Journal of Information and Data Management, 2025, 13:1 doi: 10.5753/jidm.2025.XXXXXX

Flood Prediction Using Machine Learning

Eduardo Eberhardt Pereira [**Universidade de Caxias do Sul** | ***eepereira@ucs.br***]
Daniel Luis Notari [**Universidade de Caxias do Sul** | ***dlnotari@ucs.br***]

Universidade de Caxias do Sul, Rua Francisco Getúlio Vargas, 1130, 95070-560, Caxias do Sul, RS, Brasil.

Abstract Heavy rainfall events that cause floods are natural and unavoidable, yet result in devastating damage to society and, therefore, must be studied to mitigate their impacts. This study presents a predictive Artificial Intelligence model capable of classifying the occurrence of flood events using Machine Learning techniques, including Multi-Layer Perceptron (MLP) neural networks and decision trees. This research was conducted in two separate case studies: one in the urban area of Hugo Cantergiani, the regional airport in Caxias do Sul, Brazil, and the second on the Faxinal microbasin, also located in the city. The model was trained using meteorological, such as rainfall, temperature, atmospheric pressure, and others, and hydrological data such as Antecedent Precipitation Index (API). The data collected from Instituto Nacional de Meteorologia (INMET) for the first case study, and provided by Serviço Autônomo Municipal de Água e Esgoto de Caxias do Sul (SAMAE) for the second. The first case study was treated as a classification problem, whereas the second was a regression problem. The performance of the model was evaluated using metrics coherent to each case study, such as accuracy and F1-Score for the first case and R^2, Mean Absolute Error (MAE), and Root Mean Square Eeviation (RMSE) for the second case study. Explainability was also explored using the Shapley Additive Explanations (SHAP) method. The INMET case study obtained 93.3% accuracy in predicting one day in the future. In SAMAE case study, it was possible to reach R^2 of 0.984 in the complete dataset, but only 0.779 for the outliers, that are the cases when floods happened.

Keywords: Flood prediction, Artificial Intelligence, Machine Learning, Neural Network, Extreme-event prediction

1 Introduction

The Intergovernmental Panel on Climate Change (IPCC) is an United Nations (UN) entity created to assess the impacts of climate change in the world. In its Sixth Assessment Report, the group reports that the average global temperature is 1.1°C higher than it was in the pre-industrial era [IPCC, 2023]. Of these, 1.07°C was caused by anthropogenic actions.

In that regard, climate changes go beyond the growth of the average temperature. The same document by IPCC states with high confidence that human intervention is contributing to increasing the frequency of extreme events. Further evidence is given by World Weather Attribution [2024], which used computer models to quantify human influence in extreme events. Of the sixteen flood events analyzed by the group, fifteen were caused by heavy rainfall resulting from anthropogenic actions.

In 2004, approximately a quarter of all registered natural disasters worldwide were flood events [Munich Re Group, 2004]. In Brazil, between 2017 and 2022, 55.5 billion reais were spent on damage repair due to heavy rainfall, and 14.8 million people were displaced or killed as a result [CNM, 2022]. Between September 2023 and May 2024, the state of Rio Grande do Sul in Brazil suffered several casualties due to three different flood events that consecutively broke historical records in terms of rainfall and urban flooding [Marengo *et al*., 2024].

Floods are among the most common, fatal, and destructive extreme events [Razavi *et al*., 2020; Marengo *et al*., 2024]. They can be defined as the temporary cover of land by water outside of its normal confinement, invading terrains not normally occupied by it [FLOODsite, 2005; de Sene and Moreira, 2012]. There are multiple different forms of floods, including heavy rainfall, snowmelt, tsunamis, a rise in the level of bodies of water, drainage system saturation, and reservoir leakages [Penning-Rowsell and Fordham, 1994]. Another type is flash floods, characterized by intense, hardly predictable, short-period rainfall [INMET, 2025].

In order to prevent damage from these disasters, efficient disaster management is essential. The process involves several phases, including preparation, response, and recovery [Sharma *et al*., 2021]. Early warning flood systems enter the preparation phase by ensuring that the public defense is not caught off guard, and prevention mechanisms are well established by the time the event occurs [Sharma *et al*., 2021].

Early warning systems can be implemented using Artificial Intelligence and Machine Learning techniques [Sharma *et al*., 2021]. Artificial Intelligence can be defined as the study and development of rational agents, which are computational systems projected to make decisions based on mathematical or logical models, without relying on explicit algorithms [Russel and Norvig, 2022]. Machine Learning is an area of Artificial Intelligence that focuses on developing models capable of learning from training data, thereby increasing their performance after analysis. In that regard, Machine Learning models can discover patterns in data and help solving problems that even humans cannot understand perfectly, serving as a nondeterministic alternative to unsolved problems [Russel and Norvig, 2022].

Therefore, floods are disasters that affect most cities in the world, and both their frequency and impact are intensified by climate change. Although impossible to avoid, those disasters can have their damage controlled by predicting the

event and acting on it. This project proposes the development of two early warning Machine Learning models to support decision-making on flood events.

The first is a binary classification model that estimates flood occurrence in the urban area surrounding Hugo Cantergiani Airport, using meteorological and hydrological data from flood events during the period from September 2023 to May 2024 as the independent variables. The second model is a regression model that estimates the water column level of the Faxinal microbasin's reservoir, also using meteorological and hydrological data from December 9, 2024, to June 30, 2025. Both areas are located in Caxias do Sul, Brazil. The first case study primarily maps flash floods and drainage system saturation caused by heavy rainfall floods, while the second estimates the water column level in a reservoir to avoid a leakage that could cause a flood. More detailed explanations of the input and output variables will be discussed in section 3.

This article is divided into five subsections. After the introduction, section 2 explores the current state of the art and analyzes how other studies are related to this one. Section 3 describes the methodology used for the development of the models. Section 4 shares the results obtained by the models, using relevant metrics to assess their performance. At last, section 5 concludes the findings and contributions obtained in this study, as well as highlights its limitations and suggests possible paths for future works to improve upon.

2 Related Works

Several studies have been conducted in trying to estimate the occurrence of flood events. This section will discuss the methodologies employed in these other works, placing this study in the context of the current literature.

First, there is a strong debate over the quantity of data necessary for a model to be reliable. Some authors defend that a model should have at least ten years of data to be assertive [Mosavi *et al*., 2018]. Others argue that climate changes make data from the past quickly unreliable [Wasko *et al*., 2021]. The authors believe that, for a flood prediction model, the time frame is less relevant than the number of extreme events within it. For example, ten years of data with only a single flood event is just as relevant as a single year of data with the same number of events. The models usually lack examples of extreme events, not normal events.

There is also a discussion about using Artificial Intelligence (AI) and statistical methods without any hydrological modeling. The critics of these methods raise a point about the lack of understanding of how the model makes its predictions, and that some level of hydrological comprehension must exist to determine if the predictions are coherent or merely a mathematical model overfitted to the training data [Klemeš, 1983; Todini, 1988].

Those discussions show how important the dataset is for these models. However, it is also found that one of the most significant barriers identified by researchers who use AI for flood prediction is having access to robust and reliable data [Mosavi *et al*., 2018]. Some studies have attempted to collect their own data using Internet of Things (IoT) devices [Furquim *et al*., 2018]. Despite the promising approach, the effort to install those equipments made it unfeasible to support many variables, resulting in a model that only uses rain and the river level for prediction, which ultimately could reach a R^2 of only 0.681 for the river level prediction and 80% accuracy for the civil defense alerts.

Besides the data, the models and metrics used are also essential parts of the AI studies. The most used models for flood prediction are Artificial Neural Networks (ANN), Decision Trees, Wavelet Neural Network (WNN), Support Vector Machine (SVM), Support Vector Regression (SVR), and Adaptive Neuro Fuzzy Inference System (ANFIS) [Mosavi *et al*., 2018]. The most used metrics are R^2 and RMSE. R^2 shows how the sum of the errors of the model compares to the sum of the errors when using the mean to predict, with 1 being the best value it can get, and anything below 0 means that the model is worse than predicting using the mean [Gao, 2023]. RMSE is the square root of the mean of the model's errors, powered by two, being 0 its best possible value, while other values indicate larger errors on the measurement scale of the output [Hodson, 2022].

MAE is also a recurrent metric. It is very similar to RMSE, but uses the absolute value to calculate the error, instead of powering it by two [Hodson, 2022]. This makes the interpretation simpler, as 0 is its best possible value, and anything above that represents the mean of the absolute values of the errors.

Tabbussum and Dar [2021] used observational data collected by Irrigation and Flood Control Department from Srinagar, India, and obtained and R^2 of 0.97. Pereira Filho and dos Santos [2006] did the same in São Paulo, obtaining 0.95 using ANNs. Aichouri *et al*. [2015] maintains the trend in Algeria, achieving 0.9 for runoff and comparing these results with the ones obtained by a simple linear regression model.

Dawson *et al*. [2002] uses artificial intelligence without any hydrological modeling and compares the results with autoregressive models. This class of studies improves on statistical flood prediction, but is absent from physical knowledge in the process.

Between the studies using hybrid models, Adikari *et al*. [2021] uses the API method, which defines the soil moisture based on recent rain absorption, for flood prediction. This method is interesting because it uses simple and accessible data, but still attempts to provide some physical explanations. It is known that, if the soil is saturated (which means a high API value), it is more likely that the rain will turn into superficial runoff and cause flooding.

Noymanee *et al*. [2017] uses ANN to predict flooding using open data of the water level in different points in the river Pattani, in Thailand. The study defends that open data is crucial to develop the potential of predictive models, and concludes that rain alone is not enough for predicting floods.

Khosravi *et al*. [2018] uses multiple tree-based models to define the susceptibility of flood in different points in the Haraz basin, in Iran. Kourgialas and Karatzas [2017] also works with this methodology in Greece, but on a national scale. This approach is very interesting for mapping risk, although it does not function as an early warning, as the present study intends.

A brief summary of the studies mentioned above can be

found at Appendix A. Table A.1 shows the methodology used for data in these articles, describing how the samples were acquired, which variables were used, and the period covered in the dataset. Table A.2, on the other hand, points out which kinds of models were used, their output variables, which metrics were used to evaluate the performance, and what values were encountered for those metrics.

3 Methodology

This study follows the Sample, Explore, Modify, Model, and Assess (SEMMA) methodology for Artificial Intelligence development [Sharda *et al*., 2019]. This method divides the process into sampling and exploring the data, modifying it to fit the application, training the model, and testing how well it generalizes the data. Since SEMMA is an iterative method, if the results of the test step are not as good as expected, the process can be restarted from any given step in order to improve the model. The general development process, led by the SEMMA methodology, is illustrated in Figure 1, and further explanations will be provided later in this section.

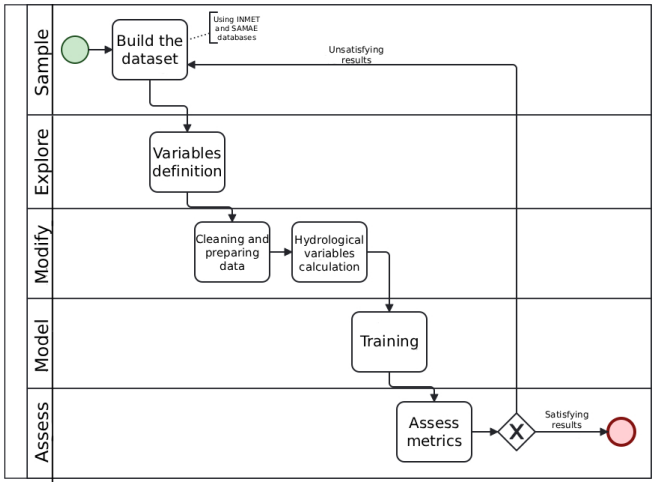


Figure 1. SEMMA methodology

This study used meteorological and hydrological data collected from Instituto Nacional de Meteorologia (INMET) [1] and data provided to the authors by Serviço Autônomo Municipal de Água e Esgoto de Caxias do Sul (SAMAE)[2], the company responsible for the management of waters in Caxias do Sul. Each dataset was used as one of the case studies.

3.1 INMET case study

The INMET data was obtained from their official website[3] in September 7, 2025. The meteorological station, as well as the rain gauge, is located at Airport Hugo Cantergiani. The data is collected by an INMET-hired observer three times a day, at 8 A.M., 12 P.M., and 6 P.M. in local time.

