

Improving Forecasting by Resampling STL Decomposition

Clara Cordeiro[1(✉)], M. Rosário Ramos[2], and M. Manuela Neves[3,4]

[1] Faculdade de Ciências e Tecnologia, Universidade do Algarve, Faro, Portugal
ccordei@ualg.pt
[2] Universidade Aberta & CEG-UAb, Lisbon, Portugal
marosram@uab.pt
[3] Instituto Superior de Agronomia, Universidade de Lisboa, Lisbon, Portugal
manela@isa.ulisboa.pt
[4] CEAUL – Centro de Estatística e Aplicações, Faculdade de Ciências, Universidade de Lisboa, Lisbon, Portugal

Abstract. The development of new forecasting algorithms has shown an increasing interest due to the emerging of new fields of application like machine learning and forecasting competitions. Although initially intended for independent random variables, *bootstrap* methods can be successfully applied to time series. The `Boot.EXPOS` procedure, which combines *bootstrap* and exponential smoothing methods, has shown promising results for forecasting. This work proposes a new approach to forecasting, which is briefly described as follows: using Seasonal-Trend decomposition by *Loess* (STL), the best STL fit is selected by testing all possible combinations of parameters. The best combination of smoothing parameters is chosen based on an accuracy measure. The time series is then decomposed into components according to the best STL fit. The `Boot.EXPOS` procedure is employed to forecast the seasonal component and the seasonally adjusted time series. These forecasts are aggregated to obtain a final forecast. The performance of this combined forecast is evaluated using real datasets and compared with other established forecasting methods.

Keywords: `Boot.EXPOS` · Forecast · Time series · Seasonal-Trend decomposition by *Loess*

1 Background and Motivation

A time series is a sequence of observations indexed by time, typically arranged at evenly spaced intervals and correlated. Time series studies have applications in almost all areas where Statistics is used, including economics, finance, the

Supported by national funds through FCT - Fundação para a Ciência e a Tecnologia under the project UIDB/00006/2020.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
O. Gervasi et al. (Eds.): ICCSA 2023 Workshops, LNCS 14112, pp. 140–149, 2023.
https://doi.org/10.1007/978-3-031-37129-5_12

environment, and medical sciences. The most interesting and ambitious objective of time series analysis is forecasting future values based on past values.

Models are commonly fitted and used to predict future values of a time series. The most widely used forecasting methods are Exponential Smoothing Methods (EXPOS, named by [17]). These methods refer to a set of techniques that can be used to model and obtain forecasts. This versatile approach continually updates a forecast emphasising the most recent experience, that is, the recent observations receive more "weight" than the older ones. Its major attributes are its simplicity and robustness.

The *bootstrap* methodology has been widely applied in numerous research areas, including time series analysis, to produce estimates. This resampling technique is a popular methodology for independent data due to its simplicity and nice properties [18]. It is a computer-intensive method that provides solutions in situations where the traditional methods are unsuccessful or difficult to apply. However, classical Efron's *bootstrap* has been shown to be inefficient in the context of dependent data, such as in the case of time series, where the dependence structure must be preserved during the resampling scheme. Nonetheless, if the time series process is driven by independent and identically distributed (IID) innovations, another way of resampling may be employed. Efron's classical *bootstrap* can be easily extended to a dependent setup. This was the rationale behind the *sieve bootstrap* [1,2].

`Boot.EXPOS` is a procedure developed to obtain point forecasts and forecast intervals by combining the *sieve bootstrap* and EXPOS. Firstly, the Akaike Information Criterion (AIC) is employed to select the "best" exponential smoothing method. Secondly, the *sieve bootstrap* scheme is performed on the EXPOS residuals, and through a backward process, a sample path is obtained. Ultimately, a replica of the original time series is generated by using the fitted values of the best EXPOS and the sample path. Forecasts are obtained by employing the exponential smoothing parameters of the "best" EXPOS. The performance of `Boot.EXPOS` was compared with two widely used forecasting methodologies, EXPOS and ARIMA, as well as with the best forecasting methods used in M3[1] time series forecasting competitions [11,23]. Given reasonable assumptions such as stationarity, `Boot.EXPOS` has been demonstrated to be an effective tool for forecasting time series [11–14].

