

Project report

Integrating Machine Learning And Deep Learning Methods For Predicting Landslides In Uttarakhand Region.

(SPARK Project)

by

VINAYAK SHUKLA

III year B.Tech Student

IIT Kanpur

Contents

1. Introduction.....	5
1.1 General.....	5
1.2 Objectives.....	6
1.3 Research Gap.....	6
2. Acquisition of data.....	7
3. Methodology.....	8
3.1 Analysis of Dataset Features.....	9
3.1.1 Landslide Trigger Classification.....	9
3.1.2 Activity of a region.....	11
3.1.3 Rainfall related dataset.....	12
3.1.4 Slope of a region	14
3.1.5 Type of Land movements.....	15
3.1.6 Soil Type.....	16
3.1.7 Land Use.....	16
3.1.8 Geological and Geomorphological features.....	16
3.1.9 Numerical details.....	17
3.2 Data preprocessing and Feature Extraction.....	18
3.2.1 Dealing with missing data.....	18
3.2.2 Dealing with categorical variables.....	18
3.2.3 Overfitting and Underfitting.....	19
3.2.4 Feature Extraction.....	19
3.2.5 Feature Scaling.....	20
3.2.6 Manually filling the data.....	20

3.3 Construction of Machine Learning models.....	21
3.3.1 Classification models.....	21
3.3.2 K Nearest Neighbor algorithm.....	22
3.3.3 Naive Bayes classifier algorithm.....	23
3.3.4 XGBoost classifier algorithm.....	24
3.3.5 Regression models.....	25
3.3.6 Linear regression.....	25
3.3.7 Random forest regression model.....	27
3.4 Artificial Neural Networks.....	29
3.4.1 Explanation of Artificial neural networks.....	29.
3.4.2 Constructing our ANN model.....	30
3.4.3 Details of the ANN model.....	31
3.4.4 Results obtained from ANN model.....	32
4. Utilization of the prediction models.....	35
4.1 Deploying Regression models through a website.....	36
4.2 Using google maps for mapping landslide red zones.....	38
5. Results and Conclusion.....	39
6. References.....	40

1. Introduction

1.1 General

A frequently used definition of landslide is “*movement of a mass of rock, earth or debris down a slope*”. They are a catastrophic phenomenon that take many lives, destroy the infrastructure and disrupt communication facilities. The Himalayan region of India is one of the most active areas of landslides, and people living here are badly affected by the frequent landslides. In India, approximately 0.51 million km-km is vulnerable to landslide hazards. Uttarakhand is chosen as the study area because the frequent landslides in the state have taken a heavy toll on life and property and have also affected the tourism industry, which is the livelihood of many people living there. The phenomenon of landslides can easily be explained by the movement of the material. Here material can be debris, rock, earth, or a mix, and movement can be slide, fall, topple, spread, and flow.

The traditional method of predicting landslides uses statistical methods to make predictions about landslides and uses landslide hazard maps to make citizens aware of landslide-prone areas. Machine learning techniques have not been fully explored to predict landslides based on historical inventory. There still does not exist any method that citizens of Uttarakhand can use to get landslide warnings and probability of occurrence in the future.

In the present study, with the help of GSI Bhukosh and IMD, an inventory of all the previous landslide occurrences in Uttarakhand since 2013 has been prepared, containing all the information such as triggering factors and rainfall received slope, and various other features.

Out of 8000 dataset points, 400 have been selected, and ML models such as naive bayes, XGBoost, random forest classifier, and ANN have been trained on the dataset to give us a landslide occurrence probability in the future if we have information of all the input features. We were able to train models with accuracy as high as 90 percent. Hence, this study can be extended further by expanding the landslide inventory to make more accurate predictions about future landslides. To utilize the outcome received from our

research, we also developed a website that the citizens of the Uttarakhand region can use to know about future landslide occurrence probability in their area.

1.2 Objectives

The whole study of this project is based on the following objectives-

- Construct a model using ML and DL to predict landslides based on the historical landslide datasets.
- Create a user interface such as a website to provide alerts to the citizens about the danger of landslides that can occur based on their location
- Create a disaster management model to help people during the time of disaster in getting aid and relief.
- To create a landslide hazard map using our constructed models and google maps.

1.3 Research Gap

A review of the above research work demonstrates that several methods have been adopted to predict landslides. Some of them are in the implementation stage as well. These are the research gaps that still exist-

- The accuracy of Deep Learning methods in predicting landslides needs to be explored.
- Lack of coordination still exists between Disaster prediction and Disaster Management. Better models which deal with both need to be formed.
- People need to be aware directly of the probability of a landslide occurring and what preventive steps they can take through a user interface such as a website or an app.

2. Acquisition of data

The successful landslide prediction depends on the preparation of a reliable database from a reliable data source. Machine Learning models are based on data handling, so it was taken into care that all dataset was collected from Indian Government agencies.

Dataset from the year 2013-2020 of the following districts of Uttarakhand -

- | | |
|----------------|-------------------|
| 1. Pithoragarh | 7. Almora |
| 2. Chamoli | 8. Tehri Garhwal |
| 3. Rudraprayag | 9. Nainital |
| 4. Bageshwar | 10. Pauri Garhwal |
| 5. Uttarkashi | 11. Dehradun |
| 6. Garhwal | 12. Champawat |

Rainfall data were obtained from the **Indian Meteorological Department**. It consists of daily rainfall measurements representing accumulated rainfall and rainfall intensity in the last 24 hours.

Other dataset features such as Activity of the region, Geology, Material of the soil, Type of soil movement, Hydrological features, Land Use, Triggering factor, Geoscientific reasons, Height, Width, Depth, Landslide area, and Landslide volume were collected from **Geographical Survey of India (GSI Bhukosh)**.

All dates with reported soil collapse, landslide, road slip, mudflow, debris flow, riprap collapse, rock fall, slope collapse, land/rock slide, and land/debris flow were classified by us as landslide events. The rest of the dates with no such remarks were classified as non-landslide events. The severity of Landslides was used to write down the Landslide probabilities.

National Disaster Management Authority (NDMA) reports on various landslides were used for dealing with disaster response and management.

3. Methodology

First aim is to use recent Machine Learning and Deep Learning advancements to construct a statistically-based prediction model. So basically, will analyze all the previous landslide dataset features and set a threshold for every feature. After this, will import regular updates of daily data features from NDMA and various sensors deployed. This data will be fed in prediction models developed in our research and would predict the occurrence of Landslide. Here is a flowchart of how the work has been done -

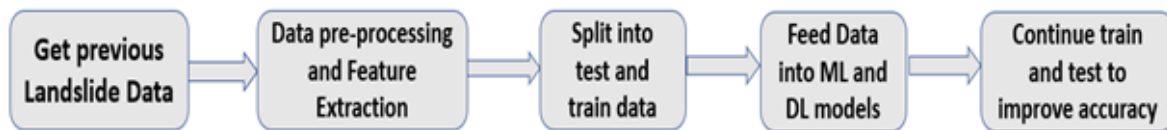


Figure 1 -Steps involved in Methodology

Using the Landslide prediction model, will warn the residents of that area if probability comes out to be in the risk zone.

Now comes the next step -How to respond to a Disaster? We Plan to incorporate the following disaster management and response systems-

Tweet and Message classification-We know that there are various government agencies coordinating disaster relief. Lots of messages are generated during a disaster requesting help and support. Due to the overwhelming number of messages that a human cannot parse, these messages may just be stacked without any reach. So, suppose we can classify which type of support the messages point to-In that case; that message can be segregated and forwarded to the respective department. The NGOs dealing with that category and help can be provided much quickly. We plan to do this using Natural language processing Techniques.

3.1 Analysis of Dataset Features

To get a better idea about dataset features, their properties, and trends, an analysis of 400 dataset points was done. It may be possible that any parameter is essential with respect to landslide occurrence for the given area, but it is also possible that the importance of the same parameter is negligible for another area. Important results of the analysis are summarized below-

3.1.1 Landslide Trigger Classification

Considering the initial cause of a landslide, we can extend the research to better predict the likelihood of a landslide occurrence. Our analysis focuses on the primary cause of a landslide, or its trigger, to determine the probability of a landslide occurring again or in the future. In our dataset, four leading causes of landslides triggers were identified in Uttarakhand -

1. Continuous Rainfall - Continuous rainfall is one of the main causes of landslide triggers which increases the landslide saturation and pore water pressure and reduces the mechanical strength of the weak layer.

2. Flooding of rivers - Flooding mostly leads to soil erosion, and sometimes this erosion is too rapid in hilly areas that may cause shallow landslides.

3. Heavy Rainfall- If there is heavy rainfall in a particular region (i.e., rainfall with high intensity for a short time), it can sometimes trigger rapid landslides in areas active to landslide risks.

4. Anthropogenic activities-Manmade activities such as road cutting, blasting of rocks, plantation, quarry, road density, and cropland trigger many landslide occurrences.

Apart from these, there were also **steep slope** and **cloud burst** occurrences as landslide triggers in dataset.

