Merging Satellite and Gauge-Measured Precipitation Using LightGBM With an Emphasis on Extreme Quantiles

Hristos Tyralis, Georgia Papacharalampous, Nikolaos Doulamis, *Member, IEEE*,
and Anastasios Doulamis, *Member, IEEE*

***Abstract*—Knowing the actual precipitation in space and time is critical in hydrological modeling applications, yet the spatial coverage with rain gauge stations is limited due to economic constraints. Gridded satellite precipitation datasets offer an alternative option for estimating the actual precipitation by covering uniformly large areas, albeit related estimates are not accurate. To improve precipitation estimates, machine learning is applied to merge rain gauge-based measurements and gridded satellite precipitation products. In this context, observed precipitation plays the role of the dependent variable, while satellite data play the role of predictor variables. Random forests are the dominant machine learning algorithm in relevant applications. In those spatial prediction settings, point predictions (mostly the mean or the median of the conditional distribution) of the dependent variable are issued. The aim of the manuscript is to solve the problem of probabilistic prediction of precipitation with an emphasis on extreme quantiles in spatial interpolation settings. Here we propose, issuing probabilistic spatial predictions of precipitation using light gradient boosting machine (LightGBM). LightGBM is a boosting algorithm, highlighted by prize-winning entries in prediction and forecasting competitions. To assess LightGBM, we contribute a large-scale application that includes merging daily precipitation measurements in contiguous United States with PERSIANN and GPM-IMERG satellite precipitation data. We focus on extreme quantiles of the probability distribution of the dependent variable, where LightGBM outperforms quantile regression forests (a variant of random forests) in terms of quantile score at extreme quantiles. Our study offers an understanding of probabilistic predictions in spatial settings using machine learning.**

***Index Terms*—Light gradient boosting machine (LightGBM), quantile regression, remote sensing, spatial interpolation.**

I. Introduction

Economic constraints limit the extent as well as the density of spatial coverage of areas with rain gauge stations. Therefore, gridded satellite datasets are used as a substitute for observed precipitation in hydrological applications. Nevertheless, gridded satellite datasets provide inaccurate estimates of actual precipitation, therefore post-processing is required by merging gridded datasets with rainfall gauge-based measurements; see the reviews by Abdollahipour et al. [1] and Hu et al. [29].

A common means to merge-gridded satellite datasets and gauge-based measurements is to apply machine learning algorithms in regression settings. In this context, satellite precipitation data play the role of predictor variables, while observed precipitation plays the role of the dependent variable. The major assumption in such settings is that the actual precipitation is represented by gauge-based measurements, albeit some studies question the accuracy of those measurements [61]. The state-of-the-art algorithm in these regression settings is Breiman's [10] random forests [3], [12], [19], [25], [26], [37], [42], [46], [68], [84]. Other machine learning algorithms also have been implemented (e.g., [35], [41], [51], [52], [59]), albeit less frequently. To better understand their performance, it is important to compare multiple algorithms using big datasets that cover large areas with dense networks of rain gauge stations [35], [51], [52].

A common characteristic of most studies merging satellite data and station observations is that spatial point predictions are issued and assessed using the squared error scoring function, the absolute error scoring function or related skill scores (e.g., the Nash-Sutcliffe efficiency and the Kling-Gupta Efficiency). The squared error scoring function is consistent (for the definition of consistency the reader is referred to Section III-D) for the mean functional of the probability distribution [22], i.e., by training a machine algorithm with a squared error scoring function, one can issue predictions of the mean of the conditional probability of the response of the regression algorithm. Similar arguments apply for the case of the absolute error scoring function (which is consistent for the median functional) as well as for the associated skill scores [22]. However, predictions are more informative when they take the form of probability distributions [23], while the requirement for probabilistic predictions in hydrology has been commented on by Papacharalampous and Tyralis [48], Papacharalampous et al. [50] and has been identified as an important problem in hydrology, in the context of uncertainty estimation in general [9].

Prediction of quantiles of the conditional probability distribution at a dense grid of quantile levels can provide an approximation of the full probability distribution [63], [64]. Our focus is on

Manuscript received 12 March 2023; revised 3 June 2023; accepted 14 July 2023. Date of publication 19 July 2023; date of current version 2 August 2023. This work was supported by the Hellenic Foundation for Research and Innovation under the "3rd Call for H.F.R.I. Research Projects to support Post-Doctoral Researchers" (Project Number: 7368). *(Corresponding author: Hristos Tyralis.)*

The authors are with the School of Rural, Surveying and Geoinformatics Engineering, National Technical University of Athens, 15780 Athens, Greece (e-mail: montchrister@gmail.com; papacharalampous.georgia@gmail.com; ndoulam@cs.ntua.gr; adoulam@cs.ntua.gr).

Digital Object Identifier 10.1109/JSTARS.2023.3297013

extreme quantiles of the conditional probability distribution (see e.g., [13], [66]), although predictions of functionals in the center of the conditional probability distribution of the response are also assessed. Extreme quantiles are of interest, given their importance for conducting flood studies. Our application includes merging the PERSIANN and the GPM-IMERG daily precipitation datasets with gauge-based daily precipitation measurements that cover the contiguous United States (CONUS).

The aim of this article is to solve the problem of probabilistic prediction of precipitation with an emphasis on extreme quantiles in spatial interpolation settings. To this end, we propose to apply the light gradient boosting machine (LightGBM) algorithm [32] trained with the quantile scoring function [33]. Applications of the quantile scoring function and the associated mean score can be found in hydrological modeling and forecasting studies, see e.g., [14], [50], [63]. LightGBM is compared with quantile regression forests (QRF, [39]), which is a variant of random forests. The assessment of LightGBM is based on its relative performance with respect to QRFs, given the strong preference for the use of random forests in spatial interpolation settings due to high predictive performance and convenience in their use [26]. Previous applications of QRF in merging satellite and gauge-based precipitation measurements include [4], [85], while applications of other machine learning algorithms that can issue probabilistic predictions in spatial interpolation settings are limited (see e.g., a deep learning application by [20]). As already mentioned, we focus on extreme quantiles while we base the comparison on skill scores whose components include quantile scoring functions. Although the assessment focuses on extreme quantiles, we further comment on quantiles at central quantile levels, where results are largely influenced by the intermittent nature of precipitation.

The remainder of this article is structured as follows. Section II presents LightGBM and QRF with emphasis on topics related to quantile regression. Section III follows with presentation of the data used in the study as well as the problem formulation and metrics for the assessment of the algorithms. Results are presented in Section IV, followed by their discussion in Section V. Section VI concludes the article.

II. Methods

We applied LightGBM to a dataset that includes gridded satellite precipitation products (see Section III-A) and gauge-based measurements (see Section III-A). Since our scope is to provide probabilistic predictions of precipitation (in particular high quantiles), LightGBM was trained using the quantile scoring function (see Section II-C). The algorithms were compared with QRF using a hold-out sample (see Section III-B). In this section, we describe LightGBM and QRF, while software implementation is provided in the Appendix. Since LightGBM and QRF are variants of boosting algorithms and random forests, respectively, we focus on certain properties that distinguish them from the respective introducing algorithms. Extended descriptions of boosting and random forests can be found in textbooks [18], [24], [31] while remote sensing scientists and technologists are familiar with them.

A. Quantile Regression Forests

QRFs [39] is a variant of random forests [10] that is used to issue probabilistic predictions. Random forest is a state-of-the-art algorithm for spatial interpolation [26] and has been extensively used in hydrological applications [68].

A random forest algorithm for regression grows an ensemble of decision trees while the prediction of the algorithm is equal to the mean of the individual trees. Building the forests of trees is done with bagging combined with randomized node optimization. Bagging (bootstrap aggregating) refers to the procedure of resampling with replacement of the training set and using this sample to train a single tree. In addition, random forests select a random subset of features at each candidate split.

QRFs define an approximation of the conditional distribution of the response variable instead of averaging predictions of trees (a procedure that approximates the conditional mean of the response variable). Properties of random forests are transferrable to QRFs. Those are summarized in [68] and include among others related to our problem at hand, high predictive performance, speed, feasibility in large-scale applications, resistance to overfitting, efficient handling of highly correlated variables, and stability.

Here we applied the R language implementation of QRFs by [74], [75], using 100 decision trees and default values of hyperparameters, since default implementation is highly efficient [68]. The specific software implementation of random forests is fast regarding computations times; however, it is extremely slower compared to LightGBM for big datasets, thus hyperparameter optimization becomes prohibitive for the large sample of the present study. Since QRFs are a regression algorithm, they are trained and predicted in the usual fashion, i.e., the training sample includes a set of observed predictor variables and a set of the observed dependent variable. The specific application is described later in Section III-B.

