Applied Intelligence (2023) 53:16107–16125
https://doi.org/10.1007/s10489-022-04319-0

FLORAS: urban flash-flood prediction using a multivariate model

Lucas A. V. Brito[1] · Rodolfo I. Meneguette[1] · Robson E. De Grande[2] · Caetano M. Ranieri[1] · Jó Ueyama[1]

Accepted: 3 November 2022 / Published online: 2 December 2022

Abstract
Hydrological models allow water levels to be predicted at critical spots when the problem of flooding is being addressed. However, these models fall short in their attempts to provide timely warnings to communities at risk as they often involve complex setup requirements and incur high computation costs. Other approaches have been adopted that make use of water level monitoring sensors for detecting floods. Although accurate in their performance, these approaches often require a high level of maintenance because their predictions rely on critical readings from sensors that have to be immersed in rivers. We recommend a machine learning-based methodology for flood detection to address this issue, called FLORAS. It makes it possible to build models that make predictions solely on the basis of meteorological data from weather stations-water height measurements are only needed to employ ground truth for the purposes of training and validation. We evaluated the methodology with current data readings from São Carlos (SP - Brazil) in experimental analyses. Water height measurements from sensors placed at sites along the river were correlated with open weather data from a reputable, local source (Climatempo – Brazilian weather). The results show that the model achieved a higher degree of of accuracy and incurred lower computational costs than SwMM, a hydrological model. These results show that the recommended methodology is suitable for systems that run with resource-scarce devices, such as the IoT systems that are usually deployed in flood detection frameworks.

Keywords Data mining · Flood identification · Machine learning · WSN

1 Introduction

A study undertaken by the United Nations (UN) states that, in the 21st Century, about 2.8 billion people have so far suffered from natural disasters, with the financial estimate of the damage exceeding USD 1.7 trillion. Between 1995 and 2015 alone, flooding events represented about 26% of the deaths caused by natural disasters. In the same period, 56% of people in the world affected by any disaster experienced flooding [1]. There are more than 40,000 flood risk areas in Brazil which can potentially affect more than 120 million people, who need to be better prepared against the hazards of floods. Another alarming fact is that the loss of life and material damage resulting from flood risks threatens to reduce more than 60% of Brazil's GDP (Gross Domestic Product). Thus, there is an urgent search being carried out for low-cost technology strategies that can better prepare towns and cities against the effects of floods [2–4]. Advanced flood forecasting can trigger damage control mechanisms to mitigate their harmful effects. At least, predictions can issue warnings so that the traffic and the people at risk can escape these hazards .

Wireless Sensor Networks (WSNs) have evolved as alternative technological systems for monitoring [5, 6]. This technology consists of a sink node and a set of autonomous wireless sensor nodes [7, 8]. The sensor nodes require limited energy resources and can be either mobile

✉ Caetano M. Ranieri
cmranieri@usp.br

Lucas A. V. Brito
lucasaugusto.vb@usp.br

Rodolfo I. Meneguette
meneguette@icmc.usp.br

Robson E. De Grande
rdegrande@brocku.ca

Jó Ueyama
joueyama@icmc.usp.br

[1] Institute of Mathematical and Computer Sciences, University of Sao Paulo, Av. Trabalhador Sao Carlense, 400, Sao Carlos, 13566-590, Sao Paulo, Brazil

[2] Department of Computer Science, Brock University, 1812 Sir Isaac Brock Way, St. Catharines, L2S 3A1, Ontario, Canada

or fixed. This means they can be randomly arranged in a dynamically -changing environment known as the detection field. In such a setup, each node is a low-power device [9–11]. However, even though sensing and monitoring the environment provides information about the real-time status of a region, there are limitations to its sensing capacity. In related work [12–20], the presence of floods is determined by the hydrological models that, in general, are defined as a mathematical representation of the water flow and its components over some part of the surface or subsurface [20]. The purpose of the models is to simulate hydrological predictions on the basis of deterministic or empirical equations [21]. However, they have a serious drawback since these hydrological models rely on several variables which can make detecting floods during torrential rainfalls a complex task as there is a need to measure the height of river bank and determine its soil properties.

Moreover, owing to environmental factors, pressure sensors require constant maintenance, which might be costly and they are prone to failure, particularly during flooding when an early warning system is essential [22]. In addition, it is difficult to maintain pressure sensors because of the location where they are placed - rivers, since river water might contain particles and pollutants. Sonar sensors are an alternative [23, 24]. However, they are power-hungry and highly sensitive, and require to be precisely focused on the river, unlike pressure sensors.

It is clear from the development and analysis of the monitoring mechanism that a more straightforward approach can successfully detect floods, particularly when there are torrential rains and flash floods.

In this article, we seek to set out the Urban Flash-Flood Prediction (FLORAS) methodology as a means of estimating the occurrence of floods. This consists of a data science pipeline that is specifically designed to make predictions based on meteorological features, by applying annotations based on water height sensors for training. This type of framework can be deployed in any river where these data are available, even with little knowledge of their physical properties, which is their most evident advantage when compared with hydrological models. FLORAS can use different machine learning techniques for forecasting, such as Naive Bayes, neural networks, random forest, KNN, and any other system.. The idea behind the methodology is that easily attainable heterogeneous data can create a model more quickly for areas affected by floods.

In addition, we implemented a real world setup, by deploying a WSN called e-NOE, in the city of São Carlos, Brazil. This monitors the height of a river by means of pressure sensors, predicts potential flooding risks, and sends warnings to the smartphones of the population at risk [25].

The aim of our methodology is to build multimodal models, which can correlate data from meteorological stations at different river height levels, by establishing a series of significant patterns through different machine learning techniques to detect flood incidents . The models can be trained to learn to establish complex correlations between meteorological data, such as temperature, humidity, wind intensity, and the actual river height – i.e. the ground-truth data are measured through pressure sensors. Only meteorological features are employed to make inferences about forecasting after the training sessions; hence, the data from water height sensors are not needed. This scenario makes it less necessary to maintain these sensors l even though it is still essential to keep track of the river height level over time - as with any machine learning process; the model might degrade over time, and thus require retraining with up-to-date measurements [26].

To the best of our knowledge, this is the first study that has put forward a low-cost flash-flood detection model and provided experimental results in South America. Our system does not require the deployment of new, complex, and costly WSNs for data collection since the meteorological data required by the models are usually made available by specialized weather services. Moreover, our methodology does not require data about river features, such as river banks and soil properties, which gives it a significant advantage when compared with hydrological models. However, we acknowledge that our strategy is only applicable if a series of measurements of both weather variables and water height sensors are available for a particular time window. Hydrological models might still be preferable for remote rivers where these kinds of data are scarce, or in situations where the temporal data are unreliable.

The main contributions of this paper are as follows:

- A data-driven framework to establish an empirical model for flood prediction based only on historical data from meteorological forecasts and water height measurements at a given location, which is thus easily replicable in different regions;
- Assessments on the performance of this model for flood detection , when compared with a well-established, physically-based hydrological model.

The remainder of this article is structured as follows. Section 2 lists and discusses other work in the field related to flood forecasting. Section 3 introduces the system modeling and describes the algorithm in detail. Section 4 outlines the experiments performed to evaluate the algorithm, and the results obtained. Finally, Section 5 concludes the work and makes recommendations for future work.

2 Related works

Flood forecasting and warning systems are essential for both rural areas and urban centers, but in particular large urban centers. Several external factors make devising precise systems for flood forecasting a challenging task, and a number of works have already explored the problem of characterizing it through models. Our research investigates model-based systems and architectures related to flood detection that are based on these previous studies. Table 1 provides a summary of the main features of the related works presented in this section.

Some of the works on flood forecasting are concerned with modeling flood events in a suitable way by relying on several sources, including databases that provide historic data about the environment. The purpose of these studies is to merge different domains and sources into a single decision-based system which can ensure a high degree of accuracy.

A flood mapping model was designed that included three urban basins in the study area: Dahongmen, Qinghe and Bahe, Beijing, China [27]. This involved the machine learning technique which transfers learning into images through the LeNet-5 architecture to improve the mapping of flood-prone areas in the metropolitan regions of China. This resulted in an improvement in Kappa metrics which were 1.4% to 12.5% in the Qinghe basin and 7.1% to 16.5% in the Bahe basin and involved different combinations of transferred modules from those of other models in the same study area.

The Participatory Model for Flood Early Warning Dissemination(MAHP) is another flood forecasting model [28]. The model attempts to predict floods in urban basins by integrating Voluntary Geographic Information (VGI) and wireless sensor networks. The MAHP model is split into independent modular components, each of which is responsible for an activity in the flood forecasting process. Although the MAHP model has several auxiliary modules, its functions can be summarized in three key categories : data acquisition; calibration of the model based on physical formulas in topological data analysis; and, finally, flood forecasting. It is worth noting that this model adopts a similar pattern for flood forecasting, where it needs periodical monitoring.

Correct topological modeling, hydrological flows, and weather conditions lead to significantly precise flood forecasting. However, forecasting systems rely heavily on real-time input information so that they can react in a suitable way to changes in the environment [29]. Thus, some other flood forecasting systems focus on factors in the wireless sensor network.

The work by Choi et al. [30] employs a methodology adapted to a hydrological model designed for a district in Seoul - South Korea. The methodology is based on 3 key factors which are: i) Flood risk analysis, ii) Historical disaster data and; iii) Study site strategies and technologies. The design of the model is based on an ArcHydro/GIS that creates different matrices of products and variables for its modeling and highlights areas that are more prone to flooding. It also defines flooding categories and allows a greater understanding of the impact of torrential rainfall and the way floods occur in these regions.

