# Responses to Reviewer 1

***In this paper, a new postprocessing approach for the correction of forecasts' systematic errors and a quantification of their uncertainty are presented. The method is based on the use of random forests. The method also uses a local regression approach to adjust the parameters that describe the dependence of the conditional probability density function of the observations on the forecast. The method is validated using daily precipitation data and forecasts from the GFSE.***

***The paper is well written, the methodology is, in general, well described, and the results are evaluated in a statistically robust way. Below, I indicate some suggestions or comments pointing out some places in which the discussion can be clarified or in which more information should be provided.***

Thank you for taking the time to read our paper and for your constructive comments! Please find our point-by-point responses below.

***L40: This statement is unclear. In the case of random forests, the resulting fit would be smoothed out by the ensemble, so the steps would not be so obvious in the output.***

Thanks for the comment, we have clarified this statement now. You are correct that the forest ensemble will smooth the step functions to a certain degree, but the result will still be a rougher approximation of a linear function compared to estimating this directly. This should be conveyed more clearly now.

***Section 2.2. Step 2. In this section, an independence test is used to identify possible dependencies between predictors and the model parameters $\theta$. There are different variables and parameters. Could the authors elaborate more on how the split-variable is selected? Is it the one with the lowest p-value with respect to any of the parameters? Which is the independence test used in this implementation?***

The split-variable is chosen that has the lowest p-value for a test statistic that assesses all parameters (i.e., MOS coefficients) simultaneously. The tests used here are permutation tests for independence using a quadratic form as the test statistic (see Hothorn et al. 2006, 2008). We have amended the text to include this information and the additional references.

***Section 2.2, Step 3: Which are the stopping criteria for the growth of the tree? What is the minimum sample size at a leaf node in the experiments reported in this paper (particularly in those experiments where a relatively small dataset is used)?***

We use the same hyperparameter values as the distributional forests of Schlosser et al. (2019). This means that nodes must have a sample size of at least 50 in order to be split again (`minsplit` = 50) and that terminal nodes (leaves) must have a sample size of at least 20 (`minbucket` = 20). The same values are used for all of the experiments, including those with only 3 years of training data. We have added this information to Sec. 3.2.1, where specific model setups for the precipitation forecasting are documented.

***Equations 6 and 7: To my understanding, these equations describe a type of local regression in which distance is measured based on the number of times two given data points belong to the same category in the different trees of a given forest. So the distance is specific to the problem at stake. This characteristic distinguishes this method from other methods that use random forests for postprocessing in the sense that $\theta$ is not directly given by the forest, but the forest provides a way to detect predictors that are close to the current predictors, and based on these neighbor predictors, a new set of parameters can be obtained (by retraining the model using only these weighted neighbors).***

Yes, this interpretation is correct. In forests with very simple "models" in the leaves (e.g., just a mean response or a success proportion) both approaches are equivalent. Thus, simply averaging mean responses from trees in a forest yields the same prediction as computing a weighted response based on the neighborhood weights. However, for parameters from more complex models (e.g., MOS coefficients or distributional parameters) the two approaches differ somewhat and the approach based on neighborhood weights is used more often in the literature.

***Based on this, I wonder:***

***What would be the performance of the proposed technique if the parameters $\theta$ provided by the forest were used directly for the postprocessing of the forecast (i.e., what is the impact of the neighbor approach on the performance of the method)?***

Thanks for the suggestion. We have compared the two approaches now: direct averaging of the predicted parameters from the individual trees vs. re-estimation of the base model using the neighborhood weights. The results are displayed in Figure 1 below. Both methods perform virtually identical when the full 24 years of training data are available. But when only 3 years of data are available the neighborhood weights perform slightly better. We have now mentioned these new results in the text.