The rain data, however, is collected only once a day, computing only the total rain in the last 24 hours, forcing a daily grouping of the data. The hypothesis is that rain and other features derived from it are strong predictors and are very likely to improve flood prediction. Therefore, this grouping should be worth it. In order not to discard two readings every day, the data for every day was grouped, gathering the maximum, minimum, and average readings of the variables. The input features are detailed in Table 1. The rational method and API method were used in this case study to guarantee some level of hydrological understanding in the model [Viessman and Lewis, 2003; Tucci, 2004; Teixeira, 2010].

Table 1. Features used in INMET case study

Feature Name	Description
Date	The day in which the data was collected (YYYY-MM-DD)
Daily rain	The total rain during the day (in mm)
Mean Temperature	Average temperature during the day (in Celsius degrees)
Max Temperature	Maximum temperature during the day (in Celsius degrees)
Min Temperature	Minimum temperature during the day (in Celsius degrees)
Thermal Amplitude	Difference between the maximum and minimum temperature during the day (in Celsius degrees)
Mean Humidity	Mean humidity during the day (in %)
Max Humidity	Maximum humidity during the day (in %)
Min Humidity	Minimum humidity during the day (in %)
Mean Atmospheric Pressure	Average atmospheric pressure during the day (in hPa)
Mean Wind Speed	Average wind speed during the day (in m/s)
Max Wind Speed	Maximum wind speed during the day (in m/s)
Mean Wind Direction	Average wind direction (in degrees)
Mean Cloudiness	Average cloud cover of the sky (in tenths)
Max Cloudiness	Maximum cloud cover of the sky (in tenths)
Insolation	Hours in which the sun was shining (in hours)
API	Antecedent Precipitation Index
Urban Runoff	Rain-runoff conversion using the rational method
Flood	Boolean target indicating the occurrence of a flood

[1] Available at: https://portal.inmet.gov.br/

[2] Available at: https://www.samaecaxias.com.br/

[3] Available at: https://tempo.inmet.gov.br/TabelaEstacoes/A001. Accessed on September 7, 2025.

The input data also passed through standardization techniques. Those techniques change the scale of the data so every feature has the same significance. When using the default scale for each feature, features like temperature that, in the studied city, usually vary between 0 and 35 degrees, can become insignificant to a neural network that also considers atmospheric pressure, which is often near a thousand hPa [Geron, 2019].

To normalize the value of each features, this study implemented a standardization technique called Z-Score, which calculates the mean and the standard deviation of each feature, and then changes the value of each instance for the amount of standard deviations that the current instance deviates from the mean. That way, every column is in the same unit of measurement, which is standard deviations away from the mean [Geron, 2019]. Z-score is also a particularly interesting choice for this work because it highlights how uncommon that value is by identifying if it is around the mean or distant from it. This capacity to determine whether a value is common or extreme could be useful for predicting extreme events.

The data was labeled using alerts from the state's civil defense agency, using news reports and articles (such as [Marengo *et al*., 2024]) as supporting references for the dates on which the events took place. The civil defense uses the terminology "warning" for potential events and "alert" for events that are already in progress. Therefore, only alerts were considered for labeling. Those labels were later shifted forward in variable time windows, creating a lag that should allow the model to learn whether the current conditions created flood conditions in the future [Geron, 2019]. Tests were made using different numbers of time steps, from one to four days in the future, allowing an analysis of the efficiency of the prediction for all these forecast horizons.

The model used was a Multi-Layer Perceptron (MLP) neural network, trained supervised with the data discussed. Since this classification problem has only two classes, the output consists of a single neuron, which should tend to 1 when there is a flood event in the time window or 0 when there is not [Geron, 2019; Russel and Norvig, 2022]. Tree-based models were preliminarily tested as well, but the MLP showed significantly better performance over them.

The dataset was also divided into training and testing, using data from September 1st, 2023, to March 31st, 2024, for training, and data from April to May 2024 for testing. This configures a dataset of 274 instances that was split into 214 (consisting of 207 normal days and 7 flood events) for training and 60 for testing (consisting of 51 normal days and 9 flood events), resulting in a split of approximately 77.7% for training and 22.3% for testing. It's also notable that there are more instances of flood events in the testing dataset than in the training dataset, an uncommon practice, but necessary to assess if the model could predict the events in May 2024 with the data of the events in September and November 2023 [Geron, 2019].

This case study used accuracy, which is the raw percentage of correct predictions the model made, and recall, which is the percentage of positive cases that were correctly predicted [Geron, 2019]. In this context, recall means the percentage of right predictions when a flood event occurred, while accuracy means the percentage of right predictions in both situations. The goal of this study is to create a model with good accuracy, but great recall, being able to always detect flood events with as few false positives as possible.

Cross-validation was used here to find the best hyperparameters, but the results were not satisfactory. The method divides the datasets into multiple folds and iterates over them, taking some of the folders as train sets and the remaining as test sets for each iteration [Geron, 2019]. By testing the cross-validation method with mulitple hyperparameters, you can estimate what configuration averages the best results.

By default, the method rewards high accuracy. Yet, with the imbalance in the dataset, high accuracy means always predicting that there will not be floods. Recall was then tested as the scoring metric, but it led to the cross-validation method rewarding models that always predicted floods. A custom measure that rewarded high recall only when there was decent accuracy was developed, but also did not lead to great results. Under these circumstances, a trial-and-error approach was used to define the hyperparameters in this case study.

As supporting metrics, precision and F1-Score were used. Precision is the percentage of times the model was correct in predicting a flood. F1-Score is the harmonic mean of recall and precision, working as a way to aggregate the two metrics into one [Geron, 2019]. They are helpful to understand not only whether the model can correctly predict the days on which a flood event happens, but also how often it incorrectly assumes a flood occurrence.

Since floods are rare events, the labels showed a considerable imbalance between classes. Only 15% of the instances were flood events, while 85% were normal days. To overcome this barrier, the oversampling algorithm Synthetic Minority Over-sampling Technique (SMOTE) was used. SMOTE uses a k-nearest neighbors algorithm to generate synthetic data based on the distance from a minority class' sample to one random neighbor of its k nearest neighbors [Chawla *et al*., 2002].

Different versions of the base dataset were created, using different lag values to move the labels and allow predictions in the other time windows (from one to four days in the future). The days on which the label became NaN due to the shift were simply replaced with 0 (No flood), as it is known that there have been no floods in the following days.

The reason the tests must stop at four days is because the data starts on September 1st, 2023, and the first event for training occurred on September 4th, 2023. Using prediction for higher windows would conflict with the oversampling method, since there would not be enough examples of flood events for the algorithm to generate synthetic examples. The k-value could be lowered for those cases, but the quality of the oversample would be compromised. The authors believe that four days is already an interesting time range to explore. Expanding the dataset was not considered to avoid creating an even bigger imbalance between common days and flood days.

Explainability was explored for both case studies using Shapley Additive Explanations (SHAP) method to test how local changes influence the model's decision [Mosca *et al*., 2022]. That way, an interpretation of how each feature contributes to the prediction can be done, and interesting forays

into explaining flood events can emerge from that.

3.2 SAMAE case study

SAMAE meteorological data were collected in their meteorological stations, and the rainfall data were collected by three different rain gauges surrounding the Faxinal basin. The water column level in the basin's reservoir was also collected and provided by SAMAE. The authors received the data in early October, 2025. Figure 2 shows the city of Caxias do Sul and its basins, as well as the location of the rain gauges at each basin. The figure was also provided by SAMAE. The Faxinal basin is painted green in the figure.

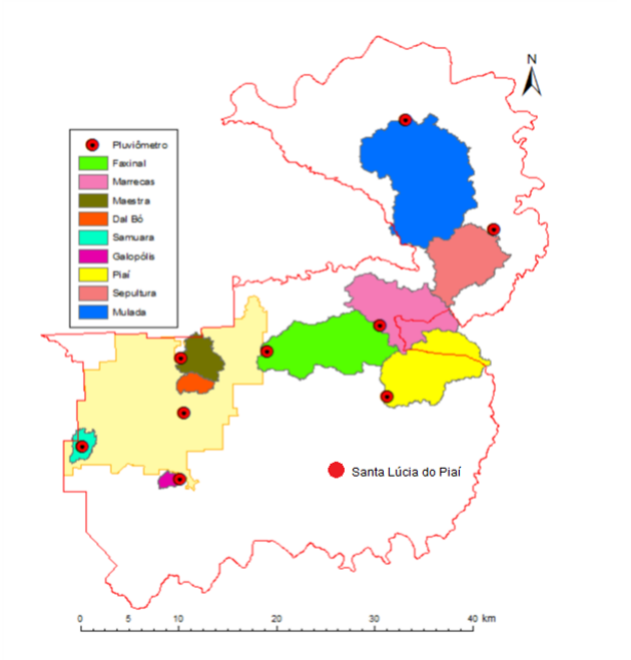


Figure 2. Caxias do Sul's basins

The meteorological and rain data serve as input for the model, while the output is the water column level in the reservoir. The model was tested first for estimation and later for prediction, first identifying if the model is capable of estimating the level of the reservoir considering the current state, and later expanding to predict the level in different time windows. The windows vary between one and three days.

Also, hydrological models were calculated and integrated as input to the model in order to achieve some level of hydrological understanding of the process. Methods used were the rational method for rainfall-runoff conversion and the API method for keeping the temporal dimension that the tested models cannot naturally store [Viessman and Lewis, 2003]. Table 2 details the features of the model.

Multiple models were preliminarily tested to evaluate their performance. The tree-based models were the ones with the highest accuracies, which converges with what the literature review showed. Tree-based models are also an interesting choice considering they are naturally explainable and dismiss the need for standardization, since the divisions those models make are scale-agnostic [Geron, 2019]. The chosen models for training were Decision Tree, Gradient-Boosted Tree, and Random Forest. The latter two were still passed through SHAP in order to understand how the different trees in the ensemble complemented each other in the decision process. The trainment was supervised with the target column discussed. A different model for each of the three types was created for each window of prediction. Since the rain data provided by SAMAE is collected daily, the prediction time slot is also daily.

Table 2. Features used in the SAMAE case study

Feature Name	Description
Date	The day on which the data was collected (YYYY-MM-DD)
Wind Speed	Wind speed at the time of measurement (in m/s)
Wind Direction	Wind direction at the time of measurement (in degrees from North)
Temperature	Air temperature at the time of measurement (in Celsius degrees)
Humidity	Relative humidity at the time of measurement (in %)
Pressure	Atmospheric pressure at the time of measurement (in hPa)
Solar Radiation	Incident solar radiation (in W/m^2)
Mean Wind Speed	Average wind speed during the day (in m/s)
Mean Wind Direction	Average wind direction during the day (in degrees)
Dew Point Temperature	Dew point temperature (in Celsius degrees)
Wind Gust Speed	Maximum wind gust speed (in m/s)
Wind Gust Direction	Direction of maximum wind gust (in degrees)
Daily Rain	Total precipitation during the day (in mm)
API	Antecedent Precipitation Index
Urban Runoff	Estimated urban runoff using the rational method
Reservoir Water Column Level	Water level of the reservoir (difference in meters from the normal operation level)

Then, the dataset was split into training and testing sets. The SAMAE dataset starts on December 9th, 2024, and goes until June 30th, 2025, configuring a dataset of 204 instances. Unlike the dataset described in subsection 3.1, this dataset

does not contain data collected during any of the three major flood events that occurred in 2023 and 2024. A less impactful but still severe event took place in mid-June 2025, but this is the only major event present in the dataset [Tomé, 2025a; GZH, 2025b]. Other more localized events took place during late-December 2024, early-January 2025 and mid-February 2025, none of which were enough to raise the water column in a significant way [GZH, 2025a, 2024; Tomé, 2025b]. Therefore, most of the outliers in this case are associated with moderately heavy rainfall, and the peaks in the water column level are not as high as they could be.

Because of that, this case does not allow the temporal division made in subsection 3.1, and the division between training and testing datasets was decided randomly, but stratified. Stratifying the splitting means that, if 25% of the data is used for testing, then each class has 75% of its instances on training and 25% on testing. This avoids a distribution where all instances of a class are in the training set and none in the test set, as well as the opposite [Singh and Mangat, 1996].

The datasets were stratified using an outlier metric. Since the goal of the model is to predict the water column height during extreme events, it's fundamental that both the training and testing datasets contain the most extreme examples in the dataset. The threshold to determine what is an outlier is 90%, which represents 20 instances.

The split was made using 80% of the normal instances for training and 20% for testing, while 70% of the outlier instances were used for training and 30% for testing. Those ratios were thought to keep as much instances for training as possible (since there's very few examples) while still retaining a considerable amount of data in the test. Other ratios were also tested, with this configuration yielding the best results. Therefore, 161 instances were used for training (147 from normal cases and 14 from outliers), and 43 instances were used for testing (37 from normal cases and 6 from outliers).

