International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/IJSRCSEIT

Flood Prediction Using Machine Learning

V. Hanumantha Rao[1], **Ms. V. S. Gayathri**[2]
PG Student[1], Assistant Professor[2]
Department of Computer Science, Chadalawada Ramanamma Engineering College, Andhra Pradesh, India

Article Info

Publication Issue :
Volume 8, Issue 5
September-October-2022

Page Number : 207-216

Article History
Accepted: 02 Oct 2022
Published: 13 Oct 2022

ABSTRACT

Flooding is the most common natural disaster on the planet, affecting hundreds of millions of people and causing between 6,000 and 18,000 fatalities every year – of which 20 percent are in India. Reliable early warning systems have been shown to prevent a significant fraction of fatalities and economic damage, but many people don't have access to those types of warning systems. So, we're building Flood prediction system Based on ML or AI. This advancement of the prediction system provides cost-effective solutions and better performance. In this, a prediction model is constructed using rainfall data to predict the occurrence of floods due to rainfall. The model predicts whether "flood may happen or not" based on the rainfall range for particular locations. Indian district rainfall data is used to build the prediction model. The dataset is trained with various algorithms like K-Nearest Neighbors, XGBoost etc.

Keywords : Supervised learning, Machine Learning, Floods, XGBoost algorithm and K-Nearest Neighbors

I. INTRODUCTION

Every year, India is the topmost flood-prone disaster place in the world. Mostly water logging in urban cities occurs in low-lying areas. Moreover, the increase in water logging is due to some fundamental points such as surface runoff, relative altitude, and not enough path of the water to drainage So, flood forecasting is essential at these places. In a recent year, there were many parts of countries which are prone to flood like Assam, Bihar, Goa, Odisha, Pune, Maharashtra, TamilNadu, Karnataka, Kerala, and Gujarat.

In the year 2015 rainfall, Chennai received 1049 millimeters (mm) of rainfall in November. Since 1918, 1088 mm of precipitation was the best recorded in November. Between October and December, the average rainfall in Kanchipuram district is 64 cm. It received the heaviest rainfall of 181.5 cm, which is 183% higher against average precipitation. In the Tiruvallur district, the average rainfall is 59 cm but recorded 146 cm of rain there was much research for prediction of flood ahead, but not many methods give the estimate with high accuracy. The flood prediction analysis majorly uses Machine Learning (ML). There are many methods in machine learning to predict the problem with higher accuracy. In this work, we have proposed to estimate the flash flood to prevent places that are prone to flood risk. The approach is to the establishment of the ML algorithm model. It incorporates the flood factor to estimate short term prediction in an urban area with higher accuracy.

Generally, the choice of features is crucial in image segmentation algorithms. Nath and Deb stated that one of the most promising features of digital images is color information. The other commonly used feature is texture or pattern information. Many researchers used these main features to identify flood events. For the detection of a flooding event, Lai and Chen employed threshold values to determine the potential foreground regions. Borges et al.introduced a probabilistic model for flood detection in videos. They combined the statistical characteristics of floods, such as color, texture, and saturation characteristics, using the Bayes classifier along with frame-to-frame changes to determine the flood presence. They then proposed a probabilistic model of flood occurrence to detect the position of flood regions, thereby significantly improving the classification performance. San Miguel and Ruiz Jr conducted a study similar to the work of Borges et al.using the thresholding method to segment the flood and non-flood regions depending on color, size, and patterns of ripples. This approach offers good flood detection capabilities but is limited by the reflections on the floodwater. Filonenko et al proposed the use of a color probability method from images obtained from a video to detect floods in real time. Lo et al employed the region growing method for flood region detection, finding it more suitable for situations in which the background and foreground shapes change over time. Therefore, all shape and size variations across the flood regions can be detected. The authors also proposed a region-based segmentation method of differentiating the foreground and background. Jyh-Horng et al proposed mean-shift and region growing approaches to develop an automated identification method for flood monitoring. The region growing algorithm is specifically used to differentiate an object within a binary mask.

