RES MILITARIS
REVUE EUROPEENNE D ETUDES EUROPEAN JOURNAL OF MILITARY STUDIES
Social Science Journal

A Predictive Model for Occurrence of Floods Using Machine Learning Techniques

By

Agnibha Sarkar
Vishwakarma Institute of Technology, Pune, India
Email: agnibha10@gmail.com

Dr. Ashutosh M. Kulkarni
Vishwakarma Institute of Technology, Pune, India
Email: ashutosh.kulkarni@vit.edu

Manish R. Khodaskar
SCTR'S Pune Institute of Computer Technology
Email: manishkhodaskar2006@gmail.com

Shubhangi Pandurang Tidake
Symbiosis Skills and Professional University, Kiwale, Pune, India
Email: shubhangi.tidake@sspu.ac.in

Rahul B. Diwate
Email: diwate.rahul@gmail.com

Abstract

Climate change is now a reality, aided by the continued and increased release of greenhouse gases such as carbon dioxide, methane, etc., due to global warming. For the flood-prone state of India, which is one of the most important catastrophic events, it aims to develop an early-warning system to reduce the risk due to floods. During the previous two decades, machine-learning (ML) methods to mimic the complex mathematical expressions of the physical processes of Flooding made a major contribution to the creation of prediction systems that offered better performance and reasonable solutions. However, by using fresh ML techniques and incorporating presently existing methods in this paper, the researchers aim to find more accurate and efficient predictive models. As a result, this paper presented the most comprehensive flood estimation methods for both long-term and short-term floods. The main objective of this paper is to improve a forecasting model for flood occurrence using an algorithm in Python programming language and display the obtained results in the form of a web application. In the future, there are possibilities for further investigation in this field to find out how much climate change will increase, and the intensity and frequency of catastrophic weather events including floods, cyclones, droughts, heat waves, etc.

Keywords: Climate Change, Data Model, Floods, Logistics, Machine Learning, Python.

Introduction

India being one of the Southeast Asian countries, lies mostly in the tropics. Not just India, all countries of Southeast Asia come under the category of tropical countries. Being a tropical country, heavy rainfall is a common occurrence in many parts of the country, throughout the year [1]. One such state in India, which receives huge quantities of rainfall and a humid tropical wet climate is Kerala, which often leads to floods and increases the

suffering of people living in the state because of the above, the disaster managers or decision makers or planners, badly require an early-warning predictive model which can work on the huge amount of observed data, to be able to identify the rainfall patterns, seasonally as well as annually, and to correlate it to the occurrence of floods for that particular year [2].

Some statistics are based on the 1901-2018-time frame for Indian rainfall and floods: India's highest estimated annual rainfall was recorded in 1961 and this was due to the nation's involvement with many cyclones in 1961. This year, Pune encountered flooding. Which is remembered as the 'PanshetFlood' [3]. The Panshet-dam developed cracks and bursts, which led to massive flooding, causing an estimated 1000 people to lose their lives. The years 1965-66 were the identical years of drought, which led to the scarcity of food and famine in the country. This led to the advent of the green-revolution in the country, production it a food-surplus nation in the years to come [4]. To avoid major damage to life and property during such natural calamities, having a model which can forecast the intensity of a calamity, long beforehand, becomes extremely useful, saving the masses from huge losses [5]. The red line in the 'Annual Rainfall in India 1901-2018' line graph, is the rainfall in India's 10-year price movement. It is evident because India's annual rainfall has more or less decreased since about the 1960s [6]. Due to global warming being one of the prime factors, the period of monsoon has also shortened and the rainfall pattern can be concluded to be erratic [7].

Literature Review

Various researchers have analysed and implemented different models and techniques, to not only predict floods but also, other natural calamities like cyclones, earthquakes, landslides, etc. which are also in a way affected due to the changes in climate and environment around us [8]. One such model, that is, the OFAS (Operational Flood Alarming System) model has been put to use in one of the systems which help to manage and predict floods. Five environmental parameters are used as input, namely: water level, temperature, precipitation, wind speed, and humidity. Human intervention plays a crucial role in changing climate through the enhanced release of greenhouse gases viz. carbon dioxide, methane, etc. [9] which leads to an increase in the intensity and frequency of cyclones in North Indian Oceans besides other oceans of the world [10]. The Theorem of the Double-Layer Multi-objective Probability Model and Multivariate-Compound-Extreme-Value-Distribution (MCEVD) makes two improvements as compared to traditional prediction methods for cyclones; by considering the annual occurring rate of cyclones, which increases its accuracy; and by describing the association midst diverse cyclone characteristic factors reasonably well. The authors in [11] stated that integrating the disaster-management assembly in India with climate adjustment tactics would make the plan as a whole, more effective. For efficiently managing natural disasters, the public should be made more aware of how their actions like indiscriminate use of fossil fuels, etc. affect the environment. More conferences, agreements, and meetings should be held to build a positive public outlook which would help considerably to mitigate the problems of global warming and climate change.

The intelligent flood disaster forecast system presented in [12], helps in monitoring or updating environmental limitations and forecasting floods, as compared to conventional attitudes [13]. SVM and k-means clustering approaches were used by the authors in this study for classifying flood-hit reasons, yielding a very good accuracy of 92%. It was also observed that the more the number of training samples, the better the accuracy. The prediction model in [14], uses the structure of neural networks and the image data sets to correctly predict floods. CNN (Convolutional Neural network) is applied to the images and the model then

extracts all the features from them. MPSO (Modified Particle Swarm Optimization algorithm) is also applied further to get the optimized parameters [15].

Landslide susceptibility has also been mapped using a Machine Learning approach with the conclusion being that the use of ML technologies can predict disastrous landslides with greater ease. Concerning the forecast of earthquakes, CNN is also being utilized in recent years. In, factors were identified those cause landslides, and the techniques that can be used to predict landslides in other regions, outside the domain of the areas studied [16]. Landslide prediction models, therefore, function as a tool for disaster-management agencies to be able to successfully implement evacuation systems for reducing loss of property, in the case of a major landslide. In the paper [17], machine learning and data mining have been collective to idea an earthquake expectation structure, based on a traffic-system. Parameters of three models, namely, KNN (k-nearest neighbours), Logistic Regression, and Decision Trees were adjusted to see if they could further improve model output after training. The next paper [18], further, takes a particular major earthquake event prediction task and compares the accuracies of various models. Each machine learning method yields answers that are distinctive from one another. The three approaches that provide the minimum positive results are KNN, Random Forest whereas SVM, KNN, MLP, and Random Forest classify higher numbers of output values correctly.

