

31 Oct. 2022, Kgs. Lyngby, Denmark.

Letter with answers to the review of the paper “onlineforecast: An R Package for Adaptive and Recursive Forecasting” received 23. July 2022.

Dear Editor in Chief, Cathrine Hurley

Sorry for not meeting the 2 months deadline, but I had some personal issues leading to a delay.

We thank a lot for the reviews and comments. We really feel they have helped to improve the manuscript! There are parts which simply have been removed, as they were general and not directly explaining the functionalities of the package, and we have modified most of the text according to the good reviewer comments. The text is now shorter and more concise, and the parts added explain more on the functionality, especially the example section has been slimmed and parts explaining the design choices and more on the functionality have been added.

We hope the new version is ready for publication, we think so!

We provide answers to the reviewers below. The answers are marked as dotted list items.

We have as requested in addition submitted a version with changes marked. Red marks deleted and Blue marks added. Note that some parts have been moved, thus they are marked as deleted where they were moved from and added where they were moved to (also some deleted parts in moved parts are marked).

Finally, we didn't manage to keep the marking in the Example Section, since too much was edited and the markup didn't work for the code chunks.

Best regards on behalf of the authors

Peder Bacher

Reviewer 1

- Thanks for the very important comments!

onlineforecast: An R Package for Adaptive and Recursive Forecasting

The `onlineforecast` packages provides a very comprehensive documentation website, which is great. The research problem on online and recursive forecasting is interesting and valuable to the community.

Major comments

1. Why do authors decide to use R6 or object-oriented programming? How does OOP benefit users or simplify the development? If it is helpful, can the authors expand more on the motivation and advantages of using R6 rather than S3, to this particular problem?
- Good point. We have added a paragraph discussing this in the beginning of the example section. We hope this clarifies the design choices.
2. The article is lengthy to the RJ standard. Since the pkgdown site is well-documented, the section on “Example with R code” can be re-written to focus on the high-level workflow rather than the interface. Like what reviewer 1 pointed out, wrapping functions into strings puts the article into a disadvantaged position.
- We have generally slimmed the text removing non essential parts.
- We agree that the paper can be shortened w.r.t. the example, so we have cut down the example section to only contain the minimum of details needed for a working example, referring to the vignettes where appropriate, it makes the section more accessible to the reader, and if some want to use it, they will go the vignettes anyway. We have added some parts, to make the functionality somewhat more transparent, e.g. how inputs are defined with simple rcode and how they are evaluated (rather than using formula class).
- We have also removed the transformations appendix, we will put that on the website, you are right those details were not needed in the paper.
- We have added a comment on the decision to use text rather than expressions for the formulas. It makes really no difference, except that we have often found ourselves writing formulas in strings, since that are easier to manipulate, and then converting them to formulas (or expressions, which of course is done when executing them for fitting a model in the package). So we decided to pass them as strings in the first place!

3. Re “Two-stage modelling procedure” section, I’d suggest the authors to shift the emphasis from some basic concepts such as transformation and regression, to online and recursive forecasting (i.e. the major contribution).
- Here we have kept some focus on the transformation and regression, since it’s explaining the functionality of the multi-horizon model set up. We think there is only a minimal of “basic concepts” in the explanation now, it’s the low-pass filter, however we find it important to keep, as it enables the “AR” dynamics of ARMAX models in the multi-horizon setup. We have taken out the transformation appendix as mentioned above.

Reviewer 2

- Thanks for the very detailed and to the point comments!

The package `onlineforecast` show some functionality for forecasting with external regressors, adding a few variable transformation and supporting recursive estimation to avoid reestimating the whole model when new data is acquired.

Content of the paper It is not clear what this package adds on top of other existing forecasting packages. For example, `forecast` package already implements most of the functionality, including time series models (ARIMA class) with external regressors. It also has functionality for fourier coefficients, and the spline transformation and low pass filtering are done in a previous step, these are also implemented in other R packages.

- It's right that there is already this functionality, however not for optimizing multi-horizon forecasts, and using forecasts as model inputs. The ARIMA optimizes the one-step forecasts, which is optimal for LTI systems and measured inputs.

I believe most of the contribution of the package is an automatic way of fitting the hyperparameters of these transformations. Another potential innovation over existing packages seems to be the recursive least squares estimation, an important speedup compared to OLS, but it is only a speedup. The additional properties claimed for recursive least squares (the forgetting factor) is just some reweighting of some of the observations in the dataset, something that is already supported in most packages that do regression on some form (e.g. the `lm` function in base R).

