Modern Journal of Statistics
2(1), 138–160
DOI:https://doi.org/10.64389/mjs.2026.02143
https://sphinxsp.org/journal/index.php/mjs/

Research article

Comparison of Machine Learning Algorithms for Flood Prediction in Puri, Odisha

Rinku Poonia[1], **Ravinder Singh**[1,*], **R.K. Bhardwaj**[2], **Vikas Kumar**[3], **Aarzoo Rani**[1]

[1] Department of Statistics, Central University of Haryana, Mahendergarh, 123031, India; rinku231808@cuh.ac.in; ravinderstats@cuh.ac.in; arzoo.malik2600@gmail.com

[2] Department of Statistics, Punjabi University, Patiala, Punjab, 147002, India; rkb_mstates@rediffmail.com

[3] Department of Civil Engineering, School of Engineering and Technology, Central University of Haryana, Mahendergarh, 123031, India; vikask@cuh.ac.in

* **Correspondence:** ravinderstats@cuh.ac.in

ARTICLE INFO

Keywords:
Flood prediction
Machine learning
Logistic regression
Decision trees
Support vector machines

Mathematics Subject Classification:
62-04, 62P12, 68T01

Important Dates:
Received: 28 October 2025
Revised: 10 December 2025
Accepted: 11 January 2026
Online: 27 January 2026

Copyright © 2026 by the authors. Published under Creative Commons Attribution (CC BY) license.

ABSTRACT

Floods pose a serious risk to communities, infrastructure, and overall regional development. Puri, a district in Odisha, India, is particularly prone to flooding due to its low-lying landscape and the heavy rains that occur during the monsoon season. In such areas, quick and accurate flood forecasting becomes crucial for protecting lives, planning evacuations, and reducing damage. This study aims to compare numerous machine learning methods, including Decision Trees, Logistic Regression, Random Forests, Support Vector Machines (SVM), and Lasso Regression, for predicting possible flood events using past flood data and environmental factors like rainfall, soil moisture, temperature, and other hydrological indicators. Soil moisture is an important variable, but its dataset was incomplete. To fill these gaps, three machine learning models were tested for soil moisture prediction. Lasso Regression performed the best, giving the lowest Mean Absolute Percentage Error (MAPE) of 0.17, and was chosen to generate the missing values. With this completed dataset, multiple algorithms were evaluated for flood prediction. Logistic Regression stood out, achieving a Recall Score of 1, a Matthews Correlation Coefficient (MCC) of 0.68, and an accuracy of 0.91. These results show that Logistic Regression is a strong and reliable choice for predicting floods in the Puri region.

1. Introduction

Floods lead to extensive loss of life and material things, such as property, infrastructure, public utilities, and livelihood systems, and therefore prove to be a major threat to a nation. The reasons contributing to this damage could be a sharp increase in population, urbanization, global warming, etc. It is concerning to know that damages caused by floods has an increasing trend. Regarding India, from the total geographic area of 329 mha, greater than 40 mha are at risk of flooding [21]. The average annual flood damage from 1996 to 2005 was Rs. 4745 crores, which was significantly more than the damage from the previous 53 years, which was Rs. 1805 crore. Major flood-affected states of India include Uttar Pradesh, Andhra Pradesh, Punjab, Haryana, Bihar, West Bengal, Odisha, Gujarat, etc. (Figure 1) [17].

Floods could be categorized into two types: Urban Floods and River Floods. Urban flood refers specifically to floods that occur within urban areas. The primary cause of these floods is excessive rainfall, which overburdens urban drainage systems and infrastructure. Rapid runoff of rainwater and decreased infiltration because of impenetrable surfaces in cities like roads, buildings, etc., increases the risk of floods. Water overflow in streets, basements, and buildings leads to infrastructure damage, interruption of transportation systems, and dangers to public safety. Overall, it disturbs urban life.

On the other hand, river floods occur due to the overflow of rivers or other water bodies. Generally, when snow-melt, heavy rainfall over an extended period, or a combination of both occur, it may lead river water levels to rise above their capacity levels, which further leads to river floods. They may cause extensive flooding in agricultural areas, floodplains, and other settlements along the river banks.

In this paper, we have taken into consideration the flash floods occurring in the Puri district of Odisha. Puri is prone to flooding mainly because of its low-lying topography as well as its monsoon season. This leads to a lot of damage to human lives and infrastructure. The factors taken into consideration for flood prediction are temperature, specific humidity, surface pressure, relative humidity, rainfall, and soil moisture.

Various techniques like GIS, HEC-RAS, and machine learning methods could be utilized for the purpose of flood prediction. These flood prediction models play an important role in the assessment of hazards and management of extreme events. Robust and precise flood forecasts are very important for future evacuation modeling, policy analysis, and strategies for managing water resources.

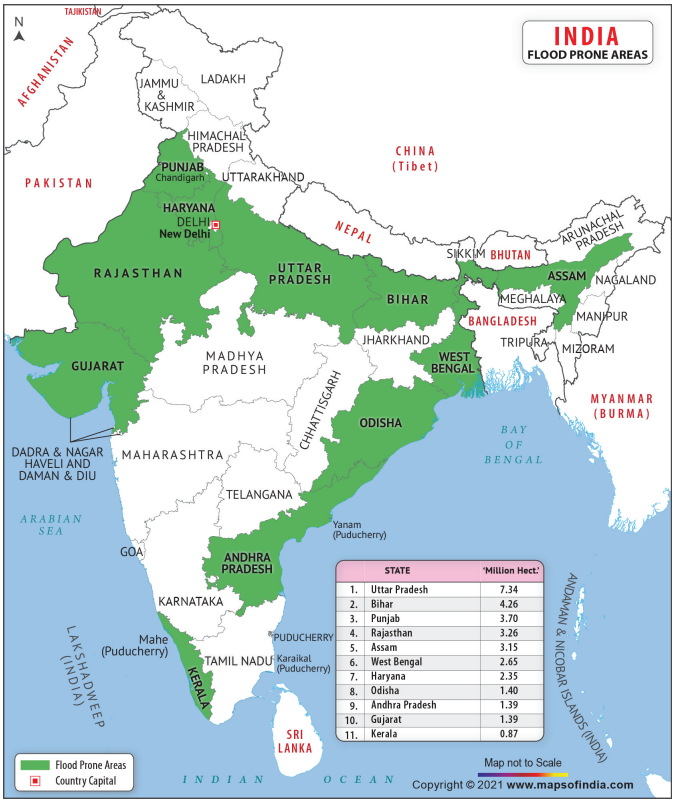


Figure 1. Top 10 Flood Prone Areas of India

We have used machine learning techniques like logistic regression, Support Vector Machine(SVM), Decision Tree, and Random Forest models for flood prediction. The data is divided into testing and training datasets. Further, machine learning algorithms are applied, and their accuracy is tested [27] and [21].

2. Literature Review

Different organizations around the world are working to make accurate and timely flood predictions. Various techniques have been used so far, like HEC-RAS, Aeronautical reconnaissance coverage Geographic Information System (Arc-GIS), and various machine learning (ML) algorithms like DT, RF, SVM, K-nearest neighbor (KNN), and Artificial Neural Network (ANN). Some use maps and digital models, while some use numerical data in their research— some work on flash floods, some on river floods, and some on urban floods. [15] aimed at the analysis of regional flood in the Mahanadi basin. ANN was used to estimate the runoff that occurred due to rainfall and water discharge. Flood inundation was estimated using the HEC-RAS model. The goodness of fit and correlated statistics were also tested. [5] aimed to increase the reliability and scalability of the flood control network through the use of IoT-based flood monitoring and ANN-based flood prediction. Temperature, river water level, humidity, rainfall, and pressure were monitored, and their periodic correlation data were found for flood prediction analysis. [2] examined the use of the logistic regression for mapping flood susceptibility in the southern Gaza Strip using six variables- rainfall, flow accumulation, topographic slope, digital elevation model (DEM), land use/land cover (LULC), and soil type. [18] demonstrated the use of ML models in flood prediction and provided insight into the most appropriate models. [9] worked on predicting areas susceptible to floods in the city of Amol, Iran, using ML algorithms. A vulnerability map was also created, and risk was assessed. [25] used a Deep Neural Network for predicting the occurrence of floods and compared the error and accuracy of deep learning models and ML models. [8] used an ANN model to make predictions regarding the depth of flood in the hour ahead. Root mean square error, average absolute error, and the coefficient of determination were calculated and used to examine the performance of the created models. [19] worked on the development of a flood prediction system using ML classifiers along with GIS techniques for urban areas. A flood susceptibility model and flood risk index were also produced. [10] employed three ML models and determined the best machine learning model using Cohen's kappa and the area under the ROC curve (AUC). [30] talks about various methods that could be used for regional flood frequency analysis. These methods include SVM, ANN, RF, hybrid models, linear regression, multiple regression, etc., which are based on artificial intelligence. [20] discussed the importance of machine learning in understanding floods in the Kerala state of India and suggested the optimal machine learning algorithm for the forecast. [7] proposed four ML models: Logistic Regression (LR), RF, MLP, and SVM to study the modeling principles of various machine learning models in forecasting flash flood vulnerability and evaluated the performance using the ROC curve. [3] reviewed flood forecasting methods to examine the development of research in the field of flood prediction, evaluate the positive and negative attributes of each method employed, and identify critical research needs. [13] did a comparative analysis of machine learning and deep learning techniques, such as one-dimensional Convolutional Neural Network (CNN), Long and Short Term Memory(LSTM), and MLP, for the Far-North region of Cameroon. [28] identified important flood causative factors and evaluated the performance of various ML algorithms, like linear regression, RF, SVM, and LSTM, for flood prediction and susceptibility mapping in the Amibara area, Ethiopia. [4] compared the performance of various machine learning algorithms and hybrid models for short-term flood forecasts for different forecast lead times. They concluded that hybrid models perform better than standalone models in almost every lead time. [1] developed 5 ensemble ML models to predict spatial-temporal water levels in an agricultural field. They suggested integration of deep learning and machine learning models for improved accuracy, and also focused on the impact of the drainage network and rainfall patterns for improving the accuracy in flood predictions.

