Answer to RC2: 'Comment on gmd-2021-368', Anonymous Referee #2, 10 Jan 2022

Dear Referee,

Thanks for your comments on our manuscript. We really appreciate them and also that you took the time to run the examples available on CRAN. We are glad to read that you consider it a valuable contribution.

As deserved, below we provide a detailed answer to each of your comments.

Kind regards,

Núria Pérez-Zanón
On behalf of all manuscript authors

> In summary, I believe this tool, even though not provide any new modelling option (but this was not the aim of the authors), is a valuable contribution and has good potential to impact several sectoral applications. Nevertheless, I believe some more details must be provided to make it really accessible to the wider public (i.e., even stakeholders not particularly expert in forecasting issues) and, in some cases, even experts. In particular, I refer to the data retrieval and formatting section, which could be very “labour-intensive” as the same authors state. All the examples provided use either link to static paths (in the paper) or already pre-processed input data (vignettes). I suggest the authors go more into detail on that and provide at least one example starting from raw data.

We added a new Appendix detailing the process followed to download and homogenize the storage for the first use case based on the code provided to download and locally store the seasonal forecast data from Copernicus CDS.

However, while the Copernicus Climate Data Store is currently one of the main sources of climate datasets, there exist other data repositories (e.g.: National Center for Environmental Information NCEI). The different data repositories can deliver the datasets in different formats for both file and structure, making it challenging to create a single software/function that considers all requirements.

In section 1.2, we have added a new first step to the list of steps to explain the climate forecast post-processing chain:

• Data collection, curation and homogenization: This includes collection of data from heterogeneous remote data sources, storage and indexing into local or organisation-accessible file systems or servers, and homogenization for all data files to comply with common internal conventions. The complexity of this step can be high, particularly if the data sources do not follow community standards. This step is out of the scope of this manuscript and the CSTools toolbox, and the use of other tools such as the cds-data-downloader (https://earth.bsc.es/gitlab/es/cds-seasonal-downloader) is suggested for this purpose.

In the same section, we also clarify that the main purpose of the package is not data retrieval:

The primary aim of CSTools is, therefore, to share post-processing methods (i.e. correction methods for forecast calibration, classification methods for multi-model forecast combination or scenario selection, downscaling methods, and visualization tools) that aren't currently available in other software packages or, whether the method exists in separate software, their inclusion facilitates the comparison of the results. Because additional steps are required (i.e. data retrieval from remote servers, storage and, indexing into local or organization-accessible file systems or servers, curation and formatting, and finally loading from the file systems or servers onto RAM memory of the processing machines), we provide extra functions and scripts in order to facilitate the use of the toolbox.

Still concerning input data, another common feature of the examples offered is that they seem to rely only on global/large scale gridded datasets. In my experience, I've learned that such datasets often don't fit adequately ground observations for specific regions. If the monitoring network (e.g. rain gauges) is dense enough, it can be used in turn to prepare one's own high-resolution (let's say) dataset. It's not clear to me if/how such datasets can be included, for example for correction or validation purposes.

Since the climate forecasts are global gridded datasets, most of the applications are built on references that are also gridded datasets, such as reanalyses. For correction or validation purposes, CSTools methods could be used to post-process a climate forecast with is-situ observations. The important step would be to create two data arrays, one for the climate forecast and another for the in-situ observations, that match the temporal and spatial dimensions: by selecting the closest grid point, by averaging observations within one gridpoint, or even, by regridding the climate forecast to better select the corresponding gridpoint. Other considerations, like the spatial representativeness of the in-situ observations, should be taken into account.

Another comment concerns the structure of the three use cases provided. I suggest describing them more homogeneously and streamlining them. The third use case is a bit sacrificed, in my opinion.

We have homogenized the text of the three use cases by following the suggested scheme of Referee #1: (i) application (why?), (ii) data required/ input and at what resolution / frequency, (iii) process required from source to model, (iv) code guide, (v) output and final data visualization and (vi) interpretation.

Finally, I suggest organizing better (in a more straightforward way) the connection between functions developed and corresponding literature references, to support the user in going into details with the theoretical aspects behind them. Maybe, some synoptic tables (even as an appendix), in addition to existing text, could help.

Following this comment and others from Referee #1, we have updated table 1 to include a description and references for each function.