Several flood event monitoring methods are available, such as gauge sensors and remote sensing technology However, gauge sensors can only provide one spatial dimension (water level), whereas remote sensing technology experiences issues with satellite revisit time. The retrieval of information may be delayed for hours depending on the method of data transmission Currently, flood warning analysis relies on water level sensors and precipitation forecasts; therefore, it is not capable of providing near-real-time and automated flood monitoring analysis As such, visual sensing systems have been introduced that have the capability to collect a vast quantity of information from a given area. Important information for many applications can be obtained from still images and video streams. The interest in visual monitoring and surveillance systems, especially in natural disaster applications, has increased with advancing surveillance technology. It is now possible to recognize events in real time with the progress in information technology For instance, Mettes et al worked on the motion of water properties using videos. Image-based approaches, combined with the appropriate image analysis techniques, may offer efficient and cost-effective methods that may be useful for managing flood events An early warning system for flood prevention and monitoring is one of the applications of visual sensing technology. Image processing is the use of computer algorithms to extract useful information from digital images, which is a vital procedure in visual sensing systems. To understand the content of an image, image segmentation is commonly used to partition it into several regions, which are often based on the attributes of its pixels. In particular, image segmentation has been applied in the fields of medical imaging, automated driving, and water management. In flood disaster applications, it may involve the separation of the foreground (in this case, the water features) from the background. Several image segmentation techniques are currently used by researchers and industry, such as thresholding, boundary-based, region-based, and hybrid techniques Some papers specifically discussed image segmentation methods handcrafted for flood disaster applications.

II. RELATED WORKS

A Hybrid Machine Learning Approach for Classifying Aerial Images of Flood-Hit Areas: Numerous parts of southern India have recently encountered severe damage to lives and properties due to floods. Floods are one among the most destructive natural hazard and recovering to normal life takes ample time. During hazards, various technologies are in use for speeding up relief operations and to minimize the amount of damage, one such being the use of drones. Many algorithms are in need for automatic analysis of remote sensing and aerial images. Nowadays, drones are being used for taking images from varied heights similar to aerial images, as they have cameras with exceptional features and effective sensors. This paper proposes a hybrid approach to classify whether a region in an aerial image is flood affected or not. A combination of Support Vector Machine (SVM) and k-means clustering proved capable of detecting flooded areas with good accuracy, classifying about 92% of flooded images correctly. Performance analysis is done by changing various kernel functions in SVM. The results show that there is a decrease in the prediction and training time when quadratic SVM is used.

Summary: In this paper, authors did several experiments on supervised and unsupervised machine learning algorithms and combine these two methods for the prediction of the floods

Different Techniques of Flood Forecasting and Their Applications: Flood forecasting (FF) is one of the most important, challenging problems in hydrology. The purpose of flood forecasting and warning is acknowledged as the most important non-structural term for reducing flood damage. A flood forecast system must provide sufficient lead time for communities to respond. Reliability of forecast is to provide as much advance notice as possible of an impending flood to the authorities and the general public. A forecast has increased in the modeling capabilities of hydrology and advancements in knowledge for analysis as well as improvements in data collection through satellite observations. This paper reviews different aspects of flood forecasting, including the models being used, techniques of collecting inputs and displaying their results, and warnings.

Summary: In this Paper, Ranit and durge implemented forecasting algorithms for flood forecasting in the future days.

Multiple Input Single Output (MISO) ARX and ARMAX model of flood prediction system: Case study Pahang: Yearend usually would see many states in Malaysia would probably hits by severe floods due to Monsoon rain especially in the east coast. Many peoples suffered properties damages and economic losses. Thus, an accurate flood water level prediction model is required as an alarm system that would warn the affected area and residence to prepare for evacuation due to the upcoming severe flood. This paper compared the prediction performances of a developed flood prediction models that were designed using Multiple-Input Single-Output (MISO) Auto regressive with Exogenous Input (ARX) and MISO Auto regressive Moving Average with Exogenous Input (ARMAX) structure. The models were designed using Matlab System Identification toolbox of parametric model. The location for the case of study was at Pahang River, Temerloh, Pahang with four upstream stations and one downstream station or observed location. The data used were obtained from the Malaysian Department of Drainage and Irrigation. Simulation results showed that the prediction performance of flood prediction model designed by ARMAX structure showed better Best Fit value and smaller rmse values as compared to the model designed using ARX structure.

Summary: Authors of this paper described how to implement ARMAX and ARX in Matlab and describes the performance and accuracies by changing the parameters in it.