The next surveyed paper [19], analyses a different type of calamity, i.e. forest fires, by presenting a risk prediction mechanism for forest fires, based on meteorological data only. This paper studies an algorithm for the classification of fire-risk, constructed on the number of fire events that have happened in the past, about certain weather conditions [20]. Further reveals that the maximum imperative feature distressing the burnt areas is temperature. The results obtained from the two methodologies (LLC-Logistic Level Count and KMC-K-Means Clustering) utilized in this study, agreed with the outcomes derived from using SVM (Support Vector Machine).

On a completely different note, just to analyse other algorithms, the next paper was surveyed [21], to anticipate the loss in revenue for a certain construction process, and a typical was constructed by using a DL-algorithm. The model which was developed was later confirmed by calculating the RMSE (Root Mean Square Error) and the MAE (Mean Absolute Error) values. On a further different note, the paper suggests that decarbonisation of transport is a necessity for maintaining a low-carbon society, with machine learning having a major role to play in various applications [22]. One of the examples which were cited includes the case of image recognition and how it could help law enforcement agencies to detect the overloading of trucks. Machine learning techniques may also be used for research and development of alternative fuels in the future.

A model has been developed for weather prediction using ensemble learning and other extrapolative like Random Forest, Linear Discriminate Analysis, Generalized Boosted Algorithm, and Support Vector Machines which provided better performance without much addition to the earlier costs [23]. On the other hand, the authors worked on the prediction of sandstorms and analyzed data for building three sandstorm prediction models using the popular data mining techniques of CART (Classification And Regression Trees), NB (Naive Bayes), and LR (Logistic Regression) [24]. The predictions were then displayed on a web application in real-time as well as up to 24 hours in advance. In the drought, the prediction problem was analysed by the authors consuming a deep-learning-based approach [25]. A DBN (Deep Belief Network) was proposed for predicting droughts on a long-term basis. Its

performance was then compared to that of standard MLP (Multi-Layer Perceptron) and SVR (Support Vector Regression) models. The DBN model provided better prediction results as compared to the MLP and therefore was concluded to be more efficient and accurate. In the previous paper papers, the authors dealt with future climate change scenarios, concerning rainfall, maximum temperature, and minimum temperature using statistical downscaling based on the Long Ashton Research Station-Weather Generator (LARS-WG) model for the regions of Gujarat [26]. Table 1 shows the summary of learning techniques, algorithms, and datasets discussed in the literature review.

TABLE 1: *Summary of Learning Techniques, Algorithms, And Data Sets Discussed In The Literature Review*

S.R. No.	Learning Techniques & Algorithms Used	Datasets Used	Summary	Limitations and Challenges
1. Swarup Mandal, Debashis Saha, Torsha Banerjee (2005) [8]	For running simulations, Neuro Solutions version 4.10 has been used. Using a training data set, a MultiLayerPerceptron (MLP) system was built and trained. The probability and non-occurrence of floods are projected using an ANN (ArtificialNeuralNetwork) model.	From the database, 50 flood events and 50 flood-free events were chosen.	The OFAS model is used to predict and manage floods at the earliest. Five environmental parameters are a charity as input, namely: water level, temperature, rainfall, wind speed, and humidity.	The key parameter for predicting floods in this study is water-level with the temperature being the least significant.
2. William W. Kellogg and Stephen H. Schneider (1978) [9]	-NA-	Worldwide statistics of CO2 release, aerosols, other trace gases, etc.	Climate changes as a result of global air/water pollution, and the main issues causing them were discussed in this paper. Predictions for up to the next 20-25 years were also stated (from the time of its publishing).	Whether the strategies implemented would be feasible, energy-consumption-wise; Whether agriculture would benefit from the changes and changes in sea level, (their impact on ice-caps and glaciers).
3. Liang Pang, Jiyi Zhou, and Defu Liu (2011) [10]	A double-layer, multi-objective, multivariate compounding extreme value distribution theorem	21-year typhoon data along the coast of Storm surge documentation for China was acquired from the China Typhoon Yearbooks (CTY) and several hydrology stations.	MCEVD has two improvements, first, by considering the annual occurring rate of typhoons It lowers the inefficiencies brought on by sampled uncertainty. Second, the multivariate model does a reasonable job of describing the relationship between the many typhoon-defining characteristics.	The MCEVD-based double-layer multi-objective probabilities ideal is a reliable process for determining the long-term likelihood of storm surge disasters spurred on by a typhoon.
4. Adil Usman (2017) [11]	-NA-	Top 10 Countries' Share of CO2 Emission in 2015,	Highlighted the structural design and circumstances present now in India. The	Challenges include educating people about activities that

		Trend in Top 10 Countries' Per Capita CO2 Emission for 2015.	purpose of this study was to unite the two independent active parts on a single interaction platform.	can harm the planet which would ultimately lead to calamities. Conferences, meetings, and negotiations related to climate change mitigation should also be aware of how it ties in with emergency management.
5. Swapnil Bande and Prof. Dr. Virendra V. Shete (2017) [12]	• Adaptive Learning Algorithm involving Gradient Descent • NARX network using Levenberg-Marquardt training algorithm (Nonlinear-Autoregressive-Network with EX-ogenous inputs)	• Data is sent to the cloud server by a single IoT node. • Spreadsheet files containing the gathered data are sent as input to MATLAB.	The proposed model monitors and predicts floods in real-time. To control the information generated by a sensor network, artificial neural networks' unreliability and the Internet of Things' scalability are integrated. Early flood forecasting is made thanks to effective interpersonal between these two parties.	The Levenberg-Marquardt training algorithm with the NARX-network gives enhanced outcomes and offers real-time flood predictions, as compared to other algorithms.
6. Akshya .J, P.L.K. Priyadarsini (2019) [13]	• Bag of Visual Words (BOVW) • The K-means algorithm • The SVM classifier • Kernel functions	A total of 200 aerial photos make up the training dataset.	SVM and k-means clustering approaches were used for classifying flood-hit areas, yielding a very good accuracy of 92%. It was also observed that the added the number of training sections, the better the accuracy.	The aspects of academics reduced training and prediction accuracy when PCA is used. By applying this classification algorithm to drone sensors, which might automate the classification and classification of flooded regions, the suggested model's upcoming projects can always be enhanced.
7. Purva Mohan Padmawar et al. (2019) [14]	CNN model, SVM, KNN, Modified Particle Swarm Optimization (MPSO) algorithm.	Testing samples have 500 photos, whereas training samples include 137 images.	The flood may be expected from the photographs that used the suggested prediction system, which uses a neural network structure. The model in this system employs CNN in the photos and retrieves all of their information. MPSO is	The accuracy of the model might well be impacted by changes in the number of epochs. The model's accuracy will be determined by the shape descriptors and their many