- Yes, in principle one can say it is only a speed up, however without this speed up, tuning parameters etc. simply becomes infeasible, especially for the multi-horizon set up.

Moreover, I believe that nowadays gradient descent is the standard method for optimizing big, costly problems, and it also have the recursive property, because one can start from the previously found solution.

- Yes, this is not wrong, however the computational costs are still considerable. With the RLS and the implemented multi-horizon approach, it's a really effective and fast way of fitting such models.

The focus on the closed-form recursive least squares also has limitations, for example, what if we want to optimize for other errors (not the squared loss), such as absolute error. What about probabilistic forecasts (and calibrating them by minimizing CRPS)?

- We are working on implementations of recursive quantile regression, which in combination with CRPS minimization, will use the same multi-horizon setup to generate probabilistic forecasts. So, that is future work.

I also believe that the time series models (ARIMA) with external features is not implemented in the recursive least squares framework?

- It's quite clear that the MA part is not directly implemented (we suggest to run an AR on the residuals), but using external features is implemented.

Overall, I would like to see a more critical comparison with the existing “literature” (packages), not only listing them in the introduction, that adds little value to the paper (the list can be easily consulted). What is onlineforecast doing that is not available on other packages? What are the differences in usability, scalability, etc.?

- We have in the literature review in the introduction added a short comparison to the onlineforecast package for each “family” of models we refer to. We hope that this clarifies which functionalities are new compared to existing packages.

**** Software usability ****

The integration with the R ecosystem seems poor, no reference to ts objects or the more recent tsibble for the tidyverse, so users need to be constantly transforming to and back from the onlineforecast data structures to do any kind of analysis in R.

- Well, yes, however there are many formats of data now, so we had to make one which supported the functionality for the multi-horizon setup. It's however easy to transform to a “flat” data.frame or data.table, from the data.list structure, which is rather simple, it's just a list of data.frames.

The package is designed in an object oriented programming way, as opposed to the common R formulas or just passing some matrices. Some examples: `model$add_inputs(I = “bspline(tday, df=5) %**% I”)` is passing strings, not R objects that can be debugged or inspected, to the formula. If users want to automate something, they need to transform from common variables and datatypes to strings. Another example of obscure syntax: `lm_fit(c(Ta__a1=0.8), model, D, scorefun=rmse, returnanalysis=FALSE)`, `Ta__a1` is referring to a parameter that was defined in a string passed to the model, and using a different string! Another example, to add interactions to the models, the package uses its own `%**%` syntax.

- We acknowledge that the syntax has its own peculiarities, which are deviating from the regular formula syntax. The reason is, that the usual syntax did not allow us to provide the functionality needed for the multi-horizon setup provided with the package. We have made a “simpler” syntax for defining inputs, as it is just R code that must evaluate on forecast matrices (or lists of forecast matrices), and eventually we can test the code by running the `“model$transform_data(D)”`, which returns the transformed data and we can debug etc. exactly as always when executing R code. We have added more attention to this in the examples section.

- Yes, the double underscore syntax is a bit peculiar... however, it's made to enable the use of the `optim()` way of defining the parameters for optimization in the first argument of the function to optimize... and to condense the code, so sometimes syntax becomes a bit special.

The format of the paper

Extremely cumbersome to read, I do not exactly know what the models implemented in the package are. A huge portion is devoted to basic concepts such as variable transformations, least squares, and the notation is very cumbersome. Moreover the explanations of well-know concepts are poor quality: For example, I do not understand what this means: "In the RLS scheme the coefficients are calculated recursively, meaning that they are updated at each time t when stepping through the period. In each update only the "newly" obtained data at t is used, thus only past data is used in the calculations, which makes the calculated RLS coefficients and predictions "out-of-sample" (opposed to LS)." I would suggest that the authors remove these parts and focus on clarifying what are the functionalities of the package.

- Yes, the notation becomes "cumbersome" when we deal with the multi-horizon models! There is really no way out of that, so since we want to describe the functionality exact with the math, so we have kept the model formulation and definitions. We have tried to clean up the text in the mentioned parts, they were not so clear, thanks for the comment.
- We think that we clarify the functionalities as good as we can! it is the purpose of the paper. We have boiled down the examples section and we focused a bit more on the functionalities in that section. The rather comprehensive vignettes covers many aspects with examples, so we think that we have found a good compromise now.