Here is a graphical analysis of Landslide trigger -

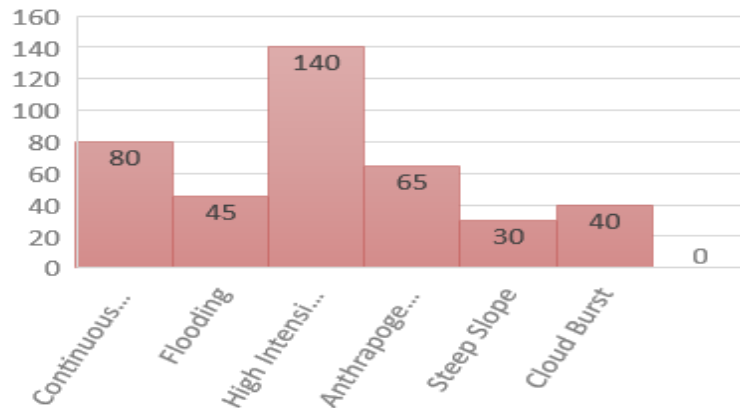


Figure 2- Trigger cause vs No. of Landslides

We can see from the above plot that High intensity rainfall was one of the primary causes of the trigger of landslides in dataset followed by continuous rainfall and anthropogenic activities. Initially the dataset was heavily biased towards heavy rainfall but that was removed through data preprocessing and feature engineering.

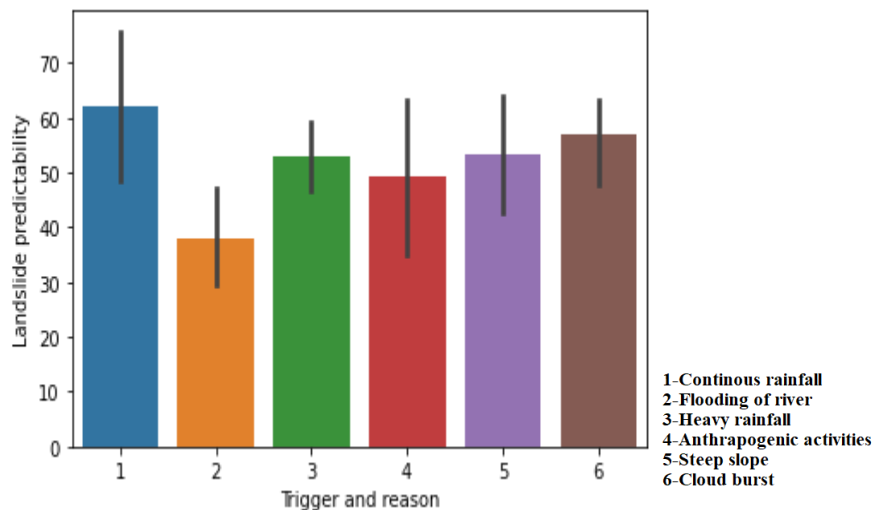


Figure 3- Landslide predictability vs trigger

From the above plot we can determine that chances of a landslide occurring (i.e. Landslide predictability) are the most in case of continuous rainfall and least in case of flooding of rivers.

3.1.2 Activity of a region

The locations in dataset were classified mainly into three regions based on the landslide risks-

1. Active region- These are the locations in which the chances of a landslide occurring are very high. They have been marked as high-risk zones on landslide susceptibility maps.

2. Reactivated regions- These are the locations that were once dormant but now have been reactivated again because of slope cutting, massive Uttarakhand disasters, and other reactivation mechanisms.

3. Dormant regions- There have been no significant landslides in these regions since 2004, therefore classified as dormant.

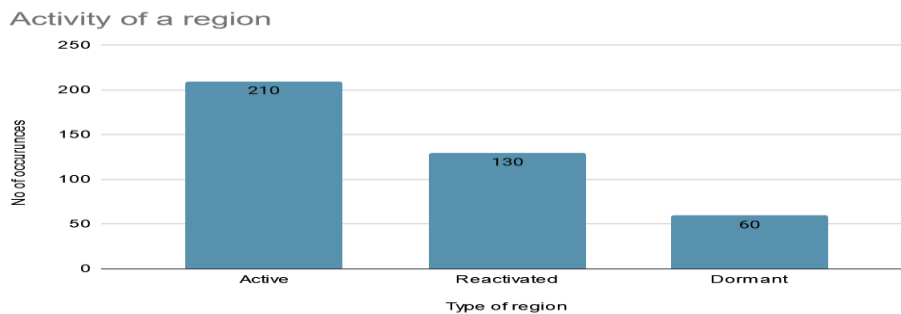


Figure 4- Activity vs no of Occurrences in data set

The above plot shows the distribution of the region in our dataset

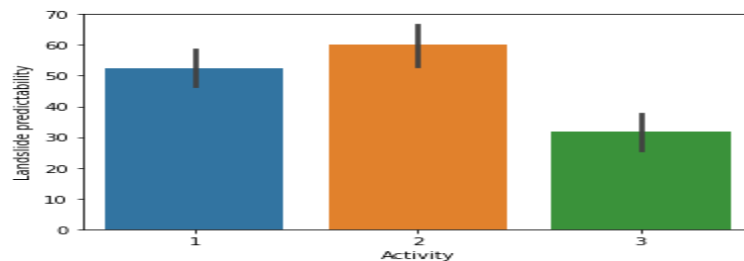


Figure 5- Activity vs Landslide predictability

From the above plot, we conclude that chances of a landslide occurring are the most in reactivated regions followed by active and significantly less in dormant areas

3.1.3. Rainfall related dataset features

Rainfall is one of the most crucial reasons behind the trigger of a landslide, and therefore proper analysis is needed on the rainfall features. There were two rainfall related variables in our dataset -

1. Cumulative Rainfall- Cumulative rainfall of 10 days before the landslide occurrence was collected from Mausam and IMD. After this rainfall event associated with major landslides was found based on the spatial distribution.

2. Rainfall Intensity- It refers to the average rainfall intensity of 24 hrs before the occurrence of landslide. If rainfall happens for less than 24 hrs, the intensity is taken for the duration of rainfall.

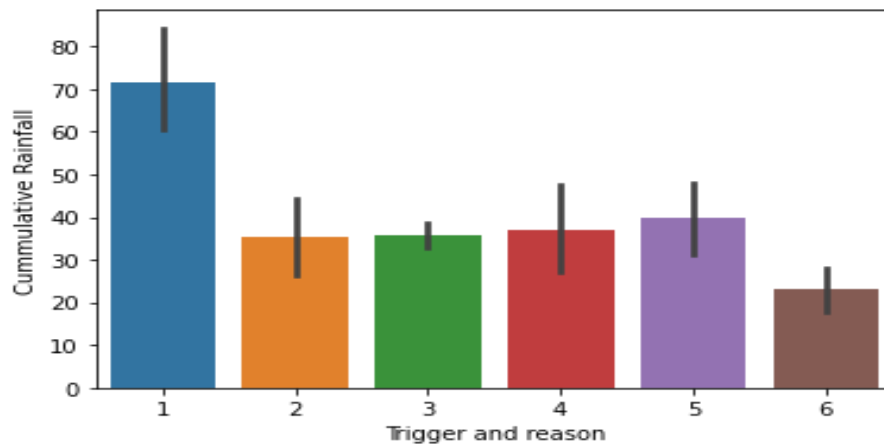


Figure 6-cumulative rainfall vs trigger and reason

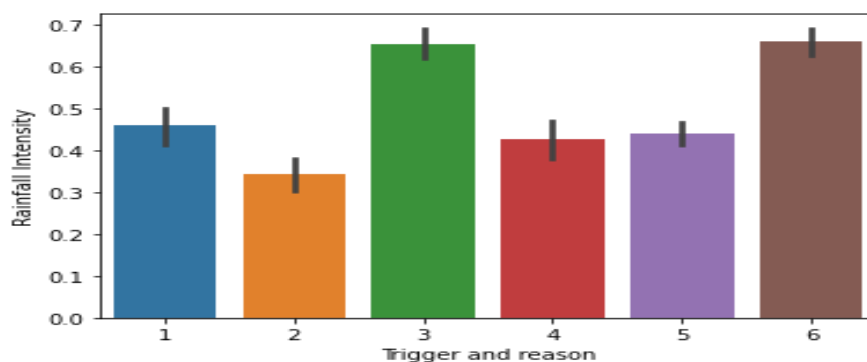


Figure 7-rainfall intensity vs trigger and reason

The above two plots show that in case of rainfall around 70-80 mm cumulative rainfall is the trigger and when intensity around 0.7-0.8 mm/hr heavy rainfall is the trigger.

Using graphical analysis we calculated cumulative rainfall and rainfall intensity thresholds for the occurrence of landslides. A threshold defines the possibility of a minimum of one landslide event in the region.