Some of the major findings can be presented in the Table 1.

Table 1. Overview of literature review

Author & Year	Dataset Used	ML Methods applied	Performance Metrics	Notable Findings
Al-Juaidi et al. (2018) [2]	rainfall, flow accumulation, topographic slope, DEM, LULC, and soil type	Logistic Regression	AUC and ROC,	Generation of a flood susceptibility map of the area. Classification of flow accumulation as the most significant, while LULC and soil type as the least significant factors influencing the occurrence of floods.
Sankaranarayanan et al. (2020) [25]	temperature and rainfall intensity	Deep Neural Network, SVM, KNN, and Naïve Bayes	accuracy and error	Deep neural network performs better and can be utilized for flood forecasting.
Dai and Cai (2021) [8]	optimum track, urban weather, tides, geographic height, water depth increment, and flood depth	Back-propagation Neural Network	mean absolute error, root mean square error, and the coefficient of determination	The model was tested for different time scales, that are 1-minute, 15-minute, 30-minute, and 60-minute, amongst which the 30-minute model achieved the best prediction accuracy.
Chen et al. (2023) [7]	elevation, slope, aspect, lithology, normalized difference vegetation index, modified normalized difference water index, surface radiation, plane curvature, profile curvature, gully density, terrain wetness index, highway density	LR, MLP, SVM, and RF	ROC and AUC	MLP model has the best flood susceptibility prediction performance. Factors- elevation, gully density, and population density have the highest impact on flood susceptibility prediction.

2.1. Literature Gap

A lot of research has been done in the field of flood prediction using a variety of variables and several techniques. Yet, there has been a gap in terms of using all the important and contributing factors. Soil moisture, which is one of the essential factors for flood prediction, has not been used yet along with other important factors. This gap has been filled in this research work. Soil moisture, along with other factors like rainfall, temperature, specific and relative humidity, and surface pressure, has been used as input variables with flood occurrence as the output variable. Machine learning algorithms have been used to complete the soil moisture data and to predict flood occurrence.

3. Study Area

Puri, one of the districts of Odisha that lies on the coast, is situated between latitudes 19°28' and 20°10' North and longitudes 86°9' and 86°25' East. With a total area of 3479 km^2 (Figure 2), Jagatsinghpur and Cuttack districts border it from the North, Ganjam from the southwest, Khordha from the west, and the Bay of Bengal from the east and southeast. The Puri Coastal Plain and the Mahanadi Deltaic Plain are the district's two main physical divisions. The Mahanadi Deltaic plain's average height is less than 100 meters, while the Puri coastal plain's average height is less than 50 meters.

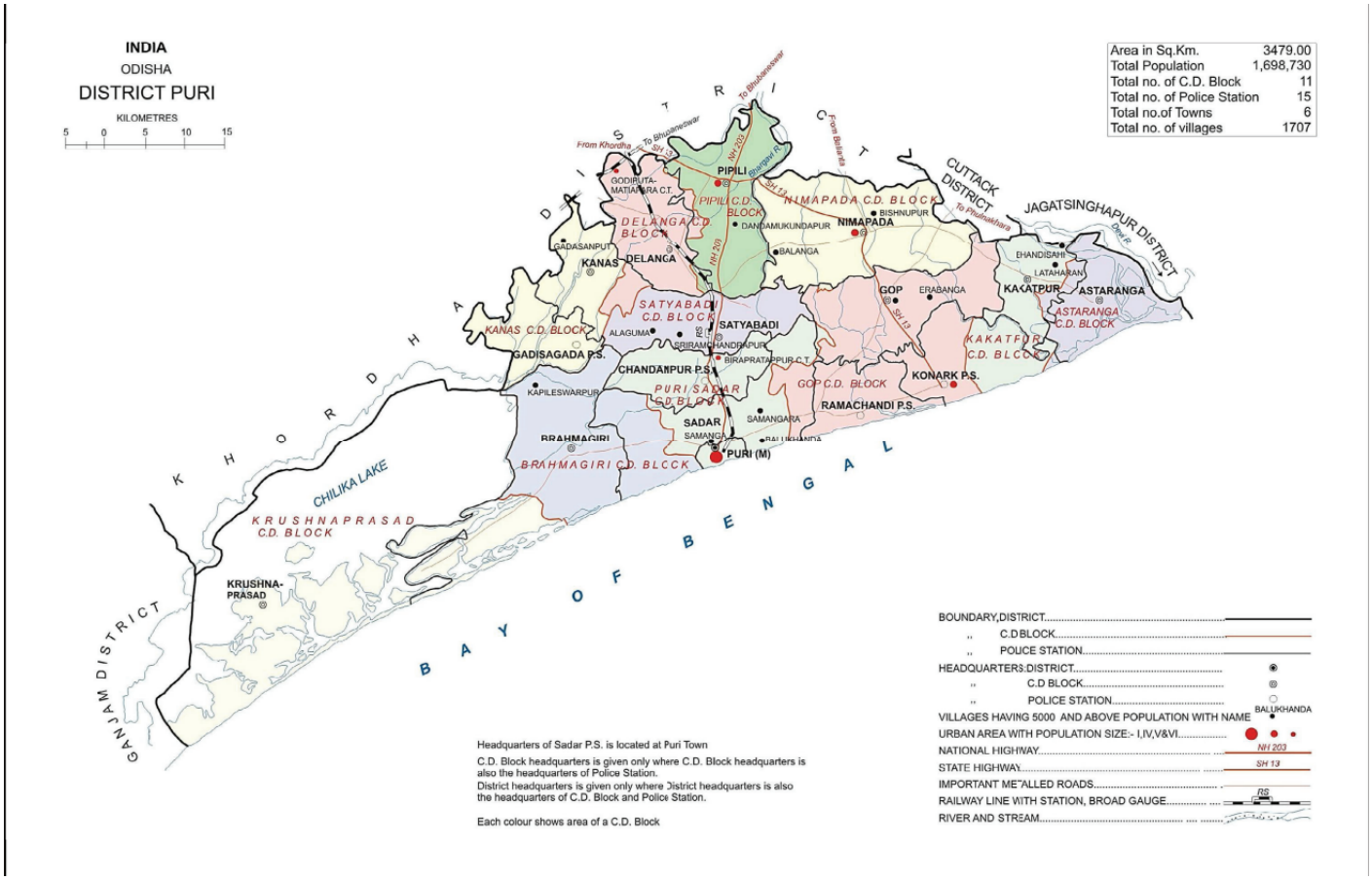


Figure 2. Map of Odisha

There are primarily three types of soil throughout this region: Aridsols, Entisols, and Alfisols. Aridsols,

which are the saline-alkali soils, cover the area near the coast. Entisols, the youngest form of alluvial soil, are found in the district's western region around Chilika Lake and also on the shoreline. Alfisols include the deltaic alluvial soil and make up the majority of the area. However, Ultisols (lateritic and laterite soils) occur over a small region in the district's northwestern sector.

The temperature is equable all year round in the district. The hot season experiences a mean maximum temperature of 31.9°C and occasionally as high as 42°C. However, the cold season experiences an average temperature of 17.1°C with minimum temperatures as low as 10.6°C. The district receives 1550 mm of rainfall on average per year.

According to the potential linked credit plan (2001-2002), submitted by NABARD, the district has an overall region of 2,64,988 hectares. The district has a total population of 1,502,682, with 1,298,654 people living in rural areas and 204,028 in urban areas. 86.42% of the district's total population resides in the village area [11].