Inspired by [25], the suggestion is to decompose the time series into components using the Seasonal-Trend decomposition by *Loess* and forecasting each sub-series. The best STL fit is chosen based on an accuracy measure using the function `stl.fit` [10,15], which runs all possible parameters combinations, culminating in the selection of the optimal smoothing parameters linked to the best fit. The sub-series that will be forecast are the seasonal component and the seasonally adjusted time series (trend and irregular components), as in [20]. This research proposes the use of the `Boot.EXPOS` procedure for forecasting those sub-series and aggregating them into a single forecast. Furthermore, accuracy measures such as RMSE, MAE, and MAPE are employed to evaluate the

[1] More information in https://en.wikipedia.org/wiki/Makridakis_Competitions.

forecasting performance of the combined procedure. The initial findings are promising, indicating that the proposed forecasting combination has the potential to be a competitive approach.

This paper is structured as follows: Sect. 2 provides a description of the methodologies used. In Sect. 3, forecasting experiments are carried out on real time series of mean sea levels in the Pacific, and the results are presented based on out-of-sample point forecast evaluation. Finally, Sect. 4 concludes with closing remarks and additional comments.

2 Methodologies in Action

This study incorporates a combination of some procedures for time series decomposition and forecasting, described in the following subsections, aiming to obtain accurate forecasts.

2.1 `Boot.EXPOS` procedure

The `Boot.EXPOS` method involves selecting an appropriate EXPOS model based on the AIC criterion, followed by investigating the residuals for any remaining patterns. The error component is isolated and checked for stationarity. If it is not stationary, data transformation or differentiation is necessary. If stationarity is present, the residual sequence is filtered by an autoregressive model to estimate the autoregressive coefficients and obtain innovations. In the case of autoregressive (AR) models, the *bootstrap* can be applied by resampling the centred residuals and generating a dataset using the estimated AR coefficients and the resampled residuals. Since the AR residuals are IID, the classical *bootstrap* can be used. It is an easy procedure that has been considered a good alternative to conventional estimation and forecasting methods. The EXPOS fitted values and the reconstructed series are used to create a sample path of the simulated data. Forecasts are generated using the initial EXPOS model. This *bootstrap* process is repeated B times, and information is recorded in a matrix. An "optimal" point forecast is obtained by averaging each column of the matrix. More details can be found in [11–14].

2.2 Time Series Decomposition by *Loess*

***Loess* Smoother.** Local regression methodology states that a regression function $g(x)$ with predictor x can be locally approximated by the value of a function in some specified parametric class. This is especially useful when a parametric function for the regression surface is not known. Such local approximations are obtained by fitting a regression surface to the data points within a chosen neighbourhood of the point x. There are several smoothing algorithms to proceed in fitting local polynomials. The focus here is on *Loess* (LOcally wEighted Scatterplot Smoothing) [4–9]. *Loess* is a computationally intensive smoothing algorithm that implements a nonparametric approach for estimating regression surfaces.

It is a very flexible procedure that does not need special assumptions about the parametric form of the regression surface and is a robust fitting method to the presence of outliers in the data. At a particular point x (or observation x_i in a data set), the loess fitting is obtained using weighted least squares to fit typically linear or quadratic polynomial functions of the predictors to a given number of points (or observations) in the neighbourhood of x. This number of points $n_\alpha = min\,(n, \lfloor \alpha n \rfloor)$, is determined by a smoothing parameter α and controls the smoothness of the estimated surface by local regression. Furthermore, α indicates the proportion of the data used for local regression and, therefore, should be a positive value between 0 and 1. The weighting scheme follows a smooth decreasing function assigning weights depending on their distance to x. In time series applications, the parameterization of loess smoothing span usually consider an integer $q = \lfloor \alpha n \rfloor$ for the number of closer neighbours (similar to the general case for $\alpha \leq 1$).

STL Decomposition and Forecast. A time series can be seen as the result of the combination of the following behaviours: T, a trend component, which represents an upward or downward movement over the time horizon; S, a seasonal component, which is a repetitive pattern over time; and I, an irregular component, remaining after the other components have been removed. One of the oldest commonly used methods is the time series decomposition method. The additive is one type of decomposition model and is described as

$$Y_t = T_t + S_t + I_t,$$

where $t = 1, \cdots N$, is the time period and N its length. The classical decomposition methods of time series allow to obtain estimates for the trend, the seasonality and the irregular components. However, these methodologies do not allow for a flexible specification of the seasonal component, and the trend component is generally represented by a deterministic time function, which is easily affected by outliers.