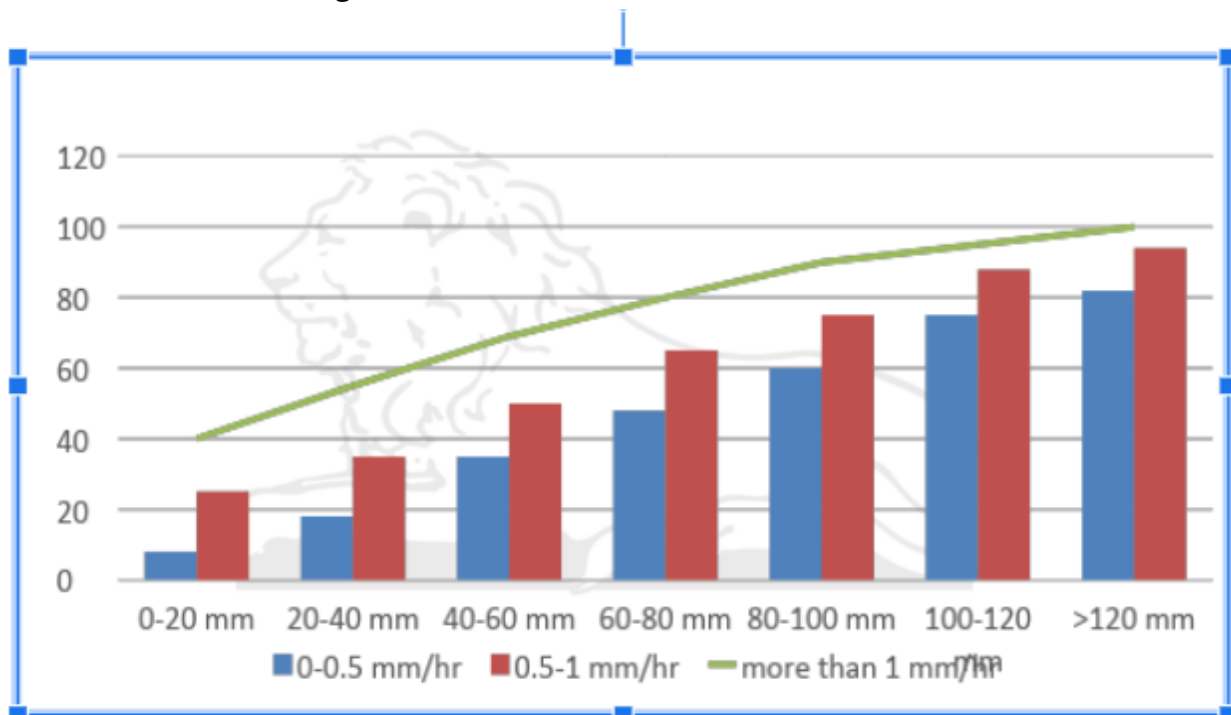


Figure 8- Calculating threshold of rainfall and intensity

In the above graph -

Vertical axis is for landslide probability.

Horizontal axis has cumulative rainfall -

For any set of cumulative rainfall the blue bar represents landslide probability when intensity is between 0-0.5 mm/hr and red bar when it is between 0.5-1 mm/hr. The green line represents continuous landslide probability when intensity greater than 1 mm/hr.

From the above graph the following thresholds were calculated-

Rainfall intensity >1 mm/hr	Cumulative rainfall >40 mm
0.5 mm/hr < Rainfall intensity < 1 mm/hr	Cumulative rainfall >70 mm
Rainfall intensity < 0.5 mm/hr	Cumulative rainfall >100 mm

This means that a cumulative rainfall greater than 40 mm of rainfall intensity >1 mm/hr can cause a minimum of one landslide. These thresholds are obtained graphically and not empirically.

3.1.4 Slope of a region

Slope angles are a key factor in estimating susceptibility to developing earth flows. The slope of the regions was obtained from the latitude and longitude of the landslide occurrence in our dataset. Even though landslides are generally connected to the steepness of the slope in geomorphological environments, it has to be taken into account that high slope angles do not always produce earth flows.

Following is a graph of the number of landslides vs. slope angle

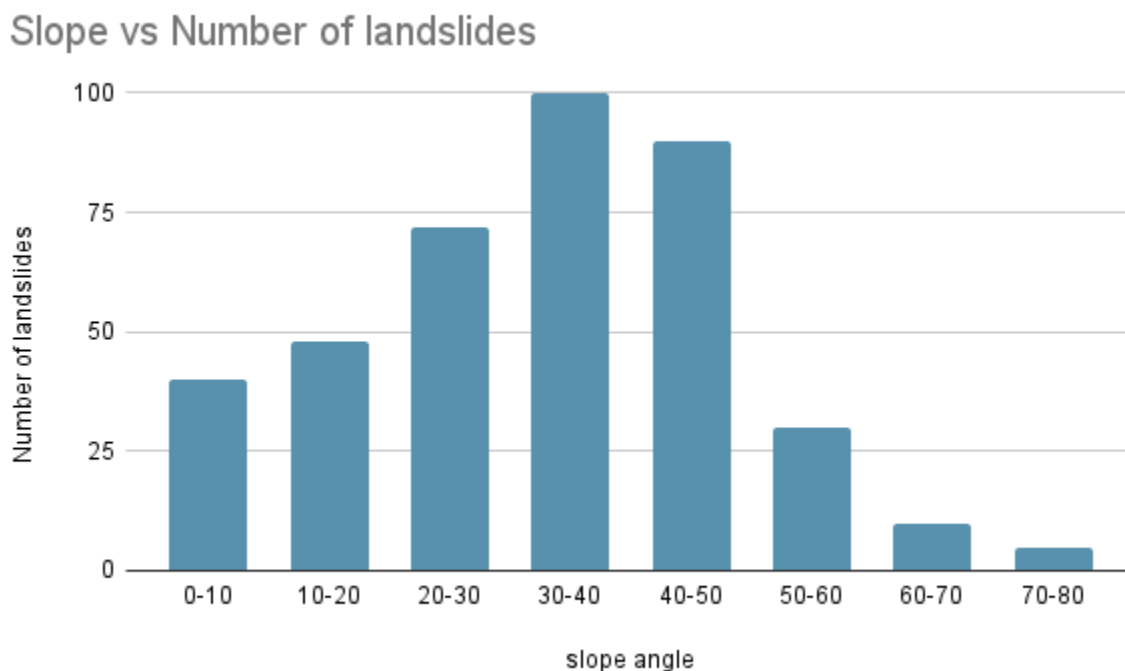


Figure 9- Slope angle vs Number of Landslides

From the above plot it can be concluded that the number of landslides increase till slope angle 30-50 degrees and then decrease. It means that regions with slope angle between 30-50 degrees are most prone to landslides.

3.1.5 Type of Land movements

There is the downward and outward movement of slope-forming materials in a landslide, including rock, soil, artificial fill, or a combination of these. The materials may move by falling, toppling, sliding, spreading, or flowing. In our dataset following types of land movements were present-

- 1. Slide-**A slide is the downslope movement of a soil or rock mass occurring dominantly on the surface of rupture or relatively thin zones of intense shear strain.
- 2. Fall -** A fall starts with the detachment of soil or rock from a steep slope along a surface on which little or no shear displacement occurs. The material then descends primarily by falling, bouncing, or rolling.
- 3. Topple-** A topple is the forward rotation, out of the slope, of a mass of soil and rock about a point or axis below the center of gravity of the displaced mass.
- 4. Flow -** A flow is a spatially continuous movement in which shear surfaces are short-lived, closely spaced, and usually not preserved after the event. The distribution of velocities in the displacing mass resembles that in a viscous fluid.
- 5. Spread -** A spread is an extension of cohesive soil or rock mass combined with general subsidence of the fractured mass of cohesive material into the softer underlying material.

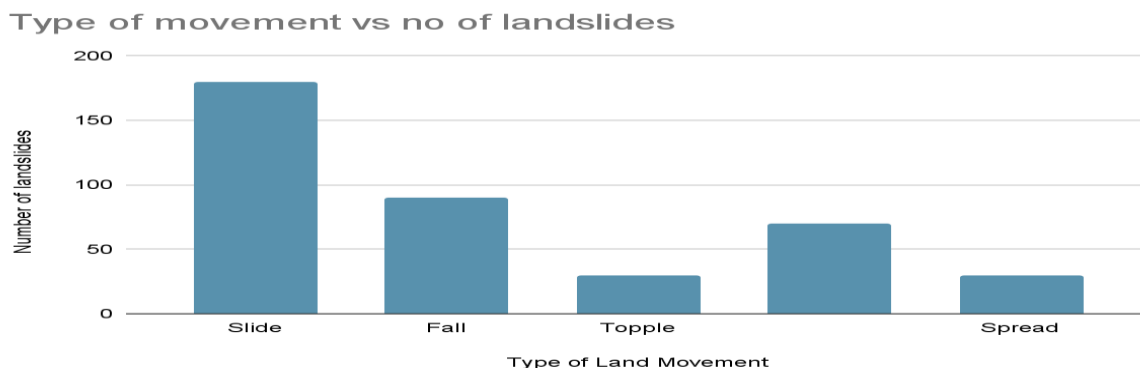


Figure 10- Type of Land movement vs number of landslides

We observe from the above column chart that slides are the most common in Uttarakhand followed by rock fall. Mudflow is found in areas near to rivers and are caused mainly by flooding of rivers.

3.1.6 Soil Type

Soil can be of various types depending on the area chosen. In our study area following categories of soil were found-moist, damp, dry, wet, saturated, and a mixture of these. As expected, from the study, it was found that areas with moist and wet soil are more prone to landslides than dry ones.