			also used by the algorithm to provide improved parameters.	kinds. As opposed to constrained ones, a variety of schemes and many patches will achieve the best outcomes.
8. Amit Juyal and Sachin Sharma (2021) [15]	Artificial Neural Network (ANN), Logistic-model tree (LMT), Random Forests (RF), Classification and Regression tree (CART), Least Square Support Vector Machine (LSSVM), and other various types of Support Vector Machines (SVM), Hodrick-Prescott decomposition, Double Exponential Smoothing (DES) method.	historical accounts, assessments of aerial pictures, and comprehensive experimental studies of landslides regions	ML technologies can be utilized to predict disappearances.	It was concluded that using a combination of research methods (ensembles) was more advantageous than using only one ML methodology for LSM. In recent years, CNN has concentrated largely on time series forecasting.
9. C. N. Madawala et al. (2019) [16]	Naïve Bayes Classifier, SupportVector Machine, the novel classifier ensemble model.	Monthly rainfall and temperature data (2012-2017).	The possible causes of landslides were established. The techniques and findings could be utilized to forecast landslides outside of the sampling locations.	Land-slide warning systems can be traditional by predicting those landslides caused by excessive rainwater, beforehand.
10. Wanjiang Han, Yuanlin Gan, Shuwen Chen and Xiaoxiang Wang (2020) [17]	K-nearest neighbour, logistic-regression, SVM, naive Bayes-algorithm, and judgment tree algorithm.	120 earthquakes have been recorded in China during the previous 20 years, with associated consequences to the transportation network and calamities.	An earthquake catastrophe prediction clustering algorithm on a traffic system is constructed by combining data mining and machine-learning technology.	To investigate whether these changes could enhance the findings of model training anymore, the three following parameters of the three models were shifted: • KNN algorithm improvement. • Logistic regression algorithm improvements. • Decision tree algorithm improvement.
11. Roxane Mallouhy et al. (2019) [18]	Logistic-Regression (LR), AdaBoost,	A single time series information set, with the inaugural reading	Every machine learning algorithm yields conclusions that are unique	Future work entails constructing casestudies based on

	Random-Forest (RF), Multilayer Perceptron, Support-Vector-Machine (SVM), Naïve-Bayes (NB), KNN, CART.	in 1967 and the latest in 2003, was obtained from an earthquake datacentre in Northern California.	among themselves. While SVM, KNN, MLP, and Random Forest appropriately classify more output, KNN, Random Forest, and MLP are the best at delivering the least false output (FP).	current intercession data and giving characteristic selection methodologies more thinking.
12. George E. Sakr, Imad H. Elhajj, George Mitri and Uchechukwu C. Wejinya (2010) [19]	Multiple-Regression (MR), Decision-Tree, Random-Forest, Neural-Networks, and Support Vector Machines are five various data mining methods.	Lebanese Agricultural Research Institute (LARI) weather information for the whole country of Lebanon (2000-2008).	Based only on weather information, a system for anticipating the danger of forest fires has just been described. With a very amazing precision of up to 96%, supporting vector machines may be used to forecast the risk of a fire in two classes. The combination of SVMs with a Gaussian kernel function produced the greatest results.	To organize fire risk into four different categories grounded on the ancient regularity of ardours and specific environmental circumstances, this article describes the prediction issue.
13. Hanchao Li et al. (2018) [20]	Multiple Linear Regression; Decision Tree; Logistic Level Count; K-mean clustering algorithm.	They collected the data set they needed from the website. "http://archive.ics.uci.edu/ml/datasets/Forest+Fires"	Temperature is the most important feature of this study. The conclusions of the two techniques are generally consistent with the SVM-derived conclusions.	The fact that each approach was only just once and on a single measurement may have been a drawback for all the experiments done and the conclusions reached.
14. Kim, J. M., Bae, J., Son, S., Son, K., & Yum, S. G (2021) [21]	Deep-Learning Algorithm Model.	Financial victims gained at construction sites (1999-2018); the total number is 1930.	A deep-learning-algorithm was every day to develop the model which could predict losses in the finance of a particular construction site. The result for this model was later verified by comparing its result with other models.	This model will contribute to reducing the number of monetary losses needed for the business. Through ongoing information gathering and future trans with new models, it may be transformed into a more consistent model.
15. David Rolnick et al. (2019) [22]	-NA-	Various use cases, like electric vehicles, electrical systems, etc.	Decarbonisation of transport is a necessity for maintaining a low-carbon society, with machine learning having a major role to play in various applications.	One of the examples which were cited includes the case of image recognition and how it could help law enforcement agencies to detect the