In Zhao et al. [31], the structuring of a model to map out possible flooding areas is shown in global terms. It uses machine - learning techniques such as K-means and 12 “capture descriptors” in 11,793 stations in an attempt to find associated groups. This enables it to generate three types of regression with algorithms such as random forest, Support Vector Machine (SVM) and power-form function (PF) and make global estimates of flood-prone areas. The

Table 1 Flood forecasting techniques proposed in related works

Technique	WSN	Warnings	AI	Multi-sources	Real-world Environment
Choi et al. [30]		✓			✓
Acosta-Coll et al. [16]		✓	✓		✓
Furquim et al. [25]	✓	✓	✓		✓
Zhao et al. [31]		✓	✓		✓
Fava [28]	✓	✓			✓
Zhao et al. [27]			✓		✓
Tiwari et al. [32]		–	✓		✓
Lammers et al. [33]		–	✓	–	✓
FLORAS	✓	✓	✓	✓	✓

The table summarizes whether the research included a WSN, if it allowed for issuing warnings of imminent floods, if the technique was based on artificial intelligence (AI), if multiple sources of data were considered, and if the models were evaluated on data from real-world environments

results showed it achieved a good performance in indicating possible areas of flooding by pointing out regions that have a tropical, temperate or arid climate.

The reliability of a flood forecasting system is closely tied to its degree of precision and efficiency in detecting floods. If the functionality of a system can be maintained in extreme circumstances of data loss, this ensures a high degree of accuracy when predicting floods.

A IoT-based Natural Disaster Detection and Prediction System (SENDI) has been devised that adopts this approach to ensure appropriate functional operations for prediction [25]. SENDI provides an architecture that is focused on fault tolerance. It relies on IoT, machine learning, and WSN to detect and predict natural disasters and issue warnings. Fault tolerance is built into the system so that the risk of communication failures and destruction of nodes during disasters can be anticipated. The use of a single variable means that the system has an integral dependency because it has to detect floods.

Tiwari et al. [32] provide a model for short-term flood prediction by testing and evaluating neural network techniques, extreme learning machines, and the M5 model tree. The data used was a 4-year historical set of river levels in the Mary River watershed in Australia and the models were evaluated by means of metrics such as coefficient of determination (R2), Nashe-Sutcliffe Efficiency (NSE), root mean square error (RMSE), percent peak deviation (Pdv), and mean absolute error (MAE). The case study results showed that the performance of extreme machine learning and the tree learning of the M5 model, was better than that of the neural network model at delivery times of 1 and 5 h, with a slight difference in accuracy between them for this window.

Lammers et al. [33] propose the design of a model based on neural networks and another model based on standard physics (WRF-Hydro), which are supplied with real-time satellite images to detect floods. Thus, the authors recommend that these algorithms should run on satellites and that they could widen the dimension of the areas that could be reached and also send warnings with a higher degree of confidence. The methodology employed was the regression technique to update flood values in the model domain on the basis of new images so that retraining could be carried out. Their results showed that the model had an accuracy of more than 90% in a simulation when compared with a physical model.

Hydrological models involve greater complexity and incur higher costs in both the development and maintenance of the model in the long term, as well as providing an acceptable degree of precision in their estimates. The physical conditions of the environment change over time, and there is always a need for recalibration.

Our methodology uses data mining to create a multimodal model with heterogeneous data sources that are easier to access. Calibration is performed by machine learning techniques that detect patterns in the data in the event of a flood, which means it can reduce the complexity and cost of calibration.

3 FLORAS: urban flash-flood prediction by means of a multivariate model

We recommend a multivariate classification model for detecting floods that is based on real-environment multi-sourced data, including wireless sensor networks (WSNs). The model, called Urban Flash-Flood Prediction (FLORAS), relies to a great extent on data mining techniques that consist of data analysis and makes use of techniques for exploration to discover new patterns and useful relationships that can represent key information. For this purpose, FLORAS is based on CRISP-DM data mining (Cross Industry Standard Process for Data Mining [34]. FLORAS thus provides real-time analysis when the scenario and conditions change, which allows immediate and customized changes at any time, together with more efficient decision-making with input data [35].

Like CRISP-DM, FLORAS also entails iteratively adjusting the 'problem model' into the contextual field by following its stages: problem/context analysis, modeling, and understanding input data (Fig. 1). Thus, FLORAS can be divided into the following categories:

- **Analysis and understanding of the problem through raw data**: project stage to analyze and understand the data problem and its correlations;
- **Preparing datasets**: design stage to prepare datasets for training the model and ensure it is well structured to find patterns in the database with regard to the problem;
- **Modeling, evaluation and application of the model**: project stage to choose the most suitable Artificial Intelligence technique that will achieve the best result for the model.

In view of this, some statistical analyses and modifications were made with the aim of finding the best possible model for flood detection . All the conducted studies and validations are explained in the next subsection.

3.1 Stages of the methodology

The methodology of this work includes an in-depth analysis of the available weather dataset to enable us to design a suitable training model in FLORAS.

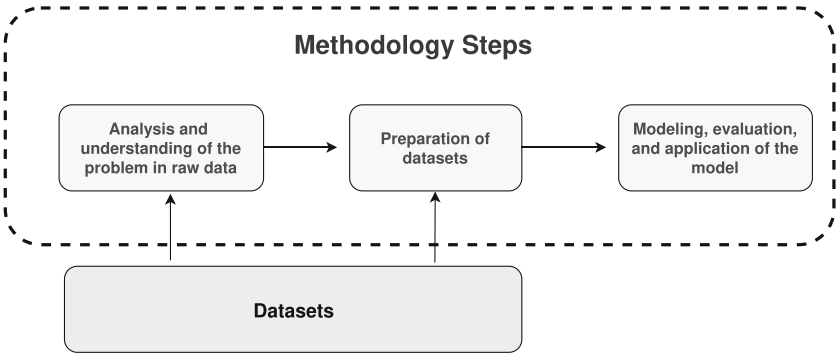


Fig. 1 An abstract view of FLORAS's methodology. Historical data, stored in a dataset, is analyzed, in order to determine the best preprocessing techniques to be employed in the data preparation step. After that, the machine learning model is parameterized, trained and evaluated

3.1.1 Dataset

The Case Study for the development of our model was São Carlos, a city in the rural area of São Paulo State where there are a large number of flood incidents . There is a WSN (Wireless Sensor Network) system, called e-NOE installed in this city to determine its river water level through pressure sensors [12, 36]. As shown in Fig. 2 [12], the monitoring points are strategically installed so that they can take into account the regions that are more susceptible to flooding in the hydrological basin.

Each point has its importance in the context of WSN, where Points 2, 3, 5 and 6 (see Fig. 2) have pressure sensors that can determine the river water level. Point 4, marked with a triangle, is formed of pressure level sensors linked to a rain gauge, and Point 1, marked with a square, is the sinking node of the network that records all the data from the other sensors and periodically saves them in the database-every 5 minutes.

This work has been built on an already existing WSN [25], which consisted of a set of sensors placed in different locations and data collected over a period of time. However, the scope of our work was restricted to point forecasts. For this reason, , we carried out our experiments by evaluating a single, but significant , point of WSN. Point 6 consists of the intersection between several rivers in the region of the city, that also experiences the most flood events [37]. The role of this data in our model is just to provide annotations to train the machine learning models by means of a supervised paradigm. In addition to the data from the WSN (Point 6), the historical data from the region's meteorological station has been correlated and this will be the input data for our model after the model for that point has been trained and calibrated.

The variables used in the weather stations were: (i) Maximum temperature, (ii) Minimum temperature, (iii) Humidity, (iv) Precipitation and (v) Wind intensity, all nominated by a ClimaTempo expert [1]. In fact , this approach can be easily replicated to other points.

The measurements that were made corresponded to a water stream which has a height that is quite low for most of the time (see Fig. 3a). When it rains with an intensity above the flow rate of this water stream, its height starts to increase (Fig. 3b). In more critical scenarios, the rain may cause a flood (Fig. 3c), which is the event which we seek to detect. In any case, after it stops raining, the height of the stream rapidly decreases, until it returns to its original state, as shown in Fig. 3d.

3.1.2 Analysis and understanding of the problem through raw data

An initial stage is essential for contextualizing and understanding the problem in its development. It includes the set of application problems that arise for the the particular features of a model. Hence, , the delimiter selected to extract the data and form the database includes the intersection point between the basins. In our case study, the basins relate to the São Carlos-SP region, which has a lower altitude and results in the largest number of flood events.

To illustrate the flooding problem, we decided to show a study that was carried out in a previous work [39]. This study shows the pattern of behavior of the river in November 2015. In that month, the river reached its highest peak which was more than 600 centimeters (6 meters) in height, according to pressure sensor readings. At this specific point in the geographical region, the systems and models consider

[1] ClimaTempo is a reputable institution that has 30-year experience in monitoring and forecasting weather conditions in Brazil and Latin America. It offers meteorological services and enables access to databases of meteorological stations installed in Brazilian territory [38]

Fig. 2 Layout of the deployment of WSN nodes in the region of the case study [12]

a flooding event to have occurred when the river water level is above 200 centimeters (2 meters). This value was determined by a study in this area reported in a previous related work [40]. Figure 4 shows in detail how the height of the river water level varies daily, where the height of the river is on the y-axis, and the x-axis represents the days of the month in November. It is worth mentioning that the average for November reached 93 centimeters, which can be regarded as high when compared with the same month in previous years.

To demonstrate the height that the river water level can reach, a new chart was created that was confined to November 23rd, 2015, when the river reached its peak. Figure 5 shows the high speed at which the river water level rises. An analysis from Origin 1 where the river is 94.2-centimeter high to Origin 2 that reaches 624.86 centimeters in height, represents an increase of 665% in a short time (45 minutes approximately), which is an anomaly that is of interest for the study.