Flood Prediction Using Multi-Layer Artificial Neural Network in Monitoring System with Rain Gauge, Water Level, Soil Moisture Sensors: Flood is one of the most destructive natural phenomena that happens on most part of the world. Notably in the Philippines, this was a major issue as it can lead to damage of properties, damage to infrastructures or even loss of lives. Current systems adhere to solve issues to prevent devastating disasters caused by floods. In this study, a system is developed to predict flood level based on real-time monitoring sensors and systems. The system predicts in advance the flood level based on the current data it gathered from sensors integrated in a real-time monitoring system. Multi-layered artificial neural network with the aid of MATLAB was used in the development of the prediction model. In the training, test, validation and overall dataset, the network showed a very good goodness-of-fit specifically 0.99889 for the training dataset, 0.99362 for the test data set, 0.99764 for the validation dataset and 0.99795 considering all the data in the dataset. The network was then programmed and integrated in the system in the actual setup. The model is validated by running trials with certain inputs and predicted flood level as the output and is compared to the actual flood level after a certain period of time. The flood prediction system showed an RMSD value of 2.2648 which signifies a small overall difference between the predicted flood level and actual flood level across the whole dataset tested in the actual setup.

Summary: This paper shows us how important is Maths in building Artificial Neural Networks from the scratch .

An Efficient Automated Hybrid Algorithm to Predict Floods in Cloud Environment: Natural and environmental sciences are one of the scientific domains which seek a lot of attention as it requires accurate real time predictions. In particular, flooding induced by heavy precipitation is one of the regular risks in Eastern Indian states. In this research work, the state of Odisha, India have been selected for predicting floods because majority of the state's districts have been exposed to floods, leading to unprecedented loss of life and property. In this paper, an optimization based feature selection Genetic Algorithm (GA) have been combined with classification algorithms to predict the occurrence of floods. The experimental results show that the GA-SVM algorithm outperforms in terms of accuracy and total execution time in comparison to other hybrid algorithms. Finally, the results are validated and compared by executing the proposed hybrid algorithm over the heterogeneous resources in Cloud environment.

Summary: G. Kaur and A. Bala compares the Genetic algorithms and hybrid algorithms in terms of both accuracies and computational time.

Developing a Flood Risk Assessment Using Support Vector Machine and Convolutional Neural Network: A Conceptual Framework: Flooding is one of the most devastating natural hazards that affect not only to infrastructures and agriculture but also to human lives. The prominent effect of global warming boasted its danger and impact in a wider range. In order to address and provide more effective measures to lessen the impact of flood hazards, it would be better to identify first the areas with such flood vulnerability. The proposed study aims to exploit the data available from the Geographical Information System (GIS) and the technology advancement in the modern world in producing a reliable flood susceptibility and probability map. Fusing ConvNet, a feedforward neural networks that specialize in image processing and prediction with SVM, a supervised machine learning for classification and regression analysis for a better image map results. Distinct image prediction output from dilated convolution and deconvolution network will be used as an input to SVM in producing its final output.

Summary: This paper describes the predicting the floods using both machine learning techniques and deep learning techniques using support vector machines and convolution neural networks.

III. Methodology

Proposed system:

In proposed system, we implement a Machine Learning algorithms for getting insights from the complex patterns in the data. This technique is computationally inexpensive because of its simple architecture.

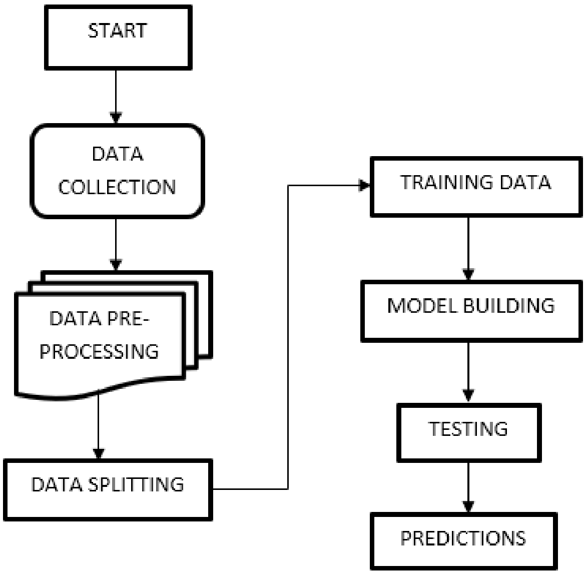


Figure 1: Block diagram of proposed method

IV. IMPLEMENTATION

The project has implemented by using below listed algorithms.

XGBoost:

XGBoost is an algorithm that has recently been dominating applied machine learning and Kaggle competitions for structured or tabular data. XGBoost is an implementation of gradient boosted decision trees designed for speed and performance.

XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. In prediction problems involving unstructured data (images, text, etc.) artificial neural networks tend to outperform all other algorithms or frameworks. However, when it comes to small-to-medium structured/tabular data, decision tree based algorithms are considered best-in-class right now.

Bagging: Now imagine instead of a single interviewer, now there is an interview panel where each interviewer has a vote. Bagging or bootstrap aggregating involves combining inputs from all interviewers for the final decision through a democratic voting process.

XGBoost and Gradient Boosting Machines (GBMs) are both ensemble tree methods that apply the principle of boosting weak learners (CARTs generally) using the gradient descent architecture. However, XGBoost improves upon the base GBM framework through systems optimization and algorithmic enhancements.

K Nearest Neighbors:

K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique.

K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.

K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K-NN algorithm.

K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

K-NN is a non-parametric algorithm, which means it does not make any assumption on underlying data.

It is also called a lazy learner algorithm because it does not learn from the training set immediately instead it stores the dataset and at the time of classification, it performs an action on the dataset.

KNN algorithm at the training phase just stores the dataset and when it gets new data, then it classifies that data into a category that is much similar to the new data.

Suppose there are two categories, i.e., Category A and Category B, and we have a new data point x1, so this data point will lie in which of these categories. To solve this type of problem, we need a K-NN algorithm. With the help of K-NN, we can easily identify the category or class of a particular dataset.

The K-NN working can be explained on the basis of the below algorithm:

Step-1: Select the number K of the neighbors

Step-2: Calculate the Euclidean distance of K number of neighbors

Step-3: Take the K nearest neighbors as per the calculated Euclidean distance.

Step-4: Among these k neighbors, count the number of the data points in each category.

Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.

Step-6: Our model is ready.

Decision Trees:

A tree has many analogies in real life, and turns out that it has influenced a wide area of machine learning, covering both classification and regression. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions. Though a commonly used tool in data mining for deriving a strategy to reach a particular goal.

A decision tree is drawn upside down with its root at the top. In the image on the left, the bold text in black represents a condition/internal node, based on which the tree splits into branches/ edges. The end of the branch that doesn’t split anymore is the decision/leaf, in this case, whether the passenger died or survived, represented as red and green text respectively.

Although, a real dataset will have a lot more features and this will just be a branch in a much bigger tree, but you can’t ignore the simplicity of this algorithm. The feature importance is clear and relations can be viewed easily. This methodology is more commonly known as learning decision tree from data and above tree is called Classification tree as the target is to classify passenger as survived or died. Regression trees are represented in the same manner, just they predict continuous values like price of a house. In general, Decision Tree algorithms are referred to as CART or Classification and Regression Trees.

So, what is actually going on in the background? Growing a tree involves deciding on which features to choose and what conditions to use for splitting, along with knowing when to stop. As a tree generally grows arbitrarily, you will need to trim it down for it to look beautiful. Let’s start with a common technique used for splitting.

Logistic Regression:

Logistic Regression was used in the biological sciences in early twentieth century. It was then used in many social science applications. Logistic Regression is used when the dependent variable (target) is categorical.

For example,

To predict whether an email is spam (1) or (0)

Whether the tumor is malignant (1) or not (0)

Consider a scenario where we need to classify whether an email is spam or not. If we use linear regression for this problem, there is a need for setting up a threshold based on which classification can be done. Say if the actual class is malignant, predicted continuous value 0.4 and the threshold value is 0.5, the data point will be classified as not malignant which can lead to serious consequence in real time.

From this example, it can be inferred that linear regression is not suitable for classification problem. Linear regression is unbounded, and this brings logistic regression into picture. Their value strictly ranges from 0 to 1.