There are several methods that deal with this type of decomposition, but some time series recalls a more flexible approach, capable of identifying a seasonal component that changes over time, can deal with nonlinear trends and is robust in the presence of outliers. With these premises, the Seasonal-Trend decomposition of time series based on *Loess* [3] was chosen. The STL decomposition is already available within the R software through function `stl` [24].

To use STL, users typically need to pre-define the *Loess* smoothing parameters either based on their knowledge of the data or using default values. However, an algorithm proposed in [15] overcomes this limitation by automatically selecting the optimal smoothing parameters that minimise the error measure. The algorithm selects the best STL model with the smallest error measure achieved using a specific combination of s.window and t.window (see [15]). This approach is particularly useful for time series studies that require evaluation of inter-annual variability. The forecasting method proposed in [25] establishes the

forecast as a linear combination of the individual forecasts for the trend, seasonal and error components obtained previously through STL decomposition. As described in [25], the decomposition procedure is supported by literature that points to improved forecasting performance of the models when distinct sub-series are used for their estimation. Assuming each sub-series follows its own structural model, the forecasting method is chosen according to accuracy measures. The final forecast is the aggregation of the extrapolations of each sub-series. In [25], four forecasting methods, namely ARIMA, Theta, Holt's Damped Trend, and Holt-Winters methods, were used as benchmarks to compare against the obtained results of the forecast approach proposed. Compared to standard statistical techniques, this approach performs relatively well in producing long-term forecasts for both the NN3 [16] and M1 [22] competition datasets. See [25] for more details.

2.3 Combining Procedures

The approach developed for the current study uses the STL decomposition [3]. As already explained, a remarkable characteristic of the STL decomposition is its robustness in the presence of outliers. Note that these observations are problematic since they influence parameter estimation (e.g., ARIMA models), which will mislead any statistical conclusion. Similar to all non-parametric regression methods, STL requires the subjective selection of smoothing parameters. The two main parameters are the seasonal (s.window) and trend (t.window) window widths. Therefore, to overcome this limitation, the `stl.fit` procedure [10,15] was used since it tries to obtain an "objective" choice of the STL smoothing parameters. Based on an accuracy measure, this algorithm selects the best STL model for each combination of s.window and t.window. This procedure has been developed to obtain "automatically" the seasonal and trend smoothing parameters by minimising an error measure.

After decomposing the time series using `stl.fit`, the `Boot.EXPOS` procedure is used to forecast the sub-series, and the "optimal" forecast will be obtained by summing the forecasts. The references [20,25] are the basis of this combined procedure, where the underlying idea was that better prediction accuracy could be achieved if the time series is split into smaller parts (or sub-series). In case of uncorrelated errors, the forecasting method is developed where each forecast combines the forecast obtained for the trend and seasonal component.

3 Mean Sea Level Case Study

Sea level trends, interannual and decadal variability, are types of patterns of sea level changes that have been studied due to their importance in forecasting mean sea-level rise for this century. This area of research is of great interest and has been the focus of numerous studies on time series of mean sea level.

This paper considers four monthly time series of the mean sea level from four locations in the Pacific Ocean[2] as seen in Fig. 1. Generally, a seasonal pattern and an upward trend are observed at different increase rates.

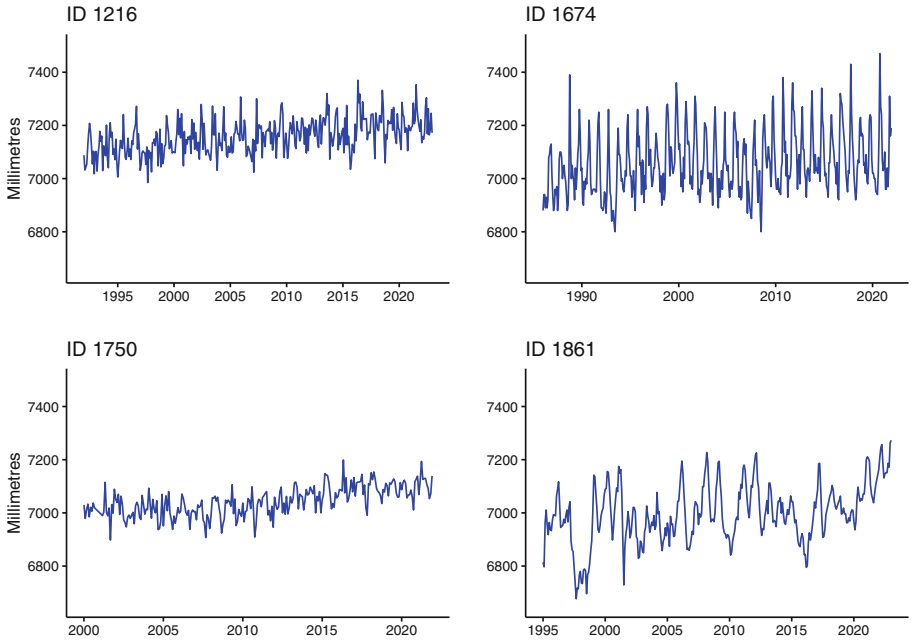

Fig. 1. The mean sea level time series from four locations in the Pacific Ocean.