4. Data Collection

Among various factors responsible for the flood, we took into consideration rainfall, temperature, specific humidity, relative humidity, soil moisture, surface pressure, and previous flood records. The data was collected from [23], [29], and [12]. Monthly data from January 1981 to December 2021 were collected for all parameters except soil moisture. Soil moisture data were available only from July 2018 to December 2021.

Table 2 provides us with a brief overview of the data variables used and their sources.

Table 2. Data Used

Variable	**Source of Data**
Rainfall	NASA Power— Data Access Viewer
Surface Pressure	NASA Power— Data Access Viewer
Humidity	NASA Power— Data Access Viewer
Temperature	NASA Power— Data Access Viewer
Soil Moisture	Ministry of Water Resources, Government of India and India Water Resources Information System
Flood Record	Department of Water Resources, Odisha

Apart from this, the census done by the Indian Government in 2014 of the district Puri gave an overview of the place in terms of area, population, weather, slope, soil, etc., which play a major role in floods. National Disaster Management Authority, Government of India, provided relevant information regarding floods, their impact, major flood-prone areas, etc.

5. Models and Methods Used

There are a variety of models/machine learning algorithms applicable to the prediction of floods. These models are explained below.

5.1. Logistic Regression

Logistic Regression is used in the case of classification problems where the aim is to predict the probability of an event belonging to a certain class. In this method, the output of the linear regression function is taken as an input, and then a sigmoid function (Figure 3) is used to estimate the probability for the provided class.
Let the independent input variables be:

$$X = \begin{bmatrix} x_{11} & \dots & x_{1m} \\ x_{21} & \dots & x_{2m} \\ \vdots & \ddots & \vdots \\ x_{n1} & \dots & x_{nm} \end{bmatrix},$$

and the dependent variable, which takes only binary values (0 or 1), is:

$$Y = \begin{cases} 0 & \text{for } Class\ 1 \\ 1 & \text{for } Class\ 2 \end{cases}$$

Now, we apply the multiple linear function to the input variables X as:

$$z = \sum_{i=1}^{n} \beta_i x_i + b,$$

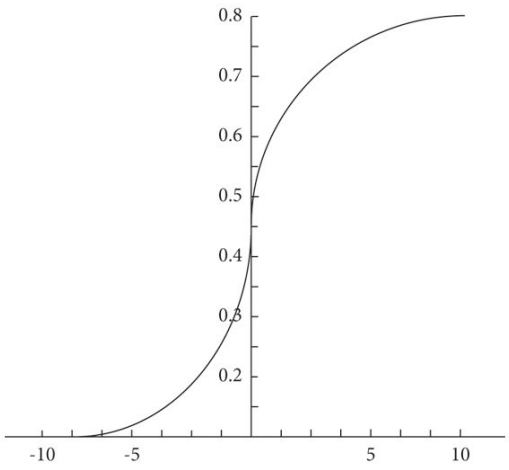


Figure 3. Sigmoid Function

where x_i is the i^{th} observation of X, β_i are the coefficients and b is the intercept term. Now, the sigmoid function (5.1) will be used to find the probability between 0 and 1, which is the predicted value of Y. The input for the sigmoid function will be z.

$$\sigma(z) = \frac{1}{1 + e^{-z}}, \tag{5.1}$$

where the likelihood of being in a class can be calculated as:

$$\begin{aligned} P(y=1) &= \sigma(z), \\ P(y=0) &= 1-\sigma(z). \end{aligned}$$

In our case, Y is "flood occurrence", and 1 denotes the occurrence of the flood while 0 denotes the non-occurrence of the flood. X has factors like rainfall, temperature, specific humidity, pressure, etc. [2] and [7].

Lasso (Least Absolute Shrinkage and Selection Operator) regression, commonly known as L1 regularization, is a linear regression technique. In this technique, the Ordinary Least Squares (OLS) cost function is penalized by the addition of the absolute values of the regression coefficients.

In Lasso regression, we find the variables and associated regression coefficients that produce a model with the least degree of prediction error. The model's parameters are restricted to reduce the regression coefficients close to zero. This is done by converting the regression coefficients' net absolute value to a value less than a predefined value (λ). The model omits variables with a regression coefficient of 0 after shrinking.

In Lasso Regression, the objective function could be expressed as:

$$\begin{aligned} & \textit{minimize}: \quad \sum_{i=1}^{n}\left(y_i - \left(\beta_0 + \sum_{j=1}^{p}\beta_j x_{ij}\right)\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j|, \\ \textit{i.e.} \quad & \textit{minimize}: \quad (RSS) + \lambda\sum_{j=1}^{p}|B_j|. \end{aligned}$$

where,

- n is the number of measurements/observations.
- p stands for the number of parameters available in the dataset.
- x_{ij} denotes the value of the j^{th} parameter associated with the i^{th} observation.
- 'RSS' or 'Residual Sum of Squares' is the variation among the actual and predicted values.
- λ is the regularization parameter that controls the amount of regularization applied. A higher lambda value results in more coefficients being set to zero.

The value of λ is frequently selected via an automated k-fold cross-validation method. For this procedure, the dataset is split randomly into k sub-samples of equal size. The prediction model developed using the (k-1) sub-samples is validated using the remaining sub-sample. Each of the k sub-samples is used for model creation, and each is utilized for model validation. This procedure is carried out k times. An overall result is produced and used to select the final model by combining the k individual validation outcomes for various values and choosing the most appealing [24].

5.2. Support Vector Machine

Support Vector Machine (SVM) is a simple machine learning algorithm that can be utilized for both regression and classification. Generally, it is used for objectives related to classification. This technique provides significant accuracy.

In this technique, we discover a hyperplane that categorizes the data points in an N-dimensional space (Figure 4). There could be several hyperplanes for doing the selection. But we need to find the hyperplane that has the largest distance among data points for both classifications, also known as the margin. Maximizing the margin distance aids in the classification of data points with more confidence.

Hyperplanes help in classifying the data points. The side where the data points lie on the hyperplane tells about the class to which they belong. The hyperplane could be a line or a two-dimensional plane, etc., based on the number of input features, i.e., 2 or 3, etc., respectively.

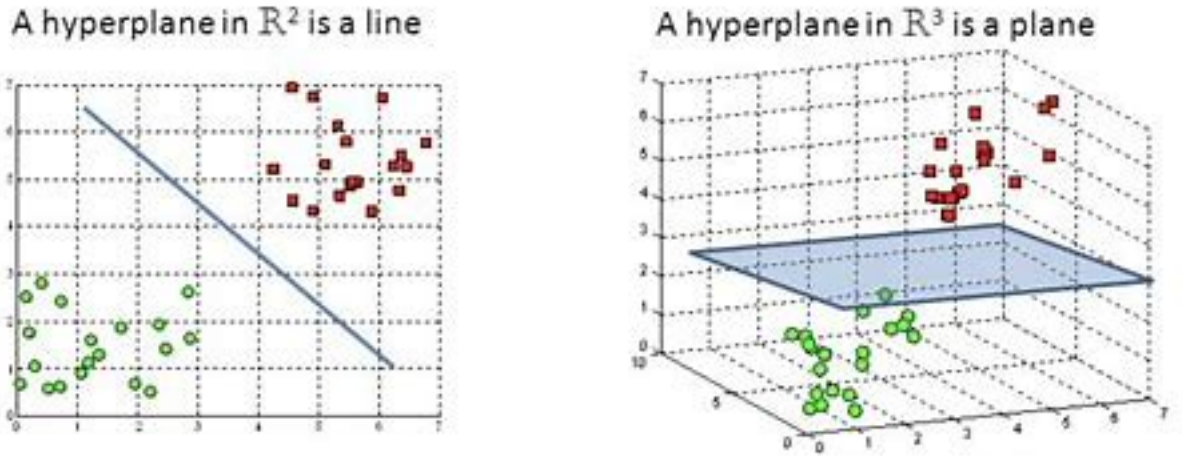


Figure 4. Hyperplanes formed in Support Vector Machine

The data points lying closer to hyperplanes are the support vectors. They have an impact on the hyperplane's positioning and position. The classifier's distance is maximized using these support vectors. The position of the hyperplane will shift when support vectors are deleted. Thus, the points aid in the development of SVM. For example, in SVM, we consider the output of the linear function, and its value being greater than k, where k is some constant, implies that it lies in class 1, and the output being $-k$ indicates its belongingness to class 2. Thus, the range $[-k, k]$ becomes the margin [14].

5.3. Decision Tree

A decision tree is one of the supervised learning techniques that can be taken into consideration in both classification as well as regression tasks. Mostly, it is considered for the case of classification. It forms a tree-like structure where each of the internal nodes denotes the features or attributes of a data set. The branches denote the decision, and every leaf node denotes an outcome, i.e., belongingness to a class. The diagrammatic representation of the Decision Tree is shown in Figure 5.