The dataset was then shifted backwards, so each instance contained the water level N days in the future, being N the lag value, between 1 and 3 days. This also assings a null value to the target column in the last N instances of the dataset. Different from subsection 3.1, it is not possible to deduce the value in the column, and therefore, those last N instances were removed from each variation of the dataset.

In most prediction systems, outliers tend to be removed for better accuracy. However, in this case, the outliers represent the flooding events. Therefore, predicting the outliers is the main goal of the model. To improve the performance of outlier prediction, oversampling techniques were also tested, but ultimately failed to enhance the model.

After running the model in the test dataset, metrics are used for assessing if the model can correctly predict the data. This study used R^2, RMSE, MAE, and R^2 for the outliers. The latter is important since the main idea is to predict flooding correctly. Therefore, the days in which flood occurs are the ones that configure an outlier, and having good predictions for the usual case but bad predictions for the flood events would give a false sense of a good performance.

Training and testing the model fit, respectively, the model and assess phases of SEMMA methodology. Since the methodology is iterative, if the assessment concludes that the model is unable to estimate the water column height correctly, the process can be restarted at any point, allowing different approaches to be tested to achieve sufficient accuracy [Sharda *et al*., 2019].

4 Results and Discussion

In order to discuss the results, this section will be divided into subsections that reconstruct the timeline of each case test. The first tests were focused in flood detection, which means being able to detect if given conditions characterize a flood event or not. Then, different windows of prediction horizons were developed. Finally, explainability was explored.

This section is divided into three subsections. The first subsection talks about the naive approaches took to develop baseline models used as comparison metrics with the actual AI models. The other two subsections discuss the results obtained for the models developed in each case study.

4.1 Naive approaches

In order to measure the quality of the predictive models, a baseline model was created for comparison. Each case study presents a distinct type of problem. While INMET case study is a classification problem, SAMAE case study is a regression problem. Therefore, the methodology used to create the baseline models varies in each case.

4.1.1 INMET case study

INMET case study used the 80% quantile threshold for rainfall and API variables. The API-based model performed better and was kept as the baseline model for this case. The baseline model had 88% of accuracy over the test dataset, but only was able to predict correctly 7 out of the 9 days in which flood happened, and wrongly assumed flood in 5 days. If the threshold was strict (90% quantile), the accuracy rose to 92% and only 1 normal day was confounded as a flood day. However, it was only able to predict 5 of the 9 days. It was only possible to predict all the 9 days by reducing the quantile threshold to 40%, which reduced the accuracy to 55%. The 80% quantile was chosen then as the best balance between recall and accuracy for the baseline model.

The detection baseline model had accuracy of 88%, with a recall of 77.78% and precision of 58.33%, characterizing a F1-Score of 0.67. The confusion matrix can be seen at Figure 3.

The one-day prediction model, however, showed a decline in every metric. The accuracy dropped to 85%, recall decreased to 67.67%, and precision is now 50.0%, characterizing a F1-Score of 0.57. The confusion matrix is shown in Figure 4a.

Figure 4b shows the two-day prediction model, that continued the trend of decline in the metrics. The accuracy dropped another 3%, going to 82%, recall decreased to 55.55% and precision to 41.67%, characterizing a F1-Score of 0.48.

The three-day model had the biggest drop in prediction quality yet, as found in Figure 4c. The accuracy went to 75%,

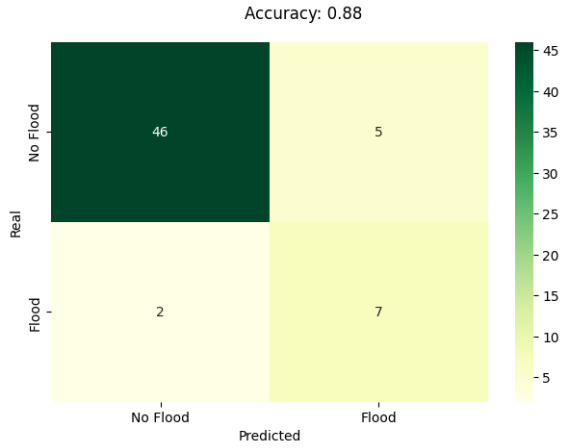
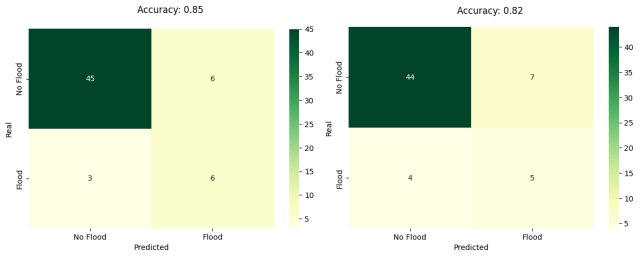


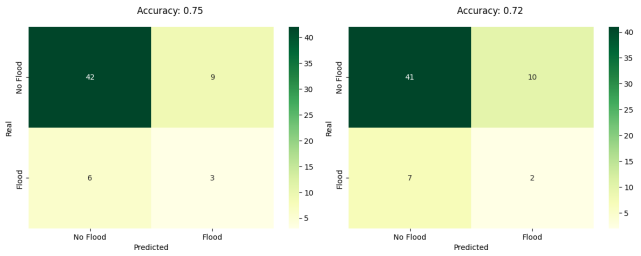
Figure 3. Baseline model for detection

recall decreased to 33.33%, and precision bottomed at 25.0%, characterizing a F1-Score of 0.29.

At last, the four day model, found in Figure 4d, shows an accuracy of 72% a recall of 22.22% and precision of 16.67%, characterizing a F1-Score of 0.19. This is the culmination of the model's decline in prediction capacity, demonstrating that using quantiles can be effective for short-term predictions but is insufficient for longer-term predictions.



(a) Baseline model for one-day prediction **(b)** Baseline model for two-day prediction

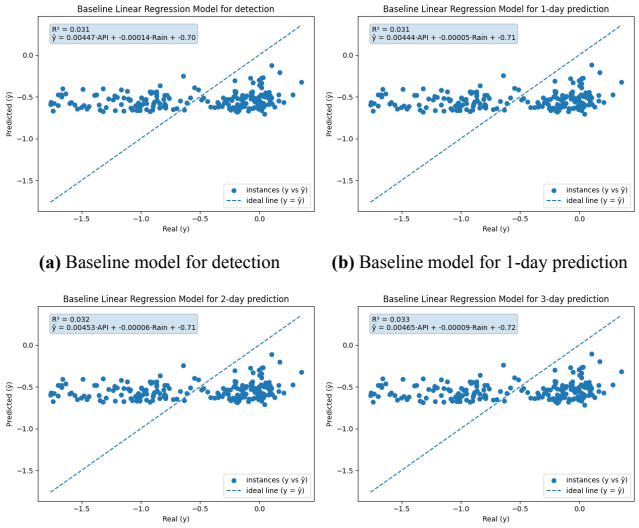


(c) Baseline model for three-day prediction **(d)** Baseline model for four-day prediction

Figure 4. Baseline prediction models for INMET case study

4.1.2 SAMAE case study

For a regression problem, however, simply using quantiles is not coherent. Because of this, the SAMAE case study's baseline model was a simple Linear Regression model using only rainfall and API as variables. This is a very simple model used only to compare with the decision tree models. Figure 5 shows that the average R^2 of the baseline models is very close to 0, which is very low, and means this model is barely any better than predicting by using the mean value for the water level.



(a) Baseline model for detection **(b)** Baseline model for 1-day prediction

(c) Baseline model for 2-day prediction **(d)** Baseline model for 3-day prediction

Figure 5. Baseline models for SAMAE case study

4.2 INMET case study

The first few tests showed several issues, reaching only around 60% of accuracy at best. Some models achieved much better accuracy, achieving even 93.75% when guessing that there would never have any flood. This revealed a significant imbalance in the dataset, as the great majority of it consisted of days on which floods did not occur.

In that perspective, this study's main obstacle was the class imbalance inherent to the problem of flood prediction. By nature, extreme events are rare, and therefore, the conditions that define flood events can be hard for an AI to comprehend, since there are very few examples. To assess this, oversampling algorithms (such as SMOTE) and undersampling algorithms (such as Near Miss) were tested to facilitate the model's learning.

Tests were made using both sampling approaches, with different degrees of success. Undersampling proved great for optimizing recall and minimizing false negatives. However, it also tended to make much more false positives, which could lead to numerous alarmist predictions that would discredit the model. Since there are very few instances, undersampling tests resulted in high accuracy, but when testing the model with the full dataset, which includes data that was cut because of the Near Miss algorithm, it proved itself to be less generic than it seemed. As such, undersampling was discarded, since it reduced a dataset that was already small and did not show considerable improvements in the performance.

On the other hand, the oversampling algorithm SMOTE obtained good results, having consistently great average recall values and being able to reach a high percentage of accuracy in recognizing a flood state. That serves as evidence to indicate that, with sufficient data on similar events, the model would be able to make accurate predictions. However, oversampling-trained models are not the most reliable, since most of the data is synthetic and might inflate the accuracy by creating too many similar cases to the ones passed as input.

To prove this inflation caused by the method, an example was made using the oversampling method with the full dataset, instead of the correct practice of using only the train-

ing dataset. When using test dataset instances in the SMOTE algorithm, new instances are created based on them, and the model is exposed to data derived from the test data [Chawla *et al*., 2002]. That way, the model is indirectly being trained with test data, making it impossible to test the model's generalization capacity. When using this approach, the model was capable of recognizing flood events with 100% accuracy, which is much more than it was capable of when trained correctly.

When isolated from the oversampling method, the test data proved itself much more difficult to predict. This approach is more correct because when the model starts being used in everyday scenarios, it'll come in contact with data that it has never seen before. To consider the model 100% capable of predicting flood events would be dangerous, because it's impossible to know how the model would react to data it has never touched, considering synthetic data based on the test data was used in its training. Even so, oversampling proved itself a promising method.

A simpler yet still efficient method to deal with imbalanced data is simply changing the threshold to compensate for the non-dominant class, in this case, the positive one. For example, in this network with a single output neuron, the default threshold considers all predictions equal or greater to 0.5 to be the positive class, and predictions below that threshold are considered negative. By changing the threshold to 0.3, for example, it's possible to make it easier for the model to predict positive cases, compensating a little the imbalance that makes the output tend to 0 [Geron, 2019].

This approach was implemented and tested, with mixed results. Without the need to generate artificial data, the model achieved very high accuracies but tended to have low recall and inconsistent predictions of flooding. That means that, even with a 0.3 threshold, the model was still not very efficient in recognizing flood events with so little examples.

With those initial tests, it was possible to establish which methods were good enough for implementation. Both oversampling and threshold-reducing methods were good enough to be tested with the periodic division, in which the training dataset consisted of data from September 2023 to March 2024, and the testing dataset consisted of data from April and May 2024.

The results indicated that the oversampling method was much more efficient for recognizing a flood state, consistently reaching high recall values and being able to keep decent accuracy. The best selected model was able to reach 83.3% of accuracy with 100% of recall. In contrast, the threshold-reduced model had a lot of trouble predicting the flood events in the test dataset, and when able to detect them well, a lot of false positives came in the process. Because of that, the selected best model for flood detection using threshold reduction had 65% of accuracy and 77.8% of recall. Figure 6 and Figure 7 show the confusion matrixes for the oversampling and threshold reduced models, respectively.

Considering the challenge of using threshold reduction to get good recall while retaining good accuracy, the oversampling model was the only one retained for testing the different prediction windows. The artificial data generated by the oversampling method is still not ideal, but using oversampling only with the training data at least retains the integrity

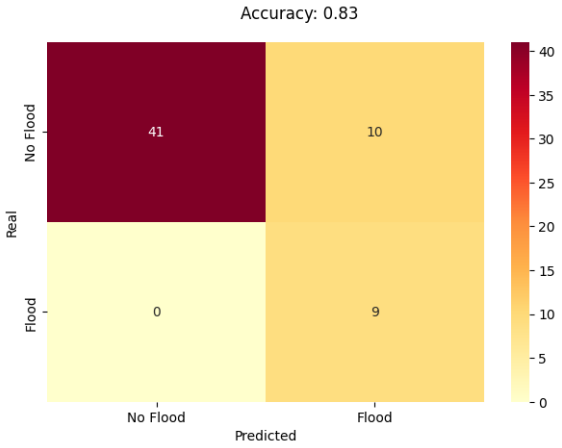


Figure 6. In-day flood detection model using oversampling

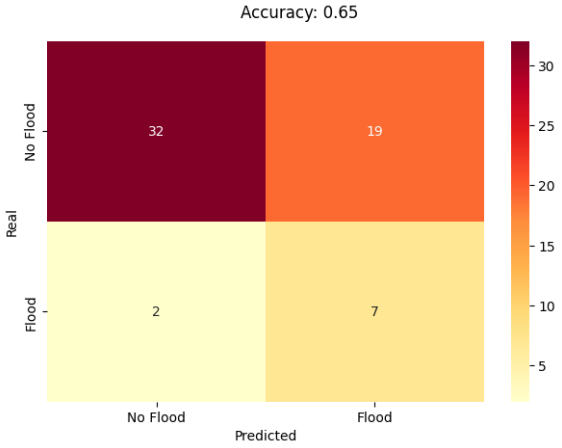


Figure 7. In-day flood detection model using threshold reduction

of the results. The scarcity of data on flood events (and the scarcity of flood events themselves) forces this limitation on the training.