3.1.7 Land Use

While determining landslide probability, it is essential to know the land use of that area. For example, areas with forest cover are less prone to landslides than roads and sparse vegetation. The below pie chart shows the land use of our study area

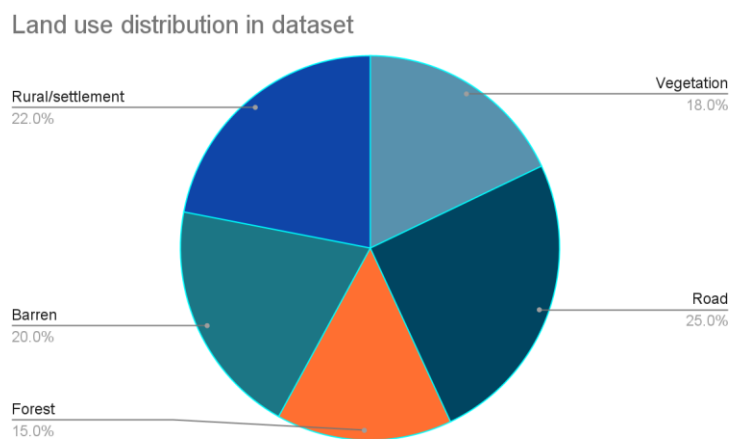


Figure 11- Distribution of Land use

So we see areas having roads and barren land have high chances of occurring of a landslide

3.1.8 Geological and Geomorphological features

Our dataset also had details about soil geology, i.e., whether it consists of dolomite, limestone, quartzite, and various other rock types.

Apart from this, geomorphological details were also present, i.e., whether it is a river terrace, hilly region, foot slope, or escarpment.

These details didn't have much weightage on landslide probability, and hence while using the ML models, they were ignored.

3.1.9 Numerical details

Our dataset had following features which were used to find the damage caused by the landslide -

1. Depth of a Landslide

2. Length, Height and Width of a Landslide

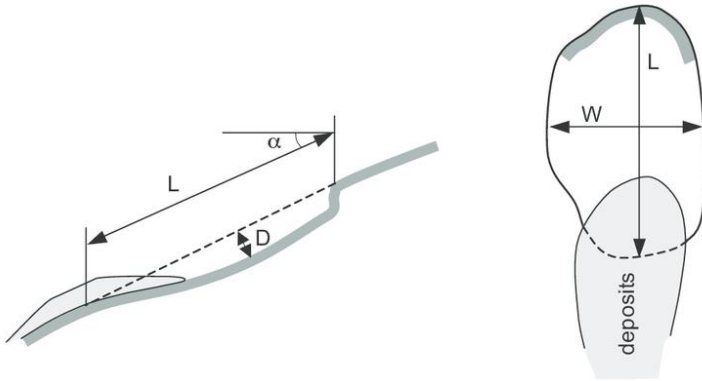


Figure 12- Depth and length of landslide

Following figure taken from Ref no.[2] clearly explains how landslide depth and length are calculated.

3. Landslide Volume - It is one of the most important feature in our ML models which we will also set as a target variable to predict the size of landslides in advance.

3.2 Data preprocessing and Feature extraction

Since models are entirely based on historical datasets, proper data processing techniques must be followed to turn the raw dataset discussed earlier into organized and sorted information that can be fed into the models. Following are some of the techniques which we used on dataset -

3.2.1 Dealing with missing data- Since missing data introduced uncertainty in our models, we handled the missing data in the following ways -

- **Using only valid data-**As specified earlier, out of 8000 dataset points, 400 landslide dataset points were selected, which had most of the values present.
- **Imputing data-**Linear regression model showed high accuracy when trained and tested on our dataset, so the remaining missing values were assigned by generating values through linear regression.

3.2.2 Dealing with categorical variables - In our dataset, variables such as activity, triggering factor, type of movement, land use were categorical variables . Since ML models work more proficiently on numerical data, categorical variables were converted into numerical values using the following two techniques -

- **Label encoding-** In this, a numerical value is assigned to a label; for example, in our case of activity, it marks active as 0, reactivated as 1, and dormant as 2. But this can add bias in our model as it will start giving higher preference to the *reactivated* parameter as $1 > 0$, and ideally, all three labels are equally important in the dataset. To deal with this issue, we tried one-hot encoding.
- **One hot encoding-**In this technique it will prepare separate columns for active, reactivated, and dormant labels for each categorical parameter. It will look like-

Active	Reactivated	Dormant
1	0	0
0	1	0
0	0	1

3.2.3 Dealing with overfitting and underfitting - When the model overfits it means that it performs well on the training set but not on the validation dataset. When the model underfits it is neither able to perform well on the training nor on the validation dataset. To see how our model is behaving we plotted a learning curve for a linear regression model as follows-

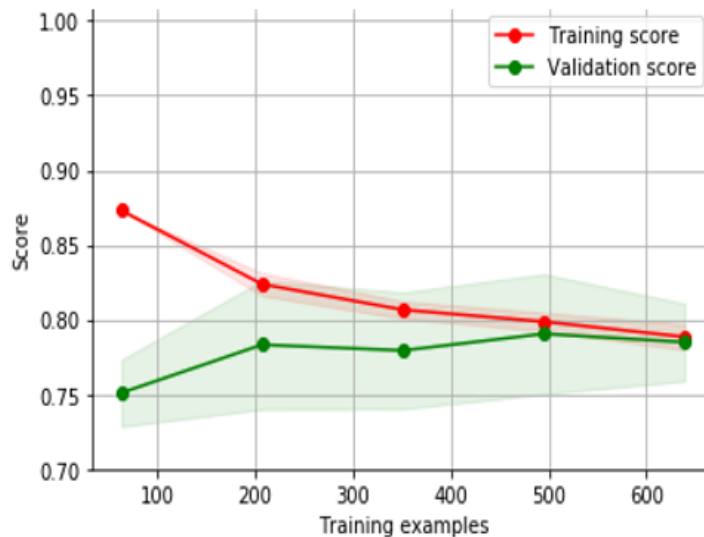


Figure 13- Comparison of Training and validation score

So from this we conclude our model is not overfitting since both the curves converge but it is underfitting because the final score is just about 0.78 .Therefore to improve the model we will hyper tune our model by doing feature extraction.

3.2.4 Feature Extraction- Feature extraction helps us attain the most informative and compact set of features, to improve the performance of machine learning models. For this, we have used two methods-

- **Feature Engineering-** We tried to improve the accuracy of our model by generating polynomials. Polynomial expansion creates interactions between features, as well as creates powers. This way, we introduce a nonlinear dimension to our data set, which can improve the predictive power of our model. In our model, we created a new variable Product, the product of cumulative rainfall and rainfall intensity. A correlation between the target variable and the square of the input features was also examined.

- **Feature Selection-** Feature selection is about choosing the relevant information and excluding irrelevant variables. For example, in our dataset, certain variables such as slide name, entry date, and geoscientific reasons were not going to affect the model in any way and therefore were dropped.

After this exploratory data analysis which we have discussed before, was performed to understand the relation between variable and landslide predictability. These variables showed a more significant impact on the target variable, i.e., Landslide probability was marked as an essential variable. In our dataset, Cumulative rainfall and rainfall intensity were observed as essential variables.

3.2.5 Feature Scaling-By using Euclidean distance between two data points, high magnitude features will weigh more in the distance calculations than features with low magnitude. So to avoid this, feature standardization is carried out, and in such cases, variables are scaled to a specific range using the min-max scalar class of sklearn library.

3.2.6 Manual filling of Data- In the case of regression models, target variable 'Landslide probability' values were to be filled manually between 0-100 after examining the landslide changes present in our dataset, damage caused by them, and numerical values such as landslide volume. Due to the requirement of manual efforts and a high level of expertise for the complete landslide inventory preparation out of 8000, only 400 dataset points were selected for the initial training and testing of the model.

3.3 Construction of Machine Learning models

3.3.1 Classification models

After preprocessing and extracting useful features from the data, it was time to construct some ML models for training and testing the dataset. We first used **classification models** to make predictions. In Classification models target variable has to be categorical; therefore, our target variable '**Landslide predictability**' had the following values-

Categorical variable value	Meaning
0	No chance of a landslide occurring
1	Moderate chances of a landslide occurrence.
2	Very high chances of a landslide occurrence.

Some important points regarding the implementation of classification models-

- Before feeding the data into classification models, we ensured that the target variable is present in equal proportions so that there is no biasedness, and the following count was obtained- (0-138,1-132, 2-130)
- In the case of classification models, all the input variables must be scaled, i.e., they all should lie in the same range; otherwise, variables such as landslide volume, which have value as high as 20000, would impact the model much more and decrease its accuracy.
- In all the classification models, input feature(X) consisted of 'Trigger,' Activity' Material' 'Movement' 'Land use' 'Cumulative rainfall' 'Rainfall intensity' 'Product' and 'Landslide volume' whereas target variable(Y) was 'Landslide predictability.'
- In all the classification models, we will **split the train and test/validation data in a ratio of 67:33**.To build the training and testing sets, four sets of data are created. **X_train**(training part of the matrix of features),**X_test**(test part of the matrix of features), **Y_train**(training part of the dependent variable associated with the X_train), and **Y_test**(test part of the dependent variable associated with the X_test).

3.3.2 K-Nearest Neighbour Algorithm

The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm mainly used in classification problems.