B. Light Gradient Boosting Machine

Gradient boosting decision trees is an ensemble learning algorithm in which decision trees are added to the ensemble sequentially. At each iteration, a new decision tree is trained with respect to the error of the algorithm so far. A gradient-descent based formulation formalized the concept of boosting [20], [38], [43]. Gradient boosting decision trees can be optimized with different scoring functions, thus they can issue predictions tailored to user's requirements. A list of properties of boosting algorithms can be found in [62]. Although boosting algorithms share some similar properties with random forests, they frequently perform better in several settings, although hyperparameter tuning is needed.

LightGBM [32] is a boosting algorithm that has some favorable properties compared to common gradient boosting algorithms. In particular, it is particularly suited for datasets with high feature dimension and large size. It uses gradient-based onside sampling that excludes data instances with small gradients (instead common boosting algorithms scan all data instances to estimate the information gain of all possible split points), thus reducing training time. Furthermore, it uses exclusive feature bundling that bundles mutually exclusive features, to reduce their number. Besides, it uses a histogram-based algorithm to find the best-split points, similar to earlier successful boosting variants (e.g., [11]).

LightGBM has multiple parameters that when tuned can increase its predictive performance. The optimization procedure is described in Section III-C. Moreover, LightGBM was trained with a quantile scoring function (see Sections II-C and III-D) to issue probabilistic predictions. Since LightGBM is a regression algorithm, it is trained and predicts in the usual fashion, i.e., the training sample includes a set of observed predictor variables and a set of the observed dependent variable. The specific application is described later in Section III-B.

C. Quantile Regression

Quantile regression algorithms are used to predict conditional quantiles in regression settings. Here, we explain how quantile regression works in practice. Hereinafter, observations will be notated with lowercase letters, while random variables will be notated by underlined lowercase letters. Let $\underline{y}$ be a random variable with cumulative distribution function $F_{\underline{y}}$ defined by

$$F_{\underline{y}}(y) := P(\underline{y} \leq y). \tag{1}$$

Then, the τth quantile of $\underline{y}$, $Q_{\underline{y}}(\tau)$ is defined by

$$Q_{\underline{y}}(\tau) := \inf\{y : F_{\underline{y}}(y) \geq \tau\}, \tau \in (0,1) \tag{2}$$

where $\inf\{\cdot\}$ denotes the infimum of a set of real numbers.

Let $F_{\underline{y}|\boldsymbol{x}}$ be the distribution of the random variable $\underline{y}$ given the p-dimensional vector $\boldsymbol{x}$

$$F_{\underline{y}|\boldsymbol{x}}(y|\boldsymbol{x}) := P(\underline{y} \leq y|\boldsymbol{x}). \tag{3}$$

Then, the τth quantile of $\underline{y}$ conditional on $\boldsymbol{x}$, $Q_{\underline{y}|\boldsymbol{x}}(\tau|\boldsymbol{x})$ is defined by

$$Q_{\underline{y}|\boldsymbol{x}}(\tau|\boldsymbol{x}) := \inf\{y : F_{\underline{y}}(y|\boldsymbol{x}) \geq \tau\}, \tau \in (0,1). \tag{4}$$

The quantile loss function $\rho_\tau(u)$ is defined as

$$\rho_\tau(u) = u(\mathbb{I}(u \geq 0) - \tau), u \in \mathbb{R}. \tag{5}$$

Here τ is the quantile level of interest and $\mathbb{I}(A)$ denotes the indicator function that is equal to 1 when the event A realizes and 0 otherwise. The quantile loss function, defined by (5), is positive and negatively oriented, i.e., the objective is to minimize it and equals to 0, when $u = 0$.

Let $\boldsymbol{\theta}$ be the parameters of a regression model (e.g., the LightGBM of Section II-B). Let $y(\boldsymbol{x}, \boldsymbol{\theta}(\tau))$ be the prediction of the regression model at quantile level τ, given values of the predictor variables equal to $\boldsymbol{x}$, and values of the model's parameters equal to $\boldsymbol{\theta}(\tau)$. To estimate $\boldsymbol{\theta}(\tau)$ for $\tau \in (0, 1)$, one should minimize the average quantile score $(1/n)\ \Sigma_{i=1}^{n}\ \rho_\tau(y(\boldsymbol{x}_i,\ \boldsymbol{\theta}(\tau)) - y_i)$, which is the core idea of linear-in-parameters quantile regression elaborated by [33]. The regression model with parameters $\boldsymbol{\theta}(\tau)$ predicts conditional quantiles $Q_{\underline{y}|\boldsymbol{x}}(\tau|\boldsymbol{x})$.

III. Data and Application

A. Data

We assessed the algorithms using daily earth-observed precipitation retrieved from the Global Historical Climatology Network daily (GHCNd) as described in Section III-A1, gridded satellite precipitation from the current operational PERSIANN (Precipitation Estimation from Remotely Sensed Information using Artificial Neural Networks) system as well as the GPM IMERG (Integrated Multi-satellitE Retrievals) late precipitation dataset (described in Section III-A2) and elevation data retrieved from the Amazon Web Services (AWS) Terrain Tiles application (described in Section III-A3)). The locations of gauges are presented in Fig. 1.

1) Earth-Observed Precipitation Data: We used total daily precipitation data precipitation retrieved from the GHCNd [16], [17], [40], NOAA National Climatic Data Center (https://www1.ncdc.noaa.gov/pub/data/ghcn/daily); accessed on 2022-02-27). In particular, data from 7261 stations spanning across CONUS (see Fig. 1) were extracted. The data cover the two-year time period 2014–2015.

2) Satellite Precipitation Data: We used gridded satellite daily precipitation data from the current operational PERSIANN system [28], [44], [45], developed by the Centre for Hydrometeorology and Remote Sensing (CHRS) at the University of California, Irvine. The PERSIANN data were retrieved from the website of the CHRS (https://chrsdata.eng.uci.edu; accessed on 2022-03-07).

Furthermore, we used the GPM IMERG late Precipitation L3 1 day 0.1 degree × 0.1 degree V06 dataset developed by the NASA (National Aeronautics and Space Administration) Goddard Earth Sciences Data and Information Services Center [30]. The GPM IMERG data were interpolated to a 0.25 degree × 0.25 degree using bilinear interpolation on CMORPH0.25 grid. Alternatives for interpolating exist, but conclusions of the manuscript are independent of the type of interpolation while they depend on the type of algorithm implemented (as explained later in Section V). The IMERG data were retrieved from the website of NASA Earth Data (https://doi.org/10.5067/GPM/IMERGDL/DAY/06; accessed on 2022-12-10).

The extracted data cover the CONUS at the two-year time period 2014–2015. The herein used GPM IMERG version of the satellite precipitation product has not used ground-based precipitation data for bias correction. The PERSIANN dataset has been corrected using ground-based data; therefore, applying a regression algorithm to the data is practically a post-processing framework that further improves the satellite dataset. That is a common approach in the field, e.g., see [3], [4] among others.

It is possible to use different or more satellite precipitation datasets. In the latter case, the available information and the number of predictor variables will increase followed by improved accuracy of the corrected precipitation product. Although using multiple satellite precipitation datasets is a common practice when building new datasets which must be accurate, that is out of the scope of the present study, which aims to provide an understanding of the properties of LightGBM when probabilistic predictions are issued. Furthermore, conclusions (see Section V)