Comparing the oversampling model with the baseline model for detection, the first thing that catches the attention is that the accuracy is lower. The baseline model had 88.3% of accuracy, against the 83.3% of the oversampling method. However, the accuracy can lead to a wrong interpretation. The oversampling model performed much better at predicting floods, being able to have 100% recall, against the 77.8% of the baseline model. The accuracy seemed better because there are more examples of the negative class than of the positive class, which inflate the numbers for the baseline model and punish the oversampling model.

When comparing both models' F1-Score, it is notable how similar they are in predictions, with the baseline model reaching around 66.5% and the oversampling method reaching 64.3%. The oversampling model is still considered an improvement for being able to keep good accuracy while improving in the prediction of flood events, which is the main goal of the study.

Finally, the labels were shifted to one day prior, and now the dataset does not indicate if there was a flood on that day, but if there was a flood on the day after. Those tests initiate the actual prediction, rather than the recognition discussed so far.

Curiously, when shifting the prediction window to 1 day, the model became much more capable of detecting flood

events. This corroborates that floods are not necessarily a direct cause of what happened on the current day, but are more influenced by what happened the day before.

Also, it's important to notice that the main goal of this case study, which was to discover whether it was possible to predict the events that happened in May 2024 using data from September and November 2023, is validated. All nine days in which the city was affected by flood events were correctly predicted by the model, achieving 100% of recall. Additionally, it was possible to avoid excessive false alarms, with only four incorrect guesses, which configures 69.2% of precision and totals 93.3% of accuracy. Figure 8 shows the confusion matrix for this test.

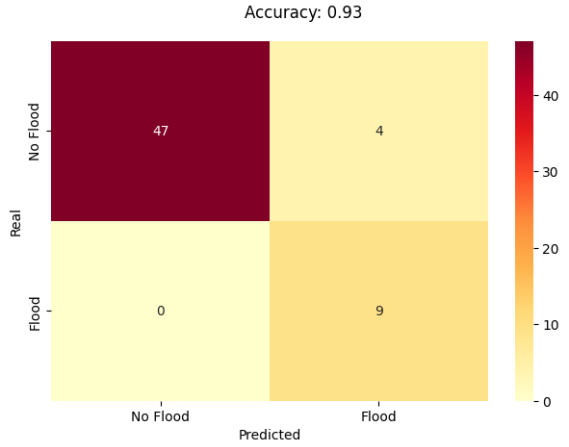


Figure 8. One-day flood prediction model

The one-day prediction model shows how the AI is important for prediction when compared to the baseline model. While the simpler quantile methods were able to achieve 85% of accuracy and detect two-thirds of the flood events, the neural network was able to reach 93.3% of accuracy and detect every single event. While the baseline models worsened as the prediction window increased, the AI model for the one-day lag was able to predict even better than its detection counterpart. Usually when the recall improves, the precision tends to diminish. Contrary to expectations, the neural network model has 69.2% precision, against the 50% precision of the baseline model.

When changing the prediction window to two days, the results become a bit less consistent. It was still viable to make the model predict the 9 flood events, but the more days it correctly recognized as flood-affected, the more days it incorrectly recognized as flood-affected as well. The best model for two days had a drop to 90% accuracy, as shown in Figure 9. The model was able to retain a good 60% precision as well.

While the baseline model showed a more obvious decrease in the quality of the predictions as the prediction windows grew, the neural network models seem to be more resistant to the passage of time. The two-day model was still able to detect every flood event, while maintaning 90.0% of accuracy and 60.0% of precision. The baseline model, however, had only 81.6% of accuracy, 55.5% of recall, and 41.7% of precision. The baseline model is not impressively good at predicting the flood events, and yet, it also predicts a lot of false positives, creating false alarms and not even guaranteeing

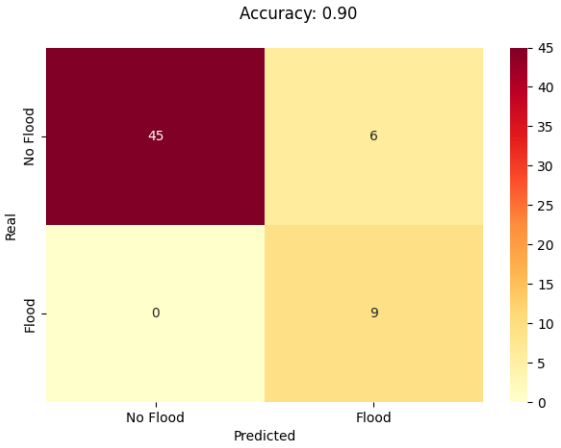


Figure 9. Two-day flood prediction model

that every event was between those detections. This shows the considerable improvement caused by the neural network.

The three-day model significantly lowered the accuracy, showing a lot of difficulty to predict all nine occurrences. That was possible to do at the cost of a great portion of the model's capacity of recognizing days without flood, indicating an overfit in the model. The best version of the model capable of recognizing all nine days had only 36.6% of accuracy, as shown in Figure 10.

However, it was possible to balance a decent accuracy with decent recall and precision for the positive class, having a model that is not perfect at predicting floods, but is coherent enough to avoid excessive alarms. It was possible to achieve 77.8% of recall with 76.6% of accuracy. The confusion matrix is shown in Figure 11.

One limitation of the three-day model, however, is that it predicts more false positives than true positives, thus having a low precision. This means that the model wrongly assumes there will be a flood with high frequency. It removes some of the reliability in the model, since only 36.8% of the times it predicts a flood are correct. This is still fairly decent for such a difficult problem and such a huge lag in the prediction, but it is debatably more reliable than an observer's verification.

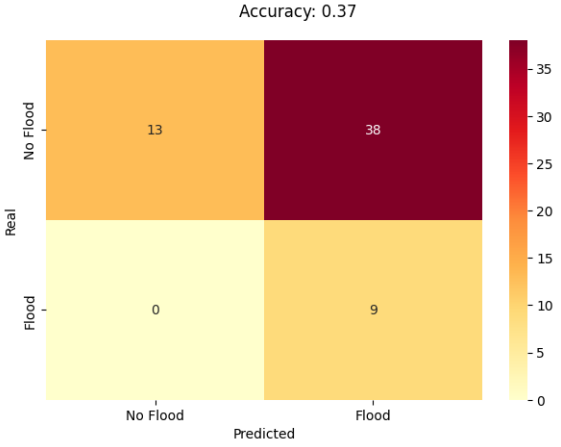


Figure 10. Three-day flood prediction model predicting all days

Even with the abrupt fall in the prediction quality, the three-day prediction model was able to maintain a decent recall of 77.8%, even if at the cost of a good precision. This model shows that three days is far too big of a horizon for this

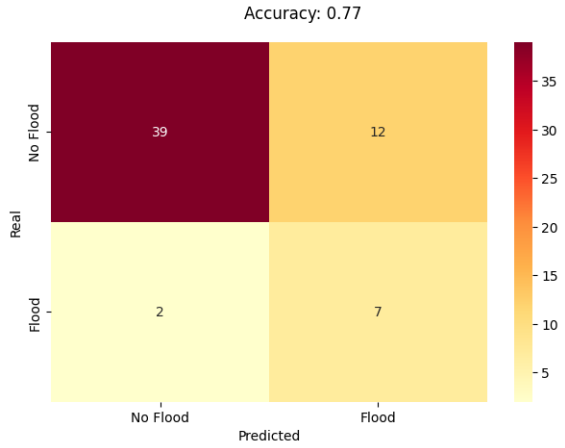


Figure 11. Three-day flood prediction model predicting some days

flood prediction model. Still, when compared to the baseline model, the prediction of the flood events is much more consistent. The baseline model had only 33.3% of recall and 25.0% of precision on predicting floods, being very close to simply guessing. In that sense, even if the neural network was unable to make certain predictions, it offers better consistency compared to simpler models, such as the baseline approach proposed.

At last, there's the four-day window. It showed more inconsistency between trainings and had more difficulty reaching its best version than the previous models, but showed very curious results. The model performed impressively well, with 88.9% of recall, 83.3% of accuracy, and 47.0% of precision. All this information is shown in the confusion matrix at Figure 12. It performed better than three-day model, even being more distant to the event. Discussions on why are explored after the explainability model SHAP was applied. When compared to the baseline model, which had 22.2% of recall and 16.7% of precision, it performed impressively well.

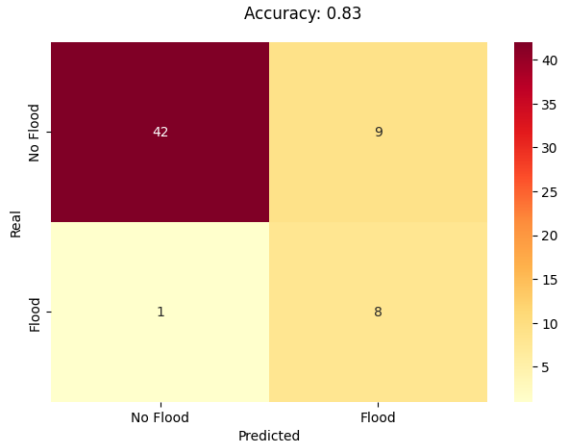


Figure 12. Four-day flood prediction model predicting some days

The results of each model are summarized in Table 3, highlighting the accuracy, the recall, and the precision of the predictions. The best model is bolded, highlighting the model's best performance.

Comparing the models side to side, it's easy to see that, while the accuracy and the recall remain relatively high in every model, the precision rarely reaches a truly great number, and only the one-day and two-day prediction models were able to overcome the 50% barrier.

Table 3. Metrics of the models

Prediction Window	Accuracy	Recall	Precision	F1-Score
0 days ahead	83.3%	100%	47.4%	64.3%
1 day ahead	**93.6%**	**100%**	**69.2%**	**81.8%**
2 days ahead	90.0%	100%	60.0%	75.0%
3 days ahead	76.6%	77.8%	36.8%	50.0%
4 days ahead	83.3%	88.9%	47.0%	61.5%

All these results were obtained for models trained with a simple architecture consisting of an input layer of 17 neurons (since the Date and Flood features were excluded from input), a single hidden layer of 8 neurons, 1 output neuron, Rectified Linear Unit (ReLu) activation function for input and hidden layers and logistic activation function for the output layer, using Adam optimizer and a constant learning rate of 0.001 (1×10^{-3}), as well as early stopping with tolerance varying between 0.0001 (1×10^{-5}) and 0.000001 (1×10^{-7}) to avoid overfitting. It's intricate to attest that the more complexity that was added to the architecture of the model, the worse it seemed to perform.

Few studies were encountered with similar classification systems as the one used in this case study in order to make comparisons with related works. Still, in Furquim *et al.* [2018], a very similar approach obtained 80.0% of accuracy and 90.0% of recall when predicting red alerts. Hence, the 93.3% of accuracy and 100.0% of recall that the one-day prediction model achieved defines excellent performance.

4.2.1 SHAP analysis

The SHAP method was used to explore the explainability of the model. SHAP uses the Shapley value, originally a concept of game theory, to estimate the contribution of each feature, considering even the interaction between features through different permutations [Mosca *et al.*, 2022]. Using this method, it is possible to identify which features are the most relevant to the model predictions, as well as defining which features better determine the behavior of each instance.

Mean SHAP values represent the absolute mean variation caused by a feature in the output. That means that this metric represents, on average, how much the output varies because of that feature [Mosca *et al.*, 2022]. This way, features with higher SHAP values are more significant to the model, and features with lower values are less significant. Optionally, the relative importance of each feature can be expressed as a percentage by normalizing its mean absolute SHAP value over the total sum of all features' mean absolute values, determining what percentage of the model's decision is defined by that feature.

