# Chapter 1

# 1. Introduction

1.1 Landslide

Landslide can be defined as the movement of rock, debris means mud, or earth downward slope. It is also called landslip. It is a type of mass wasting. Mass wasting mainly denotes any downward slope movement of soil due to gravity. It can also denotes downward movement of rock under the influence of gravity. It can occur underwater, in which case they are called submarine landslide, and in costal and onshore environments. It refers to several forms of mud wasting that include a huge movement of ground such as debris, mud or rock falls, slope failures which is deeply seated, mudflows and debris flows. It surrounds five modes which are modes of slope movement. These modes are falls, topples, glides, spreads, and flows. They are divided by different types of geologic material. Debris flows (commonly mudflow or mudslide) and rock fall are some examples of common landslide types. Almost all landslides have more than one causes. Slope movement is one reason of landslide. It occurs when forces acting on downward slope mainly for gravitational force exceed the strength of the earth materials then it devises the slope. So, that factors are added that increases the effects of downward slope forces which also factors which contributes to low or reduced strength. There are also some other reasons which causes landslide.  Landslides can be due to the verge of movement by rainfall. It can also occur for changing in water level, changing in ground water. Some other reasons are earthquakes, disturbance by human activities, volcanic activity or any combination of these factors. Landslide can be happen on earth surface or underwater due to earthquake and other factors. These types of underwater landslide is called submarine landslides. It sometimes causes tsunamis that damage coastal areas. However we are working with the landslide occurs in Bangladesh. Bangladesh is a small south-asian country. Every year many natural calamities effect Bangladesh. Landslide is one of them. In Bangladesh landslides mainly occur due to heavy rain, heavy pressure on earth surface, deforestation, cultivation, construction, vibration from traffic, earthquake etc. From all of reason heavy rainfall is the important one. It is very common hazard especially in Chittagong. It is the second largest city. Landslide has become a terror here. Chittagong division is full of hill tracts. Some landslide occuring data of Chittagong City Corporation are given below:

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Location** | **Rainfall sequence (Cumulated rainfall)** | **Consequence** |
| 13 August 1999 | Gopaipur , Kotowali Thana , Chittagong | 4535 mm-12 days  2-13 Aug 1999 | 10 people killed |
| 3 August 2005 | Nizam Road Housing Society Of The Port City’s Panchalish Area. | 25 mm-2 days  2-3 Aug. 2005 | 2 people killed and 12 injured |
| 11 June 2007 | Mati Jharna Colony Of Lalkhan Bazar , Chittagong | 610 mm-8 days  4-10 Sept 2007 | 128 people killed and 100 injured |
| 10 September 2007 | Nabi Nagar In Chittagong | 452 mm-7 days  4-10 Sept 2007 | 2 people killed |
| 18 August 2008 | Matijharna In Chittagong | 454 mm -11 days  8-18 August 2008 | 11 people killed and 25 injured |
| 26 June 2012 | Chittagong (Lebubagan Area And Foys Lake Surrounding’s) | 889 mm -8 days  19-26 June 2012 | 90 people killed and 150 injured |
| 01 July 2011 | Batali Hill, Tigerpass interswction | 3-4 days  01 July 2011 | 15 people killed and 2 injured |
| 28 July 2013 | **Lalkhan Bazar (**Akbar Colony in the Tangkir Pahar area | 1 day  28 July 2013 | 2 people killed |