				overloading of trucks. Machine learning techniques may also be used for research and development of alternative fuels in the future.
16. N. Sravanthi et al. (2020) [23]	Ensemble and predictive algorithms such as Support vector machines, Linear Discriminate Analysis, Random Forest, and Generalized Boosted algorithms.	Parameters: temp., stickiness, pressure, precipitation, etc Values are based on Guntur, Andhra Pradesh, and the rest of the dataset values were taken from past data of the city's climatic conditions.	This study has established a weather statistical method that may be used to improve the performance without imposing excessive extra costs and to lower forecasting variables.	Although meteorologists and forecasters can estimate the weather and possible improvements, the weather is still unexpected.
17. Hadil Ahmed Shaiba et al. (2018) [24]	Logistic Regression (LR); Naive-Bayes (NB); Classification and Regression Trees (CART).	The following weather channel website is where the data was obtained. https://www.wunderground.com/. There is a "weather data" component in the computer language that is obtainable from either the weather network.	The authors worked on the prediction of sandstorms and analysed data for building three sandstorm prediction models using the popular data mining techniques of CART, NB, and LR. The predictions were then displayed on a web application in real-time as well as up to 24 hours in advance.	The plan is to encompass investigates by adding more cities and trying other data mining practises to enhance the predictions. The user is also to be alerted about the duration and occurrence of sandstorms, as early as possible.
18. Norbert A Agana & Abdollah Homaifar (2017) [25]	Artificial Neural Networks (ANNs), Restricted Boltzmann Machines (RBMs), Contrastive Divergence algorithms, and Deep Belief Networks (DBNs).	The Gunnison River Basin functions as the site of something like the case study (Upper Colorado River Basin). Training data involves years from 1912 to 1992. Test data involves years from 1993 to 2013.	It was established that the DBN model offers good forecasting outcomes, records lesser prediction inaccuracies than the MLP model, and is consequently more consistent and successful.	Due to the lack of sufficiently large sample numbers to completely use the deep architecture of the DBN model, improvements over the SVR were less meaningful.
19. Jayanta Sarkar, J. R. Chicholikar, and L. S. Rathore (2015) [26]	A pseudo-random number synthesizer called LARS-WG is used to simulate environmental data at a single location for both the present and the future.	The data series used in this study is from 1969 to 2013 from three locations, namely Bhuj, Kandla, and Naliya of Kutch district, and obtained from IMD, Ahmedabad.	The authors dealt with future climate change scenarios, concerning rainfall, maximum temperature, and minimum temperature using statistical downscaling based on the LARS-WG (Long Ashton Research Station-Weather Generator) model for the	Though there is very little uncertainty concerning future temperature prediction; however, for future rainfall prediction there are always more uncertainties. There is always

			regions of Gujarat.	scope for development of more refined statistical downscaling models for better results.
20. Jayanta Sarkar and J.R. Chicholikar (2015) [27]	LARS-WG is a pseudo-random number generator that simulates the weather at a single place simultaneously in the current and the future.	Historical base daily weather data for a period of 45 years (1969-2013) was used to generate long-term (2014-2063) synthetic meteorological data for picked stations on maximum and maximum temperatures together with rainfall.	The authors dealt with future climate change scenarios, concerning rainfall, maximum temperature, and minimum temperature using LARS-WG (Long Ashton Research Station-Weather Generator) model-based statistical path that leads to the Gujarati areas.	Scope for development of more refined statistical downscaling models for better results.

Datasets Used

For exploratory data analysis, 115 years of regularly, and annual-rainfall data of India were taken, from 1901 to 2015, which consists of a total of 19 columns from Kaggle.

For building the predictive model, the data set of Kerala, from the years 1901 to 2018 was exclusively selected, this time, containing a total of 20 columns, with one additional column indicating the occurrence of floods in Kerala for that particular year from Kaggle.

Methodology

For understanding the datasets better, the data was first imported, processed, and analysed for finding out occurrences of missing data, and also to gain an overall insight. The summary of the overall datasets was obtained using the 'streamline' and 'pandas-profiling' modules in Python, which was then displayed on a web application. The missing values were worked upon, and the datasets were then visualized using the required Python visualization libraries (as explained further). After the exploratory data analysis was completed, five algorithms were fitted to the training data of the Kerala rainfall dataset, and then tested, for evaluating and comparing their accuracy scores. The classifier/algorithm with the highest accuracy score was then selected for further implementation in a web application, where it could then take input values from the user based on some decided parameters, and predict the possibility of the occurrence of floods for that particular year.

Exploratory Data Analysis

Importing and preprocessing data

The necessary modules were first imported into a Python file. Some of them include 'NumPy', 'pandas', 'matplotlib', 'seaborn', 'PIL' (Python Imaging Library), 'scikit-learn' etc. After importing the modules, the main dataset was imported using the 'pandas' command (pd.read_csv) from a CSV (Comma-Separated Values) file, which contains 115 years of Indian rainfall data. On printing out the summary of the dataset imported, the following

statistics were obtained: Rows: 4116, Columns: 19, Features : "SUBDIVISION, 'YEAR', 'JAN', 'FEB', 'MAR', 'APR', 'MAY', 'JUN', 'JUL', 'AUG', 'SEP', 'OCT', 'NOV', 'DEC', 'ANNUAL', 'Jan-Feb', 'Mar-May', 'Jun-Sep', 'Oct-Dec'", Missing values: 134. The means of the respective columns have been employed to fill in the gaps left by that of the missing data.

Visualizing data:

Using appropriate, module-specific Python commands, the required forms of visualization of data have been obtained (here, in the form of line-plots). The rainfall patterns in India over a time period of 115 years have been analyzed, annually as well as seasonally. The rainfall patterns, both annual as well as seasonal, are erratic as mention in Fig. 1 and Fig. 2 in nature.