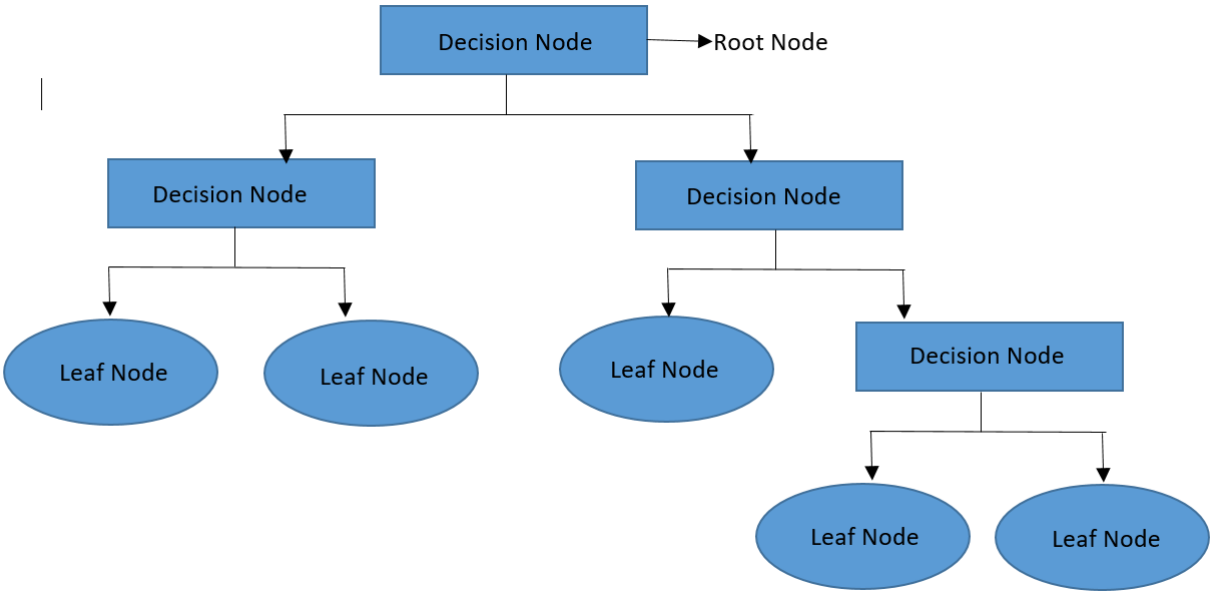


Figure 5. Working of the Decision Tree Algorithm

In this method, the training data is repeatedly split into subsets on the basis of attribute values until a criterion to stop the further split is met. During training, the best attribute is selected for splitting the data by using the DT algorithm, which is based on entropy. The aim is to get the attribute that would maximize the information gain. Entropy measures the degree of randomness in the data based on the distribution of class labels. The more the homogeneity of the class labels, the less the entropy and the more the information gain, and that attribute is therefore taken as the splitting attribute. The entropy and the information gain are as follows:

$$Entropy = H(s) = \sum_{i=1}^{n} -p_{c_i} log_2 p_{c_i},$$

where,

$$n = \text{number of categories},$$
$$p_{c_i} = \text{probability of being in category } i.$$

$$\text{Information Gain} = Gain(S, f_i) = H(S) - \sum_{v \in value} \frac{|S_v|}{|S|} H(S_V).$$

where,

$$H(S) = \text{Entropy of root node.}$$
$$H(S_v) = \text{Entropy of } v^{th} \text{ category following root node.}$$
$$S = \text{Entire sample from feature on root node.}$$
$$S_v = \text{Entire sample from } v^{th} \text{ category following the root node.}$$

The decision-tree algorithm could be written as follows:

1. Start building the tree beginning from the root node, say *R*, that consists of the entire dataset.
2. Search for the attribute that is the best in the dataset.
3. Split the root node (R) into subsets that contain all the possible values of attributes.
4. Select the best attribute and create the decision tree node.
5. Repeat the steps using the subsets of the dataset obtained in step 3 and create new decision trees.
6. Keep doing this process until no further classification of nodes is possible, and name the final node as a leaf node.

5.4. Random Forest

Another machine learning technique is Random Forest, which uses regression or classification trees for prediction. It is an ensemble of decision trees that were developed by using a data-driven bootstrap sample. Each decision tree in this technique predicts a class, and the class with the maximum votes is considered our predicted class. This technique could be used to train large databases efficiently and give clear classification results [16]. Figure 6 shows the working of the Random Forest Algorithm diagrammatically.

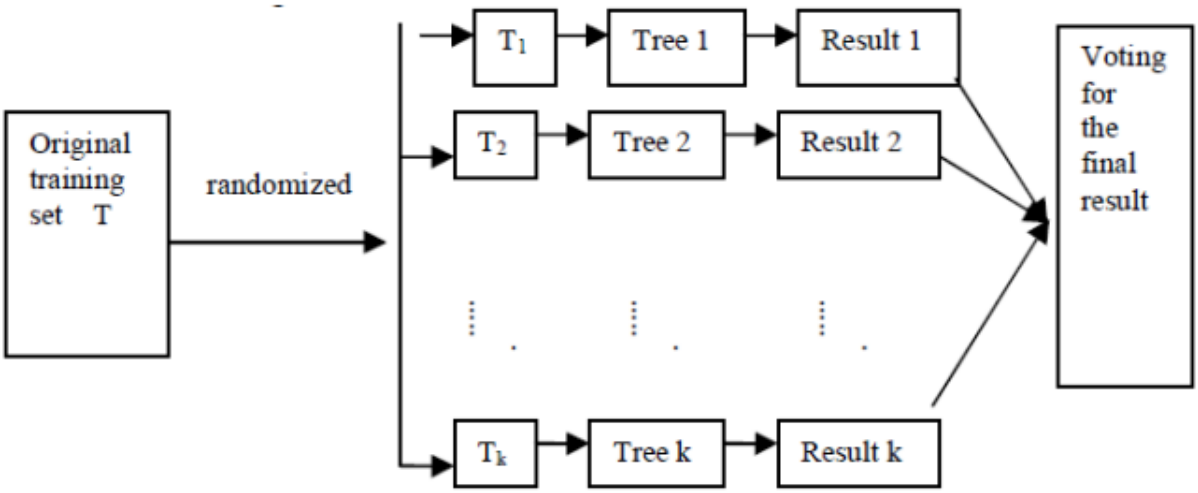


Figure 6. Working of Random Forest Algorithm

The Random Forest algorithm works as follows:

1. Randomly select *N* bootstrap samples from the training dataset (overlapping allowed).
2. Each bootstrap sample trains a decision tree, and *d* attributes are selected randomly at each node without permitting any duplication. Tree nodes are divided in a way that they create the best partition with respect to the objective function.
3. Perform steps 1 and 2 recursively *k* times.
4. Each decision tree's performance is observed, and the majority is checked to assign the class to the majority.

Generally, random forest optimizes Gini impurity, but for this model, we have specified the objective function to maximize the information gain in each segment [7] and [9].

6. Method

The entire process can be explained in the steps below:

1. **Data Collection:**
   First of all, we need to understand the factors responsible for floods in general and the factors of our concern for flood prediction. Having a proper understanding of these factors, we collect data on them. We were able to collect monthly data for the period January 1981 to December 2021.
2. **Data Pre-processing and Cleaning :**
   Once the data is available, we need to do pre-processing and cleaning of the data. Data cleaning is the technique used to 'clean' data. It is done by removing outliers (if any), replacing missing values (if any), smoothing noisy data, and correcting inconsistent data. An outlier is a point in our data that is significantly different from other observations in our data. It may arise because of variability in the measurement or as a consequence of experimental error.
3. **Data Scaling:**
   The process by which input variables/data are converted in a way to follow a distribution is known as scaling. It is important to do so before we train a machine learning model, as it helps to enhance performance. There are various methods that could be used for scaling as per our needs, like MinMax scaling, Standardization, Robust Scaling, etc. Data is split into dependent and independent datasets, and scaling is done on the data. We used Standard Scaler to do the scaling. This technique scales the features such that the features have zero mean and unit variance. It does so by subtracting the mean from each value and then dividing the value by the standard deviation ([20].
4.) **Applying machine learning algorithms on training and testing data:**
   The data is automatically split into training and testing datasets by the machine. Generally, this ratio is 70:30, i.e., 70% data is used to train the dataset, and our machine learns the working and evaluation by this training data. The purpose of the training dataset is that it is utilized to fit the model, and the values of the training dataset are known. The second dataset, known as a testing dataset, is used solely for prediction purposes. The library in Python called 'sci-kit-learn' provides us with the module 'model_selection', which provides us with the function train_test_split(), which splits the dataset into training and testing datasets ([20]).
5. **Testing and comparing accuracy :**
   We do the testing of our machine learning algorithms on testing datasets. Different algorithms are applied for training and testing, resulting in different accuracies. The percentage of instances that are correctly classified out of all instances is measured by accuracy. Its mathematical interpretation is :

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}.$$

   We also check the recall score ([6]) and Matthew's correlation coefficient ([26]).
   Recall is a metric that evaluates the number of correct positive predictions that are correctly recognized by the model. Its value lies between 0 and 1. Higher recall means that the model is effective at identifying positive instances. Its formula is:

$$Recall = \frac{TP}{TP + FN}.$$

MCC is a single-value metric that summarizes the confusion matrix. It considers true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) to provide an overall measure of classification performance. Its value lies between -1 and +1, where +1 denotes a perfect classification, 0 is a random classification, and -1 is a total disagreement between predicted and true values. Its formula is :

$$MCC = \frac{TN * TP - FP * FN}{\sqrt{(TN + FN)(FP + TP)(TN + FP)(FN + TP)}}.$$

These different values of recall score, accuracy, and MCC are compared to check which model gives the most accuracy and is best to be used.