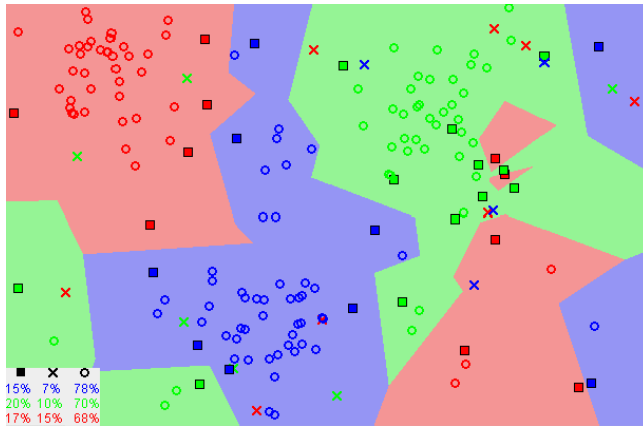


Figure 14-Working of KNN algorithm

The above image shows the implementation of the K nearest neighbor algorithm, and it can be concluded from it that most of the time, similar data points are close to each other. KNN captures the idea of similarity (sometimes called distance, proximity, or closeness) with some mathematics — calculating the distance between points on a graph. When we trained the KNN algorithm with our dataset, we chose the right K for our data by trying several Ks and picking the one that works best (i.e., has the least number of errors).

Despite this, when we trained the KNN model on our dataset, it **achieved only an accuracy of 65.15 percent**. This is because of the fact that KNN works best when there are 4-5 input variables, but in our case, there were nine input variables, and hence the result was ambiguous.

To improve the accuracy of our KNN model, we trained it once again, but this time only on five input features 'Trigger' 'Activity' 'Cumulative rainfall' 'Rainfall Intensity' and 'Landslide volume' and observed that **accuracy increased considerably to 84.85 percent**. **The K nearest algorithm is not a suitable choice for our dataset because when the training dataset increases, the algorithm will get significantly slower.**

3.3.3 Naïve Bayes Classifier Algorithm

The Naïve Bayes algorithm is a supervised learning algorithm based on the Bayes theorem and is used to solve classification problems. It is a probabilistic classifier, which means it predicts based on the probability of an object.

We are using a naive bayes classifier for our dataset because it assumes that the occurrence of a particular feature is independent of the occurrence of other features. Such as if the landslide probability is identified based on activity, trigger, and rainfall, each feature individually contributes to landslide probability. It also behaves well in the case of multi-class categorical variable predictions.

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)}$$

using Bayesian probability terminology, the above equation can be written as

$$\text{Posterior} = \frac{\text{prior} \times \text{likelihood}}{\text{evidence}}$$

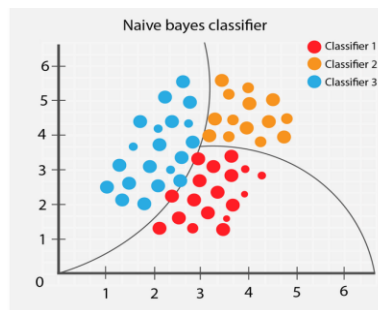


Figure 15- Working of naive bayes algorithm

From the above image it is clearly understood how a naive bayes algorithm works and how it groups different classifiers.

In our classification models confusion matrix will have the following representation-

	Actual- 0 (No landslide chances)	Actual -1 (Moderate chances of landslides)	Actual-2(Very high landslide chances)
0(No landslide chances)	True predictions	False predictions	False predictions
1(Moderate chances of landslide occurrence)	False predictions	True predictions	False predictions
2(Very high chances of landslide occurrence)	False predictions	False predictions	True predictions

This means that the diagonal values indicate our model's number of true predictions for that value.

When we trained the naive bayes classifier model on our dataset, we obtained the following confusion matrix.

0	22	1	0
1	2	21	1
2	0	1	17

From the confusion matrix, we can say that for target value 0, our model is able to predict it correctly 22 times and incorrectly just one time. Thus, the confusion matrix clearly shows that the naive bayes classifier works on our dataset with pretty high accuracy.

Figure 16-Confusion matrix for our naive bayes model

The **accuracy** obtained from the naive bayes classifier is **90.91 percent** which is pretty high, and this model, after further training, can be used to predict landslides based on historical records.

3.3.4 XGBoost Classifier Model

XGBoost is the most popular machine learning algorithm these days. Regardless of the data type (regression or classification), it is well known to provide better solutions than other ML algorithms. It has both linear model solver and tree learning algorithms. We have used XGBoost to train our dataset because of its capacity to do parallel computation on a single machine. Therefore XGBoost will be the preferred choice when we have to train with a large amount of datasets.

XGBoost's objective function is a sum of a specific loss function evaluated over all predictions and a sum of regularization term for all predictors (KK trees).

Mathematically, it can be represented as :

$$obj(\theta) = \sum_i^n l(y_i - \hat{y}_i) + \sum_{k=1}^K \Omega(f_k)$$

XGBoost handles only numeric variables.

When dataset was trained with the XGBoost model, its prediction accuracy was 92.42 percent. Therefore, the XGBoost model was the best out of the three classification models we trained on our dataset, closely followed by the naive bayes classifier model.

3.3.5 Regression Models

The problem with classification models is that they only tell us whether a landslide will happen or not. It would be better if we knew the probability of the landslide occurring between 0-100 percent. To overcome this problem, we have trained some regression models which give us an exact percentage between 0-100 of a landslide occurrence. In Regression models, the target variable Y is 'Landslide predictability,' which has a value between 0-100 and has been manually filled as explained earlier in the data preprocessing section. Since regression models work best on 4-5 input features. **Input variables X are chosen as 'Trigger ', 'Activity', 'Cumulative rainfall' 'Rainfall Intensity' and 'Product'.**

Following regression models were trained on dataset -

3.3.6 Linear Regression

A linear regression model presents a primary type of predictive model, which estimates the equation that best describes the relationship between landslide probability and the input features. Linear regression actually finds the best fit line between the input and the target variables.

Out[5]:

	Landslide Volume	Landslide predictability	Cummulative Rainfall	Rainfall Intensity	Product
Landslide Volume	1.000000	0.426045	0.679600	0.180497	0.697622
Landslide predictability	0.426045	1.000000	0.877177	0.528274	0.908826
Cummulative Rainfall	0.679600	0.877177	1.000000	0.302122	0.953964
Rainfall Intensity	0.180497	0.528274	0.302122	1.000000	0.521604
Product	0.697622	0.908826	0.953964	0.521604	1.000000

Figure 17- Correlation characteristics of input variable

The above table shows the linear correlation between various variables in our dataset. It must be noted that Landslide predictability is linearly most dependent on product followed by cumulative rainfall and rainfall intensity. Since the correlation values are pretty high, the linear regression model might work well with our dataset.

Using the sklearn library dataset was split into **training and test data in a ratio of 75:25.**

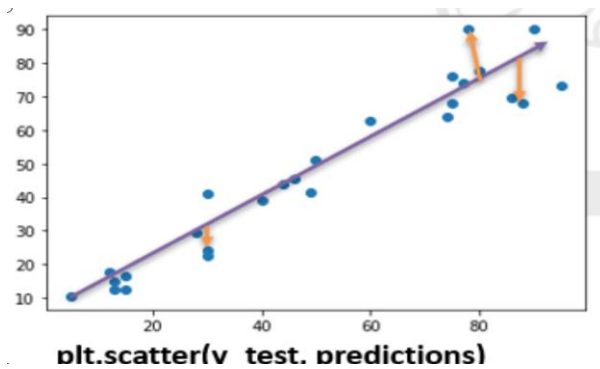


Fig 18- Test values vs predicted values for linear regression model

The above plot shows the best fit line obtained from our model after training on initial test data. The orange arrows indicate the errors.

1. Coefficient of Determination

The coefficient of determination, commonly known as R-squared (or R²), explains how much variability of one factor can be caused by its relationship to another factor. This measure is represented as a value between 0.0 and 1.0, where a value of 1.0 indicates a perfect fit and is thus a highly reliable model for future forecasts, while a value of 0.0 would indicate that the model fails to accurately model the data at all.

When the linear regression model was fit on our dataset, the coefficient of determination came out to be 0.921189435, which is very close to 1, and hence our model is near accurate.

2. Root mean squared error and mean absolute error

Mean absolute error (MAE) measures the average magnitude of the errors in a set of forecasts without considering their direction and measures accuracy for continuous variables. The mean absolute error for our linear regression model when trained and fitted was 5.75, which shows that our model predicts landslide occurrence probability values with a deviation of about 6 percent. It can be regarded as a good performance from the model.

Root mean squared error (RMSE) is a quadratic scoring rule that measures the error's average magnitude.

It is given by the formula $RMSE = \sqrt{\frac{\sum (P_i - O_i)^2}{n}}$ where P_i is the predicted value and O_i is the observed value. The lower the value of RMSE, the better is the model. For our machine learning linear regression model, the RMSE was found to be 49.48, which is considered good, and hence the L.R model can be used to make predictions.

3.3.7 Random Forest regression model

Random forests is a technique based on a standard machine learning algorithm called a decision tree. Decision trees are simple and valuable for interpretation but are typically not competitive in prediction and accuracy. The random forest model is basically a collection of decision trees. Each decision tree produces its own prediction, and at the end, the average of all predictions is taken, known as the random forest prediction.