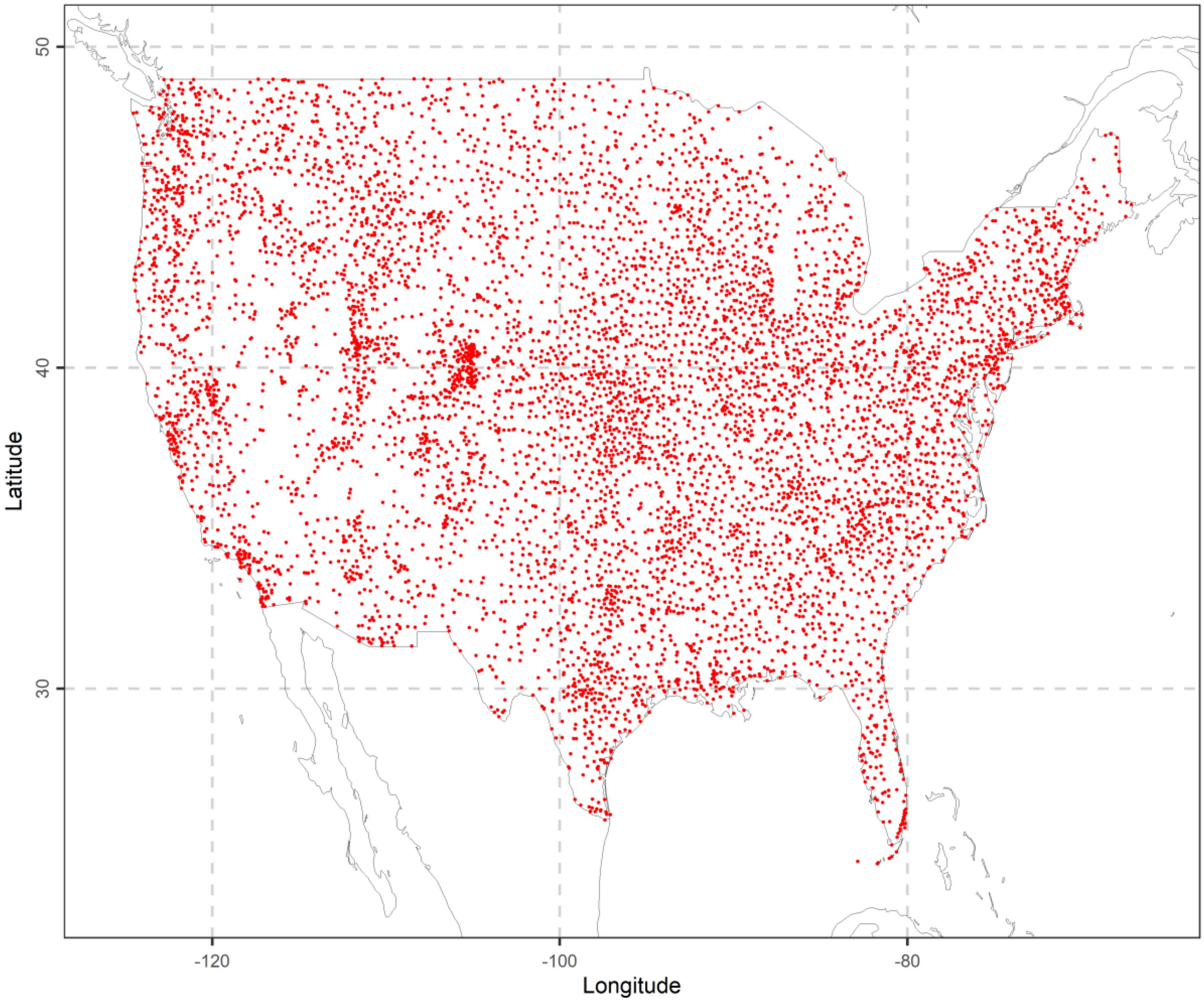

Fig. 1. Map of the geographical locations (red points) of the earth-located stations that offered data for this work. Here, latitudes measure distance north of the equator and longitudes measure distance west of the meridian in Greenwich, England in degrees (°).

will not be affected, since results are due to theoretical properties of the two implemented algorithms as discussed later in Section IV. Nevertheless, including two datasets from which the one is already corrected, but the other is not, has an additional advantage. In particular, as already shown in [52], the uncorrected dataset includes more information compared to the corrected one (in fact the latter provides less significant improvements regarding the accuracy of the final product). Therefore, the diversity of the products might serve to understand better the properties of the correction algorithms.

3) Elevation Data: Elevation is a useful predictor variable when merging gauged-based and satellite precipitation data [81]. Therefore, we computed the elevation of the stations in Section III-A1 using the AWS Terrain Tiles (https://registry.opendata.aws/terrain-tiles; accessed on 2022-09-25) application.

B. Problem Formulation and Assessment of the Algorithms

The setting of the problem has been formulated as follows similarly to the procedures proposed in [51] and [52]. The total daily station precipitation is the dependent variable in a regression problem. Predictor variables are the total daily precipitations from the closest grid points to the station. In particular, there are four predictor variables corresponding to the PERSIANN dataset and another four predictor variables corresponding to the GMP IMERG dataset. Furthermore, we computed the distances between the station and the closest grid points for each satellite dataset; thus, we obtained eight more predictor variables. The station elevations and their longitude and latitude also play the role of predictor variables. Possible interactions between predictor variables do not affect the performance of random forests (and their variants) as well as boosting (and its variants) as explained earlier in Section II-A and II-B, respectively.

To understand how we related station precipitation to the gridded satellite precipitation, we designed Fig. 2. For a single grid (e.g., the PERSIANN), we determined the closest four grid points to each precipitation station, we computed the distances d_i, $i = 1, 2, 3, 4$ from these grid points and ordered in increasing order $d_1 < d_2 < d_3 < d_4$ (see Fig. 2). When we refer to the PERSIANN dataset, the distances d_i, $i = 1, 2, 3, 4$ are called

TABLE I
INCLUSION OF PREDICTOR VARIABLES IN THE PREDICTOR SETS EXAMINED IN THIS WORK

Predictor variable	Predictor set
PERSIANN value 1	✓
PERSIANN value 2	✓
PERSIANN value 3	✓
PERSIANN value 4	✓
IMERG value 1	✓
IMERG value 2	✓
IMERG value 3	✓
IMERG value 4	✓
PERSIANN distance 1	✓
PERSIANN distance 2	✓
PERSIANN distance 3	✓
PERSIANN distance 4	✓
IMERG distance 1	✓
IMERG distance 2	✓
IMERG distance 3	✓
IMERG distance 4	✓
Longitude	✓
Latitude	✓
Station elevation	✓

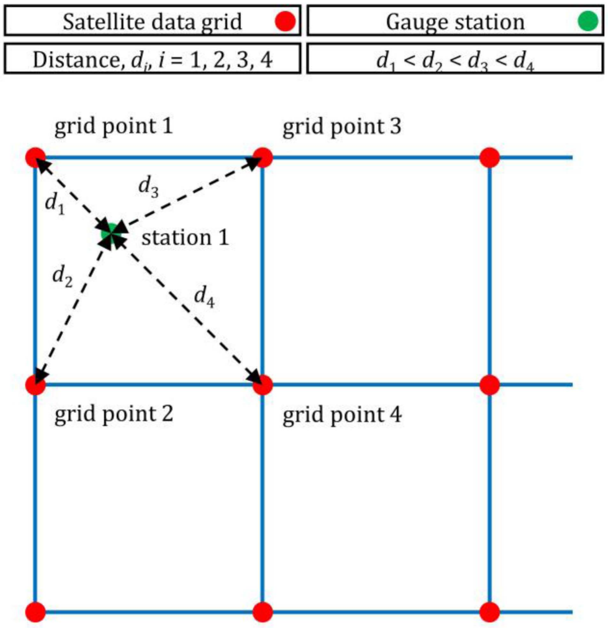

Fig. 2. Setting of the regression problem. Note that the term “grid point” is used to describe the geographical locations with satellite data, while the term “station” is used to describe the geographical locations with ground-based measurements. Note also that, throughout this work, the distances d_i, $i = 1, 2, 3, 4$ are also, respectively, called “PERSIANN distances 1−4” or “IMERG distances 1−4” (depending on whether we refer to the PERSIANN grid or the IMERG grid) and the daily precipitation values at the grid points 1−4 are called “PERSIANN values 1−4” or “IMERG values 1−4” (depending on whether we refer to the PERSIANN grid or the IMERG grid).

“PERSIANN distances 1−4” while when we refer to the IMERG dataset, the distances are called “IMERG distances 1−4.” The respective daily precipitation values at the grid points 1−4 are called “PERSIANN values 1−4” or “IMERG values 1−4.”

Table I presents the predictor variables for the regression setting of the problem. In particular, the predictor variables are the PERSIANN values 1−4, the IMERG values 1−4, the PERSIANN distances 1−4, the IMERG distances 1−4, and the station’s longitude, latitude, and elevation. The final dataset includes 4 833 007 samples. We split the dataset into three equally sized folds randomly. Random forests were trained in the union of the first two folds and were tested in the third fold (that includes 1 611 002 samples). LightGBM was trained in the first fold and was validated in the second fold. We implemented this procedure to estimate its hyperparameters (see Section III-C). After estimating the LightGBM’s hyperparameters, we retrained the algorithm using the final parameters in the union of the first two folds. The algorithm was then tested in the third fold. Predictions of LightGBM lower than 0 were transformed to 0 (it is well known that precipitation is a positive quantity), while QRF did not issue negative precipitation predictions.