Reviewer 3

- Thanks for the rather nice comments!

This article is a well-written good quality article introducing a great contribution to the R community, an online forecasting package. The article is well-detailed with sufficient descriptions of the overall approach/methodology, explanatory code examples and good documentation. I would like to suggest the following minor feedback for further improvement of the paper and the relevant software.

Background:

I found the background section ‘Time series modelling and forecasting in R’, a bit mixed up capturing many concepts such as forecasting, model evaluation etc in a single section. Also, given that the, most important contribution of the software is the online learning counterpart, I would like to see a subsection of literature dedicated for this. Even models such as ETS, SETAR have the functionality to update their coefficients over time, although done in an offline fashion, not online. Details of these would enrich this discussion in the paper.

- Yes, the mentioned models have also these functionalities, however we feel it’s a bit different scope, so we haven’t found the space for this, as we have tried to shorten down the paper, since it’s already a bit long!

Also, the authors mention multi-step forecasting and application of non-linear non-parametric modelling as novelties in this work. Here I would like to mention that other software such as GluonTS [1] and DARTS [2] from the Machine Learning community already incorporates these capabilities in their models. So it is best to cite them in the paper.

- We have now included the reference to the GluonTS article, since it’s available in R via the modeltime package, which we already cited. DARTS is not an R package on CRAN, so we didn’t cite that one.

Apart from that, since the work is about modelling streaming data, other related R packages such as oddstream [3] and Hoeffding Trees [4] also deserve some discussion, although they are not directly performing forecasting.

- Although this is useful, we feel it’s a bit out of scope for the article, again, it’s already long, so we wouldn’t add more on this in order to keep the focus.

Both Prophet and its most recent version NeuralProphet [5] can be cited for non-linear models.

- We already cited Prophet, so we decide to keep that.

Please cite all the R packages that have been used as dependencies of the implemented R package (eg: R6, digest etc.).

- We have now added citation of the packages which has an article or paper.

Code:

In terms of the code, I find the functions well-documented. However, following best practices, the `object_verb` naming style for functions is preferred [6]. The authors have used mixed styles for function names (eg: `plot_ts`, `rls_predict`)

- Well, yes, it’s right. The only “reversed” naming is the related to the two regression schemes, “lm” and “rls”, so the function related to them have the prefix, so it’s to keep those functions together, but yes, it is a bit in-consistent, but sometimes it’s better to leave a bit of that to keep other things less scattered.

I have found some of the variable names are not very explanatory. For example, the `Dbuilding` variable, it is hard to understand that this refers to a dataset without referring to the documentation. So more explanatory variable names would be preferred throughout.

- It’s the only included data in the package, so we think it works out keeping the name, which yes, could be longer, however it’s a very minor detail.

The authors could also add a citation of their R package, in addition to the paper [7].

- We have already done this to the arxiv version, i.e. with `citation()`, but we have not cited other packages directly to CRAN, only if they have a corresponding article or book.

References:

- [1] Alexandrov, Alexander, Konstantinos Benidis, Michael Bohlke-Schneider, Valentin Flunkert, Jan Gasthaus, Tim Januschowski, Danielle C. Maddix, et al. 2020. “GluonTS: Probabilistic and Neural Time Series Modeling in Python.” *Journal of Machine Learning Research: JMLR* 21 (116): 1–6.
- [2] Herzen, Julien, Francesco Lässig, Samuele Giuliano Piazzetta, Thomas Neuer, Léo Tafti, Guillaume Raille, Tomas Van Pottelbergh, et al. 2022. “Darts: User-Friendly Modern Machine Learning for Time Series.” *Journal of Machine Learning Research: JMLR* 23 (124): 1–6.
- [3] Talagala P (2019). *odddstream: Outlier Detection in Data Streams*. R package version 0.5.0, <https://CRAN.R-project.org/package=odddstream>.
- [4] <https://weka.sourceforge.io/doc.dev/weka/classifiers/trees/HoeffdingTree.html>
- [5] Triebe, Oskar, Hansika Hewamalage, Polina Pilyugina, Nikolay Laptev, Christoph Bergmeir, and Ram Rajagopal. 2021. “NeuralProphet: Explainable Forecasting at Scale.” *arXiv [cs.LG]*. *arXiv*. <http://arxiv.org/abs/2111.15397>.
- [6] <https://devguide.ropensci.org/building.html>
- [7] <https://www.carlboettiger.info/2012/03/20/citing-r-packages.html>