7. Prediction of Soil Moisture

After collecting the relevant data, we realized that some soil moisture data was unavailable. Machine Learning algorithms were utilized to forecast the values of this missing data. The steps followed were as follows:

1. Import Data: The available data is imported into the Jupyter Notebook. There are a total of 42 input values of the variables. These input variables are Rainfall, Temperature, Specific Humidity, Relative Humidity, and Surface Pressure.
2. Cleaning of Data: The imported data is checked for outliers, skewness, and correlation. Figure 7 is the heatmap formed showing the correlation among variables. However, the applied algorithms are not affected by the correlation among independent variables.

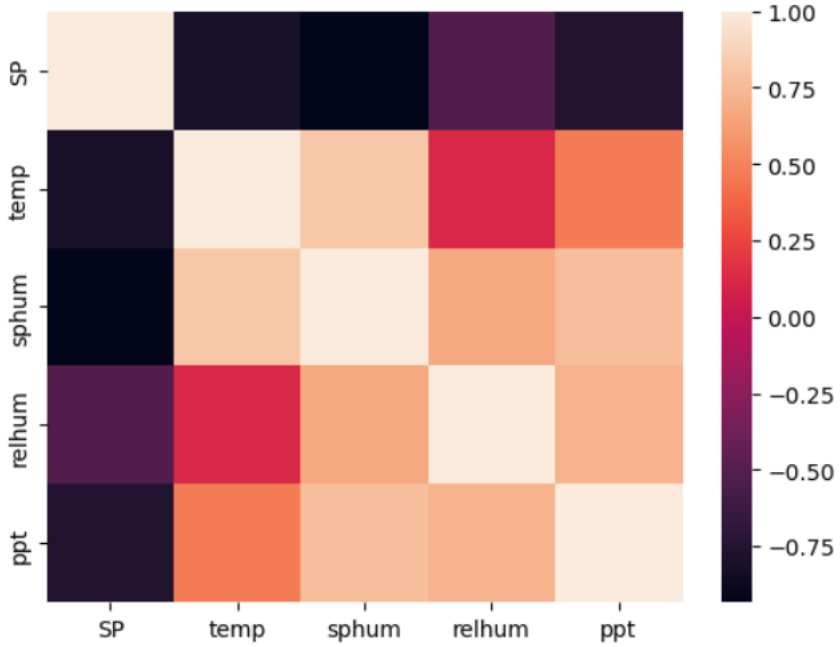
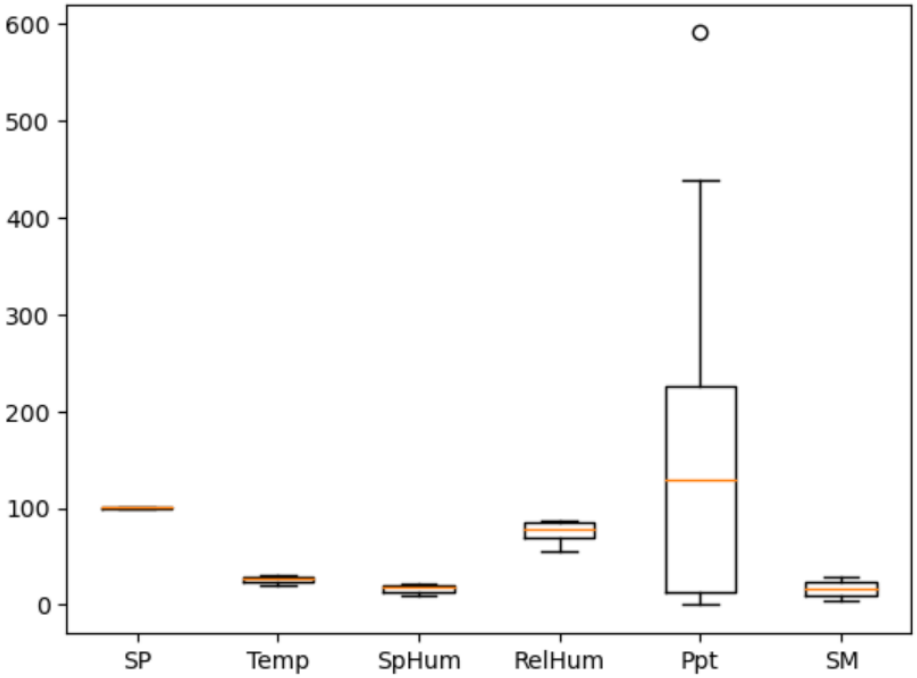


Figure 7. Heatmap showing correlation among independent variables

The presence of an outlier is spotted, which is removed by using the ”Inter-Quartile Range” method. Figure 8 and Figure 9 show the boxplots before and after the treatment of outlier,s respectively.



l

Figure 8. Boxplot of independent variables before treating the outlier

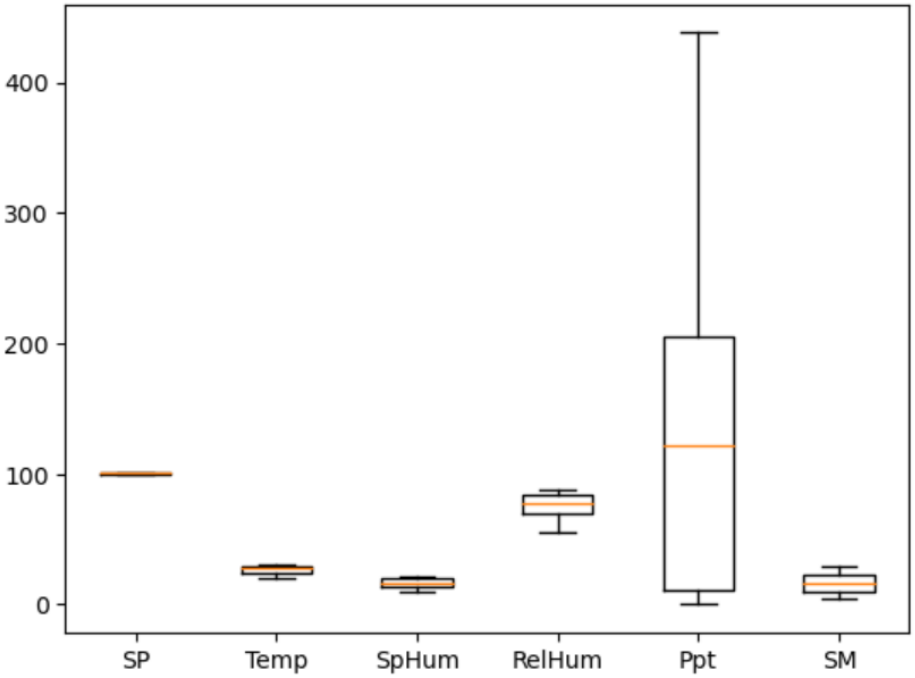


Figure 9. Boxplot of independent variables after removing outlier

3. Scaling of Data: The new data is Z-score normalized. This technique transforms the data to have a

zero mean and unit variance. The formula for standardization in this method is : $X_{scaled} = \frac{(X - X_{mean})}{X_{std}}$.

4. Applying Machine Learning Algorithms and comparing their accuracy: Machine Learning algorithms, namely, Lasso Regression, Random Forest, and Decision Tree, are applied to the training and testing data. Their mean absolute % error, mean absolute error, and root mean square error are compared (see Figure 10 and Table 3). In comparison, we observe that Lasso Regression is the most accurate and consistent model.