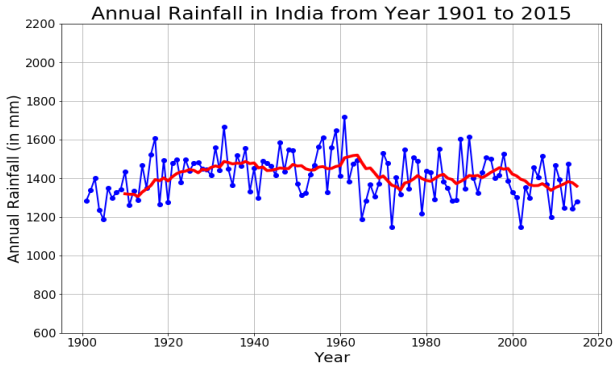

Fig. 1: *A very erratic line plot depicting the Annual Rainfall in India (1901-2015), X-axis: Year, Y-axis: Annual rainfall values (in mm)*

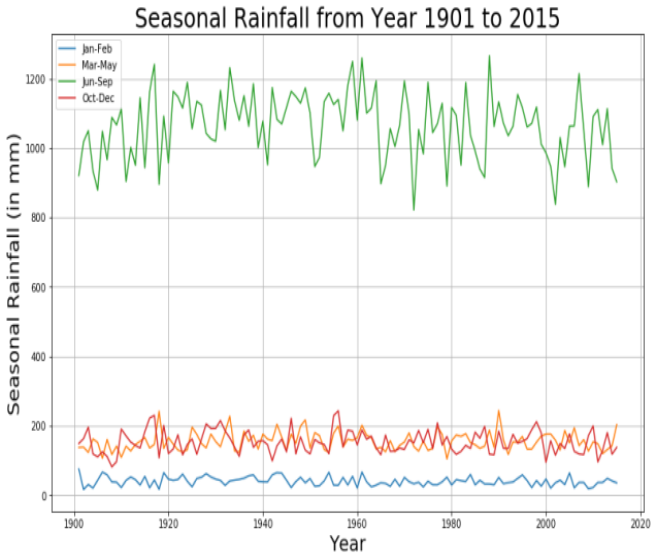

Fig. 2: *An equally erratic line plot depicting the Seasonal Rainfall in India (1901-2015), X-axis: Year, Y-axis: Monthly rainfall values (in mm)*

The monthly rainfall pattern across India has also been presented in the form of a bar chart which mention in Fig. 3, which is using the 'matplotlib' library in Python.

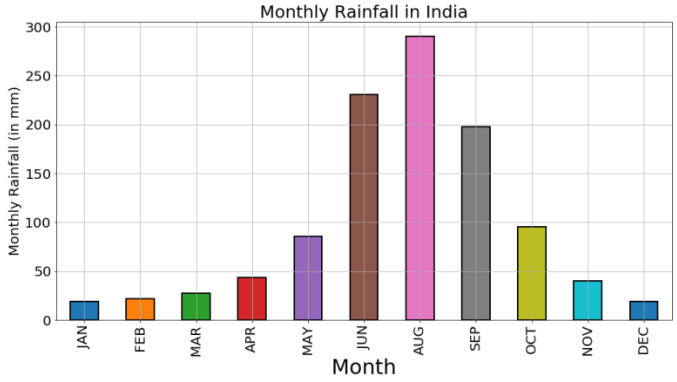

Fig. 3: *A bar plot showing the Monthly Rainfall values in India (1901-2015), X-axis: Month, Y-axis: Monthly rainfall values (in mm)*

A. Comparing Prediction Algorithms

B. Importing and preprocessing data

Using the 'pandas' command (pd.read_csv), 118 years (1901-2018) of rainfall and flood occurrence data of Kerala was imported. There was no missing data in this case, as this dataset is a subset of the already existing and transformed 115-year-varying Indian rainfall dataset, used in Exploratory Data Analysis, with the addition of three years (from 2016-2018). The 'FLOODS' column consists of Yes/No values, indicating whether a flood occurred in the state of Kerala, for that particular year. As it is not feasible to make calculations for the model/algorithm using boolean values, the Yes/No values were converted into 1s/0s respectively, using a one-liner Python command, data['FLOODS'].replace(['YES', 'NO'],[1,0],inplace=True).

C. Implementing and comparing the algorithms

5 algorithms have been chosen for implementation on the dataset. These include:

i. KNN-Classifier,
ii. Logistic Regression,
iii. Decision Tree Classification,
iv. Random Forest Classification,
v. Ensemble Learning (this technique was applied at the end, which combined the KNN, Logistic Regression, and Random Forest classifiers to generate an improved accuracy score).

D. KNN Classifier

A sample is categorised based on its 'k' closest neighbours. The class name given to the new pattern is the majority class of these 'k' closest neighbours.

Euclidean $\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$ (1)

Manhattan: $\sqrt{\sum_{i=1}^{k}|x_i - y_i|}$ (2)

E. Logistic Regression (LR)

It is the process of modelling the probability of a discrete outcome when input variables are provided. Logistic regression generally models a binary outcome, which can

take two values such as yes/no, true/false, and so on.

F. Decision Tree Classification (DT)

Classification models are created by building individual decision trees. Nodes in a decision tree evaluate the importance of various attributes, while edges and branches that are consistent with the findings of tests link to further leaflets and nodes; and leaf nodes, which predict the outcome by representing the class label. Leaf nodes are also called terminal nodes.

G. Random Forest Classification (RF)

It is a supervised machine learning classification algorithm. For it to work, decision trees are built on various samples, after which it considers the majority vote (classification case), or mean of the outcomes (regression case).

H. Ensemble Learning (EL)

Those algorithms which combine the predictions from two or more models fall under ensemble learning. This is done to improve all the performance of the model. In this revision, the three classifiers, namely Logistic Regression, Random Forest Classifier, and KNeighbors Classifier were combined to obtain an improved accuracy value. There are numerous ensemble methods using which predictive models can be built, but the 3 most significant methods include boosting, bagging, and stacking. The accuracy, recall, and ROC (Receiver Operating Characteristic) scores of the 4 individual classifiers (before the implementation of the Ensemble Learning technique) were calculated, and the scores were recorded in the form of Table 2; as given below.

Table 2: *Comparison of Individual Classifier Scores, Excluding Ensemble Learning*

	Accuracy	**Recall**	**ROC**
CNN	0.7083	0.6667	0.7083
Logistic Regression	0.9583	0.8750	0.9375
Decision Tree	0.6250	0.6250	0.6250
Random Forest	0.6666	0.7500	0.6875
Best Score	Logistic Regression	Logistic Regression	Logistic Regression

The Ensemble Learning classifier was the last to be applied as it combined 3 classifiers (KNN, LR, and RF) which generated a higher and improved accuracy score (the only exception being LR, where the score went down from 0.9583 to 0.9167 for a particular test-run). The algorithm generating the highest accuracy/score would be further implemented in the machine learning web application, as explained in the next section. The required libraries for each algorithm were imported, and the models were instantiated and fit the training data. Their accuracies were measured using the test data. On comparing all 5 algorithms, Logistic Regression emerged as the most accurate of all, with an accuracy/score of 0.9583 (before applying EL) and 0.9167 (after applying EL). The 'seaborn' library was used for plotting the accuracies of all the classifiers in the form of a bar chart, which is mention in Fig. 4.