C. LightGBM Hyperparameter Optimization

LightGBM has multiple parameters that can be tuned. We selected to optimize some of them, while we kept the default values in the `R` software implementation [60] for the remaining parameters. The selection of hyperparameters to be tuned was directed by the algorithm’s documentation as well as the experience in practical applications, since the algorithm has been part of prize-winning solutions in international prediction competitions. The parameters selected for tuning along with their description based on software’s documentation (https://lightgbm.readthedocs.io/en/v3.3.2/Parameters.html) are shown in Table II.

The parameter space includes a grid with all possible combinations of parameter’s values, excluding a set of parameters where $\texttt{num\_leaves} > 2^{\texttt{max\_depth}}$. Furthermore, we applied the algorithms with early stopping, setting the parameter `early_stopping_round` equal to 20. In this case, the algorithm stops before reaching the specified number of iterations, if for 20 iterations there is no improvement in the score. Early stopping serves in reducing training time.

D. Performance Metrics and Assessment

We compared QRFs and LightGBM using the quantile scoring function defined by

$$S_\tau(x, y) := \rho_\tau(x - y) \tag{6}$$

here y is the materialization (observation) of the spatial process and x is the predictive quantile at level τ. Hydrological predictions should be probabilistic in nature taking the form of probability distributions (see e.g., [23], [48]). Predicting quantiles of the probability distribution at multiple levels is a nice substitute of the full probability distribution. The quantile scoring function is strictly consistent for the quantile functional

TABLE II
LightGBM Parameters

Parameter	Description	Values
`max_depth`	Max depth for tree model. `max_depth` can be used to limit the tree depth explicitly	6, 8, 10
`min_data_in_leaf`	This is a very important parameter to prevent over-fitting in a leaf-wise tree. Its optimal value depends on the number of training samples and `num_leaves`. Setting it to a large value can avoid growing too deep a tree, but may cause under-fitting. In practice, setting it to hundreds or thousands is enough for a large dataset	20, 100, 200, 500, 1 000
`learning_rate`	Shrinkage rate. As a general rule, if one reduces `num_iterations`, then he should increase `learning_rate`	0.02, 0.05, 0.1
`num_iterations`	Number of iterations. The `num_iterations` parameter controls the number of boosting rounds that will be performed. Since LightGBM uses decision trees as the learners, this can also be thought of as "number of trees"	400
`num_leaves`	Max number of leaves in one tree. This is the main parameter to control the complexity of the tree model	20, 40, 60, 80, 100, 200, 500

of the predictive distribution, in the sense that if one receives a directive to predict a quantile, the expected quantile score is minimized when following the directive [22]. Therefore, when receiving a directive to predict a quantile functional, it is natural to train a model using the quantile scoring function as already mentioned in Section II-C.

The performance criterion for the machine learning algorithms at quantile level τ takes the form

$$\bar{S}_\tau := (1/n) \sum_{i=1}^{n} S_\tau (x_i,\ y_i) \tag{7}$$

where $\{x_i, y_i\}$, $i = 1, \ldots, n$ are the predictions and observations for the ith sample and n is the size of the test fold. We computed $\bar{S}_\tau$ at several quantile levels τ and for both algorithms. We considered predictions issued by QRFs as reference predictions and we computed a skill score for the reference algorithm at the specified quantile level defined by

$$S_{\tau,\text{skill}} := 1 - \bar{S}_{\tau,\text{LightGBM}} / \bar{S}_{\tau,\text{QRF}}. \tag{8}$$

In general, $S_{\tau,\text{skill}} \le 1$, while for an excellent forecast at level τ, we have $\bar{S}_{\tau,\text{LightGBM}} = 0$ and $S_{\tau,\text{skill}} = 1$. When $S_{\tau,\text{skill}} > 0$, LightGBM outperforms QRFs, while the higher the $S_{\tau,\text{skill}}$, the better the LightGBM. We did not compare the algorithms using alternative scoring functions (e.g., the squared error scoring function, or a related skill score, e.g., the Nash-Suttcliffe efficiency) because such functions are not consistent for the quantile functional [22].

In addition, we computed a score for each algorithm that characterizes how well each algorithm issues predictions with the nominal frequencies. In particular, the respective score is defined by

$$\overline{\mathrm{FR}}_\tau := \left| (1/n) \sum_{i=1}^{n} \mathbb{I}\left(y_i \le x_i\right) - \tau \right|. \tag{9}$$

To better understand the score, let $\tau = 0.95$. Assuming that perfect predictions have been issued, then $(1/n) \sum_{i=1}^{n} \mathbb{I}(y_i \le x_i)$ should be equal to 0.95 (i.e., 95% of observations should be lower or equal to respective predictions) and $\overline{\mathrm{FR}}_\tau$ should be equal to 0. Again, we computed the respective skill score, with QRFs as reference algorithm

$$\mathrm{FR}_{\tau,\text{skill}} := 1 - \overline{\mathrm{FR}}_{\tau,\text{LightGBM}} / \overline{\mathrm{FR}}_{\tau,\text{QRF}}. \tag{10}$$

In general, $\mathrm{FR}_{\tau,\text{skill}} \le 1$, while for an excellent forecast at level τ, we have $\overline{\mathrm{FR}}_{\tau,\text{LightGBM}} = 0$ and $\mathrm{FR}_{\tau,\text{skill}} = 1$. When $\mathrm{FR}_{\tau,\text{skill}} > 0$, LightGBM outperforms QRFs, while the higher the $\mathrm{FR}_{\tau,\text{skill}}$, the better the LightGBM.

IV. Results

Results of the applications of the algorithms are presented here, while those results will be discussed in detail in the next section, followed by respective explanations. Regarding the performance of the algorithm in the test set, we tested them at quantile levels $\tau \in \{0.5, 0.6, 0.7, 0.8, 0.9, 0.95, 0.97, 0.99, 0.999\}$. Recall that skill scores higher than 0 indicate that LightGBM outperforms QRF.

Regarding the case of the frequency skill score [recall the explanation of values of the frequency skill score FRτ, after (10)], presented in Fig. 3(a), the performance of both algorithms is almost equal for $\tau \le 0.8$, while there is a fluctuation around 0 for $\tau \in (0.8, 0.95)$ and LightGBM outperforms QRF in higher quantile levels. Recall from Section III-D, that the scoring functions related to frequencies are not consistent for a function of interest; therefore, a rigorous assessment of the algorithms is possible using the quantile scoring function. Skill score values for the quantile scoring function [recall the explanation of values of the quantile skill score $S_{\tau,\text{skill}}$ after (8)] are presented in Fig. 3(b), where it seems that LightGBM outperforms QRF for $\tau \ge 0.97$, while the performances of both algorithms are approximately equal at lower quantile levels. In both cases of skill scores, the score increases with τ increasing when $\tau \ge 0.97$.

It is of interest to understand how the algorithms behave when the observed values in the test set are equal to 0. Zero precipitation corresponds to approximately 72% of total daily observations, while intermittency in time and space is a dominant property of precipitation. Related skill scores for frequencies as well as quantile scoring functions are presented in Fig. 4. Regarding frequencies, the performances of both algorithms are equal [Fig. 4(a)]. That is expected, since the algorithms issue always predictions (for the case of LightGBM, after adjustment of the predictions; see Section III-B) that are equal or higher than 0; see also discussion in the next section. However, QRF seems to outperform LightGBM for $\tau \ge 0.97$, regarding the quantile scoring function; see Fig. 4(b).

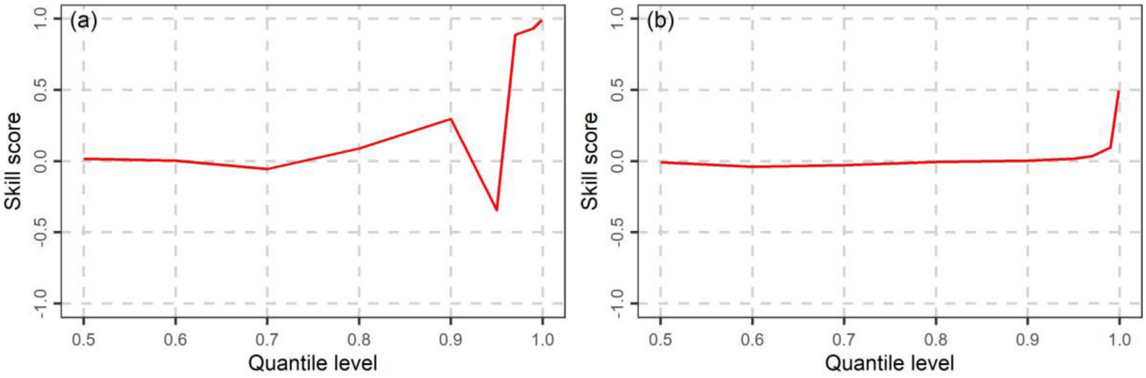

Fig. 3. Skill scores for (a) frequencies and (b) quantile losses at different quantile levels τ for complete data in the test set.