The four time series are decomposed using `stl.fit` procedure, but it was necessary to check the existence of outliers according to [3]. Only the time series ID 1861 shows outliers, as seen in Fig. 2.

Furthermore, nine forecasting algorithms are used in this case study, and they are the following:

1. `auto.arima`: Fit best ARIMA model to univariate time series with AIC;
2. `ets`: Exponential smoothing state space model;
3. `bats`: Exponential smoothing state space model with Box-Cox transformation, ARMA errors, Trend and Seasonal components;
4. `baggedETS`: Bagging Exponential Smoothing Methods using STL Decomposition and Box-Cox Transformation;
5. `Boot.EXPOS`: Exponential smoothing state space model (`ets`), AR errors and classical *bootstrap* [11,12];
6. `stl + Boot.EXPOS`: `stl` followed by `Boot.EXPOS`;

[2] Permanent Service for Mean Sea Level (PSMSL), 2023, "Tide Gauge Data", Retrieved 7 April 2023 from http://www.psmsl.org/data/obtaining/.

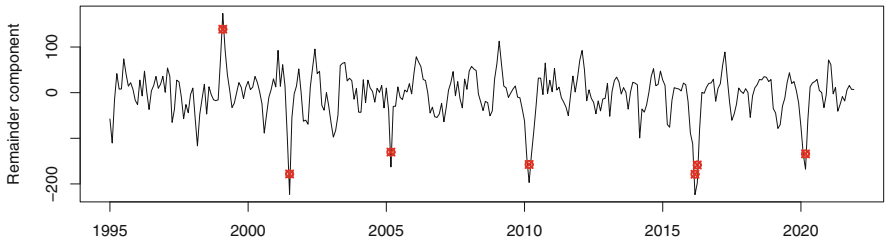

Fig. 2. ID 1861: outliers detection according to [3].

7. `stl.fit + Boot.EXPOS`: `stl.fit` followed by `Boot.EXPOS`;
8. `stlf + ARIMA`: `stlf` forecast followed by ARIMA;
9. `stlf + ets`: `stlf` forecast followed by `ets`.

The procedures 1, 2, 3, 4, 8 and 9 are available in package `forecast` [19,21] in R software. In the previous procedures, the model selection was made using the AIC criterion, the *Mean Squared Error* (MSE) was the criterion adopted for the estimation of the parameters. Moreover, for algorithms 6 to 9, the robust approach was carried out for time series ID 1861 due to the presence of outliers, as seen in Fig. 2.

In order to evaluate and compare the performance of the algorithms, the time series is separated into two parts: the fitting set and the validation set, as seen in Table 1. All the procedures enumerated are applied to the fitting set $\{y_1, \cdots, y_{n-h}\}$ and forecasts are obtained. The validation set $\{y_{n-h+1}, \cdots, y_n\}$ is used to evaluate the forecasting capacity using some accuracy measures presented in Table 2. Let y_t denote the observation at time t and $\hat{y}_t$ the forecast of y_t, $t = 1, \cdots, n$. The forecast error is defined by $e_t = y_t - \hat{y}_t$. The forecasts are computed for a hold-out period $\hat{y}_n(1), \cdots, \hat{y}_n(h)$ and compared with the true values (the validation set) using the criteria in Table 2.

Table 1. The sea level time series.

ID	Station name	Fitting set	Validation set
1216	Spring Bay, Australia	January 1992–December 2021	Year 2022
1674	Quarry Bay, Hong Kong	January 1986–December 2020	Year 2021
1750	Napier, New Zealand	January 2000–December 2020	Year 2021
1861	Honiana-B, Solomon Islands	January 1995–December 2021	Year 2022

Table 2. Accuracy measures.