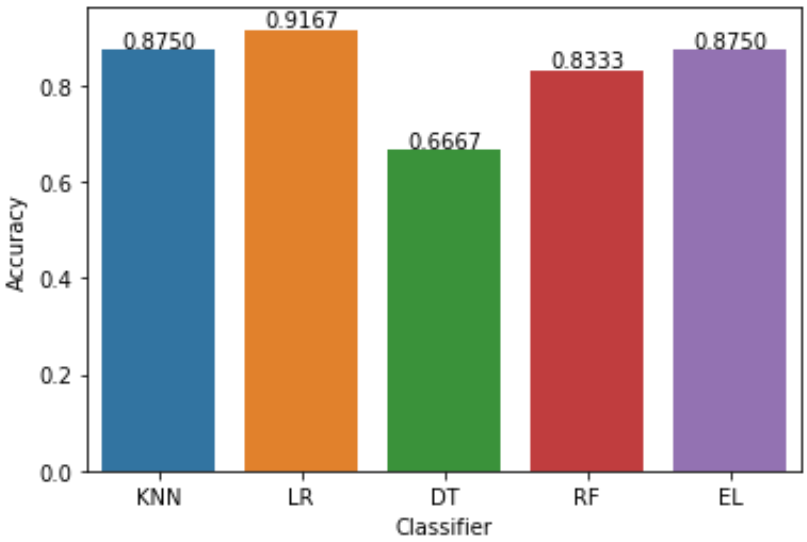

Fig. 4: *Comparison of all 5 algorithms/classifiers using a multi-colored bar chart, Logistic Regression (in Orange) has the highest accuracy score of 0.9167 (obtained after applying Ensemble Learning)*

I. Flood Prediction Model (using Logistic Regression)

J. Importing, preprocessing, and fitting data:

The Kerala dataset was imported initially. For the model, 3 additional features were required (collected and transformed from the original dataset itself) apart from the 'Annual Rainfall' feature, which was then fed to the model as training data. These include i. the 'Mar-May' column (total rainfall values from March to May), ii. The 'avg_jun' column (containing rainfall data for every 10 days in June from 1901 to 2018), iii. The 'sub' column' (containing the increase in values of rainfall from May till June 1901 to 2018), and iv. The 'ANNUAL RAINFALL' column (annual rainfall values in Kerala throughout the year, from 1901 to 2018) as displayed in Fig. 5. These features are also the input features, which would be entered into the web application by the user, for finding out the prediction of whether severe floods may occur in that particular year, or not.

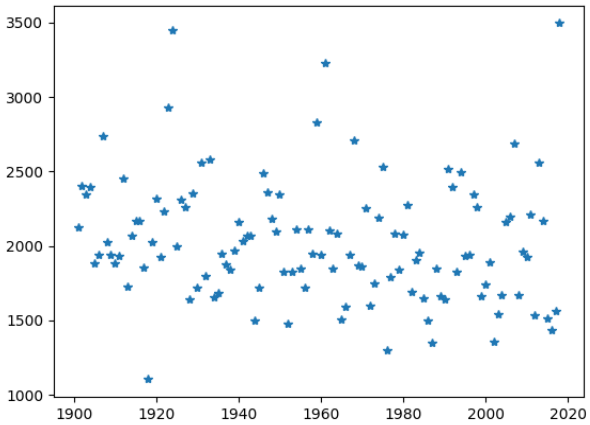

Fig. 5: *Scatter plot, Year (X-axis) v/s June-September Rainfall Value (1901-2018) (Y-axis)*

The 'FLOODS' column was selected as the target variable, which then finally could be fit into the Logistic Regression model (as every model requires both, the input and target variables).

K. Deploying the model using Flask API

2 files (model.py and app.py) were created to integrate the machine learning model with a web application, using Flask, which is a very light web framework. The development and training of the model were done in the 'model.py' file, and in the 'app.py' file, POST requests were handled, and the final prediction result was returned.

i. model.py: The libraries were imported ('numpy' and 'pandas' for manipulating matrices and data respectively, 'sklearn.model_selection' for splitting data into train and test set, 'sklearn.linear_model' to train the model using Logistic Regression and 'pickle' to save the trained model to the disk. Pickle is used for serializing and de-serializing a Python object structure (a Python object is converted into the byte stream. The dump () method dumps the object into the file specified in the arguments).

The object is instantiated as 'lr' of class LogisticRegression() and qualified using X_train and y_train (in this case, X_train = X, y_train = y1).

```
lr = LogisticRegression()
lr.fit(X, y1)
```

The model needs to be saved so that it can be used by the server. So the object 'lr' was saved to the file named 'model.pkl'.

```
pickle.dump(lr,open('model.pkl','wb'))
```

ii. app.py: The instance of the Flask () API was created and loaded the 'model.pkl' file into the model variable.A separate 'templates' directory containing the 'index.html' file was created, to render the homepage of the web-app.

```
from flask import Flask, request, render_template
@app.route('/')
def home ():
return render_template('index.html')
```

According to the Fig. 6; the 'index.html' file contains the text placeholders for the user to enter the 4 input features, along with the 'Predict' button. On hitting the button, the result is displayed to the user, in the form of a binary output value, on whether the chances of floods occurring are severe, or none. It is displayed on a different URL, with the '/predict' ending, on receiving the POST request after the button is clicked.

```
@app.route('/predict', methods=['POST'])
```

This is done by defining the predict () method, and the server is then ready to serve requests from the user.