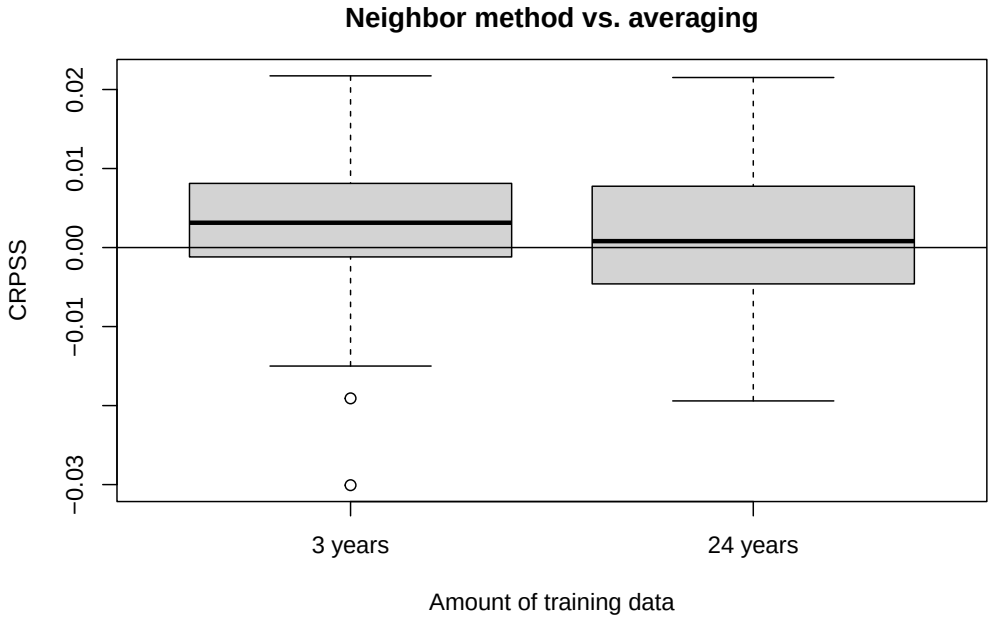


Figure 1: CRPSS of neighbor method compared to averaging method for 3 and 24 years of training data.

***What would be the performance of the method if the distance metric were replaced by the classical Euclidean norm (like in the classical nearest neighbors approach)?***

The classical nearest neighbor approach (*k*-NN) with a Euclidean norm could make sense for a distributional forest, since the terminal nodes of each individual distributional tree would be expected to contain similar values for at least some of the predictors. The approach is less suited for a MOS forest, where terminal nodes instead correspond to separate MOS models and thus require a larger range of values for (certain) predictor variable(s). Another point that would need to be considered is that the Euclidean norm is not ideal for dealing with a high dimensional predictor space. Subsequently, some form of dimension reduction would likely be required for our application.

***What is the variability of the weights? Particularly in the small training sample cases. If the weight variance is not too high in the small training sample scenarios, then this may help to increase the robustness of the method because model parameters would be trained with a relatively larger sample than in the other methods. Is there a way in which this variability can be controlled and eventually tuned as a hyperparameter to maximize the performance of the method?***

Weights of a MOS forest are generally more variable than those of a distributional forest. Unsurprisingly, the average variance of an observation weight in the training data for a specific day in the test data (Fig. 2) is significantly larger for models trained on 3 years of data. This is because larger training data sets allow for more terminal nodes – given a fixed minimum node size – and thus generally have a larger fraction of weights equal to 0. The variability of the weights cannot be tuned directly, but can be influenced by changing the number of terminal nodes in a tree. For a given training dataset, this could be achieved by changing the minimum necessary size for a terminal node.