Acronyms	Definition	Formula
RMSE	Root Mean Squared Error	$\sqrt{(mean(e_t^2))}$
MAE	Mean Absolute Error	$mean(\lvert e_t \rvert)$
MAPE	Mean Absolute Percentage Error	$mean(100 \left\lvert \frac{e_t}{y_t} \right\rvert)$

Table 3. Results of the accuracy measures.

ID	Procedure	RMSE	MAE	MAPE
1216	auto.arima	53.62	48.56	0.67
	ets	**50.03**	**44.76**	**0.62**
	bats	55.79	49.73	0.69
	baggedETS	**49.08**	**43.66**	**0.61**
	Boot.EXPOS	**49.15**	**44.26**	**0.62**
	stl + Boot.EXPOS	50.81	45.18	0.63
	stlfit + Boot.EXPOS	66.16	56.44	0.79
	stlf + ARIMA	57.26	50.05	0.70
	stlf + ets	62.73	53.20	0.74
1674	auto.arima	76.29	65.90	0.93
	ets	102.48	83.02	1.18
	bats	78.76	63.23	0.90
	baggedETS	81.07	63.08	0.90
	Boot.EXPOS	**68.63**	**53.56**	**0.76**
	stl + Boot.EXPOS	**69.19**	**54.07**	**0.77**
	stlfit + Boot.EXPOS	**66.50**	**53.16**	**0.75**
	stlf + ARIMA	75.47	61.91	0.88
	stlf + ets	93.71	77.77	1.10
1750	auto.arima	39.31	32.81	0.46
	ets	46.48	38.84	0.54
	bats	40.01	33.66	0.47
	baggedETS	42.10	34.37	0.48
	Boot.EXPOS	48.02	40.37	0.57
	stl + Boot.EXPOS	**33.65**	**27.12**	**0.38**
	stlfit+Boot.EXPOS	**33.00**	**28.81**	**0.40**
	stlf + ARIMA	**30.30**	**24.43**	**0.34**
	stlf + ets	41.81	36.70	0.51
1861	auto.arima	127.18	118.01	1.64
	ets	68.49	61.60	0.86
	bats	126.59	117.22	1.63
	baggedETS	78.86	73.35	1.02
	Boot.EXPOS	**67.67**	**60.53**	**0.84**
	stl + Boot.EXPOS	**63.89**	**55.55**	**0.77**
	stlfit + Boot.EXPOS	**63.99**	**55.73**	**0.77**
	stlf + ARIMA	68.53	61.35	0.85
	stlf + ets	68.72	61.56	s 0.85

Note: The three best results are highlighted in **bold**.

Forecasts for twelve months (h=12) are obtained using the nine procedures, and their results are presented in Table 3. Based on these results, there is evidence that the forecasts achieved by the proposed combined procedures are in the top three best methods, except for the time series ID 1216. Concerning this

time series, one possible solution for addressing this weak point could be to incorporate the option of implementing data transformation during pre-processing to stabilise the variance, such as the Box-Cox transformation.

4 Comments and Concluding Remarks

In this study, some procedures for time series decomposition and forecasting were combined in order to obtain accurate forecasts. The approach is a computational procedure for forecasting based on the combination of two procedures, `stl.fit` and `Boot.EXPOS`. The aim was to take advantage of flexible methods with fewer assumptions and provide properties for the estimators considered. This combination was applied to four real time series of mean sea levels with different lengths, seasonal patterns, trends and types of variability. The results were compared with other competing and established forecasting procedures, such as exponential smoothing and ARIMA models.

Furthermore, future work will involve the implementation of this procedure on various sets of time series and compare it with other forecasting procedures. This will constitute a major challenge to evaluate the effectiveness of the combined forecasting approach.

Acknowledgments. The authors thank the three referees for their constructive comments and valuable suggestions, which led to improvements to this work and will contribute to future work. The authors are partially financed by national funds through FCT – Fundação para a Ciência e a Tecnologia under the project UIDB/00006/2020.