Finally, the server is run, and the application is served using the following code section:

```
if __name__ == "__main__":
  app.run(debug=True)
```

Fig. 6: *The rendered 'index.html' file (homepage), with empty text placeholders for the user to enter the values for prediction*

Result

The five prediction algorithms were compared to find out which algorithm/classifier was providing the best accuracy/score on the given dataset. The accuracy scores of each classifier are as follows: KNN with 0.8750 (0.7083 before EL), LR with 0.9167 (0.9583 before EL), Decision Tree with 0.6667 (0.6250 before EL), Random Forest with 0.8333 (0.666 before EL), and Ensemble Learning with 0.8750. According to the Fig. 7; the accuracy scores of KNN, LR, and RF here are obtained after the application of the Ensemble Learning technique, thereby having higher values as compared to when their accuracy scores were calculated on an individual basis. Logistic Regression thus emerged as the most accurate algorithm.

	Name	Score
0	KNN	0.875000
1	LR	0.916667
2	DT	0.666667
3	RF	0.833333
4	EL	0.875000

Fig. 7 *All 5 algorithms were fit to the training data, and then tested for values using the test data, with Logistic Regression having the highest accuracy score of 0.9167 (obtained after applying Ensemble Learning)*

This classifier was then chosen to build the predictive model, and fit it to the training data. The trained model was then saved to the disk uses the 'pickle' library, as a pickle file, to be used by the server later on.

The user now has to input the values in the placeholders accordingly and click on the 'Predict' button. The user can take values from previous years' rainfall data. Even if they want to predict a flood situation in the future, the user needs to make sure that the necessary rainfall estimates are provided, and filled appropriately, for the model to be able to generate the correct result. The units need to be taken care of as well, the user needs to input the rainfall values in millimeters (mm).

In the following snapshots Fig. 8 and Fig. 9; displayed the figures for the year 1901 have been entered, the values being (in mm): March-May = 386.3, average rainfall every 10 days in June = 130.3, May to June increase = 649.9, and annual rainfall value in 1901 = 3248.6. According to the output of the model, the chances of flood-occurring would be severe, and rightfully so, floods did indeed occur in Kerala, in the year 1901.

Fig. 8 *User inputs the values accordingly, as can be seen, in the available text placeholders*

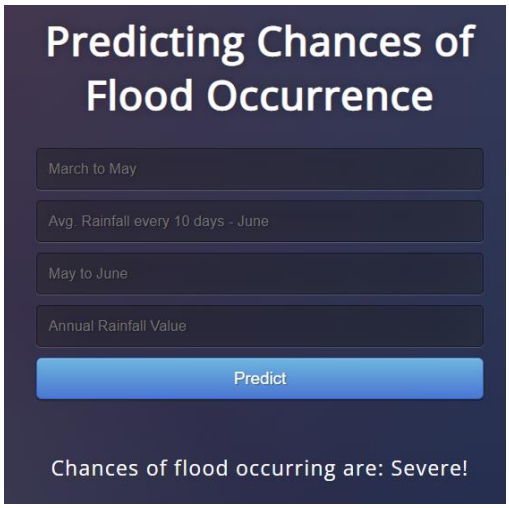

Fig. 9 *The result is displayed, the chances of the flood occurring being either 'Severe!' or 'None!'*

Conclusions

Exploratory data analysis makes it easy to understand the facts and figures hidden inside enormous datasets, providing the interpreter with visual and pictorial representations, enabling them to reason out the cause-effect relationship between various parameters in an accurate fashion, as can be seen, while observing the erratic rainfall patterns in India over approximately 120 years. A machine learning/data scientist can then interpret this data, and appropriately assign dependent and independent variables, while choosing the best algorithm for the prediction model, with the highest accuracy. Prediction models (for any calamity) make it easier for early-warning system analysts, to be able to work more accurately, and make predictions beforehand, to be able to devise the necessary emergency plans and evacuation systems, which can thus, save a considerable number of lives, as well as prevent huge losses and damage to property and livelihood. Therefore, this model will serve as a

decision support system (DSS) for the decision-makers, planners or policymakers.

As proposed for this model, the amount of rainfall over a region is the single most important parameter, with all four input variables/features being quantities of rainfall in some form or the other, be it quarterly rainfall values, average values every 10 days for a particular month, and even annual rainfall values. Therefore, including other factors (like temperature, topography, soil moisture holding capacity etc.) could add more dimensions to the model, making more real predictions based on natural attributes. This would perhaps increase the model accuracy even further, taking into account a greater number of input features that could influence the flood occurrences in a region. Also, apart from the five algorithms mentioned in this study, if an algorithm with an even higher accuracy can be developed/derived, it would better the model's predictions even further, which would further allow analysts to be more confident about their predictions, as can be expected, from greater accuracy levels.

Acknowledgment

I would like to acknowledge and give my warmest thanks and sincerest regards to my guide, Professor Rahul Diwate, who made this work possible. His guidance and advice carried me through all stages of writing this paper.

I would also like to give special thanks to my family and friends for their continuous support and understanding when undertaking my research.

References

U. C. Nkwunonwo, M. Whitworth, and B. Baily, "A review of the current status of flood modelling for urban flood risk management in the developing countries," Sci. African, vol. 7, p. e00269, Mar. 2020, doi: 10.1016/j.sciaf. 2020.e00269.

P. Arulbalaji, D. Padmalal, and K. Maya, "Impact of urbanization and land surface temperature changes in a coastal town in Kerala, India," Environ. Earth Sci., vol. 79, no. 17, p. 400, Sep. 2020, doi: 10.1007/s12665-020-09120-1.

R. K. Suryawanshi, S. S. Gedam, and R. N. Sankhua, "Inflow forecasting for lakes using Artificial Neural Networks," in WIT Transactions on Ecology and the Environment, May 2012, pp. 143–151. doi: 10.2495/FRIAR120121.

N. S. Ibrahim, S. M. Sharun, M. K. Osman, S. B. Mohamed, and S. H. Y. S. Abdullah, "The application of UAV images in flood detection using image segmentation techniques," Indones. J. Electr. Eng. Comput. Sci., vol. 23, no. 2, p. 1219, Aug. 2021, doi: 10.11591/ijeecs. v23.i2. pp1219-1226.

M. Bordbar, H. Aghamohammadi, H. R. Pourghasemi, and Z. Azizi, "Multi-hazard spatial modeling via ensembles of machine learning and meta-heuristic techniques," Sci. Rep., vol. 12, no. 1, p. 1451, Jan. 2022, doi: 10.1038/s41598-022-05364-y.