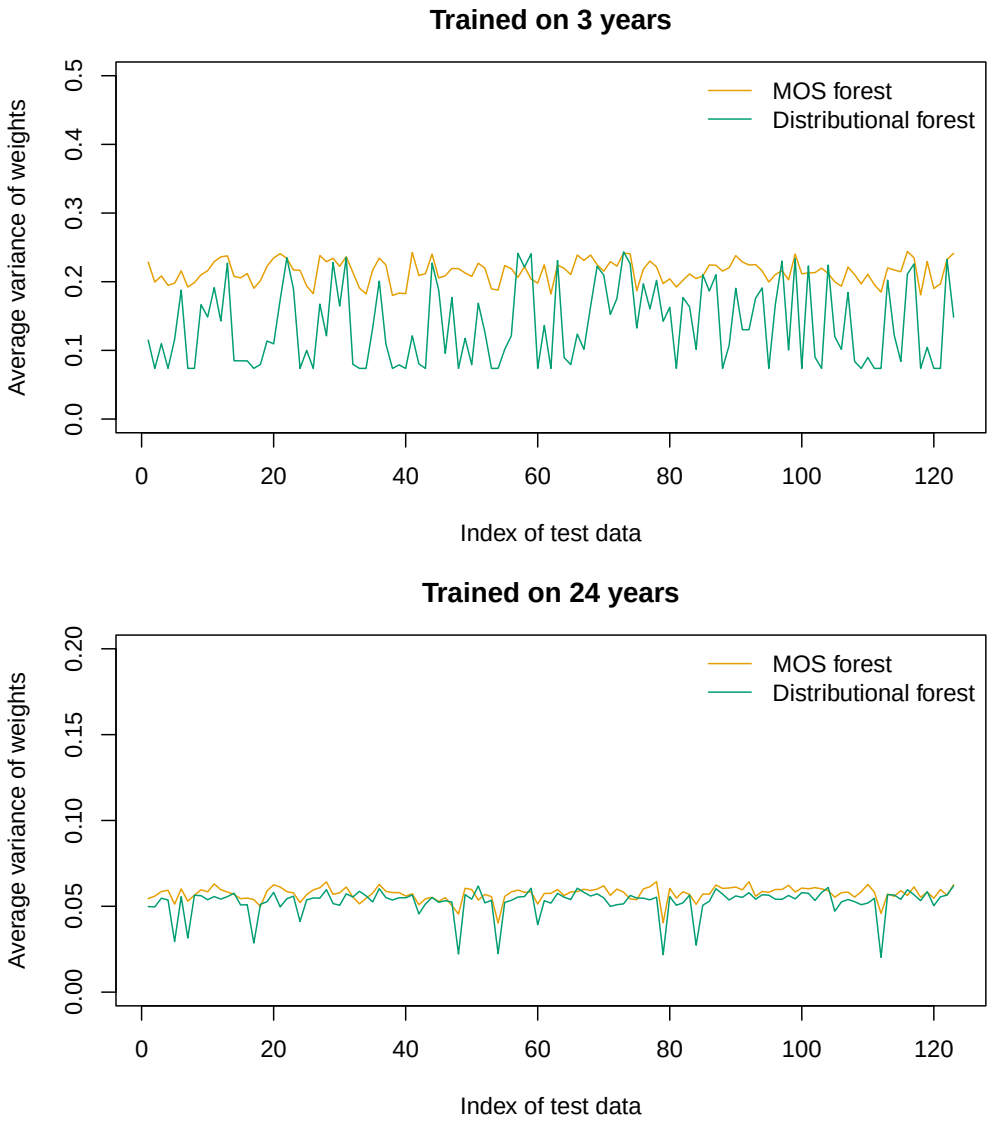


Figure 2: The average variance of the weights at Axams.

***Table 1: Could the authors elaborate more on why `tppow_mean` is excluded from the splitting variable list? It is not clear to me why that should be the case. Also, in the results section, the variable associated with the root split is the total column liquid condensate, which I assume is closely related to the precipitation rate (so the system is indirectly trying to use ttpow as a splitting variable)***

The ensemble mean of total precipitation `tppow_mean` is the "direct" weather prediction of our observations and contained within the base MOS model. It was excluded from the splitting variables to emphasize that MOS forests are able to account for additional (i.e., non "direct") variables during postprocessing. As you suspected, results are comparable when including `tppow_mean` among the splitting variables (Fig. 3), presumably due to correlations within the data.

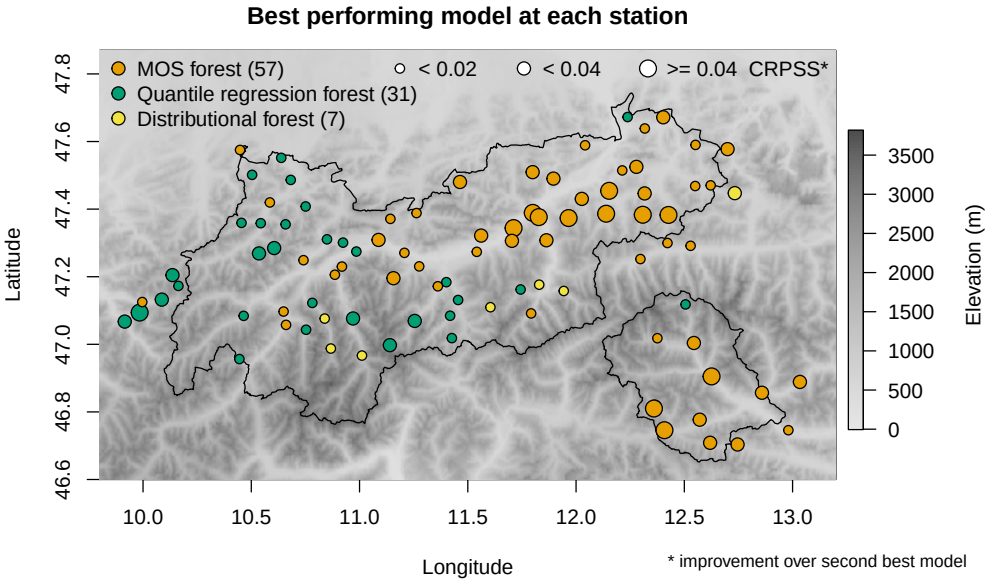


Figure 3: The best performing method at each station when including `tppow_mean` among the MOS forest splitting variables.