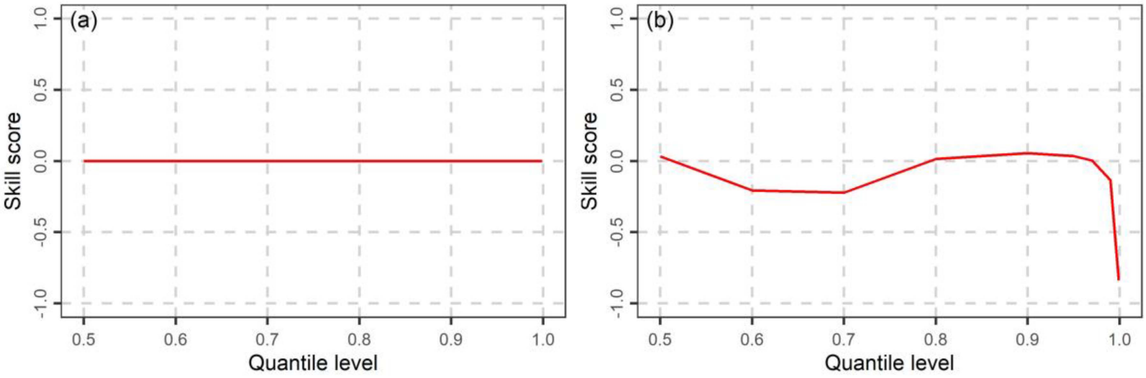

Fig. 4. Skill scores for (a) frequencies and (b) quantile losses at different quantile levels for observed precipitation equal to zero in the test set.

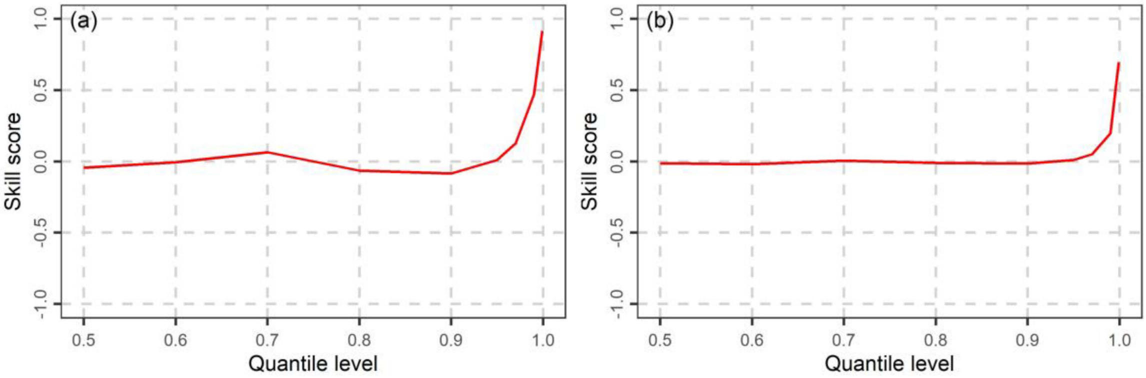

Fig. 5. Skill scores for (a) frequencies and (b) quantile losses at different quantile levels for observed precipitation higher than zero in the test set.

The overall picture changes when testing the algorithms in observed precipitation higher than 0; see Fig. 5. Here, LightGBM seems to outperform QRF for $\tau \geq 0.95$ referring to both frequency and quantile scoring function based skill scores. In both cases of skill scores, the score increases with τ increasing when $\tau \geq 0.97$.

It is also of interest to understand how the algorithms perform at each station separately; see Fig. 6. Here, we examine the case of the quantile scoring function based skill score; recall that the quantile scoring function is consistent for the quantile functional. Stations with skill scores lower than –1 were removed from Fig. 6. The reason is that, some skill score values were as low as –10 or less, which would create some artifacts in the representation of the results. The conclusions are not affected by the removal, given that skill scores are skewed, since they cannot exceed the value of 1, although they can be equal to $-\infty$. Furthermore, we removed stations where both algorithms had a mean score equal to 0 (in which case the skill score is not defined). In Fig. 6, we observe that the skill score increases as the quantile level $\tau \to 1$. Furthermore, we observe that the skill

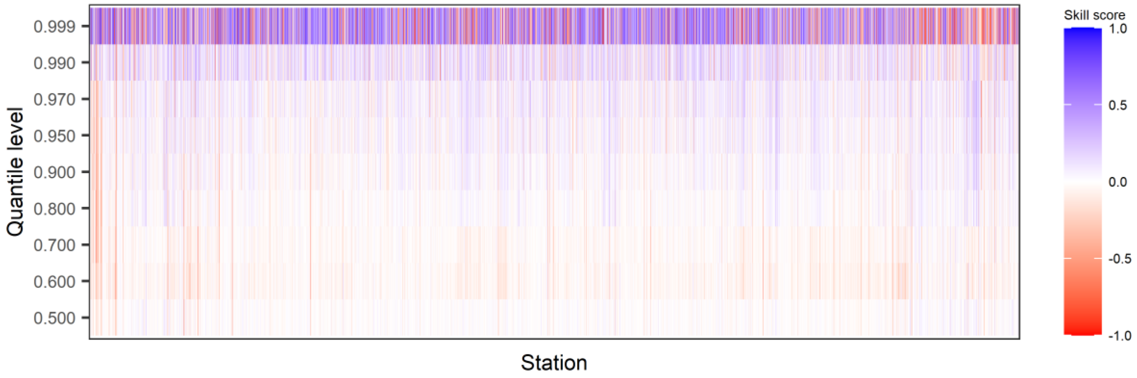

Fig. 6. Heatmap of skill scores for quantile losses at different quantile levels and each station.

score varies between stations at the same quantile level, although the variation is relatively small. A notable departure of the skill scores from 0 is observed for quantile levels $\tau \geq 0.97$.

V. Discussion

Regarding the overall performance of the two methods, LightGBM is an algorithm particularly useful in large datasets with a high number of dimensions. Furthermore, given that it belongs to the highly parametrized family of boosting algorithms that are characterized by high flexibility, it is not surprising that it outperforms random forests on average. That is evident in the case of complete data in which LightGBM, in general, performs better when assessed with the quantile scoring function. However, LightGBM does not uniformly outperform QRF at all quantile levels. At lower quantile levels, the two algorithms seem to behave similarly, while at higher quantile levels LightGBM clearly outperforms QRF.

A possible explanation for the behavior at lower quantile levels is based on the high proportion of zeros in the dataset. In particular, QRF is an algorithm based on bootstrapping, therefore, it is possible to resample zero values. On the other hand, LightGBM is based on the minimization of the quantile scoring function that may be suboptimal when the dataset is highly intermittent. For instance, the median of the conditional distribution of precipitation should be zero (since the number of zero values in the dataset is higher than 50%). The quantile scoring function at level $\tau = 0.5$ is equivalent to the absolute error scoring function which, in turn, may not be suitable in cases where one should predict a median value of a probability distribution with mass at zero. Nevertheless, the overall performances of LightGBM and QRF at quantile level 0.5 remain similar. That may be due to that, while QRF can better predict zeros, they fail at a higher degree when they issue nonzero predictions.

At higher quantile levels, LightGBM clearly outperforms QRF with regards to all skill scores while the difference increases with increasing τ, while the skill score tends to 1 as $\tau \to 1$. A possible explanation is that QRF cannot predict values that are not in the range of the training set [26]. The weakness becomes more pronounced at higher quantile levels, where high values in the training set become rarer and the algorithm tends to shift toward lower values. On the other hand, LightGBM is based on addition of base learners to previous errors and despite base learners being decision trees, it seems that it can better extrapolate beyond the range of the training set. Furthermore, the conditional probability distribution of precipitation at higher quantile levels seems to comply with regularity conditions, under which the quantile scoring function's properties seem to hold.

While QRF outperforms LightGBM with regards to the quantile skill score at higher quantile levels when observed precipitation is zero, the inverse happens when observed precipitation is higher than zero. The performance of both algorithms in the complete test set favors LightGBM, since absolute values of quantile scores are lower in general when observed precipitation is zero compared to nonzero observed precipitation, consequently the largest part of the average score belongs to nonzero values.