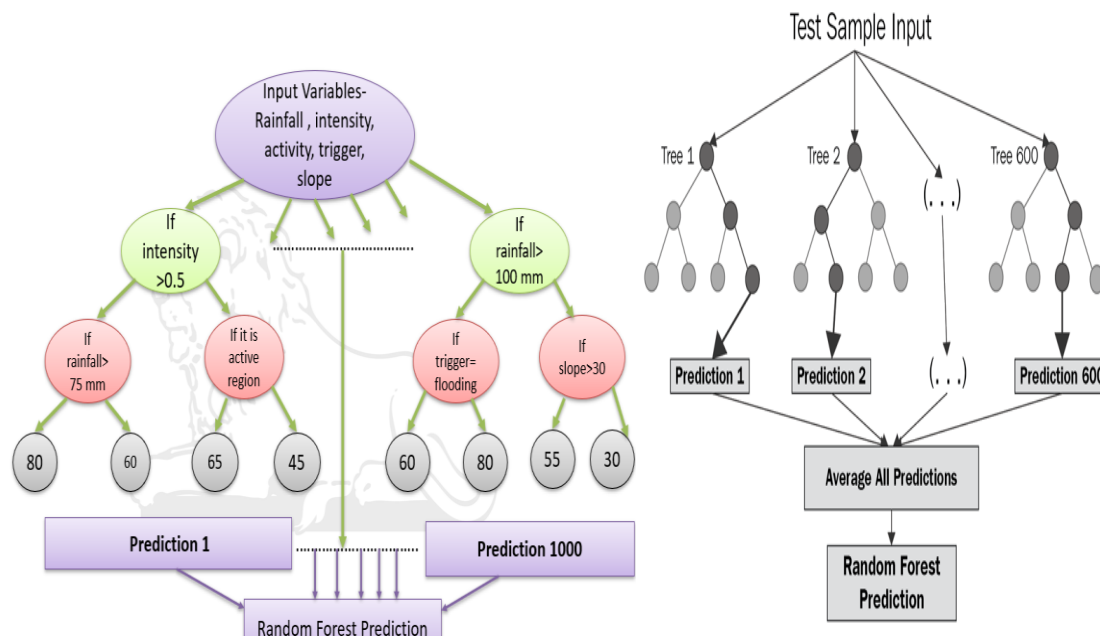


Figure 19- Working of random forest regression model

The above two diagrams show how a random forest regression model will look and work on our dataset.

- Train and test data was split in a **ratio of 75:25** using the sklearn library
- Random state =42 was chosen in our model as it gave the best accuracy.
- We initialized n_estimators=1000, which means 1000 decision trees will be made before averaging the final prediction.
- **The mean absolute error was found to be 3.31 percent.**

- The **accuracy** obtained after the random forest model was trained and predictions were made using it is **85.74%**

An accuracy of 85.74 percent is entirely reasonable to make predictions. With further training and testing of the model, accuracy will increase further, and we will be able to make accurate predictions about the chances of landslides in the future.

Using a random forest model, we can also get the importance of input variables in predicting the target variable. Here is a list of the importance of variables-

Variable	Importance
Rainfall Intensity	0.5
Cumulative rainfall	0.35
Product	0.30
Landslide predictability	0.15
Activity	0.1
Trigger	0.01
Landslide volume	0.001

The above table concludes that rainfall intensity is the most important variable in our random forest regressor, followed by cumulative rainfall. This variable importance and correlation values were used for Feature selection (i.e., selecting the most important input features in an Input model).

The overall analysis of the ML model made us come to the conclusion that the following variables - Trigger, Continuous Rainfall, Rainfall Intensity, Product of Cumulative rainfall and Rainfall Intensity, Activity of the region hold the most importance in our dataset, and they must be taken for better working of the models.

3.4 Artificial Neural Networks

After constructing several ML classifications and regression models and making predictions with them, we now explore deep learning methods such as Artificial neural networks to make predictions about landslide probability and look to train them on our dataset.

3.4.1 Explanation of Artificial neural networks

Artificial Neural Network primarily consists of three layers:

- Input Layer: As the name suggests, it accepts inputs in several different formats provided by the user.
- Hidden Layer: The hidden layer presents in-between input and output layers. It performs all the calculations to find hidden features and patterns.
- Output Layer: The artificial neural network takes input and computes the weighted sum of the inputs, and includes a bias. This computation is represented in the form of a transfer function.

The artificial neural network takes input and computes the weighted sum of the inputs, and includes a bias. This computation is represented in the form of a transfer function. These weights are also updated when ANN back propagates. Weight update methods are often called optimizers.

$$\bullet \quad \sum_{i=1}^n W_i * X_i + b$$

It determines that the weighted total is passed as an input to an activation function to produce the output. Activation functions choose whether a node should fire or not. Only those who are fired make it to the output layer. There are distinctive activation functions available that can be applied upon the sort of task we are performing.

Loss function- It is a performance metric on how well the NN manages to reach its goal of generating outputs as close as possible to the desired values.

Back propagation-Backpropagation is the central mechanism by which artificial neural networks learn. It is the messenger telling the neural network whether or not it made a mistake when it made a prediction. Backpropagation takes the error associated with a wrong guess by a neural network and uses that error to adjust the neural network's parameters in the direction of less error. The direction of less error is known by gradient descent.

3.4.2 Constructing our ANN model

Now since we have understood all the terminology associated with artificial neural networks lets make our ANN model and train on our dataset.

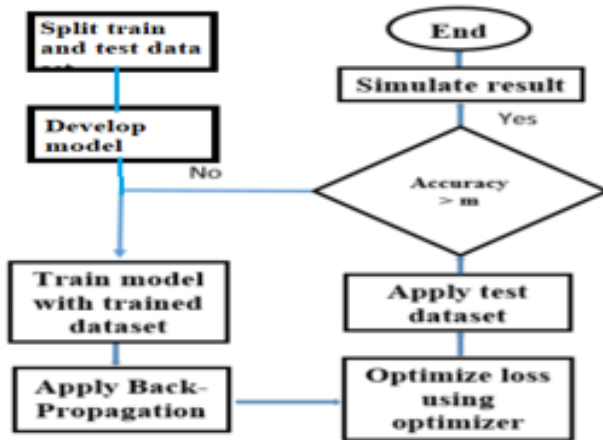


Figure 20 -Flowchart of our ANN model

These were the steps involved in construction of our ANN model and the value of m was taken to be 85 percent.

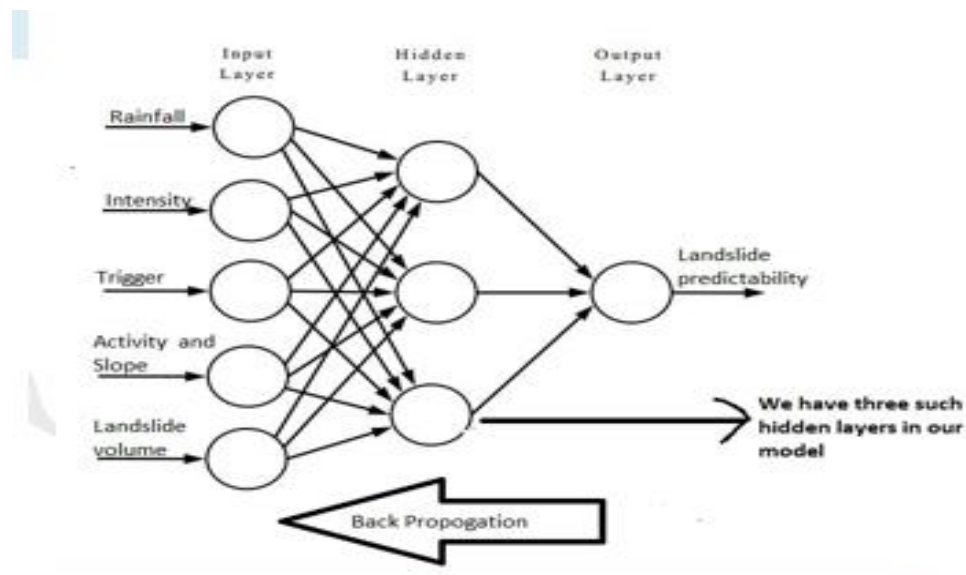


Figure 21-Our ANN model

This is how our ANN model looks like. It consists of three hidden layers, five input features and one output layer which gives landslide probability.

3.4.3 Details of the ANN model

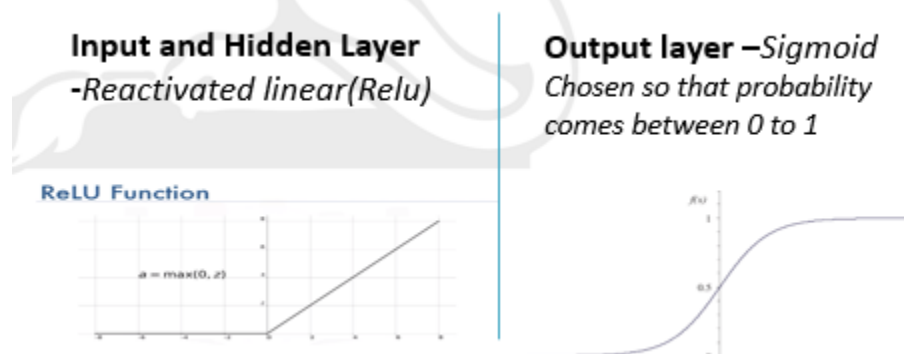
Figure 22 -Developed ANN model summary

Layer (type)	Output shape
dense_1 (Dense)	(None, 16)
dense_2 (Dense)	(None, 32)
dense_3 (Dense)	(None, 64)
dense_4 (Dense)	(None, 32)
dense_5 (Dense)	(None, 1)

Total params:393
Trainable params:393

Here dense_1 is the input layer while dense_2,dense_3 and dense_4 are the three hidden layers followed by output layer dense_5.