Category	**Function**	**Method description**	**Original code version**	**Reference**
Retrieval and transformation	CST_Load	Retrieves experiment and reference data from files stored in a common format. Includes regriding options.	Wrapper from s2dverification	Manubens et al., 2018
	CST_Anomaly*	Calculates anomalies from experiment and reference data with or without cross-validation.	Extended method from s2dverification	Manubens et al., 2018
	CST_SaveExp	Saves experimental data (with ensemble dimension) into NetCDF files (one for each start date).	CSTools development	
	CST_MergeDims	Transforms the data array with named dimension by merging two requested dimensions.	CSTools development	

	CST_SplitDim	Transforms the data array with named dimensions by splitting a requested dimension following a user-defined frequency or pattern.	CSTools development	
	as.s2dv_cube	Converts data loaded using the startR package or s2dverification Load function into a 's2dv_cube' object.	CSTools development	
	s2dv_cube	Returns a s2dv_cube object by providing the data and metadata through its arguments.	CSTools development	
Classification	CST_MultiEOFS	Applies an EOF analysis over multiple variables retaining the minimum number of principal components needed to reach the user-defined variance.	CSTools development	
	CST_WeatherRegimes*	Applies a cluster analysis based on the user-defined number of clusters. A PCA could be requested to reduce the dimensionality of the dataset.	CSTools development	Cortesi et al., 2019; Torralba, 2021

	CST_RegimesAssign*	Matches patterns with a set of reference maps (i.e. clusters from CST_WeatherRegimes) based on the minimum Eucledian distance or the highest spatial correlation.	CSTools development	Cortesi et al., 2019; Torralba, 2021
	CST_CategoricalEnsCombination	Converts a multi-model ensemble forecast into a categorical forecast by giving the probability for each category. Different methods are available to combine the different ensemble forecasting models into probabilistic categorical forecasts:	CSTools development	
		“pool” for ensemble pooling where all ensemble members of all forecast systems are weighted equally;		DelSole et al., 2013
		“comb” for a model combination where each forecast system is weighted equally;		DelSole et al., 2013
		“mmw” for model weighting.		Rajagopalan et al. 2002; Robertson et al. 2004; Van Schaeybroeck and Vannitsem, 2019

	CST_EnsClustering*	Groups ensemble members according to similar characteristics and selects the most representative member for each cluster. The user chooses which feature of the data is used to group the ensemble members: time mean, maximum, a certain percentile (e.g., 75 standard deviation) or trend over the time period.	Adaptation to CSTools	Straus et al., 2007
Downscaling	CST_Analogs*	Searches for days with similar large-scale conditions (i.e. analogs) to provide downscaled fields.		Yiou et al, 2013
	CST_RainFarm*	Implements the Rainfall Filtered Autoregressive Model which is a stochastic downscaling procedure based on the nonlinear transformation of a linearly correlated stochastic field.		Rebora et al. 2006a,b; D'Onofrio et al. 2014; Terzago et al. 2018
	CST_RFTemp	Downscales a temperature field by using a simple lapse rate correction.	CSTools development	

	CST_AdamontAnalogs	Identifies analog fields in a reference dataset, based on corresponding weather types (requires CST_AdamontQQcor beforehand)	Adaptation to CSTools	Verfaillie et al., 2017
	CST_AnalogsPredictors	Downscales precipitation and maximum and minimum temperature using analogs and considering synoptic situations and significant predictors	Adaptation to CSTools	Peral García et al., 2017
Correction	CST_BEI_Weighting*	Returns a weighted ensemble mean (or weighted terciles probabilities) according to the skill of individual members at predicting a climatological index (e.g.: NAO) (requires BEI_PDFBest and CST_BEI_Weighting beforehand).	CSTools development	Sánchez-García et al. 2019
	CST_Calibration	Member-by-member bias correction. Different methodologies are available.	CSTools development	
		"bias" corrects only the mean bias.		Torralba et al. 2017

		"evmos" applies a variance inflation technique to ensure the correction of the bias and the correspondence of the variance between forecast and observation.		Van Schaeybroeck and Vannitsem, 2011
		"mse_min" corrects the bias, the overall forecast variance and the ensemble spread by minimizing a constrained mean-squared error.		Doblas-Reyes et al. 2005 and Torralba et al., 2017
		"crps_min" corrects the bias, the overall forecast variance and the ensemble spread and minimizing the Continuous Ranked Probability Score (CRPS).		Van Schaeybroeck and Vannitsem, 2015
		"rpc-based" adjusts the forecast variance, ensuring that the ratio of predictable components (RPC) is equal to one.		Eade et al. 2014
	CST_QuantileMapping	Quantile mapping adjustment for daily (or sub-daily) data.	Extended from qmap package	Gudmundsson et al., 2012; Gudmundsson, 2016

	CST_DynBiasCorrection	Applies a bias correction between the model and the observations using the division into terciles of the local dimension 'dim' or inverse of the persistence 'theta'. Model values with lower 'dim' will be corrected with observed values with lower 'dim', and similarly for theta (requires Predictability and CST_ProxiesAttractor beforehand).	CSTools development	Faranda et al., 2017; Faranda et al., 2019
Verification	CST_MultiMetric*	Computes correlation, root mean square error and the root mean square error skill score for individual models and multi-model mean.	Extended method from s2dverification	Manubens et al. 2018. Mishra et al., 2019
	CST_MultivarRMSE*	Calculates the RMSE using multiple variables simultaneously.	CSTools development	
Visualization	PlotCombinedMap*	Plots multiple lon-lat variables in a single map according to a decision function.	CSTools development	Mishra et al., 2019; Torralba et al., 2020