Types of Logistic Regression

1. Binary Logistic Regression

The categorical response has only two 2 possible outcomes. Example: Spam or Not

2. Multinomial Logistic Regression

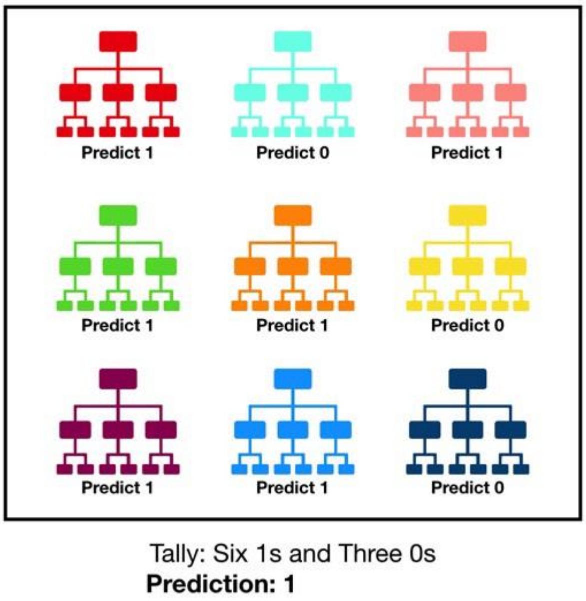
Three or more categories without ordering. Example: Predicting which food is preferred more (Veg, Non-Veg, Vegan)

3. Ordinal Logistic Regression

Three or more categories with ordering. Example: Movie rating from 1 to 5

Random Forest:

Random forest, like its name implies, consists of a large number of individual decision trees that operate as an essemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model's prediction (see figure below).



Visualization of a Random Forest Model Making a Prediction

The fundamental concept behind random forest is a simple but powerful one — the wisdom of crowds. In data science speak, the reason that the random forest model works so well is:

A large number of relatively uncorrelated models (trees) operating as a committee will outperform any of the individual constituent models.

The low correlation between models is the key. Just like how investments with low correlations (like stocks and bonds) come together to form a portfolio that is greater than the sum of its parts, uncorrelated models can produce ensemble predictions that are more accurate than any of the individual predictions. **The reason for this wonderful effect is that the trees protect each other from their individual errors** (as long as they don't constantly all err in the same direction). While some trees may be wrong, many other trees will be right, so as a group the trees are able to move in the correct direction. So the prerequisites for random forest to perform well are:

1. There needs to be some actual signal in our features so that models built using those features do better than random guessing.
2. The predictions (and therefore the errors) made by the individual trees need to have low correlations with each other.

The wonderful effects of having many uncorrelated models is such a critical concept that I want to show you an example to help it really sink in. Imagine that we are playing the following game:

- I use a uniformly distributed random number generator to produce a number.
- If the number I generate is greater than or equal to 40, you win (so you have a 60% chance of victory) and I pay you some money. If it is below 40, I win and you pay me the same amount.

- Now I offer you the the following choices. We can either:

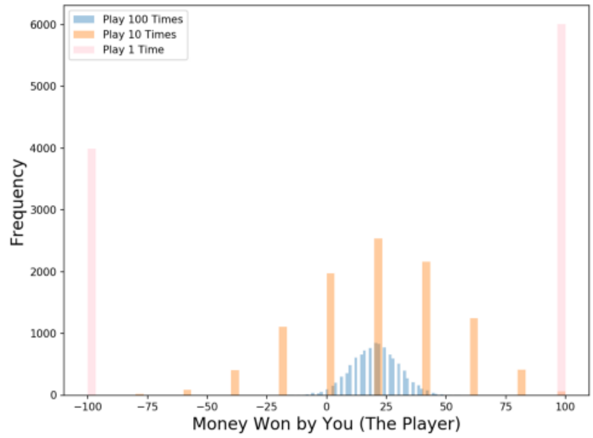
1. **Game 1** — play 100 times, betting $1 each time.
2. **Game 2**— play 10 times, betting $10 each time.
3. **Game 3**— play one time, betting $100.

Which would you pick? The expected value of each game is the same:

Expected Value Game 1 = (0.60*1 + 0.40*-1)*100 = 20

Expected Value Game 2= (0.60*10 + 0.40*-10)*10 = 20

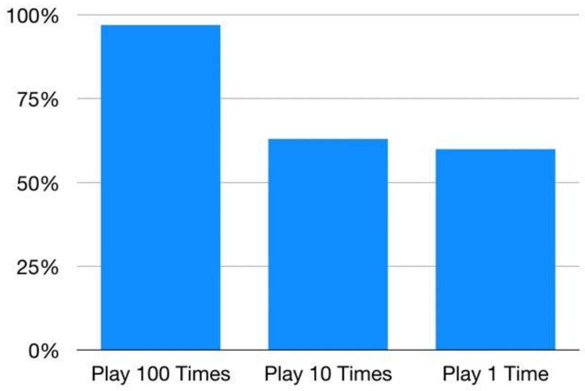
Expected Value Game 3= 0.60*100 + 0.40*-100 = 20