References

1. Alonso, A.M., Peña, D., Romo, J.: Forecasting time series with sieve bootstrap. J. Stat. Plan. Infer. **100**(1), 1–11 (2002). https://doi.org/10.1016/s0378-3758(01)00092-1
2. Bühlmann, P.: Sieve bootstrap for time series. Bernoulli **3**(2), 123 (1997). https://doi.org/10.2307/3318584
3. Cleveland, R.B., Cleveland, W.S., McRae, J.E., Terpenning, I.: Stl: a seasonal-trend decomposition. J. Off. Stat **6**(1), 3–73 (1990)
4. Cleveland, W.S.: Robust locally weighted regression and smoothing scatterplots. J. Am. Stat. Assoc. **74**(368), 829–836 (1979). https://doi.org/10.1080/01621459.1979.10481038
5. Cleveland, W.S., Devlin, S.J.: Locally weighted regression: an approach to regression analysis by local fitting. J. Am. Stat. Assoc. **83**(403), 596–610 (1988). https://doi.org/10.1080/01621459.1988.10478639
6. Cleveland, W.S., Devlin, S.J., Grosse, E.: Regression by local fitting: methods, properties, and computational algorithms. J. Econometr. **37**(1), 87–114 (1988)
7. Cleveland, W.S., Grosse, E.: Computational methods for local regression. Stat. Comput. **1**, 47–62 (1991)
8. Cleveland, W.S., Grosse, E., Shyu, W.M.: Local regression models. In: Statistical Models in S, pp. 309–376. Routledge, Abingdon (2017). https://doi.org/10.1201/9780203738535-8

9. Cleveland, W.S., Loader, C.: Smoothing by local regression: principles and methods. In: Contributions to Statistics, pp. 10–49. Physica-Verlag HD (1996). https://doi.org/10.1007/978-3-642-48425-4_2
10. Cordeiro, C.: stl.fit(): Function developed in Cristina et al. (2016). https://github.com/ClaraCordeiro/stl.fit
11. Cordeiro, C., Neves, M.: Forecasting time series with boot.expos procedure. REVSTAT-Stat. J. **7**(2), 135–149 (2009)
12. Cordeiro, C., Neves, M.M.: Boot.EXPOS in NNGC competition. In: The 2010 International Joint Conference on Neural Networks (IJCNN). IEEE (2010). https://doi.org/10.1109/ijcnn.2010.5596361
13. Cordeiro, C., Neves, M.M.: Predicting and treating missing data with boot.expos. In: Advances in Regression, Survival Analysis, Extreme Values, Markov Processes and Other Statistical Applications, pp. 131–138. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-34904-1_13
14. Cordeiro, C., Neves, M.M.: Forecast intervals with Boot.EXPOS. In: Pacheco, A., Santos, R., do Rosário Oliveira, M., Paulino, C.D. (eds.) New Advances in Statistical Modeling and Applications. STAS, pp. 249–256. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-05323-3_24
15. Cristina, S., Cordeiro, C., Lavender, S., Goela, P.C., Icely, J., Newton, A.: MERIS phytoplankton time series products from the SW Iberian peninsula (Sagres) using seasonal-trend decomposition based on loess. Remote Sens. **8**(6), 449 (2016). https://doi.org/10.3390/rs8060449
16. Crone, S.F., Hibon, M., Nikolopoulos, K.: Advances in forecasting with neural networks? empirical evidence from the NN3 competition on time series prediction. Int. J. Forecast. **27**(3), 635–660 (2011). https://doi.org/10.1016/j.ijforecast.2011.04.001
17. DeLurgio, S.A.: Forecasting Principles and Applications. McGraw-Hill/Irwin, New York (1998)
18. Efron, B.: Bootstrap methods: another look at the jackknife. Ann. Stat. **7**(1) (1979). https://doi.org/10.1214/aos/1176344552
19. Hyndman, R., et al.: forecast: Forecasting functions for time series and linear models (2023). https://pkg.robjhyndman.com/forecast/. R package version 8.21
20. Hyndman, R.J., Athanasopoulos, G.: Forecasting: Principles and Practice, 3rd edn. OTexts, Melbourne (2021). https://otexts.com/fpp3/. Accessed Mar 2023
21. Hyndman, R.J., Khandakar, Y.: Automatic time series forecasting: the forecast package for R. J. Stat. Softw. **26**(3), 1–22 (2008). https://doi.org/10.18637/jss.v027.i03
22. Makridakis, S., et al.: The accuracy of extrapolation (time series) methods: results of a forecasting competition. J. Forecast. **1**(2), 111–153 (1982). https://doi.org/10.1002/for.3980010202
23. Makridakis, S., Hibon, M.: The m3-competition: results, conclusions and implications. Int. J. Forecast. **16**(4), 451–476 (2000). https://doi.org/10.1016/s0169-2070(00)00057-1
24. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2023). https://www.R-project.org/
25. Theodosiou, M.: Forecasting monthly and quarterly time series using STL decomposition. Int. J. Forecast. **27**(4), 1178–1195 (2011). https://doi.org/10.1016/j.ijforecast.2010.11.002