4.

R Core Team (2020). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. Retrieved from https://www.R-project.org/.

Qu, F., Wnag, Y., Hou, W., Zhou, X. Wang, X., Li, J. & Wang, J.(2022). Forecasting of Automobile Sales Based on Support Vector Regression Optimised by the GreyWolf Optimizer Algorithm. Mathematics 10, 2234.10.3390/math10132234.

Shaub, D. & Ellis, P. (2020). forecastHybrid: Convenient Functions for Ensemble Time Series Forecasts. Retrieved from https://github.com/ellisp/forecastHybrid.

Shpak, N., Ohinok, S., Kulyniak, I., Sroka, W., Fedun, Y., Ginevičius, R., & Cygler, J. (2022). CO2 Emissions and Macroeconomic Indicators: Analysis of the Most Polluted Regions in the World. *Energies, 15*, 2928. https://doi.org/10.3390/en15082928

Elliott, G., Granger, C. W. J., & Timmermann, A. (Eds.). (2006). *Handbook of Economic Forecasting* (Volume 1). Elsevier.

Suhartono, R., Prastyo, S. P., Wijayanti, D. D., & Juliyanto, D. G. P. (2017). Hybrid model for forecasting time series with trend, seasonal and calendar variation patterns. *Journal of Physics: Conf. Series, 890*(2017), Article 012160. https://doi.org/10.1088/1742-6596/890/1/012160

Thomson, M.E.; Pollock, A.C.; Önkal, D. & Gönül, M.S. (2019). Combining forecasts: Performance and coherence. *International Journal of Forecasting*, Volume 35, Issue 2, 2019, Pages 474-484. 10.1016/j.ijforecast.2018.10.006.

Timmermann, A (2006). Forecast Combinations. In: Handbook of Economic Forecasting. Ed. by G Elliott, CWJ Granger, and A Timmermann. Vol. 1. Elsevier. Chap. 4, pp.135–196. 10.1016/S1574-0706(05)01004-9.

Vavliakis, K., Siailis, A. & Symeonidis, A. (2021). Optimising Sales Forecasting in e-Commerce with ARIMA and LSTM Models. In Proceedings of the 17th International Conference on Web Information Systems and Technologies (WEBIST 2021), pages 299-306. 10.5220/0010659500003058.

Wang, Q., Li, S. & Pisarenko, Z. (2020). Modelling carbon emission trajectory of China, US and India. *Journal of Cleaner Production*, Volume 258, 2020, 120723, ISSN 0959-6526, 10.1016/j.jclepro.2020.120723.

Weiss, C.E., Raviv, E. & Roetzer, G. (2018). Forecast Combinations in R using the ForecastComb Package. *The R Journal* Vol. 10/2. 10.32614/RJ-2018-052.

Wintenberger, O. (2017). Optimal learning with Bernstein online aggregation. *Machine Learning* 106 (1), 119-141. 10.48550/arXiv.1404.1356.

Xiao, Y., Xiao, J., & Wang, S. (2012). A hybrid model for time series forecasting. *IO Press, 1 Jan., 2012*, 133–143. https://doi.org/10.3233/HSM-2012-0763

Zhuang, X., Yu, Y., & Chen, A. (2022). A combined forecasting method for intermittent demand using the automotive aftermarket data. *Data Science and Management, 5*(2), 43–56. https://doi.org/10.1016/j.dsm.2022.04.001