	PlotForecastPDF*	Plots the probability distribution function of several ensemble forecasts. Can include tercile and extreme (above P90 and below P10) categories, individual members and a corresponding observation.	CSTools development	Soret et al., 2019; Lledó et al., 2020a
	PlotMostLikelyQuantileMap*	Plots the probability for the category with the maximum probability in each grid point.	CSTools development	Lledó et al., 2020a; Torralba, 2019
	PlotPDFsOLE	Plots two probability density gaussian functions and the optimal linear estimation (OLE) resulting from their combination.	CSTools development	Sánchez-García et al., 2019
	PlotTriangles4Categories*	Function to convert any 3-d numerical array to a grid of coloured triangles.	CSTools development	Torralba, 2019; Verfaillie et al., 2020; Lledó et al., 2020b

Below I provide some specific comments (and highlight some typos). I recommend careful re-reading of the manuscript. I hope my review helps improve the overall quality of the manuscript and makes more accessible the interesting toolbox developed.

We acknowledge this detailed review and we answer each comment below.

L 66: as illustrated in Fig. 1

Corrected.

L100: R-based

Corrected.

L104: please check this sentence

The sentence "CSTools could nonetheless be useful to research scientists, as it is made compatible some of the aforementioned R packages." has been corrected as "CSTools could nonetheless be useful to research scientists, as it has been designed to be compatible with some of the aforementioned R packages."

L130: maybe "each function"?

Corrected, thanks.

L191: to automatically interpolate

Corrected.

L193: lotlan_data for temperature? Please check

The name of the data object could be changed to something similar to "lonlat_temp" which would be more appropriate to reflect the fact that it is a temperature data sample and more coherent with the lonlat_prec data sample name. To fix this problem, we have opened an issue in the gitlab repository (https://earth.bsc.es/gitlab/external/cstools/-/issues/84) and included this change in the package.

L197: downloaded into (or simply "in")

This line has been removed.

L244: "The amount of categories can be changed and are taken as…" please check this sentence. To which subject is the verb "are" referred? To the categories?

The sentence has been checked.

Previous version: "The amount of categories can be changed and are taken as the climatological quantiles (e.g. terciles), extracted from the observational data."

New version: "The user can set up the total number of categories that will be used to define the observed climatological quantiles."

L291: not clear: is this function available only for the Iberian Peninsula? Will it be available for other areas in the future?

The mention of the Iberian Peninsula has been removed since it was originally developed for the Iberian Peninsula and generalized in CSTools to accept inputs for any region in which high-resolution observational datasets are available.

L301: not clear: here, too, is this function available only for NAO?

We have re-written these lines as follows:

Sánchez-García et al. (2019) used the North Atlantic Oscillation (NAO) to improve the skill of the seasonal precipitation forecast over the Iberian Peninsula. Given that this methodology could be explored to improve the skill of different climate variables that are led by other climate indices, the method has been generalized and named Best Estimate Index (BEI).

L375: A comparison … IS also possible

We thank the reviewer for finding out this error. Corrected.

L386: three example case studies

Corrected.

L399: the link does not work. However, I would prefer some more technical link than that to a newspaper

The link has been removed.

L401: I guess IP stands for Iberian Peninsula. But his term is used only some words before, so please check the sentence and rephrase

We have fixed it.

Previous version: Very high wind speeds were later recorded over large part of the Iberian Peninsula due to 4 cyclones going across the IP (AEMET, 2018).

New version: Very high wind speeds were later recorded over large parts of the Iberian Peninsula due to the passing of four cyclones (AEMET, 2018).

L453: by?

Removed.

L503: “only one member”: it’s better to tell how many members make up the ensemble

We completely agree with this comment and have added the clarification “out of 25” to the text.

L509: please explain what “ensemble dressing” means.

We have described the ensemble dressing procedure in section 2.2.6, and referenced that section in this part of the manuscript.

L545: I would write “agriculture and industry, while meltwater shortage …”

The suggestion has been included in the text.

L597: “the result is” (better) or “the results are”

While reviewing the text, this line has been re-shaped to: “The monthly spectral slopes obtained”. Thanks for noticing.

Figure 6a: I guess this map shows one of the 25 possible precipitation fields for 11 December 1993 given by the SEAS5 ensemble

Yes, you are right. We have corrected it as follows: “Figure 6: One out of the 25 ensemble members of the original ensemble”.