Figure 23-Activation functions



For the output layer sigmoid activation function is chosen as we need output(probability) in the range of 0-1.

Optimizer chosen was ADAM and Mean squared error was chosen as the loss function.

3.4.4 Results obtained from Artificial neural network

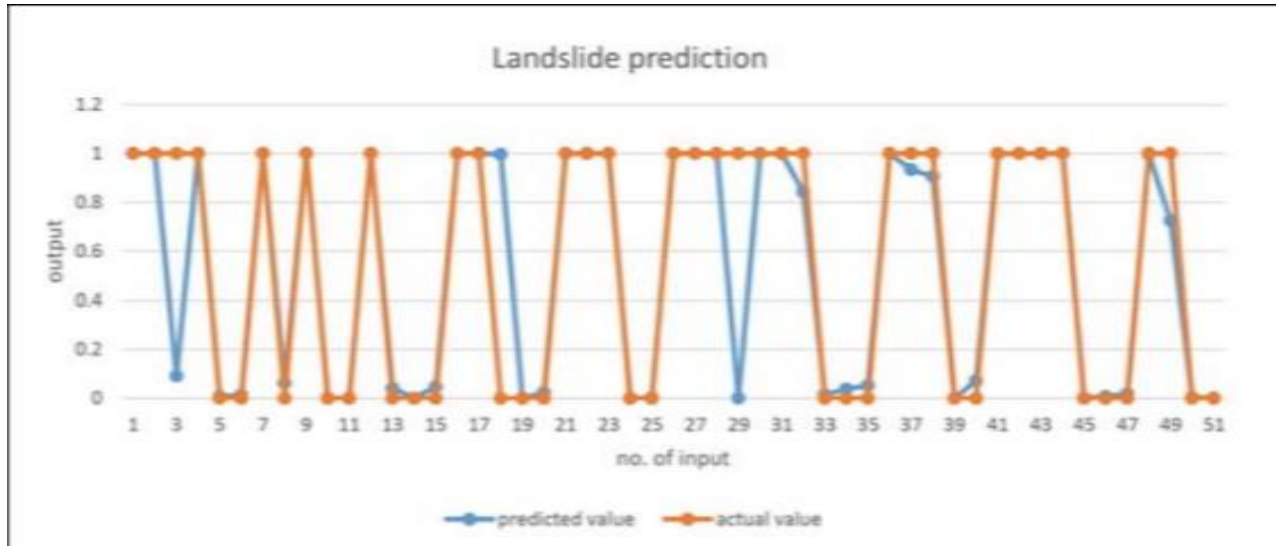


Figure 24- Predicted vs actual value

Following is a plot of predicted value vs actual values .We can see most of the predicted value by ANN coincide with the actual values.

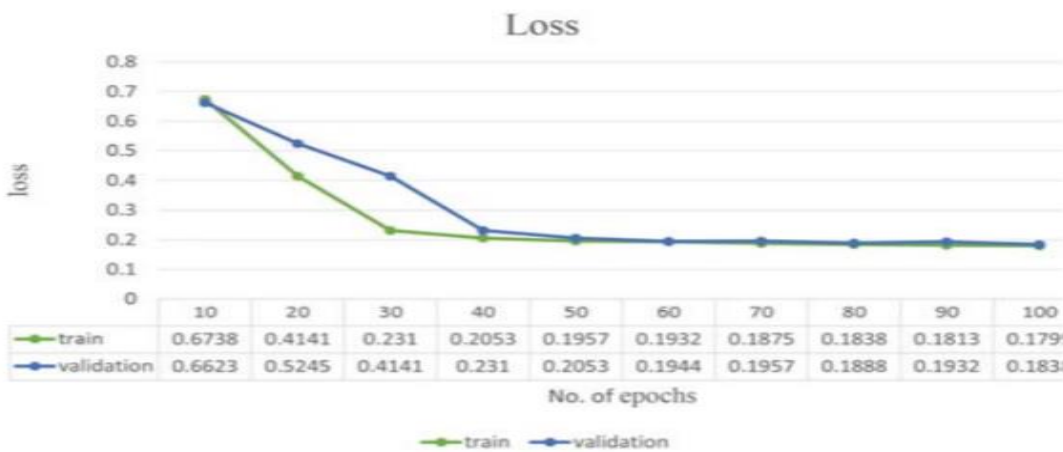


Figure 25-Loss vs no of epochs

Epoch is the number of time training data is fed into ANN. Number of epochs were set to 100 and from the above plot we can see that as number of epoch increases the loss value decreases i.e. the model is able to retain more information now and adjusts itself according to it.

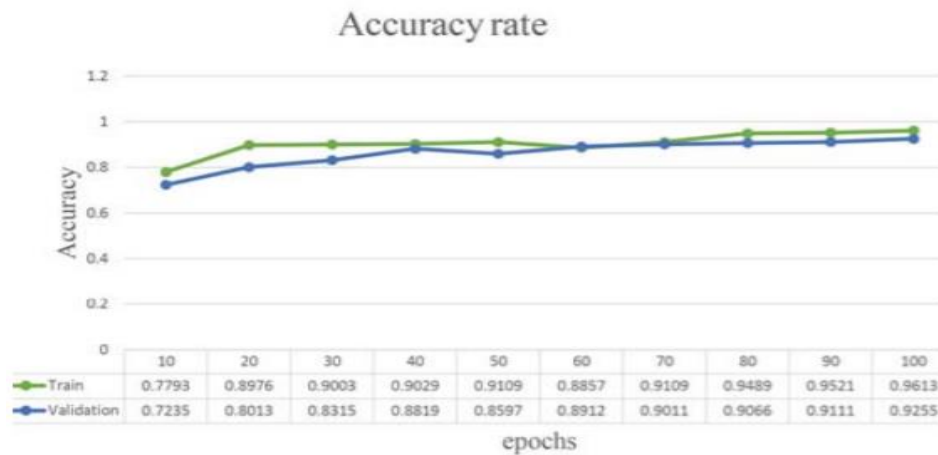


Figure 26 -Accuracy vs no of epochs

We conclude from the above plot that accuracy keeps on increasing with the number of epochs.

Accuracy of Our ANN model comes out to be **89.87%** .We also constructed ANN model without certain input variables to understand the importance of that particular variable in our ANN model.

Factor	Accuracy
Without cumulative rainfall	76.78
Without rainfall intensity	78.25
Without activity and slope	83.84

Therefore cumulative rainfall and rainfall intensity hold the most importance in our ANN model.

4. Utilization of the Prediction models

In the previous section of this study, we have constructed various ML models and artificial neural networks (ANN), which can predict future landslides if they have the rainfall data and other information such as the slope of the region, whether it is a landslide active or dormant zone. In Fact, if we set Landslide volume as the target variable in our model, we can also predict the damage caused by a landslide owing to the characteristic of ML models learning from historical landslide datasets.

One of the common mistakes in most research projects is that the outcome never reaches the target audience, and it takes many years for the research outcome actually to cause some change.

One of the main objectives behind this research work was to help the citizens of Uttarakhand deal with the problem of frequent landslides every year. It would be great if they could get early warnings of landslides and hence take precautionary steps accordingly.

Landslide prediction is a very tough task, and the Uttarakhand government has taken many steps in this direction. Our model is a positive step in this direction, and with consistent efforts and an improved dataset, we can undoubtedly predict landslides with considerable accuracy in the near future. To make our research work available for citizens of Uttarakhand, we must deploy our constructed models using a website or an app.

Here is a flowchart of how we planned to utilize our construction models.

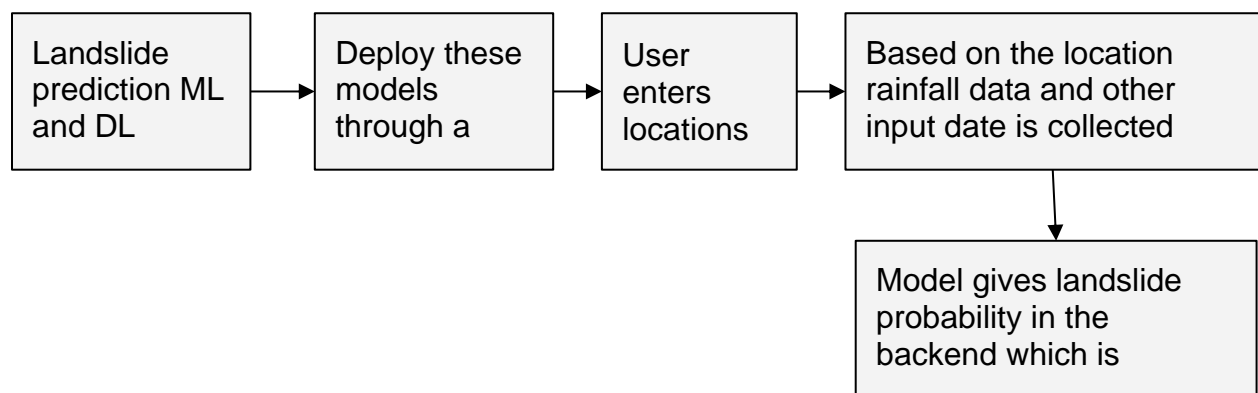


Figure 27- Flowchart of utilizing our research work

4.1 Deploying regression models through a website

The deployed website can be found at <https://landslides-model.herokuapp.com/>

Have deployed regression models linear regression and random forest through a website. Following is a snapshot of website-

The screenshot shows a web form with the title "Input values:". It contains three input fields: "Cumulative_rainfall:" with the value "75", "Intensity:" with the value "1", and "Product:" with the value "69". Below these fields is a "Submit" button. The output section displays the input values as "Cumulative_rainfall : 75 Intensity : 1 Product : 69", followed by the label "Probabability of Landslide Occuring:" (note the typo). The final output is a large number: "69.25919019503031".