Examining single algorithms is important to understand their properties at hand. However, the combination of algorithms (ensemble learning in machine learning literature, [57]), may result in higher improvements. That has been proved in practice using simple combinations (e.g., [47]) or stacking [69] for predicting the mean functional of the conditional probability distribution. Combinations of algorithms for probabilistic prediction (e.g., stacking, [36], [73], [82]), perhaps including spatial features (e.g., [49]) is a topic worth examining and has been proven successful in other hydrological applications [50], [67] as well as in merging gauged-based and satellite precipitation datasets (see [83] for the case of the mean functional). Predictions of extreme quantile based on extreme value theory are also worth examining, but it remains to assess in practice whether the conditional distribution in spatial settings is heavy-tailed. Nevertheless, extremal quantile regression has been applied in post-processing applications in the time domain [66]. Assessing other algorithms, e.g., deep learning ones [34], [58] in probabilistic predictions of precipitation might also be a topic worth examining.

VI. Conclusion

We proposed issuing probabilistic predictions of daily precipitation in spatial settings by merging gauge-based measurements and satellite precipitation products using LightGBM. LightGBM outperforms the state-of-the-art in such settings QRFs (a variant of random forests) when predicting extreme quantiles of the conditional probability distribution of the response variable, while both algorithms show similar performance when predicting quantiles at the center of the probability distribution. The difference in the performance of the methods increases in favor of LightGBM as the quantile level (at which the methods are compared) increases and tends to 1. Confidence in the results is built through the comparison of the algorithms in a large dataset that includes observed precipitation in the CONUS.

An intuitive explanation of the results is also provided, according to which LightGBM can better predict extreme quantiles due to the inability of random forests to extrapolate beyond the range of the training set combined with the improved ability of LightGBM to issue accurate predictions due to its structure. On the other hand, QRFs are equal to LightGBM when predicting quantiles at the center of the conditional probability distribution, due to the highly intermittent nature of precipitation, combined with their bootstrap-based structure, which seems to be more suitable in this case compared to algorithm structures that are based on the quantile scoring function.

Appendix

We used the R programming language [56] to implement the algorithms and to report and visualize the results.

For data processing and visualizations, we used the contributed R packages data.table [15], elevatr [27], ncdf4 [55], rgdal [8], sf [53], [54], spdep [5], [6], [7], and tidyverse [70], [71].

The algorithms were implemented by using the contributed R packages ranger [74], [75] and lightgbm [60].

The performance metrics were computed by implementing the contributed R package scoringfunctions [64], [65].

Reports were produced by using the contributed R packages devtools [72], knitr [76], [77], [78], and rmarkdown [2], [79], [80].

Acknowledgment

This work was conducted in the context of the research project BETTER RAIN (BEnefiTTing from machine lEarning algoRithms and concepts for correcting satellite RAINfall products).

References

[1] A. Abdollahipour, H. Ahmadi, and B. Aminnejad, “A review of downscaling methods of satellite-based precipitation estimates,” *Earth Sci. Inform.*, vol. 15, no. 1, pp. 1–20, 2022, doi: 10.1007/s12145-021-00669-4.

[2] J. J. Allaire et al., “R markdown: Dynamic documents for R,” R package version 2.18, 2022. [Online]. Available: https://CRAN.R-project.org/package=rmarkdown

[3] O. M. Baez-Villanueva et al., “RF-MEP: A novel random forest method for merging gridded precipitation products and ground-based measurements,” *Remote Sens. Environ.*, vol. 239, 2020, Art. no. 111606, doi: 10.1016/j.rse.2019.111606.

[4] M. A. E. Bhuiyan, E. I. Nikolopoulos, E. N. Anagnostou, P. Quintana-Seguí, and A. Barella-Ortiz, “A nonparametric statistical technique for combining global precipitation datasets: Development and hydrological evaluation over the Iberian Peninsula,” *Hydrol. Earth Syst. Sci.*, vol. 22, pp. 1371–1389, 2018, doi: 10.5194/hess-22-1371-2018.

[5] R. S. Bivand, “Spdep: Spatial dependence: Weighting schemes, statistics,” R package version 1.2-7, 2022. [Online]. Available: https://CRAN.R-project.org/package=spdep

[6] R. S. Bivand and D. W. S. Wong, “Comparing implementations of global and local indicators of spatial association,” *TEST*, vol. 27, no. 3, pp. 716–748, 2018, doi: 10.1007/s11749-018-0599-x.

[7] R. S. Bivand, E. Pebesma, and V. Gómez-Rubio, *Applied Spatial Data Analysis with R*, 2nd ed. Berlin, Germany: Springer, 2013, doi: 10.1007/978-1-4614-7618-4.

[8] R. S. Bivand, T. Keitt, and B. Rowlingson, “RGDAL: Bindings for the ‘Geospatial’ data abstraction library,” R package version 1.6-2, 2022. [Online]. Available: https://CRAN.R-project.org/package=rgdal

[9] G. Blöschl et al., “Twenty-three unsolved problems in hydrology (UPH)–A community perspective,” *Hydrological Sci. J.*, vol. 64, no. 10, pp. 1141–1158, 2019, doi: 10.1080/02626667.2019.1620507.

[10] L. Breiman, “Random forests,” *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[11] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[12] C. Chen, B. Hu, and Y. Li, “Easy-to-use spatial random-forest-based downscaling-calibration method for producing precipitation data with high resolution and high accuracy,” *Hydrol. Earth Syst. Sci.*, vol. 25, no. 11, pp. 5667–5682, 2021, doi: 10.5194/hess-25-5667-2021.

[13] S. Curceac, P. M. Atkinson, A. Milne, L. Wu, and P. Harris, “Adjusting for conditional bias in process model simulations of hydrological extremes: An experiment using the North Wyke farm platform,” *Front. Artif. Intell.*, vol. 3, no. 82, 2020, doi: 10.3389/frai.2020.565859.

[14] N. Dogulu, P. López López, D. P. Solomatine, A. H. Weerts, and D. L. Shrestha, “Estimation of predictive hydrologic uncertainty using the quantile regression and UNEEC methods and their comparison on contrasting catchments,” *Hydrol. Earth Syst. Sci.*, vol. 19, no. 7, pp. 3181–3201, 2015, doi: 10.5194/hess-19-3181-2015.

[15] M. Dowle and A. Srinivasan, “Data.table: Extension of ‘data.frame’,” R package version 1.14.6, 2022. [Online]. Available: https://CRAN.R-project.org/package=data.table

[16] I. Durre, M. J. Menne, and R. S. Vose, “Strategies for evaluating quality assurance procedures,” *J. Appl. Meteorol. Climatol.*, vol. 47, no. 6, pp. 1785–1791, 2008, doi: 10.1175/2007JAMC1706.1.

[17] I. Durre, M. J. Menne, B. E. Gleason, T. G. Houston, and R. S. Vose, “Comprehensive automated quality assurance of daily surface observations,” *J. Appl. Meteorol. Climatol.*, vol. 49, no. 8, pp. 1615–1633, 2010, doi: 10.1175/2010JAMC2375.1.

[18] B. Efron and T. Hastie, *Computer Age Statistical Inference*. Cambridge, U.K.: Cambridge Univ. Press, 2016, doi: 10.1017/CBO9781316576533.

[19] C. A. Fernandez-Palomino et al., “A novel high-resolution gridded precipitation dataset for Peruvian and Ecuadorian watersheds: Development and hydrological evaluation,” *J. Hydrometeorol.*, vol. 23, no. 3, pp. 309–336, 2022, doi: 10.1175/JHM-D-20-0285.1.

[20] J. H. Friedman, “Greedy function approximation: A gradient boosting machine,” *Ann. Statist.*, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.

[21] L. Glawion, J. Polz, H. G. Kunstmann, B. Fersch, and C. Chwala, “SpateGAN: Spatio-temporal downscaling of rainfall fields using a cGAN approach,” *ESS Open Arch.*, 2023, doi: 10.22541/essoar.167690003.33629126.

[22] T. Gneiting, “Making and evaluating point forecasts,” *J. Amer. Stat. Assoc.*, vol. 106, no. 494, pp. 746–762, 2011, doi: 10.1198/jasa.2011.r10138.

[23] T. Gneiting and A. E. Raftery, “Strictly proper scoring rules, prediction, and estimation,” *J. Amer. Stat. Assoc.*, vol. 102, no. 477, pp. 359–378, 2007, doi: 10.1198/016214506000001437.

[24] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Berlin, Germany: Springer, 2009, doi: 10.1007/978-0-387-84858-7.

[25] X. He, N. W. Chaney, M. Schleiss, and J. Sheffield, “Spatial downscaling of precipitation using adaptable random forests,” *Water Resour. Res.*, vol. 52, no. 10, pp. 8217–8237, 2016, doi: 10.1002/2016WR019034.