L719 (and elsewhere): please check throughout the text if there are shifts using tenses (from the present to the past and vice versa)

We appreciate this comment and we have revised the text while re-structuring the use cases.

LL719-720: not clear if these operations were made through CSTools (please refer also to main comments)

At the time of writing this manuscript, these operations were done using other software. However, the functions CST_RFTemp could be used to post-processing the temperature dataset and CST_Load (which allows regridding with CDO) could be used to do the bilinear interpolation of the rest of the variables.

L723: “the SNOWPACK model is run for each of the 21 seasonal forecasts over the hindcast period 1996-2016”. Only here the objective of the use case is clearly stated. I suggest declaring it at the beginning of the section.

We have re-written the use case and added the following sentence at the beginning to explain the objective:

“The post-processing of seasonal precipitation forecast in the Alps to be used as input for the SNOWPACK model is shown in this use case, as well as, the result of the SNOWPACK model snow depth.”

L741: again, for what period? State clearly the objectives of the exercise at the beginning of the section.

Given the rewritten of the use cases following the previous reviewer's suggestion, we consider the information is now clear.

L794: at the end of this section, I realize that the fact that the SCHEME hydrological model is used is not so relevant, after all. The case study could be generalized to any (semi-distributed or even distributed) hydrological model requiring precipitation and temperature forecasts.

Indeed. The use case is post-processing temperatures and precipitation seasonal forecasts. We have re-written the use case to follow the structure suggested by reviewer #1 but also to clearly show the target users of this use case.

L804: “(see e.g. Fig. 4)” I would remove this test in brackets.

Removed.

L813: also, agricultural issues are involved (drought, irrigation needs, water resources management, etc.)

The sentence has been modified to include this comment as:

“Similarly, these use cases are relevant for risk management of high wind speed, coastal and flooding events, as well as, agricultural issues implied by droughts, irrigation needs or water resources management.”

L815: what about the other features? I think this sentence underestimates other aspects of the tool. Please explain in more detail.

We really appreciate this comment. We have improved the sentence to highlight that apart from the methods, other aspects of the software are valuable.

Please note: Appendixes A and B are not referred to in the main text. They should be and contextualized.

We appreciate the careful review. Appendices are now cited in the text.

Appendix B: Details on data collection, curation, homogenization, and requirements for CST_Load

In order to use CST_Load, the storage needs to be homogenized. CST_Load accepts several parameters to configure the loading and interpolation of data. The CST_Load documentation in the reference manual is linked to the s2dverification (Manubens et al., 2018) reference manual where the description of all parameters is detailed.

Basically, CST_Load requires path patterns pointing to the NetCDF files or OPeNDAP URLs requested via the other parameters. A variable with a matching name must be present in the files. The path patterns, one for each experimental/observational dataset to be loaded, express the set of files comprising each dataset. Therefore, a path pattern is a string containing some specific wildcards that are recognised and replaced by the corresponding values by CST_Load. The most commonly used wildcards in a path pattern specification are “$START_DATE$”, “$STORE_FREQ$”, and “VAR_NAME”. For example, given a dataset that consists of the following files:

- /data/datasetA/monthly/tas_20180101.nc
- /data/datasetA/monthly/tas_20180201.nc
- /data/datasetA/monthly/tas_20180301.nc

The path pattern to express the set of files would be as follows:

“/data/datasetA/$STORE_FREQ$/VAR_NAME_$START_DATE$.nc”

The use case 1, “Assessing the odds of an extreme event”, directly loads wind speed (in surface for the case of SEAS5 and at 100m for the case of ERA5). However, this variable is not directly available in the Copernicus CDS while u and v wind components are in 6 hourly and monthly frequencies. In order to get the monthly wind speed, the 6 hourly frequency components are used to calculate the 6 hourly wind speed and, then, calculate the monthly average of the wind speed using CDO (Schulzweida, 2019). Notice that averaging the monthly wind components may lead to a different result. To automatise this calculation on all the files in a folder, the following bash code could be adapted:

```
path_output="./data/monthly/sfcWind_f6h/"

path_component="./data/6hourly/"
for year in {1993..2018}
do
for month in 01 02 12
do
output_file=${path_output}"sfcWind_${year}${month}01.nc"
uas_file=${path_component}"uas/uas_${year}${month}01.nc"
vas_file=${path_component}"vas/vas_${year}${month}01.nc"
cdo -L -b F64 -f nc -setunit,'m/s' \
  -setname,"sfcWind" \
  -sqrt \
    -add \
      -mul -selname,uas $uas_file -selname,uas $uas_file \
      -mul -selname,vas $vas_file -selname,vas $vas_file \
  ${output_file}_tmp
```

```
cdo monmean ${output_file}_tmp $output_file

done
done
```

An equivalent script, using CDO dailymean operator, can be used to convert the data downloaded into daily mean values for use case 2 and 3.