These analyses show how important it is to design mechanisms to detect floods. An interesting factor worth noting is that the use of a single variable makes the model-dependent. In a hypothetical scenario, a failure in this single variable can have an adverse effect on the flood detection. . Another significant factor should be taken into account with regard related to the scenario – urban rivers – where timely flooding detection is crucial. Floods are usually caused by severe storms that can either last minutes or hours. Even in fast flooding events like the one described in Fig. 5, hydrological models take a long time to discover and issue a warning about the increase in the river water level. The reason for this delay is that the computational cost is very high simply because of the large amount of data processing. Our model is designed to incur a low computational cost

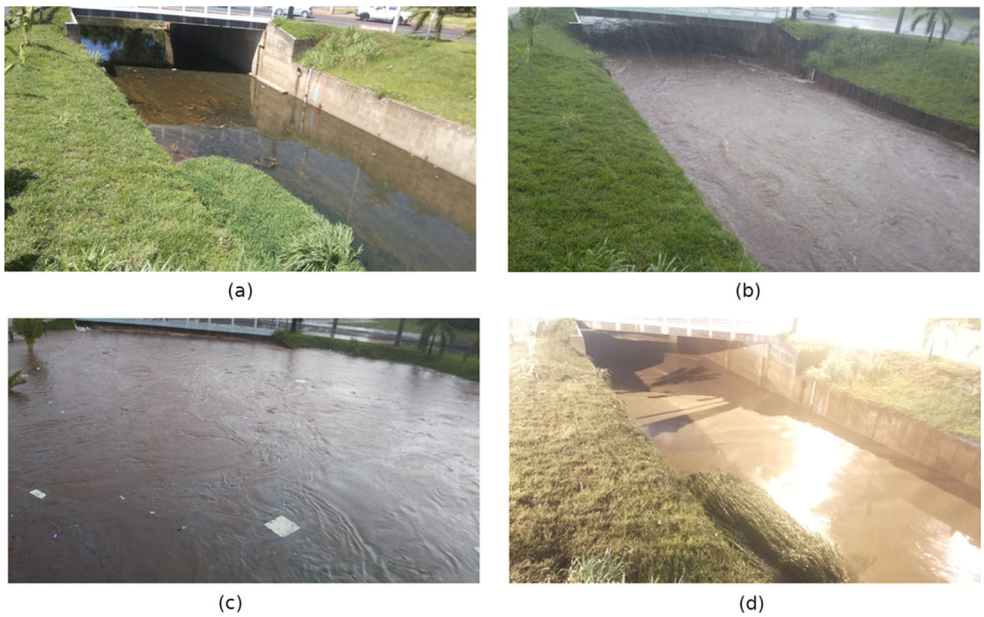


Fig. 3 Water stream at the point analyzed in our experiments. (a) The water stream remains at a low height for most of the time; (b) when it rains with a certain intensity, the height of the stream rises; (c) if there are more extreme events, a flood might occur ; (d) a couple of hours after a flood, the water stream returns to its original state

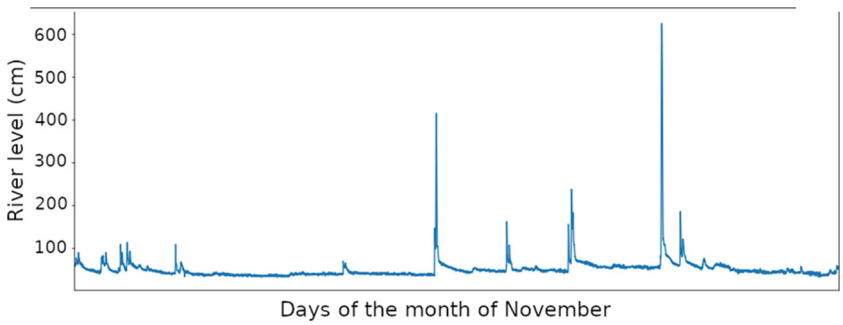


Fig. 4 Water level readings and variation in November, 2015. This data was obtained from the pressure sensors present in the river, deployed as part of the e-NOE project [36]

while employing several data sources . Thus, this model brings together all the improvements necessary to make it an essential tool for society and the responsible stakeholders, such as civil defence departments in municipalities, so that it can mitigate the possible problems that floods can cause across the city and the need for early warning systems.

Our methodology aims to design models that can assist in flood forecasting while being based only on meteorological variables and water height measurements in a given period, without the need for specialized knowledge about the watershed. It should be noted that this kind of knowledge might be important for predicting what kind of meteorological variables might be needed for supplying the model, which in our case was provided by a third-party meteorological service (i.e., Climatempo).

In our case study, we carried out experiments based on data obtained from the sensor located at the most critical location in the area of interest – Point 6, which represents the lowest point in the city. The objective was to create a proof of concept based on these specific sensor data, to assess whether our approach would achieve a satisfactory performance for flood prediction. The same methodology can be applied to different points, though without modeling the dependencies between them. It could also assist strategic flood management without the need for gathering a lot of knowledge from the watershed, as in physical models, which could be useful for remote regions with limited resources.

3.1.3 Preparation of data sets

This stage involves all the activities that are involved in constructing the final dataset. We should emphasize that the methodology took account of the variables from the local sensors, in particular data from hydrometers, and meteorological data provided by a reliable source. These meteorological variables included maximum temperature, minimum temperature, humidity, precipitation and wind intensity. The model considered these variables together with local variables so that it could order its data points sequentially in time.

This dataset plays a vital role in the machine learning training of the models , and hence assessing the performance of the flood event prediction ; inevitably, several ongoing optimizations are involved when preparing the dataset. As a result, we divided the process into three stages where each was necessary for all the data sources, both in the river level database and in the weather station database [41]. The preparation of the data undergoes the following procedures:

1. **Cleaning**: this operation eliminates spurious characters, standardizes formats, reduces inconsistencies and restores missing data. The reading frequencies of several data sources were different.
2. **Transformation**: in this operation, we removed data from the dataset when the information was merged, such as river level and precipitation, to eliminate

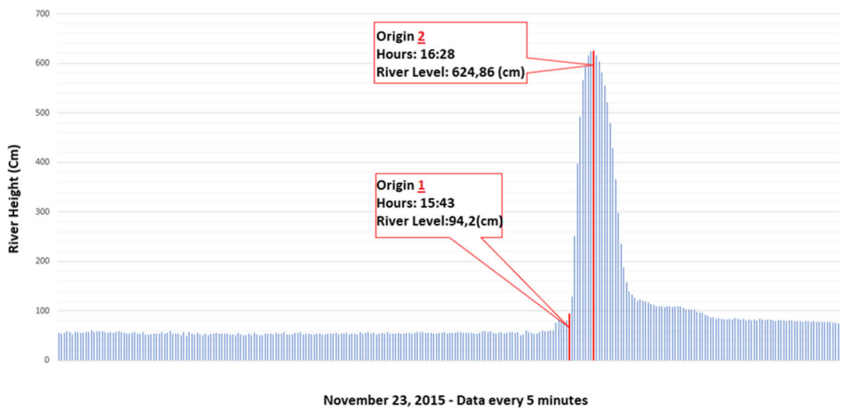


Fig. 5 Flood Analysis on November 23, 2015

redundancies. In addition, as there were different collection times, the frequency of these collections was standardized.

3. **Database integration**: in this stage , all the information from each database is collated . The information is linked in a single database with several variables through this compilation, which makes it more complete and useful for training the model. Figure 6 illustrates the process by showing and comparing the database before and after the necessary and precise treatment.

The river level data were collected every 5 minutes, and the meteorological station data were gathered every hour. Thus , it was necessary to carry out a standardized procedure in the collection to ensure the same time scale occurred every 5 minutes. This meant that the the datasets are all standardized to an updated period of 5 minutes, with the aim of maintaining the river level data at a constant rate. Thus the flood characteristics represented in these data are not lost. After this temporal structuring, the data were recorded on a label to facilitate the classification of the river level. Thus, the Label that indicates flood is derived from the river water level (data from e-NOE). These data are numeric and represent the height of the river. The threshold that indicates the existence of a flood event is set to 200-centimeters high, in compliance with the recommendations of SENDI [25]. Thus, over 200 cm shows that a flood event is taking place.

Label 1 defines the flood, and Label 0 represents the readings below the threshold, and defines a no-flood. It is worth noting that the location where all the readings are recorded is from Point 6 in the e-NOE project. The location relates to an open market which is susceptible to recurrent flooding. This variable is essential for the model in the training phase as its classification was used to overcome the flooding problem. The classification of the training requires data in a time series where there are periods both with and without flooding events; thus, the model can correctly determine the conditions that lead to flooding on the basis of new input data.

As shown in Fig. 7, flooding events do not occur immediately after the onset of rain. A period of rainfall accumulation is required for floods and to cause streams to overflow.

To enable the model to adapt to this concept, the Cross-Correlation [2] technique is applied between two data sources: rainfall data and river level data.

According to Boyd [42], cross-correlation is used to calculate and graph the correlations between two time series X and X. This is defined by the ratio between the sample covariance $\gamma i, j$ and the product of the variances of the time series, σ_i^2 and σ_j^2, as in (1).

$$\rho_{i,j} = \frac{\gamma_{i,j}}{\sigma_i^2 \sigma_j^2} \tag{1}$$

The sample covariance $\gamma_{i,j}$ is expressed in (2), where N is the number of data values:

$$\widehat{\gamma} i, j = \frac{1}{N} \sum_{t=1}^{N} \left[(X_i^t - \overline{X} i)(Xj^t - \overline{X}_j) \right] \tag{2}$$

As a result, the sample cross-correlation is defined by the ratio described in (3):

$$\widehat{\rho} i, j = \frac{\sum_{t=1}^{N} \left[(X_i^t - \overline{X} i)(X_j^t - \overline{X} j) \right]}{\sqrt{\sum_{t=1}^{N} \left(X_i - \overline{X} i \right)^2 \sum_{t=1}^{N} (X_j - \overline{X}_j)^2}} \tag{3}$$

Both data sources relate to Point 6 in the region and the period of three years (that cover 2014, 2015, and 2016). The technique is employed to find the Lag between them, which represents the maximum correlation between the variables. The value found for Lag was 24, and shows the maximum relationship between them. With the aid of this value, we are able to analyze two factors:

- **TC (Concentration Time)**: as shown in (4), where TC is the new variable created (i.e., concentration time), P represents the precipitation data, and N is the size of the database.

$$TC_j = \sum_{i=N-24}^{N} P_i \tag{4}$$

  This formula can be explained as follows: with each new data from the TC, the latest 24 values of the historical precipitation data are added, which will enhance the TC by correctly demonstrating the amount of rain that has accumulated from each new series of data in exact time. Thus, this calculation can assist the model in flood detection.
- **Overflow Interval**: it was empirically discovered through observations of the collected datasets that the basin takes an average of approximately 2 hours to reach the overflow point. This interval specifically applies to the particular region in the case study at its location number 6. The overflow interval can be roughly estimated by multiplying the value 24 by 5. This value of 5 represents the frequency at which the rainfall data and other weather station data were modeled in the final database, with regard to the frequency displayed in the river level data.