The analysis of the mean absolute SHAP values presented in Table C.1 is a key aspect of this study, as it intends to integrate hydrological models with AI, and learning how the AI is making the predictions is of extreme importance to understand how the model's findings relate to the physical and hydrological knowledge that exist.

For the detection model, 15.25% of its decision is defined by API, which shows the strong relevance of the hydrological modeling to flood detection, and validates the hypothesis that rain and other features based on it could improve the prediction performance. Even being a very simple model, API can input the memory of the past day's rain into the model, providing a context otherwise impossible for an MLP network. It is also 66.66% more relevant than the second most important feature, which is mean pressure, showing how powerful this variable is for identifying flood.

Other features, like wind, temperature, cloudiness, and of course, rain, were also important, showing that the conditions of the atmosphere are really important for flood detection. However, the rain alone is not capable of identifying flood dates, and is not even remotely the most important feature for this prediction. This makes sense, since the basin response time makes rain cause late damage, and the city's runoff system can handle heavy rain until a certain point, failing when saturated by consistent rain.

The summary plots of the SHAP values for every model can be found at Appendix B. Those are particularly great ways to understand the behavior of the model, as it scatters every test instance and compares, for each local analysis, the SHAP value of each feature with the feature value of the instance. With that, it is possible to comprehend if high values mean positive or negative influence for the model's decision of a flood occurring. It is also possible to see the SHAP values for each feature across the models in Appendix C. Those resources are provided in the appendix for better visualization.

The summary of the detection model shows a very linear behavior for most of the features, with most variables showing that higher values represent a higher chance of flood and lower values represent lower chances, and some others have the inverse. Maximum wind speed and pressure, as well as mean humidity, are the only variables that show a slight overlap between behaviors, but they are still well separated.

The linear behavior is seen in API, wind direction, mean temperature, maximum cloudiness, mean wind speed, rain, rainfall-runoff, maximum temperature, insolation, mean cloudiness, maximum pressure, and mean humidity. The inverse behavior, on the other hand, appears in mean pressure, thermal amplitude, minimum temperature, minimum humidity, and maximum wind speed.

Cloudiness, pressure, humidity, and rain play the most important roles for the one-day prediction. This means that the current rain added by the rain forecast the model is making represents flood in the near future better than the accumulated rain. The difference between the most relevant features shows that the model is really predicting a flood event, instead of recognizing it like the previous model did.

It's also curious to see how pressure and humidity are so important and so closely significant. That aligns really well with the Crossing Humidity, Dewpoint, and Atmospheric Pressure (CHRUDA) model's discovery of how pressure, humidity, and dewpoint correlate to predicting rain around nine hours in the future. The authors believe that the model recognizes that those features are the most significant to predicting flood because they are reliable variables to predict rain, which leads to flood [Gutierrez-Lopez *et al*., 2019].

The two-day prediction also does not take API much in consideration, but is significantly different from the one-day model. The wind direction variable is curiously the most relevant to the model's decision, representing around 13.6% of the variance in the decision. Cloudiness, temperature, and rain follow it, and isolation, humidity, thermal amplitude, and wind speed are similarly relevant. Those variables indicate that the model is still analyzing the imminence of rain in order to predict floods.

Also, the wind direction variable being the most relevant could mean two things: a significant discovery about the behavior of flood events in the city, or a signal of overfitting. The authors of this study tend to believe in the second. If every flood event is marked by wind coming from a specific area in the world, such as the Atlantic Ocean, that could be a good lead to focus monitoring in those areas. However, if it is merely a coincidence that most flood events in this dataset originate from rain in a specific area, that could simply represent the model's overfitting. It's also curious how the models interchangeably consider it relevant or not, which could mean that it is possible to predict without this information. If so, that would be another signal of overfitting.

Exploring the API values in those two models, it is possible to see how high values of API are spread around leaning the model towards predicting flood and not flood. That means the API does not have a linear influence, as it did in the detection model - higher values can contribute negatively to the prediction, depending on other variables. One possible conclusion is that the model believes that, when there's no imminent rain and the API is really high, a flood event just took place, and the chance of another happening is lower. If the API was lower, than even with no immediate rain, there's a greater chance of a flood event taking place because there has not been any flood in the immediate past days.

The three-day model also uses a similar prediction core to the one-day model, giving high significance to pressure, humidity, temperature, and rain. However, API once again rises as the most important feature by a large margin, representing 13.56% of the behavior and having the most linear distribution that the detection model had, meaning high API values could lead to a flood. This suggests that, while the imminence of heavy rain is an important factor for predicting floods in several days in the future, more important than that is how saturated the soil already is. This is consistent with the logic of a basin's response time, as well as the logic of the city's response to flooding [Tucci, 2004]. Heavy rain might harm the city if it happens suddenly, but it will be dealt with as fast as it starts. However, if the soil and the city's runoff system are saturated, any rain could turn into superficial runoff and, therefore, be classified as flooding. So, for long horizons, API was the most reliable of the variables for flood prediction.

The four-day model works quite differently than the other models. At first sight, it might appear to be a very similar case to the three-day and detection models in the sense that API is the most relevant feature. It truly represents 29.63% of the model's decision, which is the most representative feature in those models. However, the linear behavior of API attested in the other models is not reproduced here. This model's API analysis consider it of inverse proportion to the

event of a flood, meaning that high API values mean little to no chance of having a flood four days in the future.

This indicates two very interesting findings. The first is that the model is most likely clustering events, and realizing that, if there's a high API value today, it is very likely that a flood is happening, and therefore, there is little to no chance that this flood will still be taking place four days in the future, as both the basin and the city were able to deal with the water by then in the examples the model was given.

This finding also implicates that a four days lag could be the limit value for this experiment. If the model is recognizing a flood in order to say that there will not be a flood, there's a very likely chance that the right predictions of the model are arbitrary coincidences or derived from overfitting, and do not really represent a good understanding of the problem. That could explain why it got better results than the three-day model.

However, it could also mean that the estimatives of the flood happening are all based on the rain prediction variables, which are also important to the model, and a high API only serves to help place the event within a temporal context of a flood event. This would mean that API helps the model avoid the prediction of a flood that lasts longer than the expected. Since rain, wind, humidity, temperature, pressure, and cloudiness variables all have a very linear behavior, the rain prediction aspect is evidently strong in the model, and more in-depth studies would have to be conducted in order to determine which of the hypothesis is the most probable.

This estimative is corroborated by Lorenz's claims that, above one or two days in the future, weather prediction becomes speculative [Lorenz, 1963]. That does not mean the prediction has no value, but rather that it can change significantly and in unpredictable ways, so considering predictions in this time window as certain is not ideal.

Overall, the SHAP analysis demonstrates that the models behave coherently with hydrological reasoning, but there are certain signals of overfitting in higher prediction windows. For detection, it emphasizes hydrological memory through API. For short-term prediction, it focuses on the signs of future precipitation on meteorological data. For longer horizons, the predictability decreases and the model relies heavily in the current occurrence of flood to determine the chance of future flood. The varying importance of API across windows reveals that it is a powerful predictor, but limited for long-term prediction, with meteorological factors being more relevant in those contexts.

4.3 SAMAE case study

SAMAE provided several datasets containing fragmented data, which, after an exploration process, were selected for this study and were integrated into a single dataset, described in Table 2.

Initial tests were made using this dataset. Those initial tests resulted in a R^2 of 0.61 for the detection model. To improve the model's context for predictions, data from previous days was incorporated, significantly enhancing it. API is an important factor for obtaining the history of the rainfall in the region, but for a regression model, this knowledge will not always be enough to estimate the water column level, and the level in the previous days is a better indicator.

While the R^2 demonstrated a lot of improvement, the model was still finding difficulty predicting outliers. Multiple different oversampling approaches were used to try to improve the model's capacity of predicting the water column level when a flood event occurs. First, SMOTE for Regression (SMOTER) was tested. However, it proved itself inefficient, as the SAMAE dataset had very little data for the k-nearest neighbors algorithm that SMOTER is dependent on. Next, some more naive approaches were tested, such as duplicating the outlier instances and duplicating them with small, random noises. Both approaches decreased the outliers' R^2 considerably and were discarded. The oversampling techniques were then dropped, and this limitation was accepted into the model.

The detection model was the first tested and showed the best results. The model that performed best with this dataset was the Decision Tree, achieving an R^2 of 0.98, MAE of 0.04, and RMSE of 0.06. The outliers' R^2 value was 0.78, which is a decently high value considering the small number of examples, but not as reliable as the models in the previous case study.

Next came the Gradient Boosting model, with an even higher R^2 of 0.99, MAE of 0.03, and RMSE of 0.04. Even if the general performance of the model was better, it was considerably worse in predicting the outliers, with an outliers' R^2 of 0.62. At last, the random forest model, with R^2, MAE, RMSE, and outliers' R^2 of 0.98, 0.03, 0.06, and 0.30, respectively, showing a consistent performance for the normal cases, but very low precision for the outliers. A summary of the results for the detection model can be found at Table 4.

Table 4. Summary of the metrics for the detection model

Model	R^2	MAE	RMSE	Outliers' R^2
Decision Tree	0.984	0.044	0.068	0.779
Gradient Boosting	0.992	0.032	0.046	0.618
Random Forest	0.986	0.037	0.063	0.299

The prediction models, however, did not share the capacity of decently predicting the Faxinal basin's reservoir water level during flood events. The one-day prediction, for example, the best model was the Gradient Boosting, which achieved a very strong overall R^2 of 0.96, as well as a MAE of 0.069, RMSE of 0.11, but an outliers' R^2 of only 0.59. This shows a vast decrease in the capacity of identifying the water level in flood events. It proves that the model is very good for prediction in normal days, but in days of extreme rainfall, it will not be as reliable. The other models performed even worse for the outliers, even if they were able to maintain the overall metrics in a good state, as seen in Table 5. The Decision Tree model is particularly interesting for having a negative R^2 for the outliers, which means the model is less precise than a prediction made only using the mean water level [Gao, 2023]. That means that the Decision Tree model, even with great overall scores, is terrible for predicting flood.

The tendency started by the Decision Tree model in the 1-day prediction spread to all models in the 2-day prediction models, as shown in Table 6. The best R^2 for outliers in this

Table 5. Summary of the metrics for the one-day prediction model

Model	R^2	MAE	RMSE	Outliers' R^2
Decision Tree	0.976	0.063	0.091	-0.600
Gradient Boosting	0.964	0.069	0.111	0.588
Random Forest	0.989	0.041	0.061	0.216

model was ironically in the Decision Tree model, which was -0.30. That shows that these models have absolutely no value for predicting flood in a horizon longer than one day in the future.

Table 6. Summary of the metrics for the two-day prediction model

Model	R^2	MAE	RMSE	Outliers' R^2
Decision Tree	0.971	0.075	0.099	-0.304
Gradient Boosting	0.940	0.098	0.142	-0.942
Random Forest	0.977	0.061	0.087	-1.107

This bad prediction remained in every other window, only worsening with the increase of the prediction window. The 3-day model was decided to be the last presented, in order to demonstrate not only the persistence of terrible prediction for flood events, but also the decline in the overall capacity of prediction even for the common days. The results obtained by this model are in Table 7.

Table 7. Summary of the metrics for the three-day prediction model

Model	R^2	MAE	RMSE	Outliers' R^2
Decision Tree	0.888	0.102	0.183	-0.76
Gradient Boosting	0.922	0.097	0.153	-0.592
Random Forest	0.948	0.080	0.124	-0.617

When comparing these tree-based models with the baseline Linear Regression model, it is very clear how much better those models performed. The worst R^2 found in the tree-based models was the 3-day prediction Decision Tree, with 0.89, which is staggeringly higher than the average of 0.03 the baseline models had. Also, even if the models with higher prediction windows were not able to make correct estimations for the outliers, the baseline models were actually much worse. The same day detection model had an outliers' R^2 of -60.34, while the prediction models' number were -60.54, -60.81, -61.10, for the one-day, two-day, and three-day prediction models, respectively. This indicates that the sum of the quadratic errors in the baseline models were around 60 times bigger than the sum of the quadratic deviation from the mean value of the outliers.

In perspective with the results obtained on the related works, the results were also very good. The best prediction model obtained R^2 of 0.964, which is competitive to the results obtained by Tabbussum and Dar [2021], Pereira Filho and dos Santos [2006], Dawson *et al*. [2002], and Kourgialas and Karatzas [2017], that ranged from 0.95 to 0.98. MAE and RMSE are harder to compare since they are sensitive to the unit of measurement of the output. Even so, for the same one-day prediction model MAE was 0.069 and RMSE was 0.111. The first metric ranged from an impressive 0.00034 (Tabbussum and Dar [2021]) to 0.12 (Noymanee *et al*. [2017]) and 0.27 (Saravi *et al*. [2019]). The second ranged from 0.018 to 0.16 in the same studies.