[26] T. Hengl, M. Nussbaum, M. N. Wright, G. B. M. Heuvelink, and B. Gräler, “Random forest as a generic framework for predictive modeling of spatial and spatio-temporal variables,” *PeerJ*, vol. 6, no. 8, 2018, Art. no. e5518, doi: 10.7717/peerj.5518.

[27] J. W. Hollister, “Elevatr: Access elevation data from various APIs,” R package version 0.4.2, 2022. [Online]. Available: https://CRAN.R-project.org/package=elevatr

[28] K.-L. Hsu, X. Gao, S. Sorooshian, and H. V. Gupta, “Precipitation estimation from remotely sensed information using artificial neural networks,” *J. Appl. Meteorol.*, vol. 36, no. 9, pp. 1176–1190, 1997, doi: 10.1175/1520-0450(1997)036<1176:PEFRSI>2.0.CO;2.

[29] Q. Hu, Z. Li, L. Wang, Y. Huang, Y. Wang, and L. Li, "Rainfall spatial estimations: A review from spatial interpolation to multi-source data merging," *Water*, vol. 11, no. 3, 2019, Art. no. 579, doi: 10.3390/w11030579.

[30] G. J. Huffman, E. F. Stocker, D. T. Bolvin, E. J. Nelkin, and J. Tan, "GPM IMERG late precipitation L3 1 day 0.1 degree x 0.1 degree V06," Edited by Andrey Savtchenko, Greenbelt, MD, Goddard Earth Sciences Data and Information Services Center (GES DISC), 2019, doi: 10.5067/GPM/IMERGDL/DAY/06.

[31] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning*. Berlin, Germany: Springer, 2013, doi: 10.1007/978-1-4614-7138-7.

[32] G. Ke et al., "LightGBM: A highly efficient gradient boosting decision tree," *Adv. Neural Inf. Process. Syst.*, vol. 30, pp. 3146–3154, 2017.

[33] R. W. Koenker and G. Bassett Jr, "Regression quantiles," *Econometrica*, vol. 46, no. 1, pp. 33–50, 1978, doi: 10.2307/1913643.

[34] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.

[35] H. Lei, H. Zhao, and T. Ao, "A two-step merging strategy for incorporating multi-source precipitation products and gauge observations using machine learning classification and regression over China," *Hydrol. Earth Syst. Sci.*, vol. 26, no. 11, pp. 2969–2995, 2022, doi: 10.5194/hess-26-2969-2022.

[36] K. C. Lichtendahl Jr, Y. Grushka-Cockayne, and R. L. Winkler, "Is it better to average probabilities or quantiles?," *Manage. Sci.*, vol. 59, no. 7, pp. 1594–1611, 2013, doi: 10.1287/mnsc.1120.1667.

[37] Q. Lin, T. Peng, Z. Wu, J. Guo, W. Chang, and Z. Xu, "Performance evaluation, error decomposition and tree-based machine learning error correction of GPM IMERG and TRMM 3B42 products in the Three Gorges reservoir area," *Atmospheric Res.*, vol. 268, 2022, Art. no. 105988, doi: 10.1016/j.atmosres.2021.105988.

[38] A. Mayr, H. Binder, O. Gefeller, and M. Schmid, "The evolution of boosting algorithms: From machine learning to statistical modelling," *Methods Inf. Med.*, vol. 53, no. 6, pp. 419–427, 2014, doi: 10.3414/ME13-01-0122.

[39] N. Meinshausen, "Quantile regression forests," *J. Mach. Learn. Res.*, vol. 7, pp. 983–999, 2006.

[40] M. J. Menne, I. Durre, R. S. Vose, B. E. Gleason, and T. G. Houston, "An overview of the global historical climatology network-daily database," *J. Atmospheric Ocean. Technol.*, vol. 29, no. 7, pp. 897–910, 2012, doi: 10.1175/JTECH-D-11-00103.1.

[41] H. Meyer, M. Kühnlein, T. Appelhans, and T. Nauss, "Comparison of four machine learning algorithms for their applicability in satellite-based optical rainfall retrievals," *Atmospheric Res.*, vol. 169, pp. 424–433, 2016, doi: 10.1016/j.atmosres.2015.09.021.

[42] A. F. Militino, M. D. Ugarte, and U. Pérez-Goya, "Machine learning procedures for daily interpolation of rainfall in Navarre (Spain)," in *Trends in Mathematical, Information and Data Sciences*, N. Balakrishnan, M. Á. Gil, N. Martín, D. Morales, and M. del Carmen Pardo, Eds., Berlin, Germany: Springer, 2023, pp. 399–413, doi: 10.1007/978-3-031-04137-2_34.

[43] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Front. Neurorobot.*, vol. 7, no. 21, 2013, doi: 10.3389/fnbot.2013.00021.

[44] P. Nguyen et al., "The PERSIANN family of global satellite precipitation data: A review and evaluation of products," *Hydrol. Earth Syst. Sci.*, vol. 22, no. 11, pp. 5801–5816, 2018, doi: 10.5194/hess-22-5801-2018.

[45] P. Nguyen et al., "The CHRS data portal, an easily accessible public repository for PERSIANN global satellite precipitation data," *Sci. Data*, vol. 6, 2019, Art. no. 180296, doi: 10.1038/sdata.2018.296.

[46] G. V. Nguyen, X.-H. Le, L. N. Van, S. Jung, M. Yeon, and G. Lee, "Application of random forest algorithm for merging multiple satellite precipitation products across South Korea," *Remote Sens.*, vol. 13, no. 20, 2021, Art. no. 4033, doi: 10.3390/rs13204033.

[47] G. Papacharalampous and H. Tyralis, "Hydrological time series forecasting using simple combinations: Big data testing and investigations on one-year ahead river flow predictability," *J. Hydrol.*, vol. 590, 2020, Art. no. 125205, doi: 10.1016/j.jhydrol.2020.125205.

[48] G. Papacharalampous and H. Tyralis, "A review of machine learning concepts and methods for addressing challenges in probabilistic hydrological post-processing and forecasting," *Front. Water*, vol. 4, 2022a, Art. no. 961954, doi: 10.3389/frwa.2022.961954.

[49] G. Papacharalampous and H. Tyralis, "Time series features for supporting hydrometeorological explorations and predictions in ungauged locations using large datasets," *Water*, vol. 14, no. 10, 2022b, Art. no. 1657, doi: 10.3390/w14101657.

[50] G. Papacharalampous et al., "Probabilistic hydrological post-processing at scale: Why and how to apply machine-learning quantile regression algorithms," *Water*, vol. 11, no. 10, 2019, Art. no. 2126, doi: 10.3390/w11102126.

[51] G. Papacharalampous, H. Tyralis, A. Doulamis, and N. Doulamis, "Comparison of machine learning algorithms for merging gridded satellite and earth-observed precipitation data," *Water*, vol. 15, no. 4, 2023a, Art. no. 634, doi: 10.3390/w15040634.

[52] G. Papacharalampous, H. Tyralis, A. Doulamis, and N. Doulamis, "Comparison of tree-based ensemble algorithms for merging satellite and earth-observed precipitation data at the daily time scale," *Hydrology*, vol. 10, no. 2, 2023b, Art. no. 50, doi: 10.3390/hydrology10020050.

[53] E. Pebesma, "Simple features for R: Standardized support for spatial vector data," *R J.*, vol. 10, no. 1, pp. 439–446, 2018, doi: 10.32614/RJ-2018-009.

[54] E. Pebesma, "SF: Simple features for R," R package version 1.0-9, 2022. Accessed: Oct. 10, 2022. [Online]. Available: https://CRAN.R-project.org/package=sf

[55] D. Pierce, "NCDF4: Interface to Unidata netCDF (version 4 or earlier) format data files," R package version 1.20, 2021. [Online]. Available: https://CRAN.R-project.org/package=ncdf4

[56] R Core Team, "R: A language and environment for statistical computing. R foundation for statistical computing," Vienna, Austria, 2022. [Online]. Available: https://www.R-project.org

[57] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdiscipl. Rev.: Data Mining Knowl. Discov.*, vol. 8, no. 4, 2018, Art. no. e1249, doi: 10.1002/widm.1249.

[58] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, 2015, doi: 10.1016/j.neunet.2014.09.003.

[59] Z. Shen and B. Yong, "Downscaling the GPM-based satellite precipitation retrievals using gradient boosting decision tree approach over Mainland China," *J. Hydrol.*, vol. 602, 2021, Art. no. 126803, doi: 10.1016/j.jhydrol.2021.126803.