[2] Cross-correlation is a standard method of estimating the highest degree in which two series are correlated.

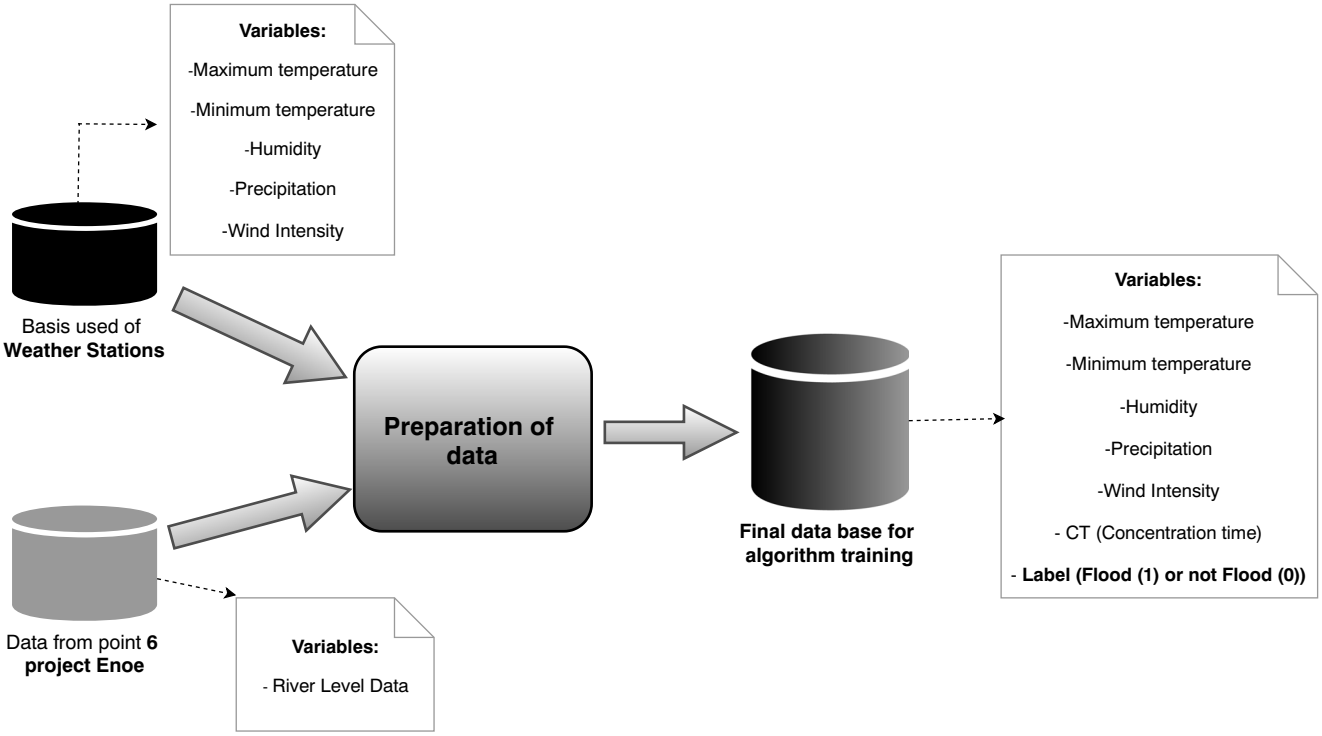


Fig. 6 Procedure for consolidating the database needed for training the machine learning model. Data from weather stations is combined with readings from water height sensors placed directly in the water stream, allowing for supervised learning (i.e., data from the weather stations are the input variables, and the corresponding river height measurements are turned into the labels)

3.1.4 Modeling, evaluation, and application of the model

The modelling determines the techniques that must be used for a particular objective, such as classification, clustering or regression. In this case, the classification technique involves adopting a reframed approach to address this problem unlike most other approaches in the literature, which treat it with the regression technique [43].

Traditional machine learning techniques, which regard each of the readings as independent data points, are still competitive for certain time series-based applications [44]. Since our work involves a proof of concept, these techniques

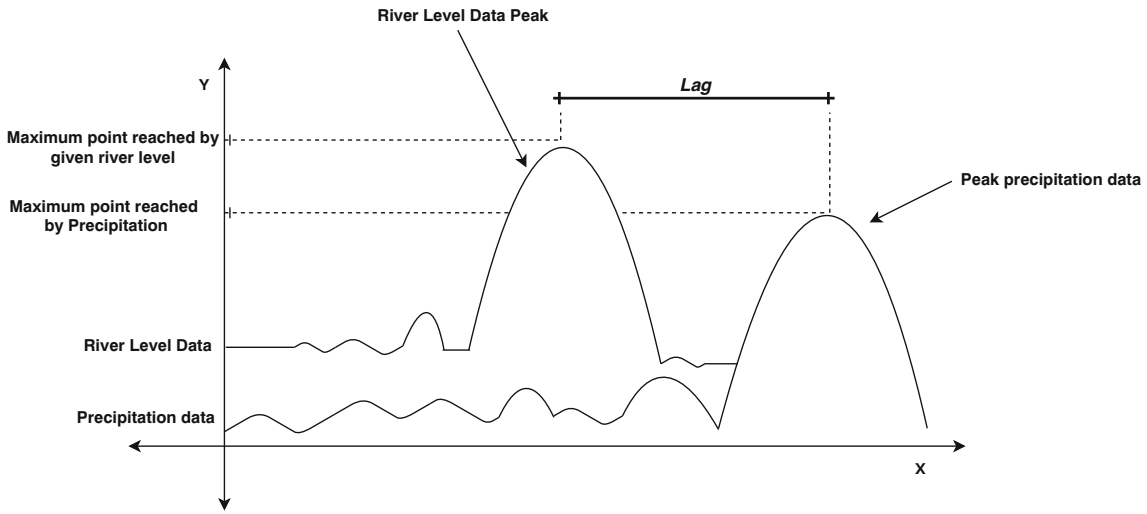


Fig. 7 Relationship between river level data and precipitation data

could provide a suitable assessment of the predictive power of meteorological variables for building machine learning models that can forecast local flooding events.

One of the main advantages of a machine learning model for flood prediction, such as ours, is that the model can be established solely on the basis of historical data. The training procedure entails processing this training data to tune the free parameters of the model. Its aim is to provide predictions based on an unknown function $f : X \rightarrow Y$, which has independent variables derived from a feature space X. Moreover, the dependent variables correspond to a class within a set Y [45]. In our case, the feature space consists of meteorological variables, and there are only two classes (i.e., flood and non-flood). Figure 8 illustrates this.