The results obtained, even if not fantastic, show progress in the estimation of the water column level in Faxinal's reservoir. When compared to simply using the average, the one-day prediction's error is 59% smaller. When compared to a simple linear regression, it is multiple times better. While there are still barriers to overcome, the results are already promising.

4.3.1 SHAP analysis

To try to understand the decision-making process of the models, explainability was explored using SHAP. For the Decision Tree models, it is possible to use only the tree the model uses for decision-making to explain it. However, Random Forest and Gradient Boosting models use several trees that complement each other, and therefore, only analyzing a single tree does not allow for a full comprehension of the model's behavior [Geron, 2019]. For those models, SHAP was used for explainability.

The Decision Tree was the best model for detection. As shown in the first divisions of the tree, at Figure 13, almost every single division is made based on the water column level of the day before. There's only two exceptions, the mean of the water column level in the last three days, and the humidity, both only appearing in the fourth level of the tree. This shows how the model heavily relies on the previous state, instead of the current atmosphere. Such behavior could mean that models that work with temporal storage, like s or Long Short-Term Memory (LSTM)s, could be more appropriate for this case, as they naturally handle sequential data [Geron, 2019].

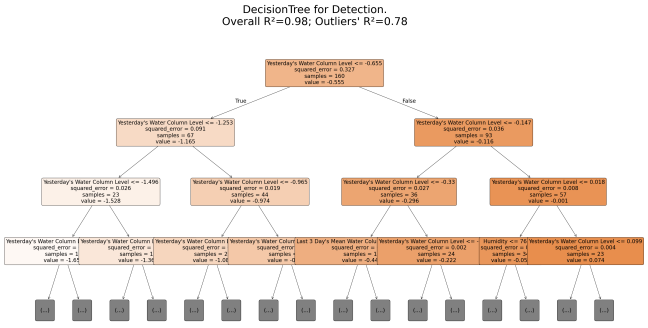


Figure 13. Best model for detection explained

The one-day prediction problem was better solved by the Gradient Boosting model. In this case, SHAP was used for explanation, since an ensemble model's behavior cannot be explained by a single tree. As shown in Figure 14, the water column level on the day of the prediction is the most important factor, just like the level on the day before was in the detection model. It is notable how the higher values are important for increasing the water level predicted, but the lower values tend to have an even more significant impact by decreasing the level predicted. The interpretation is that if the level is high today, it has a higher chance of being high tomorrow, but if it is low, it's significantly more likely that it will be lower.

It is also evident how several other variables have little to no impact on the prediction, with the difference between each instance being indistinguishable for several features. Rain and urban runoff, though less impactful than other features,

also make an important distinction between normal values and outliers, demonstrating its importance. Other variables, such as humidity, are clearly distinguished into two levels of contribution, which could imply that there's a single instance of a decision node including that feature.

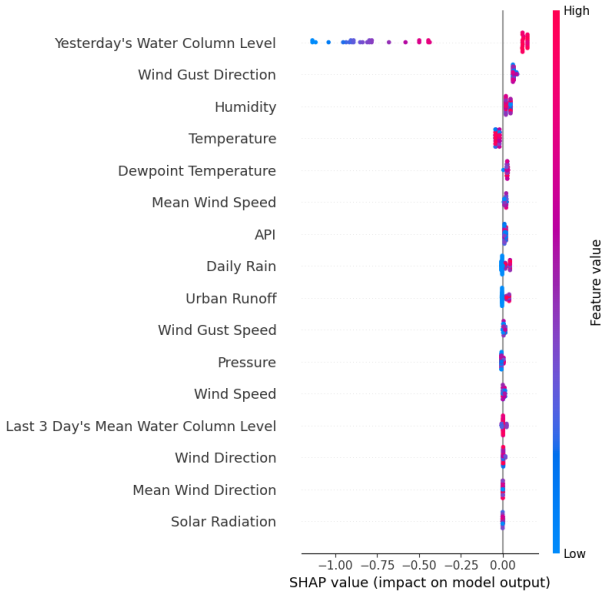


Figure 14. Best model for one-day prediction explained

The two-day prediction problem was better represented by the Decision Tree model at Figure 15. It's essential to note, however, that this model's predictions for flood days were inferior to those obtained by simply using the average. The tendency to use the current day's water level as the primary source of prediction continues, but several different factors start getting relevance. Pressure, temperature, solar radiation, and the average water level in the last three days all appeared in the third and fourth levels of the tree, and rain also appeared in the fourth level. This bigger variance is coherent, since the more days pass, the less the water column level of today will be determinant for the future, and the atmospheric conditions start to play an important role. Those variables were probably helpful for the normal day predictions, but they were not enough to help the model succesfully predict the outliers with the given samples.

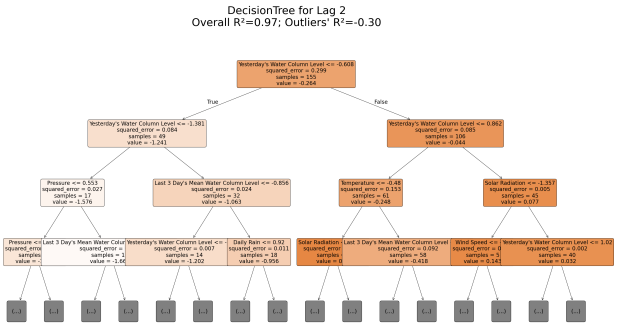


Figure 15. Best model for two-day prediction explained

At last, the three-day prediction was better solved by the Gradient Boosting model, which also did not achieve decent standards for the flood events. The level of the current day was still the most important feature, but the outliers did not impact the model as much as they did in the one-day prediction, for example. Pressure, the water column level in the last three days, and wind direction were also relevant for the prediction. API, dewpoint temperature, and temperature follow, but different feature values share similar impact, which can raise suspicions on how determinant they actually are. The remaining features follow with less impact in Figure 16.

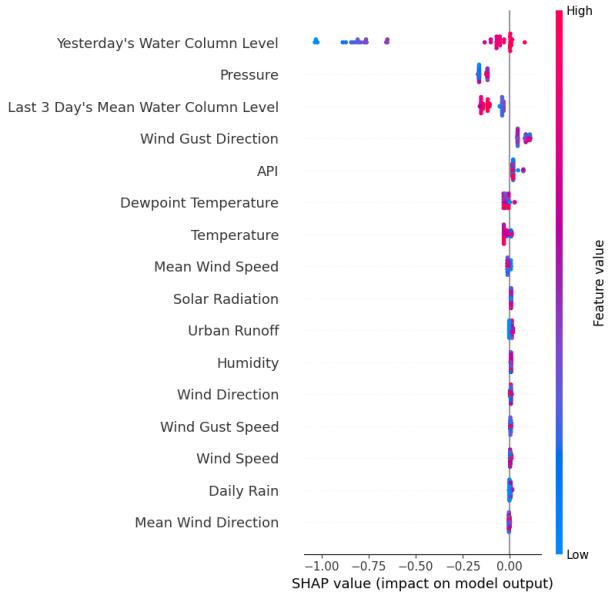


Figure 16. Best model for three-day prediction explained

5 Conclusions

This study, through two different case studies, was able to achieve decent to great prediction in flood events. In the INMET case study, it was proved that, using the data of the floods that occured in September and November of 2023, it is possible to completely predict the events in May 2024 with up to 2 days of antecedence. It was also possible to predict a good portion of the events for longer windows, though overfitting suspicions rose.

The SAMAE case study had more modest findings, but proved that it is also possible to decently predict how the level of a reservoir is affected by extreme rainfall up to one day in the future. This case study was strongly affected by the lack of data collected during flood events, and deserve further investigation in how to deal with this imbalance and improve the outlier prediction.

In comparison to the baseline models proposed in this study, the AI models developed showed great improvement, both in general terms and when analyzing the flood events isolated. The models were also on par with the state of the art reviewed. For example, Furquim *et al.* [2018] used a very similar classification model to the INMET case study, and obtained 80% of accuracy for civil defense's red alerts, while the models developed achieved between 77.7% and 93.3%. Dawson *et al.* [2002], Tabbussum and Dar [2021], and Pereira Filho and dos Santos [2006] achieved 0.98, 0.97, and 0.96 of R^2 for runoff prediction, which is very close to the results obtained for the level of the reservoir in SAMAE case study. These studies, however, do not specify how well

the model performed with the outliers, which limits the comparison in this regard.

The following subsections will be discussing some of the major limitations on the models developed for this study. Those acknowledgments help to better define what the model can and cannot do, as well as pave the way for further work that can improve on the results obtained so far.

5.1 Labeling uncertainty

This study used data from the state's Civil Defense to label the flood events for the INMET case study. This is not a flawless methodology, as the warnings they provide not always represent the exact days in which the city was harmed by flood. Sometimes, the warnings merely represent risk, rather than the actual occurrence of an event. That's the reason why this study used only alerts from Civil Defense, which are strictly triggered when there's actual devastation happening, instead of warnings. The articles used for support are also very unclear about dates, with long date ranges and usually talking about the month as a whole instead of specific days. This lead to a strictness in the labeling that could cause confusion for the model.

This uncertainty regarding the dates could lead to ambiguities. If the model is taught that a day was affected by a flood even if it was not, the model will make wrong assumptions, and this will lead to a wrong comprehension of the problem. Therefore, the accuracy might be high, but if the labels are inherently wrong, that means nothing for the model's capacity of prediction. As stated by Saravi *et al*. [2019], a model is just as good as the data that it was trained on.

The uncertainty in the labeling is a known limitation of the model, but the authors defend that, given the information freely available at the internet of the incidents, this method is as close as possible from reality. For future works, a personal data collection when the events are taking place would be more appropriate.

Another way to overcome this limitation is to improve the classification system to something similar to the work of Furquim *et al*. [2018], which uses the different levels of alert from Civil Defense as the output of the model.

5.2 Spatial and Temporal Limitations

The data collected for the training of the INMET case study was obtained at a specific point in the city, the Salgado Filho Airport. However, the labels consider flood in the city as a whole, since the Civil Defense data has little spatial accuracy. This can lead to some uncertainty in the prediction, since the data is from a point in the city that was not necessarily damaged by the flood, and that is not reflected in the labeling.

Also, the classification method of this case study has a spatial limitation that makes it specific for cities in which the flooding was directly caused by the rain in its own region. The floods in Porto Alegre in this same period, for example, were caused by the level of Lake Guaíba rising because every river in the surrounding area flows into Guaíba. Therefore, for cases similar to Porto Alegre, the rain in other regions as well as the level of the surrounding rivers would most likely be fundamental, and the method applied here would need some adjustments to truly fit the problem.

Another limitation of the model, also related by the lack of data available, is the temporal limitation. Since the rainfall data is only collected once a day in the city, the lags are in days, instead of hours. However, floods can probably be better predicted within the same day, few hours prior. Twenty-four hours is a huge lag for prediction, considering how rapidly the weather can change, and data collected more frequently could lead to better accuracy, as well as very short-term predictions. For future works, the model can be tested without rain data, but as shown in the SHAP analysis, rainfall and API are very relevant features in the INMET model.

This temporal limitation is also true to the SAMAE model, and seemed to affect the prediction more strongly than it did in the INMET model, as the lack of examples could be diminished without the daily aggregation made to match the temporality of the rainfall data. In this case, the rainfall-related variables seemed less relevant, and a test without those features looks promising, and should be further explored.

For the INMET model, even collecting meteorological and hydrological data hourly would be insufficient to solve the temporal limitation, since the labeling would continue to be daily. To properly solve this issue, the time precision of the flood events would have to be better defined as well.

5.3 Prediction Policy

In this study, the authors opted for prioritizing correctness on flood prediction rather than overall accuracy. This tends to lower the overall accuracy as flood events are more rare than days in which the city is not flooded. This leads to the model wrongly predicting several cases of flood. This could be considered a conservative or alarmist approach, as it will always choose to predict a flood when in doubt.

The alarmist nature of the model, while not ideal, is believed to be better in comparison with a less conservative model. The authors defend that it is better to be prepared for damages that will not happen than to be taken by surprise. While it would be incorrect to say that the events that took place in May 2024 were a surprise, given the previous events that served as alerts, the cities in Rio Grande do Sul were certainly unprepared for the event. The damage could have been diminished with some degree of precaution, and the authors believe models like these can help structuring early warning systems that can lead to more readiness in the case of flood events.