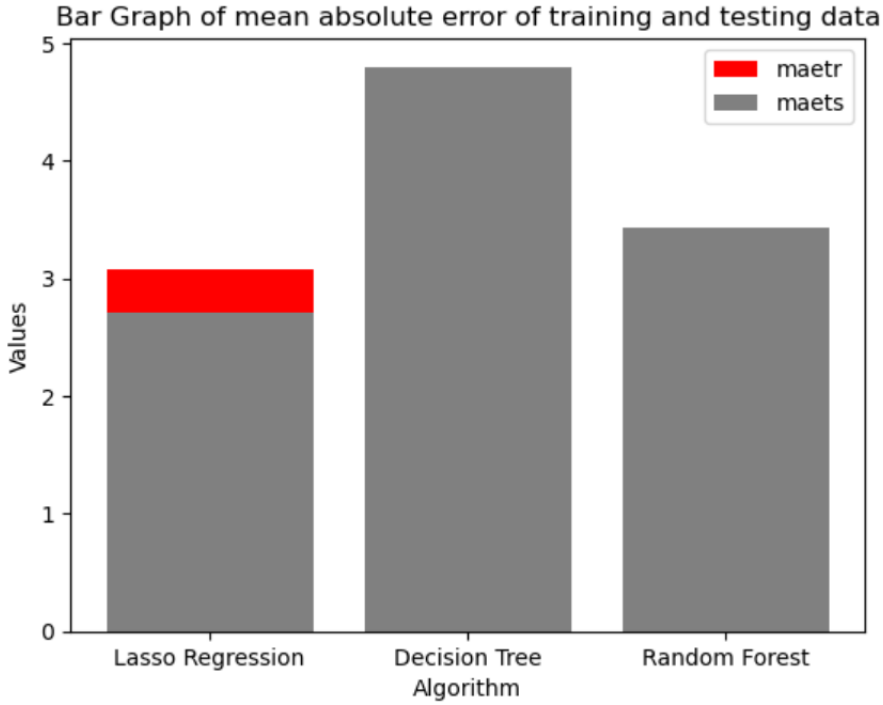


Figure 10. Comparison of Mean Absolute error of the three algorithms for soil moisture prediction

Table 3. Comparison of MAE, RMSE for soil moisture

	Lasso Regression	Decision Tree	Random Forest
MAE of training dataset	3.07	0	0.09
MAE of testing dataset	2.70	0.17	0.18
RMSE of training dataset	3.66	0	1.68
RMSE of testing dataset	3.37	4.36	4.09

5. Prediction of Soil Moisture: Decision Tree and Random Forest overfit. Lasso Regression gives the best results. So, we use it to make predictions of incomplete data regarding soil moisture. The predicted values are compiled into an Excel file and then joined to the actual data. Since the soil moisture data were studied and estimated from a small dataset, the data produced and used may influence the final flood predictions.

8. Flood Prediction Analysis

Once the data is completed by predicting unavailable values of soil moisture, we move forward to apply machine learning algorithms for the prediction of floods. The steps followed are as follows:

1. Import Data: The completed data is imported into the Jupyter Notebook. The input variables in this case are Rainfall, Temperature, Specific Humidity, Relative Humidity, Surface Pressure, and Soil Moisture.

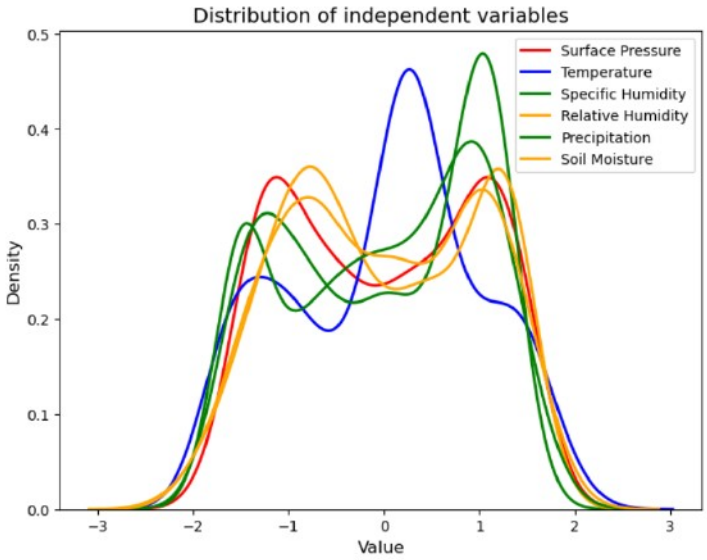


Figure 11. Probability distribution of the independent variables of the dataset

2. Cleaning of Data: The imported data is checked for outliers, skewness, and correlation. We can analyze the data curve as in Figure 11. Figure 12 shows that no outliers are present in the data.

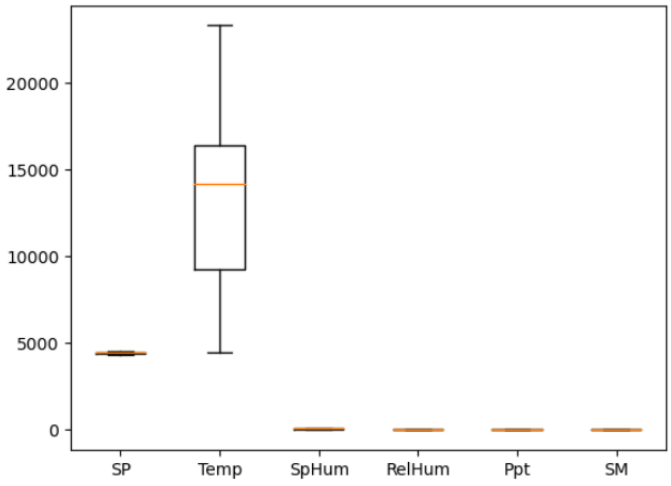


Figure 12. Boxplot of the independent variables

Skewness in the data is treated using the Box-Cox transformation, whose formula could be presented

as:

$$y(\lambda) = \begin{cases} \frac{y^{\lambda}-1}{\lambda} & \text{if} \lambda \neq 0 \\ logy & \text{if} \lambda = 0 \end{cases}$$

Where y denotes the original data and λ is the transformation factor which determines the type and magnitude of the transformation. The heatmap given below shows the correlation among the variables (Figure 13). Multicollinearity affects the coefficients and p-values, but it does not influence the predictions, the precision of the predictions, and the goodness-of-fit statistics. So, we can use the dataset for logistic regression [22].

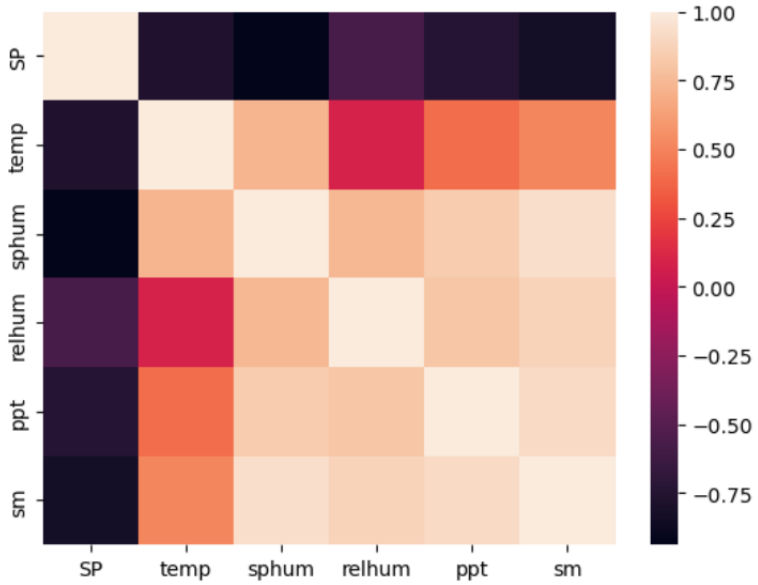


Figure 13. Heatmap showing correlation among independent variables

3. Scaling of Data: Just like the case of soil moisture, here also the data is Z-score Normalized, transforming the data to have zero mean and unit variance. Figure 14 and Figure 15 depict the relation among different variables before and after normalization. We can see that the dataset has been normalized, and the relationship between some of the variables has become linear post doing standardization.
4. Applying Machine Learning Algorithms: Machine Learning algorithms, namely, Logistic Regression, Support Vector Machine, Decision Tree, and Random Forest, are applied and tested on the training and the testing data, respectively. The number of trees (n_estimators) and maximum features used for the random forest classifier are 100 and 2, respectively.
5. Comparison of applied algorithms: Accuracy, Recall Score, Confusion matrix, and Matthews's Correlation Coefficient corresponding to all the applied models are compared to know about the best-fit model (Table 4).
   We realize that, as compared to the other applied models, Logistic Regression is the best-fitted model. The fitted coefficients of the model are presented in Table 5, showing that soil moisture, surface pressure, temperature, and precipitation play a vital role in making flood predictions.