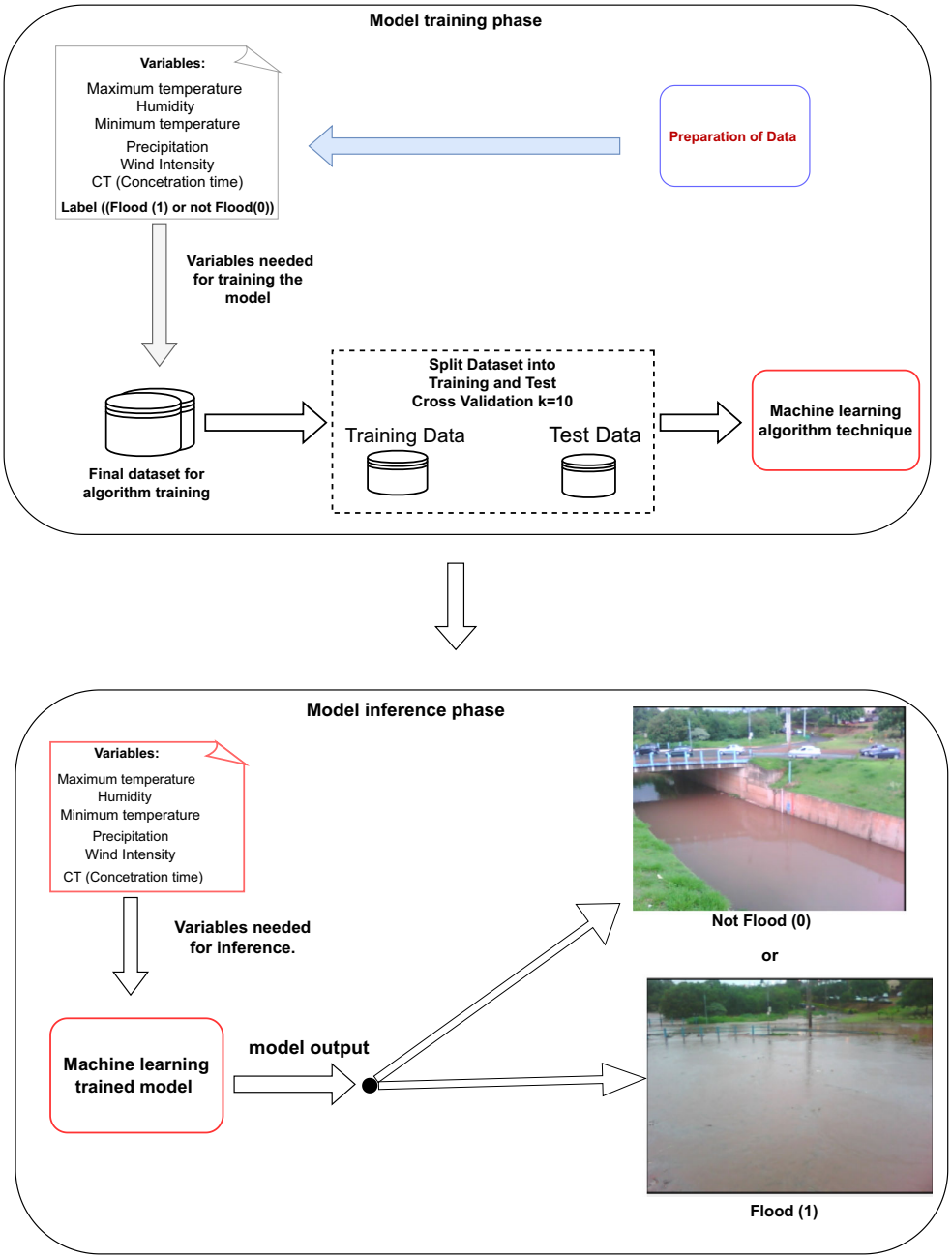


Fig. 8 Machine learning model during the training and inference phases. For training, historical data is split into training and test sets, so that the free parameters of the model are optimized and the validations are carried out. For inference, only the meteorological variables are needed, allowing for a binary prediction to be made (i.e., whether there is an imminent flood or not)

Thus, unlike the physical models, our methodology allows models to be designed without necessarily studying the physical factors that lead to flooding, such as topography, vegetation, soil, and evaporation, provided historical data is available for both the meteorological variables and water height measurements at a given location. We performed the aggregation of heterogeneous data and the model used variables that had been strategically defined by experts [19, 20].

The methodology was designed so that, although the readings from the local sensors are important for training the model, a trained model is only supplied with the meteorological variables in order to provide inferences (i.e., to forecast flooding events). This means that, if any sensor fails, the approach is still capable of making predictions. Hence, no critical issues can occur if Sensor 6 fails, as long as the up-to-date meteorological measurements are still being made available (e.g., by a third-party meteorological service such as Climatempo).

Nonetheless, it is important to fix any problems arising from the sensor as soon as possible, because its readings through time are important to allow the model to be retrained in the future, as data and concept drift can occur [26]. In other words, as in any machine learning model in operation , the distribution of the data may change as the result of different factors (e.g., changes to the environment or to the methods employed for measuring the variables), and the machine learning model might be adapted accordingly so that it can keep making reliable predictions. This adaptation usually involves retraining the model with more up-to-date data.

In summary, the methodology for training the models involved the following stages:

- Data from the local sensors, such as time series with height measurements in centimeters, are used as annotations. Since the flood prediction task has been framed here as a classification problem, a threshold of 200 centimeters was set, above which a flooding event is characterized. This threshold was based on studies in the region [40];
- Meteorological variables were collected and synchronized with the information obtained from the sensors (i.e., the annotations), in order to train the machine learning model.
- Inferences are made by only supplying the model with meteorological variables.

Figure 8 displays the schema of the training phase and also the training phase of the model in more detail. In the next section, the entire process of choosing and applying machine learning algorithms will be explained in more detail.

4 Performance analysis

A performance analysis was conducted to evaluate the accuracy and resilience of our approach. Our model was implemented by means of the Weka tool [46], which is a machine learning tool widely used by the scientific community. Two sets of assessments were made. The first evaluation examined the flood detection and identification algorithms. The second evaluation sought to check the resilience of the methodology, by taking account of failures that can occur if there is a lack of variables in the pattern of behaviour of the model. However, "shallow models" are still competitive for certain time series-based applications [44, 47, 48].

4.1 Assessment of Flood detection

A set of experiments was carried out to evaluate the ability of our methodology to detect flood events. The results were compared with a physical model, which was employed as a reference-point for performance. The methodology was assessed by examining three machine learning techniques. These algorithms and their respective parameters are listed as follows:

- **Probability-based algorithm**: Naive Bayes was utilized where there was no parameter adjustment, as it is non-parametric.
- **Function-based algorithm**: Multilayer Perceptron was employed with a learning rate of 0.1, a momentum of 0.8, and 10 hidden neurons. SVM was also explored with radial base kernel function (RBF - Radial Basis Function). Its gamma was 0.75, and its constant C was 1000;
- Search-based algorithm. K-Nearest Neighbors (KNN) was used with $k = 10$.
- **Decision tree-based algorithm**: Random Forest was employed with maximum depth of 10, and the numeration of 500.

We also examined some important metrics to check the accuracy of our model. Thus, a more detailed analysis took note of precision, recall, and F-Measure. These metrics are obtained from the confusion matrix and are defined as follows:

- **Precision**: this metric is statically determined, and it defines the ratio between the true positives and all elements classified as positives;
- **Recall**: this value is also statically defined, and consists of the ratio between the true positives and all elements that are actually positives;
- **F-Measure**: This metric is obtained by computing the harmonic average between precision and recall.

When evaluating FLORAS, we analyzed the performance of the algorithms by means of the Experimental Planning and Evaluation technique [49]. In this experiment, our objective was to compare the performance of different machine learning techniques in the context of our methodology. To this aim, we considered a balanced subset of the data, designed so that each flooding event was included along with the "no-flood" tuples immediately before and after it. In other words, for each flooding event i of length t_i, we included all the data points correspondent to the flood plus the tuples correspondent to the intervals of length $t_i/2$ right and right after it, ending up with a balanced dataset. This analysis was important for the specific scenario that we have analyzed, in which the water stream remains at a very low level most of the time, and its height only rises when there is heavy rainfall (see Fig. 3).

In this case, k-fold cross-validation is employed with $k = 10$ (i.e., one partition of the dataset is used for evaluating it, and the remaining 9, for training). Thus, it is possible to measure the error estimate more accurately, since the performance metrics tend to converge as k increases. The results demonstrate that the Random Forest algorithm allows a more accurate classification than the other selected classifiers. Figure 9 graphically depicts the analysis in boxplots, that refer to the classifications performed. It should be noted that the item highlighted in red refers to the results of Random Forest and displays the median of the upper accuracy achieved by the other classifiers. The median values can be found in Table 2, and the highest of them is highlighted in bold.

A number of statistical analyses were conducted to validate the results shown in Table 2. Initially, the *Shapiro Wilk* method [50] was used to determine its adequacy and normality and, hence , to conduct parametric or non-parametric tests. Shapiro Wilk is a general test designed to detect all deviations from normality. The test rejects the hypothesis of normality when the value p is less than or equal to 0.05. Therefore, since the values obtained in p were not all greater than 0.05 (Table 2), we considered that the hypothesis of normality with 95% reliability should be rejected. This means the non-parametric test is the most suitable for the subsequent analyses.

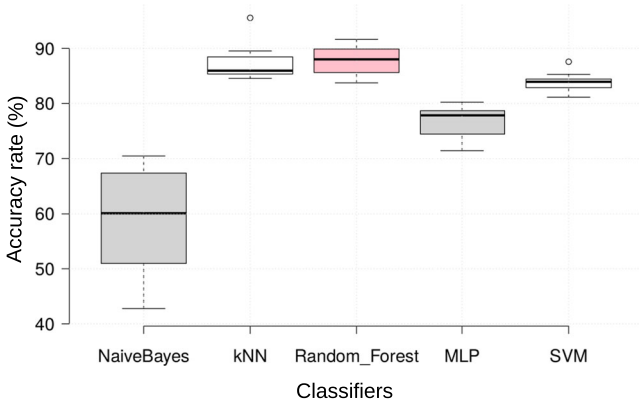


Fig. 9 Boxplot analysis of the degree of accuracy achieved by the classifiers for the detection of floods

Table 2 Average (%) of accuracy and the p-values of the result sets for the detection of floods

Classifiers	Accuracy Avg (%)	Shapiro Wilk p-value
N.Bayes	58,9	0.2966
kNN	87,1	0.0031
R_Forest	87,8	0.9717
MLP	76,6	0.1755
SVM	83,9	0.7846

In the case of the non-parametric test, we made a pair of comparisons with the aid of the Wilcoxon Rank Sum test to determine the statistically significant difference between the groups of results. The result of this analysis is summarized in Table 3. The test states that values less than 0.05 indicate a statistically significant difference between the groups of results.

The results indicate that the Random Forest and KNN classifiers do not show a statistically significant difference between each other for the dataset used for flood detection (i.e., $p > 0.05$), although the classification made by Random Forest on average has a higher degree of accuracy than KNN, as shown in Fig. 9 and Table 2.

The Random Forest algorithm achieves the highest average accuracy; it has an accuracy higher than the error and a greater F-Measure than the other algorithms, as shown in Table 4. These results confirm that Random Forest is the best algorithm for our model. In this work, finding the most suitable search algorithm is an essential factor in adopting our approach, since in the previous analysis, the suitability of the algorithm was determined by its low error rate.