Therefore, while a limitation of the model, its alarmist nature can also be seen as something positive to city planners who could use the model to assess the possibility of a flooding. The model urges for precaution, just like the population.

5.4 Contributions

This paper contributes to the literature by applying the current Machine Learning techniques in early prediction of flood events in a territory that lacks this kind of modeling. It also employs a binary classification system that produces a straightforward, easily interpretable result to aid decision-makers. This differs from most runoff predictions in the

current state of the art, and follows the approach taken by Furquim *et al*. [2018]. Moreover, this study uses hydrological reasoning and the SHAP method to improve explainability of how these models work, which is a gap in the literature. Finally, this study takes an isolated look at the prediction of outliers, guaranteeing that the models are actually making good predictions for the days of extreme rainfall and not only achieving good metrics because of the normal cases.

Acknowledgements

This study recognizes the collaboration of SAMAE team, especially Márcio Adami, for acquiring and sanitizing the data used for the SAMAE case study. It also acknowledges the importance of the open data of the INMET team and the warnings by the Civil Defense, which made the study possible with great data available for free and for everyone. The future of data is open data.

References

Adikari, K. E., Shrestha, S., Ratnayake, D. T., Budhathoki, A., Shanmugam, M. S., and Dailey, M. N. (2021). Evaluation of artificial intelligence models for flood and drought forecasting in arid and tropical regions. *Environmental Modelling & Software*, 144:105136. DOI: 10.1016/j.envsoft.2021.105136.

Aichouri, I., Hani, A., Bougherira, N., Djabri, L., Chaffai, H., and Lallahem, S. (2015). River flow model using artificial neural networks. *Energy Procedia*, 74:1007–1014. DOI: 10.1016/j.egypro.2015.07.832.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357. DOI: 10.1613/jair.953.

CNM (2022). Prejuízos causados pelas chuvas no brasil entre 2017 e 2022 ultrapassam r55, $5bilhes, $revelacnm.

Dawson, C., Harpham, C., Wilby, R., and Chen, Y. (2002). Evaluation of artificial neural network techniques for flow forecasting in the river yangtze, china. *Hydrology and Earth System Sciences*, 6(4):619–626.

de Sene, E. and Moreira, J. C. (2012). *Geografia geral e do Brasil: espaço geográfico e globalização*, volume 1. Editora Scipione, São Paulo.

FLOODsite (2005). Language of risk.

Furquim, G., Filho, G. P. R., Jalali, R., Pessin, G., Pazzi, R. W., and 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. DOI: 10.3390/s18030907.

Gao, J. (2023). R-squared (r2) – how much variation is explained? *Research Methods in Medicine Health Sciences*, 5(4):104–109. DOI: 10.1177/26320843231186398.

Geron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Sebastopol, 2 edition.

Gutierrez-Lopez, A., Cruz-Paz, I., and Mandujano, M. M. (2019). Algorithm to predict the rainfall starting point as a function of atmospheric pressure, humidity, and dewpoint. *Climate*, 7(11):131. DOI: 10.3390/cli7110131.

GZH (2024). Moradores de vila s ao pedro, no interior de caxias do sul, voltam a ficar ilhados por conta da chuva. *GZH — Pioneiro*.

GZH (2025a). Bento gonçalves e caxias do sul registram estragos causados pela chuva. *GZH — Pioneiro*.

GZH (2025b). Temporal no rs provoca mortes, bloqueio de estradas e alagamentos em v'arios munic'ıpios. *GZH*.

Hodson, T. O. (2022). Root-mean-square error (rmse) or mean absolute error (mae): when to use them or not. *Geoscientific Model Development*, 15:5481–5487. DOI: 10.5194/gmd-15-5481-2022.

INMET (2025). Sobre meteorologia.

IPCC (2023). Climate change 2023: Synthesis report.

Khosravi, K., Pham, B. T., Chapi, K., Shirzadi, A., Shahabi, H., Revhaug, I., Prakash, I., and Bui, D. T. (2018). A comparative assessment of decision trees algorithms for flash flood susceptibility modeling at haraz watershed, northern iran. *Science of The Total Environment*, 627:744–755. DOI: 10.1016/j.scitotenv.2018.01.266.

Klemeš, V. (1983). Conceptualization and scale in hydrology. *Journal of Hydrology*, 65:1–23.

Kourgialas, N. N. and Karatzas, G. P. (2017). A national scale flood hazard mapping methodology: The case of greece – protection and adaptation policy approaches. *Science of the Total Environment*, 601–602:441–452. DOI: 10.1016/j.scitotenv.2017.05.197.

Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20(2):130–141. DOI: 10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2.

Marengo, J. A., Dolif, G., Cuartas, A., Camarinha, P., Gonçalves, D., Luiz, R., Silva, L., Alvava, R. C. S., Seluchii, M. E., Moraes, O. L., Soares, W. R., and Nobre, C. A. (2024). O maior desastre climático do brasil: chuvas e inundações no estado do rio grande do sul em abril-maio 2024.

Mosavi, A., Ozturk, P., and wing Chau, K. (2018). Flood prediction using machine learning models: Literature review. *Water*, 10(11):1536. DOI: 10.3390/w10111536.

Mosca, E., Szigeti, F., Tragianni, S., Gallagher, D., and Groh, G. (2022). Shap-based explanation methods: A review for NLP interpretability. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 4593–4603, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Munich Re Group (2004). Topics geo annual review: Natural catastrophes 2004.

Noymanee, J., Nikitin, N. O., and Kalyuzhnaya, A. V. (2017). Urban pluvial flood forecasting using open data with machine learning techniques in pattani basin. 119:288–297. DOI: 10.1016/j.procs.2017.11.187.

Penning-Rowsell, E. C. and Fordham, M. (1994). *Floods across Europe. Flood Hazard Assessment, Modelling and Management*. Middlesex University Press, London.

Pereira Filho, A. J. and dos Santos, C. C. (2006). Modeling a densely urbanized watershed with an artificial neural network, weather radar and telemetric data. *Journal of Hydrology*, 317(1-2):31–48. DOI: 10.1016/j.jhydrol.2005.05.007.

Razavi, S., Gober, P., Maier, H. R., Brouwer, R., and Wheater, H. (2020). Anthropocene flooding: challenges for science and society. volume 34, pages 1996–2000. DOI: 10.1002/hyp.13723.

Russel, S. J. and Norvig, P. (2022). *Inteligência Artificial: Uma Abordagem Moderna*. gen LTC, 4 edition.

Saravi, S., Kalawsky, R., Joannou, D., Casado, M. R., Fu, G., and Meng, F. (2019). Use of artificial intelligence to improve resilience and preparedness against adverse flood events. *Multidisciplinary Digital Publishing Institute*.

Sharda, R., Delen, D., and Turban, E. (2019). *Business intelligence e análise de dados para gestão do negócio*. Bookman, 4 edition.

Sharma, K., Anand, D., Sabharwal, M., Tiwari, P. K., Cheikhrouhou, O., and Frikha, T. (2021). A disaster management framework using internet of things-based interconnected devices. *Mathematical Problems in Engineering*, 2021(2629):1–21. DOI: 10.1155/2021/9916440.

Singh, R. and Mangat, N. S. (1996). *Stratified Sampling*, volume 15 of *Kluwer Texts in the Mathematical Sciences*, pages 102–144. Springer, Dordrecht. DOI: 10.1007/978-94-017-1404-4$_5$.

Tabbussum, R. and Dar, A. Q. (2021). Performance evaluation of artificial intelligence paradigms—artificial neural networks, fuzzy logic, and adaptive neuro-fuzzy inference system for flood prediction. *Environmental Science and Pollution Research*, 28(20):25265–25282. DOI: 10.1007/s11356-021-12410-1.

Teixeira, C. A. (2010). Apostila de hidrologia aplicada. Curitiba.

Todini, E. (1988). Rainfall-runoff modeling – past, present and future. *Journal of Hydrology*, 100:341–352.

Tomé, B. (2025a). Choveu mais de 160 milímetros em cerca de 36 horas em caxias do sul. *GZH — Pioneiro*.

Tomé, B. (2025b). Moradores de caxias do sul protestam após danos causados pela chuva no bairro rio branco. *GZH — Pioneiro*.

Tucci, C. E. M. (2004). *Hidrologia: Ciência e Aplicação*. Editora da UFRGS, Porto Alegre, 4 edition.

Viessman, W. J. and Lewis, G. L. (2003). *Introduction to Hydrology*. Prentice Hall, Upper Saddle River, New Jersey, 4 edition.

Wasko, C., Westra, S., Nathan, R., Orr, H. G., Villarini, G., Herrera, R. V., and Fowler, H. J. (2021). Incorporating climate change in flood estimation guidance. *Philosophical Transactions of the Royal Society A*, 379(2195). DOI: 10.1098/rsta.2019.0548.

World Weather Attribution (2024). When risks become reality: Extreme weather in 2024.

A Related Works Summary

Table A.1. Data Used in Related Works

Article	Data Source	Variables	Data Period
Noymanee *et al*. [2017]	Open data	Basin's level, precipitation	2015-2017
Saravi *et al*. [2019]	Governmental departments	Meteorological, rain duration, material damage, dead and hurt people	1994-2018
Furquim *et al*. [2018]	Sensoring	Precipitation, river's level, and images	2013-2014
Tabbussum and Dar [2021]	Governmental departments	Precipitation, runoff, evapotranspiration, and vegetation	1990-2018
Pereira Filho and dos Santos [2006]	Governmental departments	Precipitation, runoff e river's level	1991-1995
Aichouri *et al*. [2015]	Meteorological stations	Precipitation and runoff	1986-2003
Dawson *et al*. [2002]	Three Gorges University	Precipitation and runoff	1991-1993
Adikari *et al*. [2021]	Open data (Australia) and governmental departments (Thailand)	Meteorological, precipitation, and runoff	1987-2014 (Australia) and 1981-2018 (Thailand)
Khosravi *et al*. [2018]	Governmental departments	Precipitation, basin's slope, altitude, and curve	1991-2015
Kourgialas and Karatzas [2017]	Governmental departments	Precipitation, runoff, vegetation, basin's altitude, slope, and erosion susceptability	1960-2014

Table A.2. Models and Results in Related Works

Article	**Metrics and Results**	**Prediction type**	**Model**
Noymanee *et al*. [2017]	MAE (0.12), RMSE (0.16), and EI	Basin's level (observational)	Bayesian Linear Regression, ANN, Decision Tree, Random Forest
Saravi *et al*. [2019]	RMSE (0.13), MAE (0.27), Accuracy (79.83%), and Confusion Matrix	Flood class	Random Forest, ANN, Naive Bayes, Logistic Regression, Lazy Learner
Furquim *et al*. [2018]	RMSE (22.92), MAE (12.97), and R^2 (0.6813) for the river level e Accuracy (80% and 66%), sensibility (90% and 70%), and specificity (88%, and 72%) for the red and yellow alerts, respectively	River level and civil defense's alerts	Neural Network
Tabbussum and Dar [2021]	NSE (0.968), R^2 (97.07%), MSE (0.00034), RMSE (0.018), MAE (0.0073), CA (0.018)	Runoff	ANN, Fuzzy models, and ANFIS
Pereira Filho and dos Santos [2006]	R^2 (0.96), and MSD (0.033)	River level and runoff	ANN
Aichouri *et al*. [2015]	R (0.902), ASE (0.00001), and MARE (1.453%)	Runoff	ANN
Dawson *et al*. [2002]	MSRE (0.007), RMSE (0.1933), CE (94.7%), AIC (1611), and R^2 (0.98)	Runoff	ANN
Adikari *et al*. [2021]	Numeric results not presented. R^2, RSE, RAE, NSE, WI	Runoff (observational)	CNN, LSTM e WANFIS
Khosravi *et al*. [2018]	Accuracy (94.3%), Kappa (0.886), RMSE (0.216), AUC (0.976)	Susceptability to flooding	Decision Tree
Kourgialas and Karatzas [2017]	RMSE (0.52), NSE (0.96), and R^2 (0.95)	Susceptability to flooding	ANN