Outcome Distribution of 10,000 Simulations for each Game

What about the distributions? Let's visualize the results with a Monte Carlo simulation (we will run 10,000 simulations of each game type; **for example, we will simulate 10,000 times the 100 plays of Game 1**). Take a look at the chart on the left — now which game would you pick? Even though the expected values are the same, **the outcome distributions are vastly different going from positive and narrow (blue) to binary (pink).**

Game 1 (where we play 100 times) offers up the best chance of making some money — **out of the 10,000 simulations that I ran, you make money in 97% of them!** For Game 2 (where we play 10 times) you make money in 63% of the simulations, a drastic decline (and a drastic increase in your probability of losing money). And Game 3 that we only play once, you make money in 60% of the simulations, as expected.



Probability of Making Money for Each Game

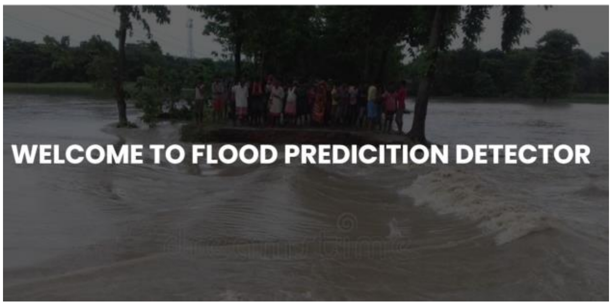
So even though the games share the same expected value, their outcome distributions are completely different. The more we split up our $100 bet into different plays, the more confident we can be that we will make money. As mentioned previously, this works because each play is independent of the other ones.

Random forest is the same — each tree is like one play in our game earlier. We just saw how our chances of making money increased the more times we played. Similarly, with a random forest model, our chances of making correct predictions increase with the number of uncorrelated trees in our model.

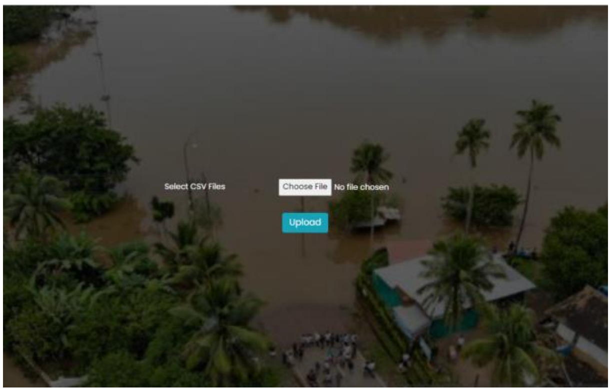
V. Results and Discussion

The following images will visually depict the process of our project.

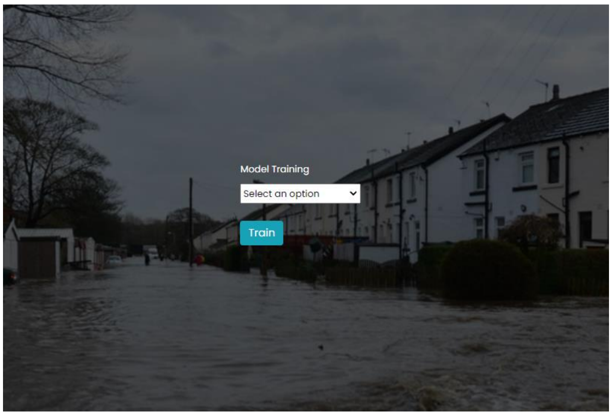
Home page: This is the home page of this project, In our project, we are detecting whether the floods will occur or not.



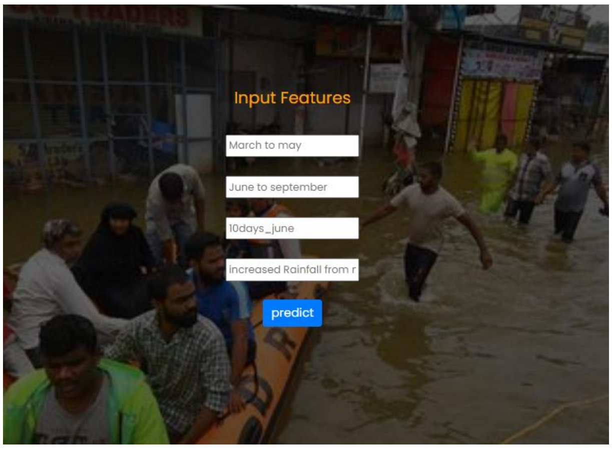
Upload: Here in this project we are uploading the dataset through which we are working.