Figure 28-Snapshot of the deployed website

For now we are taking three inputs from the user- Cumulative rainfall, Intensity and product. The probability of landslide occurring which is printed as output has been predicted from the regression models.

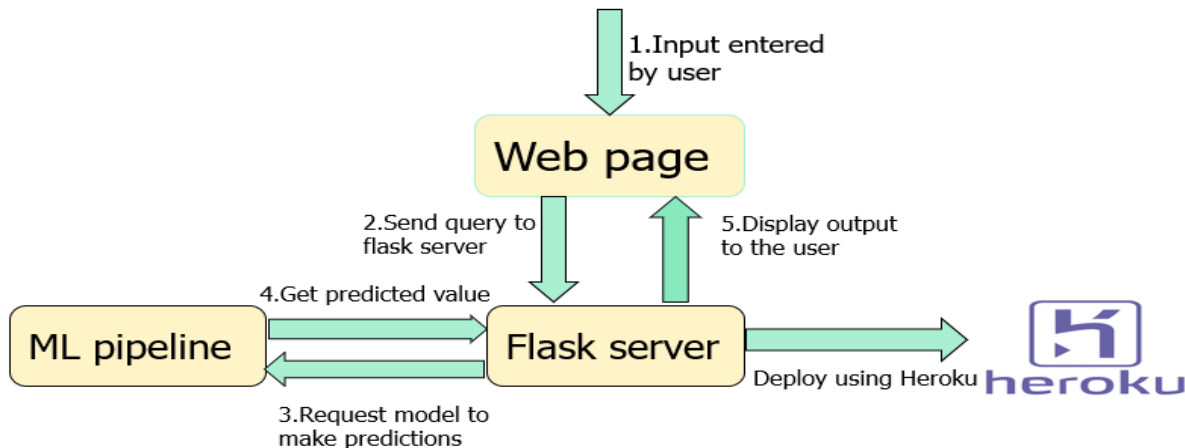


Figure 29- Steps involved in deploying a website

These are the steps which are involved in working on website. Have used **Flask** to deploy website. Flask is a web application framework written in Python, and it provides us with tools, libraries, and technologies that allow us to build a web application.

Flask is used to deploy models on a localhost server, but that was of no use to us. Therefore we have used **Heroku**-a cloud service platform to host our website so that anyone can access it.

One issue with the website is that we have to manually enter input variables values such as cumulative rainfall. A citizen who will use this website would not have these values. Therefore, the backend of the website needs to be updated. The user only has to enter its location, and all these input variables are collected from the API of government agencies that keep a regular update of these data..

4.2 Using Google maps for mapping Landslide red zones

In our dataset, we have the latitude and longitude of the regions. When our constructed models make predictions about the landslide probability, we can get latitudes and longitudes of those places where probability is more than 50 percent. For such areas, the citizens need to be made aware of the danger of landslides in the future. We have achieved this by marking the location of such places on google maps. Hence, citizens of Uttarakhand would just have to navigate through google maps, and if their site is shown in the red zone, it means that

there are chances of a landslide happening in the near future. Following is a snapshot of google earth implementation on our test dataset.

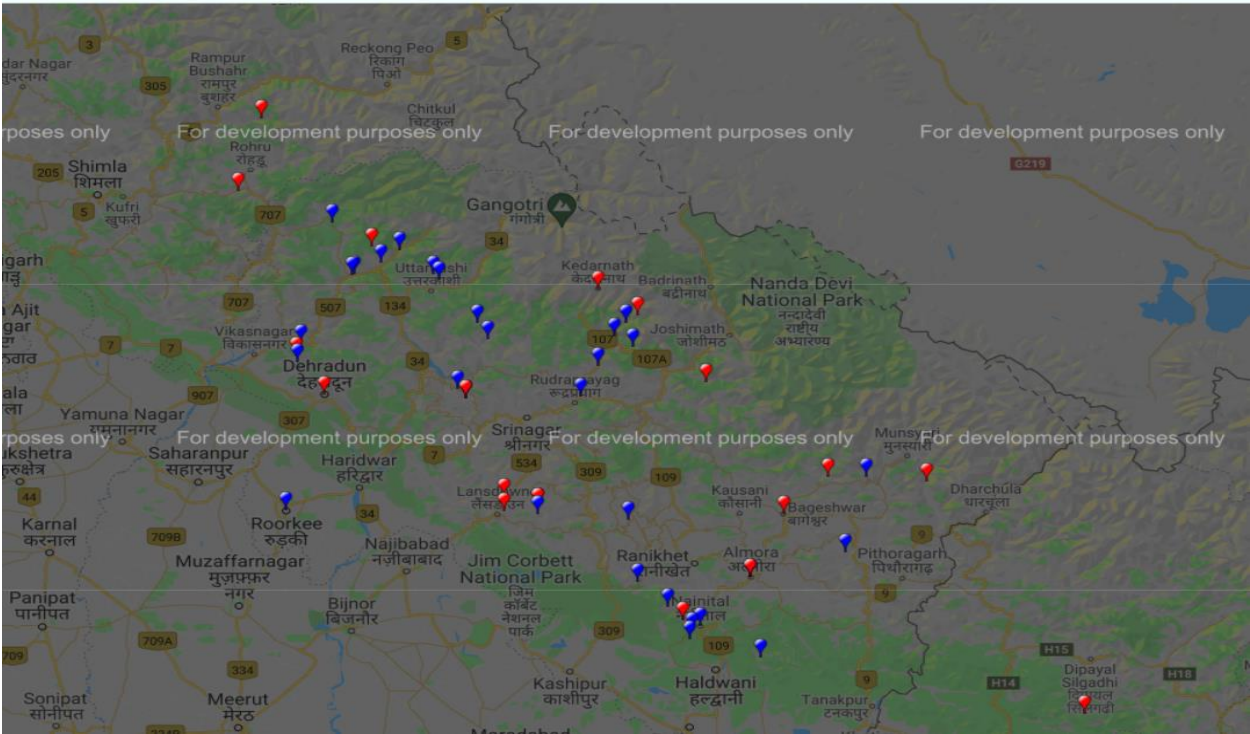


Figure 30- Landslide danger zone on google maps

The regions marked with red are the ones that have a high probability of occurring of a landslide, and with blue are those who are safe from landslides during that given time of which input data is taken. The areas shown in the snap above have been marked according to the landslide probability calculated from the input dataset in our test case

5. Results and Conclusion

At the end of project, I was able to achieve the following results-

- 1. Successfully constructed classification and regression models, which could predict landslides with considerable accuracy. Following is a comparative analysis of all the ML models constructed-

Model	Accuracy	Type
-------	----------	------

K Nearest Neighbour	65.61 %	Classification
Naive Bayes Classifier	90.91 %	Classification
XGBoost Classifier	92.42 %	Classification
Linear Regression	M.A.E =5.75	Regression
Random Forest model	85.74 %	Regression

2. Also constructed an artificial neural network (ANN) model to see how it works on dataset. It was able to predict landslides from the test dataset with an accuracy of 89.87%.

3. Using constructed models, did a case study on the June 2013 Uttarakhand Disaster. Model was able to correctly predict 81 percent of the landslides that happened during those times.

4. Were able to deploy prediction model on the website successfully.

5. Using google maps, were able to mark landslide danger zones on the map according to the landslide probability predicted from model, thus enabling users to know whether there are chances of landslide in the region.

References

Following references were used while doing the study-

[1] S. K. Shukla, S. K. Chaulya "Real-Time Monitoring System for Landslide Prediction Using Wireless Sensor Networks", International Journal of Modern Communication Technologies & Research (IJMCTR)

[2] Mark E. Reid, Richard G. LaHusen, "Real-Time Monitoring of Landslides", U.S. Geological Survey

[3] Ahmed MohamedYoussef · Hamid RezaPourghasem “Landslide susceptibility mapping using machine learning algorithms and comparison of their performance at Abha Basin, Asir Region, Saudi Arabia”

[4] Fausto Guzzetti , Stefano Luigi Garian “Geographical landslide early warning systems”

[5] Y. P. Sharda, “Landslide Studies in India”, Glimpse of Geoscience Research in India, the Indian report to IUGS 2004-2008, Indian National Science Academy, Silver Jubilee, 2009, pp. 98-101.

[6] M. T. Brunetti, S. Peruccacci, M. Rossi, S. Luciani , D. Valigi & F. Guzzetti, “Rainfall thresholds for the possible occurrence of landslide in Italy”, Natural Hazards and Earth System Sci, 10, pp. 447 – 458, 2010