B SHAP summaries

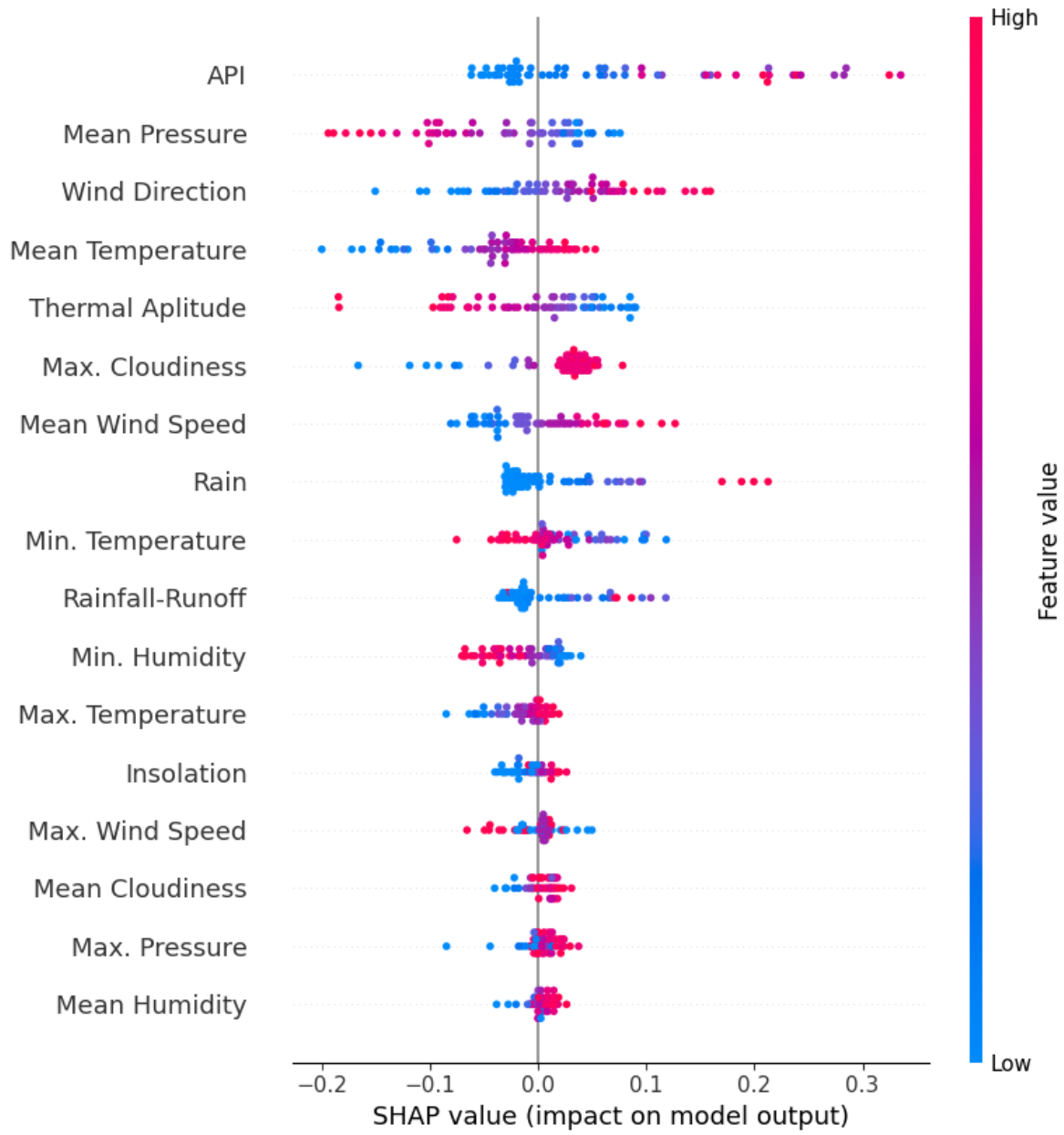


Figure B.1. SHAP summary of the Flood Detection Model

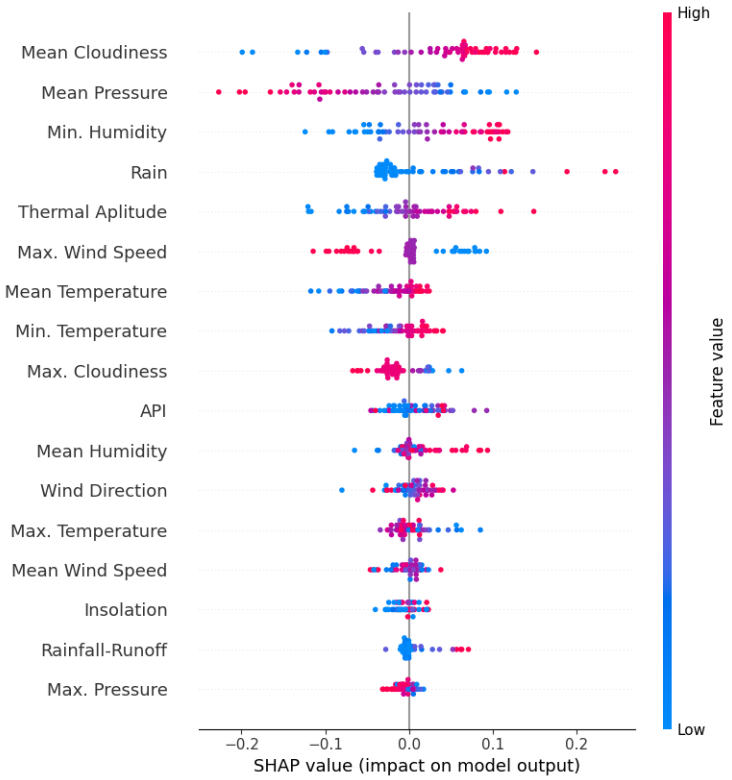


Figure B.2. SHAP summary of the One-Day Flood Prediction Model

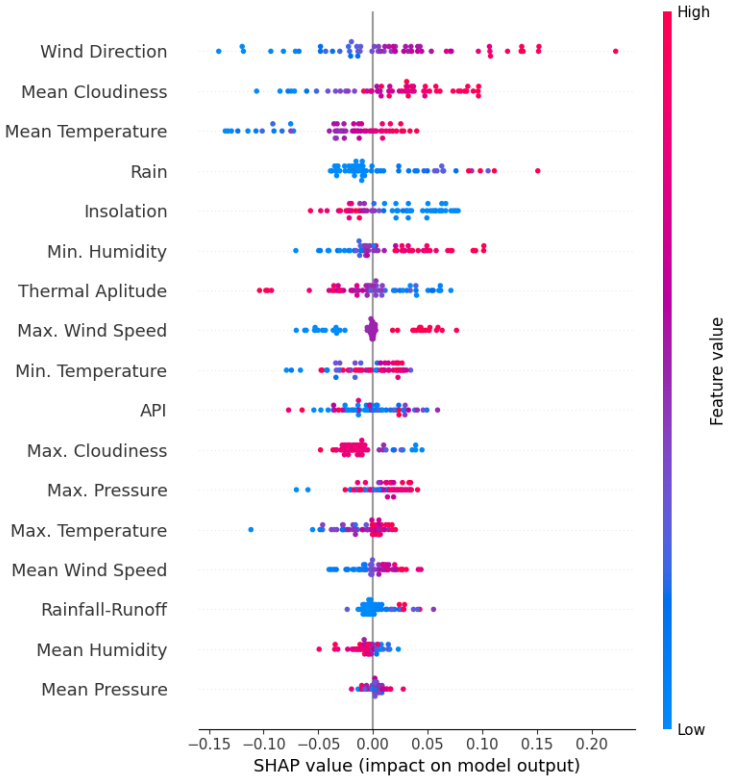


Figure B.3. SHAP summary of the Two-Day Flood Prediction Model

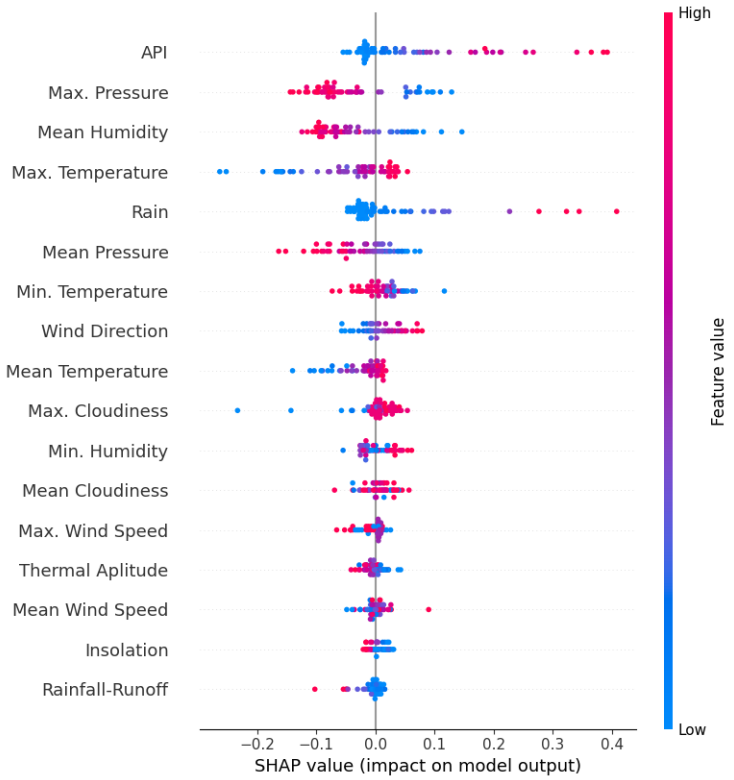


Figure B.4. SHAP summary of the Three-Day Flood Prediction Model

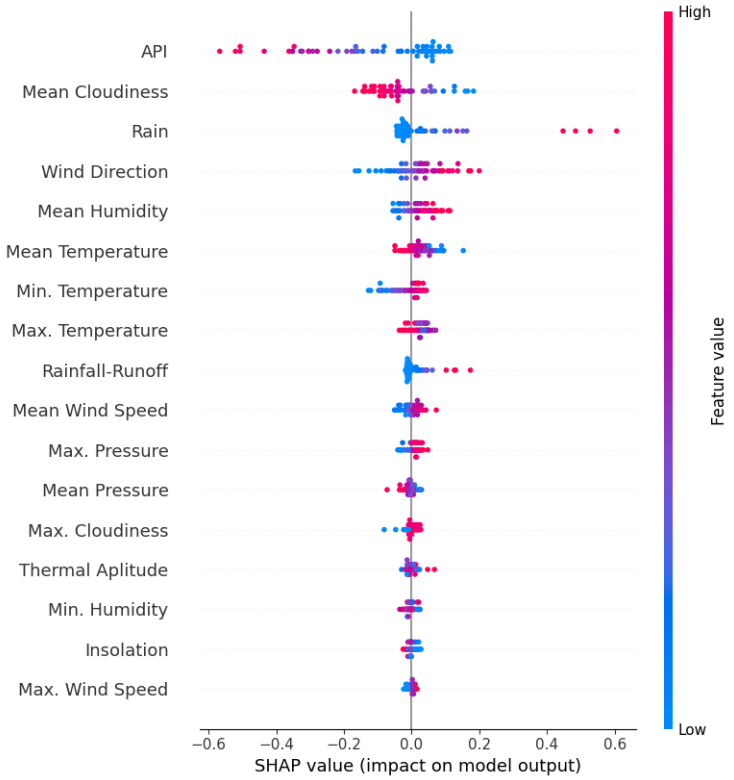


Figure B.5. SHAP summary of the Four-Day Flood Prediction Model

C SHAP summaries

Table C.1. Mean of the global absolute SHAP values for different prediction windows

Feature	0-day	1-day	2-day	3-day	4-day	Mean
API	**0.095**	0.023	0.022	**0.085**	**0.164**	**0.078**
Rain	0.039	0.048	0.034	0.059	0.071	0.050
Mean Cloudiness	0.012	**0.076**	0.043	0.018	0.078	0.045
Wind Direction	0.054	0.017	**0.056**	0.027	0.061	0.043
Mean Pressure	0.059	0.072	0.006	0.044	0.012	0.039
Mean Temperature	0.051	0.028	0.038	0.026	0.037	0.036
Thermal Aplitude	0.050	0.042	0.029	0.012	0.011	0.029
Mean Humidity	0.008	0.020	0.010	0.067	0.043	0.030
Min. Humidity	0.028	0.060	0.033	0.021	0.011	0.031
Max. Pressure	0.012	0.010	0.018	0.075	0.018	0.027
Max. Temperature	0.018	0.016	0.016	0.059	0.031	0.028
Min. Temperature	0.031	0.026	0.023	0.027	0.034	0.028
Max. Cloudiness	0.042	0.025	0.020	0.024	0.011	0.024
Max. Wind Speed	0.015	0.040	0.026	0.013	0.008	0.020
Mean Wind Speed	0.040	0.012	0.014	0.012	0.019	0.019
Rainfall-Runoff	0.031	0.010	0.012	0.012	0.024	0.018
Insolation	0.015	0.011	0.033	0.012	0.009	0.016