P. H. Hrudya, H. Varikoden, and R. Vishnu, "A review on the Indian summer monsoon rainfall, variability and its association with ENSO and IOD," Meteorol. Atmos. Phys., vol. 133, no. 1, pp. 1–14, Feb. 2021, doi: 10.1007/s00703-020-00734-5.

M. Moishin, R. C. Deo, R. Prasad, N. Raj, and S. Abdulla, "Designing Deep-Based Learning Flood Forecast Model with ConvLSTM Hybrid Algorithm," IEEE Access, vol. 9, pp. 50982–50993, 2021, doi: 10.1109/ACCESS.2021.3065939.

S. Mandal, D. Saha, and T. Banerjee, "A neural network-based prediction model for flood in a disaster management system with sensor networks," in Proceedings of 2005 International Conference on Intelligent Sensing and Information Processing, 2005., 2005, pp. 78–82. doi: 10.1109/ICISIP.2005.1529424.

W. W. Kellogg and S. H. Schneider, "Global Air Pollution and Climate Change," IEEE Trans. Geosci. Electron., vol. 16, no. 1, pp. 44–50, Jan. 1978, doi: 10.1109/TGE.1978.294524.

Liang Pang, Jiyi Zhou, and D. Liu, "The probability prediction of storm surge disaster along south-east coasts of China," in 2011 International Conference on Multimedia Technology, Jul. 2011, pp. 1827–1830. doi: 10.1109/ICMT.2011.6002851.

A. Usman, "Integrated disaster risk management in Indian environment: Prediction, prevention and preparedness," in 2017 IEEE Global Humanitarian Technology Conference (GHTC), Oct. 2017, pp. 1–6. doi: 10.1109/GHTC.2017.8239246.

S. Bande and V. V. Shete, "Smart flood disaster prediction system using IoT & neural networks," in 2017 International Conference On Smart Technologies For Smart Nation (SmartTechCon), Aug. 2017, pp. 189–194. doi: 10.1109/SmartTechCon.2017.8358367.

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

P. M. Padmawar, A. S. Shinde, T. Z. Sayyed, S. K. Shinde, and K. Moholkar, "Disaster Prediction System using Convolution Neural Network," in 2019 International Conference on Communication and Electronics Systems (ICCES), Jul. 2019, pp. 808–812. doi: 10.1109/ICCES45898.2019.9002400.

A. Juyal and S. Sharma, "A Study of Landslide Susceptibility Mapping using Machine Learning Approach," in 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Feb. 2021, pp. 1523–1528. doi: 10.1109/ICICV50876.2021.9388379.

C. N. Madawala, B. T. G. S. Kumara, and L. Indrathilaka, "Novel machine learning ensemble approach for landslide prediction," in 2019 International Research Conference on Smart Computing and Systems Engineering (SCSE), Mar. 2019, pp. 78–84. doi: 10.23919/SCSE.2019.8842762.

W. Han, Y. Gan, S. Chen, and X. Wang, "Study on Earthquake Prediction Model Based on Traffic Disaster Data," in 2020 IEEE 11th International Conference on Software Engineering and Service Science (ICSESS), Oct. 2020, pp. 331–334. doi: 10.1109/ICSESS49938.2020.9237667.

R. Mallouhy, C. A. Jaoude, C. Guyeux, and A. Makhoul, "Major earthquake event prediction using various machine learning algorithms," in 2019 International Conference on Information and Communication Technologies for Disaster Management (ICT-DM), Dec. 2019, pp. 1–7. doi: 10.1109/ICT-DM47966.2019.9032983.

G. E. Sakr, I. H. Elhajj, G. Mitri, and U. C. Wejinya, "Artificial intelligence for forest fire prediction," in 2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Jul. 2010, pp. 1311–1316. doi: 10.1109/AIM.2010.5695809.

H. Li, X. Fei, and C. He, "Study on Most Important Factor and Most Vulnerable Location for a Forest Fire Case Using Various Machine Learning Techniques," in 2018 Sixth International Conference on Advanced Cloud and Big Data (CBD), Aug. 2018, pp. 298–303. doi: 10.1109/CBD.2018.00060.

J.-M. Kim, J. Bae, S. Son, K. Son, and S.-G. Yum, "Development of Model to Predict

Natural Disaster-Induced Financial Losses for Construction Projects Using Deep Learning Techniques," Sustainability, vol. 13, no. 9, p. 5304, May 2021, doi: 10.3390/su13095304.

R. Mukherjee, D. Rollend, G. Christie, A. Hadzic, S. Matson, A. Saksena, and M. Hughes, "Towards Indirect Top-Down Road Transport Emissions Estimation," in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Jun. 2021, pp. 1092–1101. doi: 10.1109/CVPRW53098.2021.00120.

N. Sravanthi, M. L. Venkat, S. Harshini, and K. Ashesh, "An Ensemble Approach to Predict Weather Forecast using Machine Learning," in 2020 International Conference on Smart Electronics and Communication (ICOSEC), Sep. 2020, pp. 436–440. doi: 10.1109/ICOSEC49089.2020.9215444.

H. A. Shaiba, N. S. Alaashoub, and A. A. Alzahrani, "Applying Machine Learning Methods for Predicting Sand Storms," in 2018 1st International Conference on Computer Applications & Information Security (ICCAIS), Apr. 2018, pp. 1–5. doi: 10.1109/CAIS.2018.8441998.

N. A. Agana and A. Homaifar, "A deep learning based approach for long-term drought prediction," in SoutheastCon 2017, Mar. 2017, pp. 1–8. doi: 10.1109/SECON.2017.7925314.

J. Sarkar, J. R. Chicholikar, and L. S. Rathore, "Predicting Future Changes in Temperature and Precipitation in Arid Climate of Kutch, Gujarat: Analyses Based on LARS-wG Model," Curr. Sci., vol. 109, no. 11, p. 2084, Dec. 2015, doi: 10.18520/v109/i11/2084-2093.

J. Sarkar and J. R. Chicholikar, "Climate change scenario in the Gujarat region-analyses based on LARS-WG (long Ashton research station-weather generator) model," Asian J. Water, Environ. Pollut., 2015.