4.2 Statistical analysis of simulated scenarios

We conducted a statistical analysis and simulations to observe the model's resilience in terms of its performance estimation. The analysis also seeks to determine the adaptability of the model. The statistical evaluation was conducted by means of the feature importance [51]. The simulation was carried out by assessing the degraded performance observed when certain variables are removed from the dataset. In this context, the following factors are addressed for a better understanding of the simulation:

- **Model**: the model is designed by following the stages of CRISP-DM, as described earlier, where it employs Random Forest as the search algorithm. Thus, once

Table 3 *P*-values of the pair comparison performed with the *Wilcoxon Rank Sum* test for the identification of floods

	Naive Bayes	R_Forest	kNN	MLP
Random Forest	0.0001	–	–	–
kNN	0.0013	0.2411	–	–
MLP	0.0001	0.0001	0.0013	–
SVM	0.0013	0.0051	0.0051	0.0013

the model has been trained and validated, it only needs meteorological data from weather stations and TC (Concentration Time) to detect floods. A section of the dataset of three years was employed to conduct the training of the model, where the section does not overlap the testing dataset;

- **Testing Data**: the data used in the simulation of the model relate to the period of a whole month in the input dataset – November 2015. It should be noted that these data were not used when training the model;
- **Simulated Scenarios**: Problems that may occur in the real world were represented by observing our model's performance during simulated situations when there may have been a gradual loss of variables. Monitoring of hydrological systems in real-world settings may fail to collect data about some key variables, which are essential to enable the model to detect floods. Thus, we simulated scenarios where these losses might occur to determine their influence on the model's accuracy and behavior. It should be noted that all possible combinations of variables were considered for each scenario. Thus, five scenarios were created based on the six available variables:
  - *Scenario 1 (C1)*: this represents the situation where there is only one variable available to the model for flood detection;
  - *Scenario 2 (C2)*: a combination of all the variables where only two of them are available at a time;
  - *Scenario 3 (C3)*: a combination of all the variables where only three of them are available at a time;
  - *Scenario 4 (C4)*: a combination of all the variables where only four of them are available at a time;
  - *Scenario 5 (C5)*: there are only five variables available to the model. Thus, only one variable is unavailable for flood detection.

The feature importance function was evaluated in the Random Forest algorithm to demonstrate the importance of each variable in the development of the model. This analysis focused on how each variable is obtained by dividing the data into subsets, which are grouped/classified; a variable is assigned to the class in which it is most related . The output of this function represents a degree of importance of each variable in the context of the model. The results obtained were (i) Maximum Temperature = 21%, (ii) Minimum Temperature = 16%, (iii) Humidity = 24%, (iv) Precipitation = 10%, (v) Wind intensity = 4%, and (vi) Concentration Time (TC) = 22%. The results of this analysis are summarized in Fig. 10.

As a result of the second experiment it was clear that the model performed well in the simulation. The significant loss in accuracy is linked to scenarios C1 and C2. The average difference is approximately 18.87% for the best case and 32.59% for the worst case scenario. These two values stand out when compared with the 87.8% obtained from all the available variables that the model needs – a scenario where no variable is removed. However, the discrepancy does not affect the capacity of the model to detect floods, which is an important point when making comparisons with other studies that are not based on multivariable analyses.

In the case of Scenarios 3, 4, and 5, the model performed as expected and with a significant degree of precision as the average difference in accuracy for the best case scenario is 3.13% and for the worst case, it is 11.92%. These two best and worst boundaries stand out when they are compared with the 87.8% obtained from all variables. When the two experiments are analyzed together, it is evident that the wind intensity variable is the one with the lowest percentage in

Table 4 Comparison of the precision results of the classification algorithms *(Precision, recall, and F-Measure)*

Algorithm	Precision	Recall	F-Measure
MLP	0.772 ± 0.045	0.767 ± 0.083	0.765 ± 0.019
KNN	0.873 ± 0.027	0.871 ± 0.036	0.867 ± 0.034
SVM	0.843 ± 0.044	0.839 ± 0.065	0.838 ± 0.011
R_Forest	0.881 ± 0.037	0.878 ± 0.048	0.878 ± 0.006
NaiveBayes	0.594 ± 0.0248	0.589 ± 0.13	0.583 ± 0.054

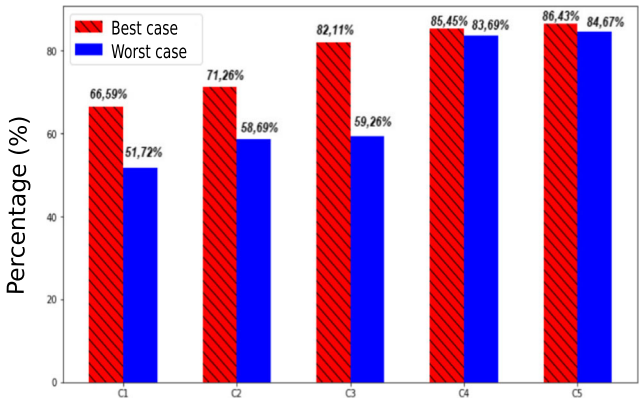


Fig. 10 Analysis of the feature importance for the Random Forest classifier in Scenarios C1, C2, C3, C4, and C5

terms of the importance of the dataset. It is the variable most representative in the worst cases, particularly Scenario 1. Variables, such as minimum temperature and precipitation, are found in the center between the best and worst cases. They need certain combinations to avoid influencing the accuracy of the model in a negative way.. In the best case scenarios, variables such as humidity, concentration time, and maximum temperature appear to be values that can combine to assist in the model's effectiveness in the context of the simulated scenarios.

In view of this, it is clear that the model did not stop working despite the shortcomings of all the scenarios. We can observe that the model is able to correctly detect floods even if half of the variables reported in Scenario 3 are not available. In Scenario 3, an accuracy of 82.45% is still obtained in the best case, which is very close to the 87.8% degree of accuracy obtained by all the variables. These two factors demonstrate the adaptability of the model in the face of adversity, which is a critical feature since it has to be implemented in actual setups to cope with natural disasters, and this makes it an essential tool for detecting floods.

4.3 Comparison with a hydrological model

The literature includes studies of hydrological models for the prediction and avoidance of disasters by floods. There are several models with different configurations and data requirements for their operation. The following models are highlighted in the proposal for the following factors:

- **Concentrated or distributed**: models are regarded as concentrated when they do not take into account the spatial variability of the basin, whereas distributed models have spatially distributed parameters and variables. The distribution may also be temporary , and the models can be classified as either permanent or transient;
- **Stochastic or deterministic**: stochastic models have features that are associated with the concept of probability , and have intervals of probability in their results, whereas deterministic models produce unique values (i.e. the same output) for the same input;
- **Conceptual or empirical**: conceptual Models are those that have physical concepts with aggregated calculation functions. They are unlike empirical models, which are based on statistical analyses such as correlation methods and regression analysis. They also fall into this category, because they employ non-linear empirical methods such as artificial neural networks and Random Forest.

The resources for each category of flood forecasting, must be taken into account when making the choice of the model to be adopted, e.g, simplicity of use, the efficiency of results, computational costs and complexity. Thus, the Storm Water Management Model (SWMM), described by Fava et al. [52], which carries out flood monitoring and forecasting in the urban basin of São Carlos-SP. He made this choice is because it is a model introduced in the same region of study as this paper. SWMM has been widely explored for several quantitative and qualitative types of study. It is a semi-distributed hydrodynamic model that has hydraulic and hydrological components. The work was developed for the São Carlos area by using the water level data from the e-NOE project and an experimental four-rainfall network as input to perform flood simulations.

The data used for the comparison were obtained in November 2015. In this month, the river reached its highest peak, exceeding 600 centimeters - or 6 meters - and is also the month with the highest average rate of flooding in the 3 years of data collection, according to the sensor reading. Thus this month was chosen to make the evaluation and a comparison of performance in flood detection. In light of all the features and operations of SWMM examined here it is important to evaluate and compare our results with the method employed in a simulation environment. SWMM yields continuous results, and as we wished to make a comparison with our model, its results were converted to binary responses (i.e., values either above or equal to 200 cm were considered floods).

Figure 11 shows the flood results obtained by SWMM and the planned method for dealing with the event that occurred on November 23, 2015, and compared this with the responses measured by the sensor. Both models discovered the flood conditions and detected a likelihood of flooding before it occurred. Our model detected the floods up to 25 minutes before they happened, while the SWMM model performed such a detection no more than 5 minutes in advance. Apart from the comparison shown in Fig. 11,

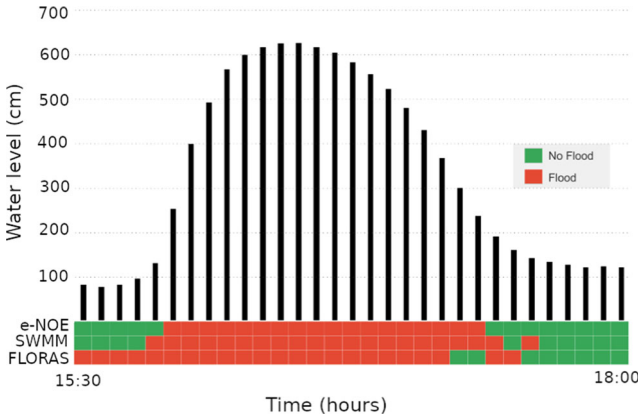


Fig. 11 Performance prediction analysis of e-NOE (i.e., ground-truth), SWMM, and FLORAS for input data on November 23, 2015. The FLORAS methodology allowed for an earlier detection of the flooding event when compared to the SWMM model

another comparison was conducted to assess the efficacy for the entire month of November.

According to Table 5, both models achieved a high degree of accuracy , which was expected because there are far fewer examples of "floods" than "no floods" (i.e., the dataset was severely unbalanced). We also examined the precision and recall measurements. Tables 6 and 7 show the confusion matrices of the models. When predicting natural disasters, such as floods, Type II errors (i.e., false negatives) are more serious than Type I errors (i.e., false positives). The potential damage of missing an event (e.g., not detecting an actual flood, hence not warning the authorities about such a critical event) is higher than the concerns of a false positive (e.g., detecting a flood that does not happen, possibly mobilizing the authorities unnecessarily). Nonetheless, a high number of false positives undermines the credibility of the model, and discourages a proper handling of potential disasters.