***Table 1: Could the authors provide here or in the text some details about the configuration of the other methods? Since overfitting is a major concern when dealing with trees and forests, indicating the tree growth stopping criteria (or any other pruning approach) would be relevant for the comparison.***

By default, distributional forests use the same growth stopping criteria as those of the MOS forests, already mentioned above: `minsplit = 50` and `minbucket = 20`. For quantile regression forests, we use the default settings of the quantregForest package with a minimum terminal node size (`minbucket`) of 10. No pruning is performed.

***Figure 2: This figure is very interesting. However, I could not find cases in which precipitation occurred without being forecast (or maybe there is only one case in node 13). Is this because of the selected nodes, or is this a general property of the dataset?***

This is a general property of the RainTyrol dataset. Although approximately 42% of the observations (across all years and stations) are zero, only 0.7% of the ensemble mean forecasts are zero. Most likely this difference in frequencies results from the bilinear interpolation scheme used to generate ensemble forecasts for each station location and from the nature of ensembles themselves. For example, to obtain a forecast of no precipitation (i.e., `tppow_mean = 0`) would require that all ensemble members at each of the four neighboring gridpoints do not predict any precipitation.

***4.1: The names given to the different predictors are not clear. For example, what does `pwat_mean_max` mean? I assume the mean is from the ensemble mean, but I cannot interpret the max. This also applies to other names: `t500_sprd_min`, `tppow_sprd1824`.***

In addition to the ensemble mean of the 24 hour precipitation forecasts between +6h and +30h, the RainTyrol dataset also contains forecasts based on 6 hourly intervals. For example, the variable `tppow_sprd1824` is the spread of the 6h precipitation forecasts issued by the ensemble for the period between 18 and 24 UTC (i.e., lead times of +18h and +24h).

Variable names can contain two underscores. The expression after the first underscore indicates how forecasts are aggregated over the ensemble dimension (i.e., `sprd` for spread of the ensemble). The expression after the second underscore describes aggregation over the lead times within the 24h accumulation period (i.e., mean, maximum or minimum over +6h, +12h, +18h, +24h, +30h). Subsequently, `pwat_mean_max` is obtained by computing the mean of the ensemble forecast of `pwat` for each lead time, and then taking the maximum of these ensemble means.

We have added information about the variables and their naming convention to the paper.

***Regarding `tppow_sprd1824`, later in the text or in a figure caption, it is said that it corresponds to the spread over the 18–24 hour lead time period. Why did the authors choose this period to characterize the ensemble spread?***

The ensemble spread over the 18-24 hour period (`tppow_sprd1824`) is one of many variables contained within the RainTyrol dataset. Summer rainfall in Tirol is often caused by convection during the late afternoon or evening hours. Including forecasts from this time period can be valuable for postprocessing since NWP biases and miscalibrations may behave differently depending on the nature of the precipitation event. We have now mentioned this in the text.

***L300 "if the variable observed is not a direct output of the NWP model". This is unclear. Why can't physical quantities other than the ones observed be used to model the conditional probability distribution parameters?***

You are right that forecasts of physical quantities other than the ones observed could be used to model the conditional probability distribution parameters, but in that case it may be difficult to specify a suitable MOS regression that works well and is still physically meaningful or natural to understand. In order to avoid any confusion though, we have removed the highlighted phrase.

***L141 $\gamma_0$ is introduced here, but it has not been defined before ( $\sigma$ is used instead in the previous discussion).***

You are correct, we have now changed $\gamma_0$ to $\sigma$.

***Equation 7: The meaning of the denominator is not clear.***

Dividing by the size of the terminal node $| \mathcal{P}_p^t |$ avoids underrepresenting trees with more (and thus generally smaller) terminal nodes when calculating the weights. We have now mentioned this in the text.

***Equation 8: Please clarify the meaning of $\phi$ and $\Phi$.***

$\phi$ and $\Phi$ refer to the probability density function and cumulative density function of a standard Gaussian distribution, respectively. This has been added to the text.

***L225 rates are***

We believe the sentence is correct as is: "The first split of the tree separates rare ($n$ = 23) weather situations ..."

***Figure 3: Please clarify the meaning of the titles of the panels ("Location" and "Scale").***

The titles refer to the location and scale parameters of the response distribution (i.e., $\mu$ and $\sigma$, respectively). This has been clarified in the figure caption.

***In L268 and Caption Fig. 5, CRPS is used instead of CRPSS***

We have changed CRPS to CRPSS at L268. The caption of Fig. 5 was technically correct since the circle sizes corresponded to differences in CRPS not CRPSS. For consistency, we have now modified the figure to instead show the CRPSS of the best method with respect to the second best method. Subsequently, new cutoffs for the circle size are approximately one order of magnitude larger.