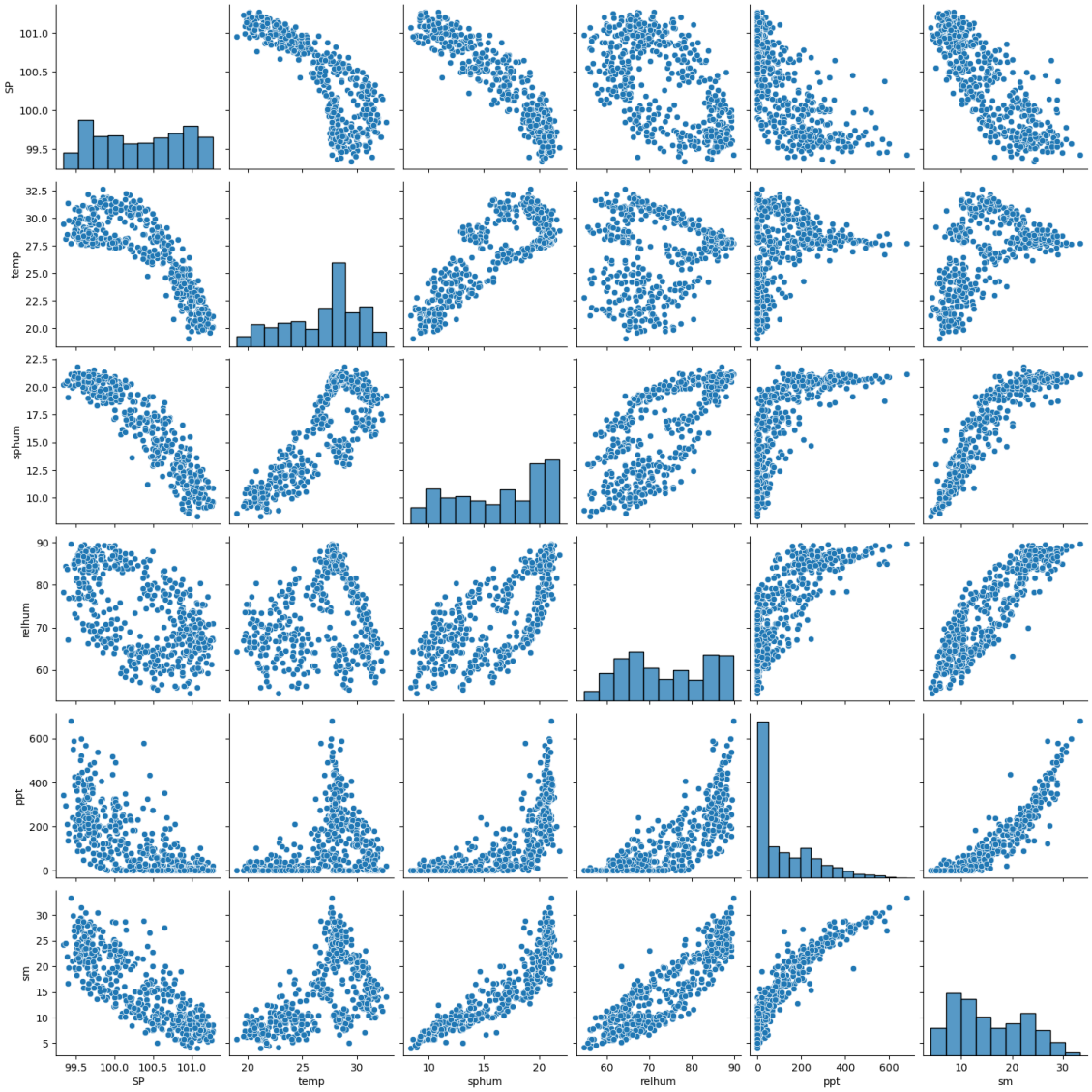


Figure 14. Relation among the independent variables before normalization

9. Conclusion and future scope of work

Lasso Regression demonstrated the strongest performance for soil moisture estimation among the models evaluated. It achieved the lowest mean absolute percentage error (MAPE), recording values of 0.19 for the training dataset and 0.17 for the testing dataset. These relatively low error rates indicate that Lasso Regression is capable of effectively capturing the underlying patterns and variability in soil moisture, even with a limited number of input features and monthly temporal resolution. This performance highlights its suitability for handling multicollinearity among predictors, which is common in environmental datasets, and for identifying the most influential variables contributing to soil moisture fluctuations.

For the flood prediction task, Logistic Regression was found to be the most effective model, outperforming other machine learning techniques applied in this study. The model achieved a Recall Score of 1, indicating that it successfully identified all positive flood cases without any false negatives—an essential

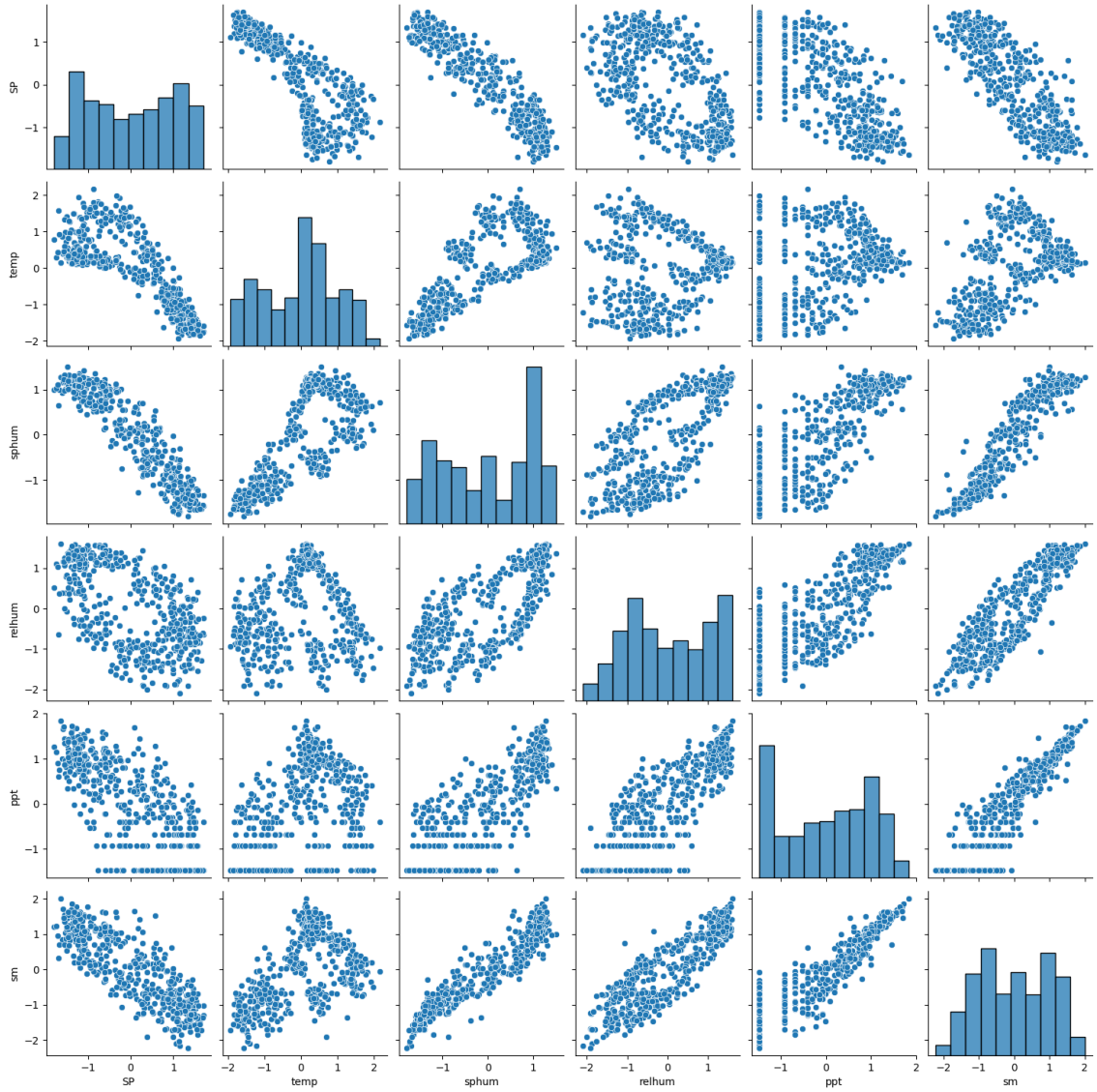


Figure 15. Relation among the independent variables after normalization

requirement in flood forecasting, where missed predictions can lead to severe consequences. Additionally, the Matthews Correlation Coefficient (MCC) value of 0.68 reflects a balanced and dependable prediction quality, considering both true and false classifications. The training and testing accuracies of 0.91 and 0.94 further demonstrate that the model generalizes well and maintains stability across unseen data. Together, these metrics suggest that Logistic Regression provides a moderate yet consistently superior predictive strength compared to the other algorithms assessed in the study.

Given its performance, Logistic Regression can be recommended as a dependable model for flood prediction based on key hydrometeorological variables, including rainfall, specific humidity, relative humidity, surface pressure, and soil moisture. These factors play a significant role in influencing flood events, and their combined contribution allows the model to capture both atmospheric and land-surface dynamics effectively.