Overall, the SWMM model was slightly more accurate than ours. SWMM showed only 33 false positive errors for the 8,259 data points, while our model showed 42 errors, with 37 false positives and 5 false negatives. As a result, the recall of our model was 85.71%, against 100% of the SWMM model. This means there was a higher probability of Type II errors. Precision was also slightly lower for our model. It should be noted that both models have shown a precision score of only around 50%, which was caused by the very high prevalence of "no flood" samples when compared with "flood" incidents. Thus, the absolute number of mistakes can be expected to be high even if the probability of a particular wrong classification is low.

Despite being less accurate than the physical model adopted as a reference-point (i.e., the SWMM model), our model has important advantages that might make it promising for future flood prediction applications. The fact that its performance was only slightly inferior to such a model is an indicatiion of its potential value , since it can be calibrated on the basis of only historical data.

In view of the complexity and developmental cost of the SWMM model, which includes the need for gathering specialist information about the environment, it leads to

Table 5 Accuracy comparative analysis of our model (random forest) and SWMM

Metrics	Our Model (Random forest)	SWMM
# of data points	8259	8259
Hits(%)	8217 (99,49%)	8226 (99,60%)
Misses	42	33
Precision	44.77%	51.47%
Recall	85.71%	100%

Table 6 Confusion matrix for our model, using the Random Forest classification technique. True positives (samples correctly classified as positive) and true negatives (samples correctly classified as negative) are highlighted in green, while the samples incorrectly classified as positive or negative are highlighted in red. The bottom row represents the total number of samples classified as positive or negative, while the rightmost column represents the total number of samples that are actually positive or negative

$n = 8259$	Predicted positive	Predicted negative	
Actual positive	30	5	35
Actual negative	37	8,187	8,224
	67	8,192	

Table 7 Confusion matrix for the SWMM physical model. True positives (samples correctly classified as positive) and true negatives (samples correctly classified as negative) are highlighted in green, while the samples incorrectly classified as positive or negative are highlighted in red. The bottom row represents the total number of samples classified as positive or negative, while the rightmost column represents the total number of samples that are actually positive or negative

$n = 8259$	Predicted positive	Predicted negative	
Actual positive	35	0	35
Actual negative	33	8,191	8,224
	68	8,191	

a longer computational time than our proposed model. The developed model obtained satisfactory results, even though it was regarded as an empirical model. According to Henonin et al. [53], the computational time of an empirical model is shorter than conventional physics-based models, such as SWMM, because its resolution is lower, which results in a smaller amount of data for its development. Hence , this model has become less complex and costly. As shown in the literature review, numerous cities in Brazil are vulnerable and susceptible to flooding and have little or no data available for flood monitoring. In view of this, the installation of an empirical model, as put forward in this paper, is very advantageous, since the required data are only those measured at a sensor level and through the rainfall data provided by the weather stations (this rainfall data is available nationwide).

Thus, this methodology proved to be efficient and can be applied to different points of a WSN network or even other regions where there is little information, so our approach should prove to be useful in the context of strategic flood management.

5 Conclusion

Floods are particularly problematic for countries in tropical areas, such as Brazil, especially during the rainy season. While we are aware that we cannot completely avoid flood events, we can mitigate the damage caused by floods through appropriate in-advance actions with the aid of computing and sensor technologies. This work showed that ML-based flash-flood detection is more lightweight and less complex than the existing hydrological models. We recommended a model that detects floods without the need for river height sensors, which often give rise to maintenance problems. Failures can become a serious hindrance, mainly because timely warnings are needed whenever a flood is detected.

We devised a model that relies on a multivariate approach to combine data from the e-NOE project and weather stations. e-NOE provides height sensor measurements that can be be used during the training phase while weather station organizations 3 provide meteorological data, such as wind intensity, precipitation, and humidity. Meteorological stations can enable us to detect and predict flooding events with data gathered through rain gauges deployed across the country. Even though several previous works have adopted hydrological models for flood detection, most of them are quite complex and costly in the long term. The high degree of complexity originates from the need for a large amount of data from the region's hydrographic basin, and this makes calibration a non-trivial task. Thus, this work strengthens the belief that it is possible to develop a less costly and complex tool without losing the efficient means of detecting floods.

The research presented in this paper consists of a proof of concept, in which data from a single point in the WSN was employed as labels for training a machine learning model fed with meteorological variables related to a broader region. A broader solution could include other points in the watershed, allowing for modeling the dependencies between them and increasing accuracy. On the other hand, such a solution would be inherently more complex for new implementations in real-world scenarios, requiring a more sophisticated infrastructure (e.g., the deployment and maintenance of more sensors), and a deeper knowledge of the terrain and climate.

In future work, we intend to add more weather variables, such as atmospheric pressure and lightning, to the model which might help to fine-tune its estimations and provide a more precise risk assessment of flooding. We believe that visual sensor data can also be incorporated in the model that can include associated factors when determining floods. These variables might significantly correlate with river water levels, while assuming that there might be skewing in the samples. Finally, we intend to explore more robust training techniques and tools, such as deep learning and recurrent neural networks, to have a more detailed understanding of the complexities that can be found in our multivariate sample data sources.

Acknowledgements This work was partially supported by Coordination of Superior Level Staff Improvement (CAPES-PROEX), Brazil, Sao Paulo Research Foundation (FAPESP), Brazil - grants 2021/10921-2 and 2020/07162-0, and NSERC Discovery grant, Canada.

References

1. Hughes D, Ueyama J, Mendiondo E, Matthys N, Horré W, Michiels S, Huygens C, Joosen W, Man KL, Guan S-U (2011) A middleware platform to support river monitoring using wireless sensor networks. J Braz Comput Soc 17(2):85–102
2. Wu W, Emerton R, Duan Q, Wood AW, Wetterhall F, Robertson DE (2020) Ensemble flood forecasting : current status and future opportunities. WIREs Water 7(3):1432. https://doi.org/10.1002/wat2.1432
3. Kao I-F, Zhou Y, Chang L-C, Chang F-J (2020) Exploring a long short-term memory based encoder-decoder framework for multi-step-ahead flood forecasting. J Hydrol 583:124–631. https://doi.org/10.1016/j.jhydrol.2020.124631
4. Zanchetta ADL, Coulibaly P (2020) Recent advances in real-time pluvial flash flood forecasting. Water 12(2):570