Model Training: In training phase system generates the model from the dataset by using python modules.



Predictions: Prediction with Input Features. Here, User needs to enter input in order to detect the desire output.



VI. Conclusion

We have successfully developed a system to predict whether the floods will occur or not in this application. This is created in a user-friendly environment with Python programming and Flask. The system is likely to gather data from the user in order to predict whether there is a chance of flood occurring or not.

VII. REFERENCES

[1]. J. Akshya and P. L. K. Priyadarsini, "A Hybrid Machine Learning Approach for Classifying Aerial Images of Flood-Hit Areas," 2019 International Conference on Computational Intelligence in Data Science (ICCIDS), 2019, pp. 1-5, doi: 10.1109/ICCIDS.2019.8862138.

[2]. A. B. Ranit and P. V. Durge, "Different Techniques of Flood Forecasting and Their Applications," 2018 International Conference on Research in Intelligent and Computing in Engineering (RICE), 2018, pp. 1-3, doi: 10.1109/RICE.2018.8509058.

[3]. F. A. Ruslan, K. Haron, A. M. Samad and R. Adnan, "Multiple Input Single Output (MISO) ARX and ARMAX model of flood prediction system: Case study Pahang," 2017 IEEE 13th International Colloquium on Signal Processing & its Applications (CSPA), 2017, pp. 179-184, doi: 10.1109/CSPA.2017.8064947.

[4]. F. R. G. Cruz, M. G. Binag, M. R. G. Ga and F. A. A. Uy, "Flood Prediction Using Multi-Layer Artificial Neural Network in Monitoring System with Rain Gauge, Water Level, Soil Moisture Sensors," TENCON 2018 - 2018 IEEE Region 10 Conference, 2018, pp. 2499-2503, doi: 10.1109/TENCON.2018.8650387.

[5]. G. Kaur and A. Bala, "An Efficient Automated Hybrid Algorithm to Predict Floods in Cloud Environment," 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), 2019, pp. 1-4, doi: 10.1109/CCECE.2019.8861897.

[6]. J. M. A. Opella and A. A. Hernandez, "Developing a Flood Risk Assessment Using Support Vector Machine and Convolutional Neural Network: A Conceptual Framework," 2019 IEEE 15th International Colloquium on Signal Processing & Its Applications (CSPA), 2019, pp. 260-265, doi: 10.1109/CSPA.2019.8695980.

[7]. Qianyu Zhang., Nttha Jindapetch.,Rakkrit Duangsoithong and Dujdow Buranapanichkit., "Investigation of Image Processing based Real-time Flood Monitoring," in Proc. of the 2018 IEEE 5th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA 2018), Songkhla, Thailand, 2018.

[8]. Ryo Natsuaki and Akira Hirose, "L-Band Sar Interferometric Analysis for Flood Detection in Urban Area –A Case Study In 2015 Joso Flood, Japan," 2018.

[9]. Shijin LI., Kaikai MA., Zhou JIN., and Yuelong ZHU, "A new flood forecasting model based on SVM and boosting learning algorithms," 2016.

[10]. Sreehari E and Satyajee Srivastava, "Prediction of climate variable using Multiple Linear regression," in 2018 4th International Conference on Computing Communication and Automation (ICCCA, 2018.

[11]. Swapnil Bande and Virendra V. Shete, "Smart flood disaster prediction system using IOT & Neural Networks," in InternationalConference On Smart Technology for Smart Nation, 2017.

[12]. Tibin Mathew Thekkil and prabakaran N, "Real-time WSN Based Early Flood Detection and Control Monitoring System," in 2017 International Conference on Intelligent Computing,Instrumentation and Control Technologies (ICICICT),, 2017.

[13]. Vinothini A and Baghavathi priya A, "Survey of Machine Learning Methods for Big Data Applications," in International Conference on Computational Intelligence in Data Science, 2017.

[14]. Wahyu Sardjono and Widhilaga Gia Perdana, "The Application of Artificial Neural Network for Flood Systems Mitigation at Jakarta City," in International Conference on Information Management.

Cite this article as :

V. Hanumantha Rao, Ms. V. S. Gayathri, "Flood Prediction Using Machine Learning ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 8 Issue 5, pp. 207-216, September-October 2022.

Journal URL : https://ijsrcseit.com/CSEIT228537