[60] Y. Shi et al., "LightGBM: Light gradient boosting machine," R package version 3.3.4, 2022. [Online]. Available: https://CRAN.R-project.org/package=lightgbm

[61] Q. Sun, C. Miao, Q. Duan, H. Ashouri, S. Sorooshian, and K.-L. Hsu, "A review of global precipitation data sets: Data sources, estimation, and intercomparisons," *Rev. Geophys.*, vol. 56, no. 1, pp. 79–107, 2018, doi: 10.1002/2017RG000574.

[62] H. Tyralis and G. Papacharalampous, "Boosting algorithms in energy research: A systematic review," *Neural Comput. Appl.*, vol. 33, no. 21, pp. 14101–14117, 2021, doi: 10.1007/s00521-021-05995-8.

[63] H. Tyralis and G. Papacharalampous, "Quantile-based hydrological modelling," *Water*, vol. 13, no. 23, 2021, Art. no. 3420, doi: 10.3390/w13233420.

[64] H. Tyralis and G. Papacharalampous, "A review of probabilistic forecasting and prediction with machine learning," 2022, *arXiv:2209.08307*.

[65] H. Tyralis and G. Papacharalampous, "Scoringfunctions: A collection of scoring functions for assessing point forecasts," R package version 0.0.5, 2022. [Online]. Available: https://CRAN.R-project.org/package=scoringfunctions

[66] H. Tyralis and G. Papacharalampous, "Hydrological post-processing for predicting extreme quantiles," *J. Hydrol.*, vol. 617(Part C), 2023, Art. no. 129082, doi: 10.1016/j.jhydrol.2023.129082.

[67] H. Tyralis, G. Papacharalampous, A. Burnetas, and A. Langousis, "Hydrological post-processing using stacked generalization of quantile regression algorithms: Large-scale application over CONUS," *J. Hydrol.*, vol. 577, 2019, Art. no. 123957, doi: 10.1016/j.jhydrol.2019.123957.

[68] H. Tyralis, G. Papacharalampous, and A. Langousis, "A brief review of random forests for water scientists and practitioners and their recent history in water resources," *Water*, vol. 11, no. 5, 2019, Art. no. 910, doi: 10.3390/w11050910.

[69] H. Tyralis, G. Papacharalampous, and A. Langousis, "Super ensemble learning for daily streamflow forecasting: Large-scale demonstration and comparison with multiple machine learning algorithms," *Neural Comput. Appl.*, vol. 33, no. 8, pp. 3053–3068, 2021, doi: 10.1007/s00521-020-05172-3.

[70] H. Wickham, "Tidyverse: Easily install and load the 'Tidyverse,'" R package version 1.3.2, 2022. [Online]. Available: https://CRAN.R-project.org/package=tidyverse

[71] H. Wickham et al., "Welcome to the tidyverse," *J. Open Source Softw.*, vol. 4, no. 43, 2019, Art. no. 1686, doi: 10.21105/joss.01686.

[72] H. Wickham, J. Hester, W. Chang, and J. Bryan, "Devtools: Tools to make developing R packages easier," R package version 2.4.5, 2022. [Online]. Available: https://CRAN.R-project.org/package=devtools

[73] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, 1992, doi: 10.1016/S0893-6080(05)80023-1.

[74] M. N. Wright, "Ranger: A fast implementation of random forests," R package version 0.14.1, 2022. [Online]. Available: https://CRAN.R-project.org/package=ranger

[75] M. N. Wright and A. Ziegler, “Ranger: A fast implementation of random forests for high dimensional data in C++ and R,” *J. Stat. Softw.*, vol. 77, no. 1, pp. 1–17, 2017, doi: 10.18637/jss.v077.i01.

[76] Y. Xie, “Knitr: A comprehensive tool for reproducible research in R,” in *Implementing Reproducible Computational Research*, V. Stodden, F. Leisch, and R. D. Peng, Eds. London, U.K.: Chapman & Hall, 2014.

[77] Y. Xie, *Dynamic Documents with R and Knitr*, 2nd ed. London, U.K.: Chapman & Hall, 2015.

[78] Y. Xie, “Knitr: A general-purpose package for dynamic report generation in R,” R package version 1.41, 2022. [Online]. Available: https://CRAN.R-project.org/package=knitr

[79] Y. Xie, J. J. Allaire, and G. Grolemund, *R Markdown: The Definitive Guide*. London, U.K.: Chapman & Hall, 2018. [Online]. Available: https://bookdown.org/yihui/rmarkdown

[80] Y. Xie, C. Dervieux, and E. Riederer, *R Markdown Cookbook*. London, U.K.: Chapman & Hall, 2020. [Online]. Available: https://bookdown.org/yihui/rmarkdown-cookbook

[81] L. Xiong, S. Li, G. Tang, and J. Strobl, “Geomorphometry and terrain analysis: Data, methods, platforms and applications,” *Earth-Sci. Rev.*, vol. 233, 2022, Art. no. 104191, doi: 10.1016/j.earscirev.2022.104191.

[82] Y. Yao, A. Vehtari, D. Simpson, and A. Gelman, “Using stacking to average Bayesian predictive distributions,” *Bayesian Anal.*, vol. 13, no. 3, pp. 917–1003, 2018, doi: 10.1214/17-BA1091.

[83] O. Zandi, B. Zahraie, M. Nasseri, and A. Behrangi, “Stacking machine learning models versus a locally weighted linear model to generate high-resolution monthly precipitation over a topographically complex area,” *Atmospheric Res.*, vol. 272, 2022, Art. no. 106159, doi: 10.1016/j.atmosres.2022.106159.

[84] L. Zhang et al., “Merging multiple satellite-based precipitation products and gauge observations using a novel double machine learning approach,” *J. Hydrol.*, vol. 594, 2021, Art. no. 125969, doi: 10.1016/j.jhydrol.2021.125969.

[85] Y. Zhang, A. Ye, P. Nguyen, B. Analui, S. Sorooshian, and K. Hsu, “QRF4P-NRT: Probabilistic post-processing of near-real-time satellite precipitation estimates using quantile regression forests,” *Water Resour. Res.*, vol. 58, no. 5, 2022, Art. no. e2022WR032117, doi: 10.1029/2022WR032117.

Hristos Tyralis received the diploma in civil engineering from the Hellenic Air Force Academy, Athens, Greece, in 2002, the Master’s degree in statistics and operations research from the National and Kapodistrian University of Athens, Athens, Greece, in 2004, and the Diploma and Ph.D. degrees in civil engineering from the National Technical University of Athens, Athens, Greece, in 2006 and 2015, respectively.

He is currently the Director with NATO Constructions Directorate, Construction Agency, Hellenic Air Force and a post-doctoral researcher with National Technical University of Athens, Athens, Greece. He has more than 20 years of experience in public works supervision and management with the Hellenic Air Force. His current research interests include probabilistic forecasting using statistical and machine learning.

Georgia Papacharalampous received the diploma in civil engineering, the M.Sc. degree in water resources science and technology for coastal zone management and the Ph.D. degree in civil engineering from the National Technical University of Athens (NTUA), Athens, Greece, in 2014, 2016, and 2020, respectively.

Since the completion of her Ph.D. degree, she has been working as a Postdoctoral Researcher. She has successfully completed her work at the University of Patras, Patras, Greece (2021), the Roma Tre University, Rome, Italy (2021), and the Czech University of Life Sciences, Prague, Czech Republic (2022), and she currently works with NTUA. Her research interests include forecasting, machine and statistical learning, and statistical hydrology.

Dr. Papacharalampous is recipient of several distinctions and awards, including the International Scientific Prize of the Dimitris N. Chorafas Foundation for her Ph.D. thesis.

Nikolaos Doulamis (Member, IEEE) received the Diploma and Ph.D. degrees (Hons.) in electrical and computer engineering from the National Technical University of Athens (NTUA), Athens, Greece, in 1995 and 2001, respectively.

He is a Professor with NTUA. He is the author of more than 350 articles in leading journals and conferences receiving more than 6 600 citations. He is involved in large-scale European projects, such as H2020 Stop-It and H2020 Beneffice. He has served as an organizer and/or a TPC in major IEEE conferences.

Dr. Doulamis is recipient of many awards (e.g., the Best Student Among all Engineers and the Best Paper Awards).

Anastasios Doulamis (Member, IEEE) received the Diploma and Ph.D. degrees in electrical and computer engineering from the National Technical University of Athens (NTUA), Athens, Greece, in 1995 and 2001, respectively.

Until January 2014, he was an Associate Professor with the Technical University of Crete, Chania, Greece. He is an Associate Professor with NTUA. He is the author of more than 360 articles in leading journals and conferences receiving more than 6 500 citations.

Dr. Doulamis is recipient of several awards in his studies, including the Best Greek Student Engineer and the Best Graduate Thesis Award. He has also served on the program committee of several major conferences of IEEE and ACM.