However, the study is inherently limited by the use of monthly data. Monthly datasets provide only

Table 4. Comparison of accuracy, recall score, MCC, and confusion matrix

	Logistic Regression	Decision Tree	Random Forest	Support Vector Machine
Accuracy of training dataset	0.91	1	1	0.90
Accuracy of testing dataset	0.94	0.9	0.92	0.92
Recall Score	1	0.66	0.66	0.75
Matthew's Correlation Coefficient	0.68	0.49	0.42	0.57
Confusion Matrix	$\begin{bmatrix} 44 & 0 \\ 3 & 3 \end{bmatrix}$	$\begin{bmatrix} 41 & 3 \\ 2 & 4 \end{bmatrix}$	$\begin{bmatrix} 43 & 1 \\ 3 & 3 \end{bmatrix}$	$\begin{bmatrix} 43 & 1 \\ 3 & 3 \end{bmatrix}$

Table 5. Fitted coefficients of Logistic Regression

Feature	Coefficient
Surface Pressure	-0.91048
Temperature	-0.69352
Specific Humidity	-0.28403
Relative Humidity	0.59154
Precipitation	0.63987
Soil Moisture	2.43550

coarse temporal granularity, which may overlook short-term fluctuations and rapid environmental changes that often precede flood events. As a result, model performance remains moderate, and the predictive capabilities may not fully reflect real-time flood dynamics. Future studies could significantly benefit from the use of daily or even hourly data, which would allow models to detect more subtle patterns, improve responsiveness, and ultimately enhance prediction accuracy.

Moreover, expanding the range of input features can also strengthen the predictive framework. Variables such as cyclone occurrences, tidal effects, storm surges, river discharge levels, and land-use or land-cover changes could provide deeper insights into the mechanisms driving flood events. Incorporating such factors would contribute to a more comprehensive model capable of handling complex environmental interactions.

Increasing the dataset size is another critical avenue for improvement. A larger and more diverse dataset would allow machine learning models—particularly those sensitive to data volume—to refine their parameter estimates, reduce overfitting, and improve generalization capability. This is especially relevant for soil moisture prediction, which plays a vital role in influencing final flood outcomes. Better soil moisture modeling can directly enhance flood prediction accuracy, given its strong link to surface runoff and infiltration processes.

In summary, while the current study offers meaningful insights into the effectiveness of Lasso Regression and Logistic Regression for soil moisture estimation and flood prediction, respectively, substantial improvements can be achieved through finer-resolution data, additional environmental features, and larger datasets. These enhancements have the potential to significantly improve the reliability, robustness, and overall predictive power of future flood forecasting systems.

Author's Contributions

All authors have equally contributed in writing and reviewing the manuscript.

Data availability

The data is available on the open sources as mentioned in the article.

Disclosure statement

No potential competing interest was reported by the authors.

Conflict of interest

The authors do not declare any conflict of interest.

Funding

This research received no external funding to carry out this research.

Human and/or animal rights

The authors declare that no human participants or animals were involved in the research.

References

1.Akinsoji, A. H., Adelodun, B., Adeyi, Q., Salau, R. A., Odey, G., and Choi, K. S. (2025). Prediction of spatial-temporal flood water level in agricultural fields using advanced machine learning and deep learning approaches. *Natural Hazards*, 121(7):7915–7940.

2.Al-Juaidi, A. E., Nassar, A. M., and Al-Juaidi, O. E. (2018). Evaluation of flood susceptibility mapping using logistic regression and gis conditioning factors. *Arabian Journal of Geosciences*, 11(24):765.

3.Antwi-Agyakwa, K. T., Afenyo, M. K., and Angnuureng, D. B. (2023). Know to predict, forecast to warn: A review of flood risk prediction tools. *Water*, 15(3):427.

4.Asif, M., Kuglitsch, M. M., Pelivan, I., and Albano, R. (2025). Review and intercomparison of machine learning applications for short-term flood forecasting. *Water Resources Management*, 39(5):1971–1991.

5.Bande, S. and Shete, V. V. (2017). Smart flood disaster prediction system using iot & neural networks. In *2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon)*, pages 189–194. Ieee.

6.Brownlee, J. (2020). How to calculate precision, recall, and f-measure for imbalanced classification. *Machine learning mastery*, 1.

7.Chen, Y., Zhang, X., Yang, K., Zeng, S., and Hong, A. (2023). Modeling rules of regional flash flood susceptibility prediction using different machine learning models. *Frontiers in Earth Science*, 11:1117004.

8.Dai, W. and Cai, Z. (2021). Predicting coastal urban floods using artificial neural network: The case study of macau, china. *Applied Water Science*, 11(10):161.

9.Darabi, H., Haghighi, A. T., Mohamadi, M. A., Rashidpour, M., Ziegler, A. D., Hekmatzadeh, A. A., and Klove, B. (2020). Urban flood risk mapping using data-driven geospatial techniques for a flood-prone case area in iran. *Hydrology Research*, 51(1):127–142.

10.Deroliya, P., Ghosh, M., Mohanty, M. P., Ghosh, S., Rao, K. D., and Karmakar, S. (2022). A novel flood risk mapping approach with machine learning considering geomorphic and socio-economic vulnerability dimensions. *Science of The Total Environment*, 851:158002.

11.Directorate of Census Operations, O. (2014). *Census of India 2011 - Odisha - Series 22 - Part XII A - District Census Handbook, Puri*. Office of the Registrar General & Census Commissioner, India (ORGI).

12.DoWR_Odisha (2010). List of past flood and area damaged by flood in orissa.

13.Dtissibe, F. Y., Ari, A. A. A., Abboubakar, H., Njoya, A. N., Mohamadou, A., and Thiare, O. (2024). A comparative study of machine learning and deep learning methods for flood forecasting in the far-north region, cameroon. *Scientific African*, 23:e02053.

14.Gandhi, R. (2018). Support vector machine—introduction to machine learning algorithms. *Towards Data Science*, 7.

15.Guru, N. (2016). *Flood frequency analysis of partial duration series using soft computing techniques for Mahanadi River basin in India*. PhD thesis, National Institute of Technology Rourkela.

16.Liu, Y., Wang, Y., and Zhang, J. (2012). New machine learning algorithm: Random forest. In *Information Computing and Applications: Third International Conference, ICICA 2012, Chengde, China, September 14-16, 2012. Proceedings 3*, pages 246–252. Springer.

17.MapsofIndia (2021). Top ten flood prone areas in india.

18.Mosavi, A., Ozturk, P., and Chau, K.-w. (2018). Flood prediction using machine learning models: Literature review. *Water*, 10(11):1536.

19.Motta, M., de Castro Neto, M., and Sarmento, P. (2021). A mixed approach for urban flood prediction using machine learning and gis. *International journal of disaster risk reduction*, 56:102154.

20.Mudau, M. (2022). Flood prediction using kerala state using machine learning.

21.National Disaster Management Authority, G. o. I. (2020). Natural hazards-floods.

22.Neter, J. (1996). *Applied Linear Statistical Models*. Irwin series in statistics. Irwin.

23.Power, D. (2021). Power—data access viewer.

24.Ranstam, J. and Cook, J. (2018). Lasso regression. *Journal of British Surgery*, 105(10):1348–1348.

25.Sankaranarayanan, S., Prabhakar, M., Satish, S., Jain, P., Ramprasad, A., and Krishnan, A. (2020). Flood prediction based on weather parameters using deep learning. *Journal of Water and Climate Change*, 11(4):1766–1783.

26.Sisters, L. (2020). Matthews correlation coefficient: when to use it and when to avoid it. In *IEEE May*.

27.Tripathi, P. (2015). Flood disaster in india: an analysis of trend and preparedness. *Interdisciplinary Journal of Contemporary Research*, 2(4):91–98.

28.Wedajo, G. K., Lemma, T. D., Fufa, T., and Gamba, P. (2024). Integrating satellite images and machine learning for flood prediction and susceptibility mapping for the case of amibara, awash basin, ethiopia. *Remote Sensing*, 16(12):2163.

29.WRIS, I. (2021). India water resources information system.

30.Zalnezhad, A., Rahman, A., Nasiri, N., Haddad, K., Rahman, M. M., Vafakhah, M., Samali, B., and Ahamed, F. (2022). Artificial intelligence-based regional flood frequency analysis methods: A scoping review. *Water*, 14(17):2677.

© 2026 by the authors. Disclaimer/Publisher's Note: The content in all publications reflects the views, opinions, and data of the respective individual author(s) and contributor(s), and not those of Sphinx Scientific Press (SSP) or the editor(s). SSP and/or the editor(s) explicitly state that they are not liable for any harm to individuals or property arising from the ideas, methods, instructions, or products mentioned in the content.