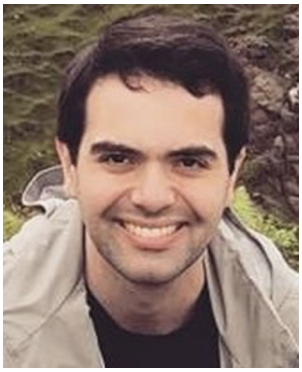
5. Chen C, Hui Q, Xie W, Wan S, Zhou Y, Pei Q (2021) Convolutional neural networks for forecasting flood process in internet-of-things enabled smart city. Comput Netw 186:107744. https://doi.org/10.1016/j.comnet.2020.107744
6. Diniesh V, Murugesan G, Jude MJA, Jayanth E, Rishikesh N, Nanthini K (2021) An experimental study and analysis of impact on mobile sink in wireless sensor networks. In: Advances in smart system technologies, pp 253–260. Springer
7. Chen X, Hu Y, Dong Z, Zheng P, Wei J (2021) Transformer operating state monitoring system based on wireless sensor networks. IEEE Sensors J, pp 1–1. https://doi.org/10.1109/JSEN.2021.3050763
8. Bendigeri KY, Mallapur JD, Kumbalavati SB (2021) Real-time monitoring of crop in agriculture using wireless sensor networks. In: Raj JS, Iliyasu AM, Bestak R, Baig ZA (eds) Innovative Data Communication Technologies and Application, pp 773–785. Springer
9. Freitas DJ, Marcondes TB, Nakamura LHV, Meneguette RI (2015) A health smart home system to report incidents for disabled people. In: 2015 International conference on distributed computing in sensor systems, pp 210–211. https://doi.org/10.1109/DCOSS.2015.28
10. Maschi LF, Pinto AS, Meneguette RI, Baldassin A (2018) Data summarization in the node by parameters (dsnp): local data fusion in an iot environment. Sensors 18(3):799
11. Filho GPR, Neto JRT, Valejo A, Meneguette RI, Villas LA, Ueyama J (2018) Um sistema de controle neuro-fog para infraestruturas residenciais via objetos inteligentes. In: Anais do XXXVI Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos. SBC
12. Ueyama J, Faiçal BS, Mano LY, Bayer G, Pessin G, Gomes PH (2017) Enhancing reliability in wireless sensor networks for adaptive river monitoring systems: reflections on their long-term deployment in brazil. Comput Environ Urban Syst 65:41–52. https://doi.org/10.1016/j.compenvurbsys.2017.05.001
13. Srinithi A, Sumathi E, Sushmithawathi K, Vaishnavi M, Muthukumaran M (2019) An embedded based integrated flood forecasting through ham communication. Asian J Appl Sci Technol (AJAST) 3:63–67
14. Ming X, Liang Q, Xia X, Li D, Fowler HJ (2020) Real-time flood forecasting based on a high-performance 2-d hydrodynamic model and numerical weather predictions. Water Resour Res 56(7):2019–025583. https://doi.org/10.1029/2019WR025583
15. Wijayarathne DB, Coulibaly P (2020) Identification of hydrological models for operational flood forecasting in St. John's, newfoundland, canada. J Hydrol : Regional Studies 27:100646. https://doi.org/10.1016/j.ejrh.2019.100646
16. Acosta-Coll M, Ballester-Merelo F, Martínez-Peiró M (2018) Early warning system for detection of urban pluvial flooding hazard levels in an ungauged basin. Nat Hazards 92(2):1237–1265
17. Mostafa E, Mohamed E et al (2014) Intelligent data classification and aggregation in wireless sensors for flood forecasting system. In: Proceedings of 2014 mediterranean microwave symposium (MMS2014), pp 1–8. IEEE
18. Chen D, Liu Z, Wang L, Dou M, Chen J, Li H (2013) Natural disaster monitoring with wireless sensor networks : a case study of data-intensive applications upon low-cost scalable systems. Mobile Netw Appl 18(5):651–663
19. Devia GK, Ganasri BP, Dwarakish GS (2015) A review on hydrological models. Aquatic procedia 4:1001–1007
20. Pandi D, Kothandaraman S, Kuppusamy M (2021) Hydrological models : a review. Int J Hydrol Sci Technol 12(3):223–242
21. Kemp KK (1993) Environmental modeling with gis: a strategy for dealing with spatial continuity. National Center for Geographic Information and Analysis (NCGIA
22. Ding Y, Zhu Y, Feng J, Zhang P, Cheng Z (2020) Interpretable spatio-temporal attention lstm model for flood forecasting. Neurocomputing 403:348–359. https://doi.org/10.1016/j.neucom.2020.04.110
23. Han S, Coulibaly P (2017) Bayesian flood forecasting methods : a review. J Hydrol 551:340–351. https://doi.org/10.1016/j.jhydrol.2017.06.004. Investigation of Coastal Aquifers
24. Le X-H, Ho HV, Lee G, Jung S (2019) Application of long short-term memory (lstm) neural network for flood forecasting. Water 11(7):1387
25. Furquim G, Filho GPR, Jalali R, Pessin G, Pazzi RW, Ueyama J (2018) How to improve fault tolerance in disaster predictions : a case study about flash floods using IoT, ML and real data. Sensors 18(3):907
26. Sculley D, Holt G, Golovin D, Davydov E, Phillips T, Ebner D, Chaudhary V, Young M, Crespo J-F, Dennison D (2015) Hidden technical debt in machine learning systems. Advances in neural information processing systems 28:2503–2511
27. Zhao G, Pang B, Xu Z, Cui L, Wang J, Zuo D, Peng D (2021) Improving urban flood susceptibility mapping using transfer learning. J Hydrol 126777:602
28. Fava MC (2015) Modelo de alerta hidrológico com base participativa usando sistema de informações voluntárias para previsão de enchentes, PhD thesis, Universidade de São Paulo
29. Zanchetta AD, Coulibaly P (2020) Recent advances in real-time pluvial flash flood forecasting. Water 12(2):570
30. Choi Y, Kang J, Kim J (2021) Urban flood adaptation planning for local governments: hydrology analysis and optimization, vol 59
31. Zhao G, Bates P, Neal J, Pang B (2020) Design flood estimation for global river networks based on machine learning models. Hydrology and Earth System Sciences Discussions 2020: 1–25
32. Tiwari MK, Deo RC, Adamowski JF (2021) Short-term flood forecasting using artificial neural networks, extreme learning machines, and m5 model tree. In: Advances in Streamflow forecasting, pp 263–279. Elsevier. https://doi.org/10.1016/B978-0-12-820673-7.00012-3. https://www.sciencedirect.com/science/article/pii/B9780128206737000123
33. Lammers R, Li A, Nag S, Ravindra V (2021) Prediction models for urban flood evolution for satellite remote sensing. J Hydrol 603:127175
34. Martínez-Plumed F, Contreras-Ochando L, Ferri C, Orallo JH, Kull M, Lachiche N, Quintana MJR, Flach PA (2019) Crisp-dm twenty years later: from data mining processes to data science trajectories. IEEE Trans Knowl Data Eng
35. Wirth R, Hipp J (2000) Crisp-dm : Towards a standard process model for data mining. In: Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining. Springer-Verlag London, UK, vol 1
36. Furquim G, Pessin G, Faiçal BS, Mendiondo EM, Ueyama J (2016) Improving the accuracy of a flood forecasting model by means of machine learning and chaos theory. Neural Comput Appl 27(5):1129–1141
37. Sarmento Buarque AC, Bhattacharya-Mis N, Fava MC, Souza FAAd, Mendiondo EM (2020) Using historical source data to understand urban flood risk : a socio-hydrological modelling application at gregó,rio creek, brazil. Hydrological Sci J 65(7):1075–1083

38. Escada P, Coelho CA, Taddei R, Dessai S, Cavalcanti IF, Donato R, Kayano M, Martins ES, Miguel JC, Monteiro M et al (2021) Climate services in Brazil : Past, present and future perspectives. Climate Services 24:100276
39. Brito LAV, Bressiani D, Ueyama J (2018) Explorando aprendizado de máquina com multivariáveis para previsao de enchentes em ambientes iots : um estudo empırico no sistema de monitoramento de rios e-noé. Anais do II Workshop de Computação Urbana (COURB) 2018, vol 2(1/2018) . SBC
40. Furquim GA (2017) Uma abordagem tolerante a falhas para a previsão de desastres naturais baseada em iot e aprendizado de máquina. PhD thesis, USP São Carlos
41. Roiger RJ (2017) Data mining : a tutorial-based primer chapman and Hall/CRC
42. Boyd DW (2001) Chapter 8 - stochastic analysis. In: Boyd DW (ed) Systems analysis and modeling, pp 211–227. Academic Press. https://doi.org/10.1016/B978-012121851-5/50008-3
43. Vieira AC, Garcia G, Pabón RE, Cota LP, de Souza P, Ueyama J, Pessin G (2021) Improving flood forecasting through feature selection by a genetic algorithm–experiments based on real data from an amazon rainforest river. Earth Sci Inform 14(1): 37–50
44. Wang Z, Hong T, Piette MA (2020) Building thermal load prediction through shallow machine learning and deep learning. Appl Energy 263:114683
45. Mello RFd, Ponti MA (2018) Statistical learning theory. Mach Learn, pp 75–128
46. Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P, Witten IH (2009) The weka data mining software : an update. ACM SIGKDD Explorations Newsl 11(1):10–18
47. Karasu S, Altan A (2019) Recognition model for solar radiation time series based on random forest with feature selection approach. In: 2019 11th International Conference on electrical and electronics engineering (ELECO). IEEE, pp 8–11
48. Luo J, Zhang Z, Fu Y, Rao F (2021) Time series prediction of covid-19 transmission in america using lstm and xgboost algorithms. Results in Physics 27:104462
49. Healey P, Rothman H, Hoch PK (1986) An experiment in science mapping for research planning. Res Policy 15(5):233–251
50. Shaphiro S, Wilk M (1965) An analysis of variance test for normality. Biometrika 52(3):591–611
51. Lakshmanan V, Robinson S, Munn M (2020) Machine learning design patterns. O'Reilly Media
52. Fava MC, Mazzoleni M, Abe N, Mediond EM, Solomatine DP (2018) An approach for urban catchment model updating 13th international conference on Hydroinformatics
53. Henonin J, Russo B, Mark O, Gourbesville P (2013) Real-time urban flood forecasting and modelling – a state of the art. J Hydroinf 15(3):717–736. https://iwaponline.com/jh/article-pdf/15/3/717/387049/717.pdf. https://doi.org/10.2166/hydro.2013.132

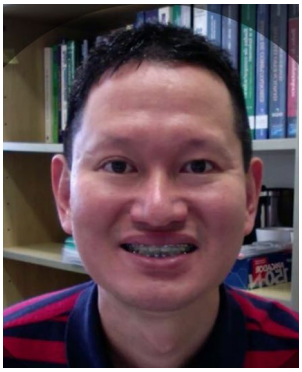
Lucas A. V. Brito is a PhD student at the Institute of Mathematics and Computer Sciences of the University of São Paulo (ICMC-USP), São Carlos. He graduated in Computer Engineering at the University of Araraquara (Uniara, 2015), where he was a PROUNI scholarship holder and received an award from CREA-SP for excellent academic performance. He obtained his Master's degree at ICMC-USP (2019). He has experience in Logic Programming and Artificial Intelligence with an emphasis on DevOps and MlOps. His research interests include Data Mining, Machine Learning, Robotics and Computer Engineering.

Rodolfo I. Meneguette is an Assistant Professor at the University of São Paulo (USP. He received his Master's degree in 2009 from Universidade Federal de São Carlos (UFSCar) and his doctorate from the University of Campinas (Unicamp), Brazil, in 2013. In 2017 he did his post-doctorate at the PARADISE Research Laboratory, University of Ottawa, Canada. His research interests are in the areas of vehicular networks, resources management, mobility flows and vehicular clouds.

Robson E. De Grande is an Associate Professor in the Department of Computer Science at Brock University, Canada. He received his PhD in Computer Science from the University of Ottawa, Canada, in 2012. His research interests include large-scale distributed and mobile systems, cloud computing, performance modelling and simulation, computer networks, vehicular networks, intelligent transportation systems, and distributed simulation systems, actively contributing to these areas. He has been a technical program and special session co-chair of several IEEE and ACM-sponsored conferences, including IEEE/ACM DS-RT, ACM MobiWac, ACM DIVANet, and IEEE DCOSS International Workshop on Urban Computing.



Caetano M. Ranieri graduated in Computer Science at the Sao Paulo State University (2013) and did his Master's degree (2016) and PhD (2021) at the Institute of Mathematical and Computer Sciences of the University of Sao Paulo (ICMC-USP). During his PhD, he worked as a visiting scholar at the Heriot-Watt University, Scotland (2020). He has experience in Machine Learning, Deep Learning, Neurorobotics, and Human-Robot Interaction. Performed research focused on classifying human behaviour based on multimodal signals and its replication on biologically-inspired robotics systems. Currently, he works as a postdoctoral research fellow at ICMC-USP, with research focused on artificial intelligence in the context of the Internet of Things.



Jó Ueyama is a Full Professor at the Institute of Mathematics and Computer Science (ICMC) of the University of São Paulo (USP). Prof. Ueyama is also a Brazilian Research Council (CNPq) fellow since 2014. He completed his PhD in computer science at the University of Lancaster (UK) in 2006. Before joining USP, he was a research fellow at the University of Kent at Canterbury (UK). Jó has published 64 journal articles and more than 100 conference papers. His main research interest includes Computer Networks, Security, and Blockchain.