Table 1: Some landslide occurring data in Chittagong

**1.2 Artificial Neural Network**

Artificial Neural Network (ANN) is an information processing system which is inspired by the biologically human nervous systems as like as human brain process information. Information processing system of this system is the key structure. Artificial Neural Network gather knowledge simultaneously with detecting the patterns and relationships among huge number of data and trained by experience, not from programming. Artificial Neural Network is formed thousands of single neuron or information processing unit. These all are connected with coefficients which is also called weights that build neural structure and organized in layers. The accuracy power of the network comes from the connection of neurons in the network. Every processing element has weighted inputs, transferring function and one output. The behavior of an artificial neural network mainly depends on the learning rule of the network, architecture and activation function. Weights are such type of parameters which can be adjustable. Then we need to multiple input value with weight. We get output of the neuron by passing the weighted sum through an activation function. This produce a output of a neuron. Activation function introduces a way of nonlinearity of in the network. When it is trained then the inter-unit connections are optimized until the error in minimized. Then the network reaches on a good level of accuracy. The network is first trained by some dataset. Then it can give new input information to predict. Neural network has remarkable ability to give a solution from complicated or imprecise data. It also can be used for extracting patterns and detection tendency which are too complicated to notice by human or normal computer techniques. A trained neural network has an expert ability to categorize data and analyze. It can handle new situations arise in predicting. The effective feature of this neural network is backtracking. Using backtracking it can adjust its weights. Which is very important. It adjust the weights to get efficient output. A neural network can create own representation of the data during learning time. It can be used in real time operation. Fault tolerance by redundant information coding is an advantage of neural network.

**1.3 Artificial Neural Network used in landslide:**

However in our module we use ANN to predict the probability of occurring landslide. First of all we need to know the reason of landslide. Some reasons of landslide are slope movement, changing in water level, rainfall, earthquakes, changing in ground water, volcanic activity. Some other reasons are disturbance by human activities or any combination of the factors that already narrated. We are predicting the probability of occurring landslide in Bangladesh. Bangladesh is a small south-asian country. Every year many natural calamities effect Bangladesh. Landslide is one of them. In Bangladesh landslides mainly occur due to heavy rain, heavy pressure on earth surface, deforestation, cultivation, construction, vibration from traffic, earthquake etc. From all of reason heavy rainfall is the important one. Landslide also depends on height of the place. It also depends on slope of the spot. Landslide occurs when there are heavy rain for few days continuously. So we need previous rain data. We use these four parameters for our module. In our dataset we collect height of the effected spot. Then collect slope information from GIS (Geographical Information Software). Rainfall data can be collected from weather forecasting website of Bangladesh. Predicting landslide is really very difficult task. Because occurrence of soil like landslide and Earthquake is uncertain. So it is difficult to predict it accurately.

We have four independent parameters and one dependent parameters. Four independent parameters are height, slope, rainfall data and precipitation data of previous five days. These parameters value will multiply with their respective weights. Then calculate the sum of weighted inputs. We use multiple regression for analysis. Before multiplication with weight we need to preprocess the input dataset. We use fit transform to preprocess the input values. We use activation function on the weighted sum. We use sigmoid activation function.

**1.4 Motivation of the work**

Bangladesh is a small south-asian country. Every year many natural calamities effect Bangladesh. Landslide is one of them. In Bangladesh landslides mainly occur due to heavy rain, heavy pressure on earth surface, deforestation, cultivation, construction, vibration from traffic, earthquake etc. From all of reason heavy rainfall is the important one. It is a major threat to both lives and property. Every year there falls a lot of rain. Bangladesh is not free from this hazard. Landslide mainly occur in Bangladesh due to heavy rainfall. Last year on 12 June 2017, heavy rain triggered a series of landslides at Rangamati, Chittagong and Bandarban three hilly districts. It causes death of a lot of people and caused power cuts and telecommunication disrupts. Since 24 June, rains have set off flash floods and landslides in the low-lying areas in the south-eastern districts of Cox’ Bazar, Bandarban and Chittagong. In 2007 Chittagong landdslides occurred in the port city of Chittagong. On 11 June 2007, heavy monsoon rainfall caused landslides that engulfed slums around the hilly areas of the city. Many tourist also fall in landslide hazard on their tour to hilly districts Rangamati, Bandarban and khagrachori. A lot of civilian also affected by the landslide. But if we predict the probabilities of landslides, we can take some pre steps we can reduce the damages. At least we can save the valuable life of people. This motivated me to do this work.

**1.5 Objectives of the project**

The major contributions of our work are illustrated in the following statements:

 To predict the probabilities of occurrence of landslide.

 To develop an online system.

## 1.6 Organization:

This report is organized in different chapters. In chapter 1, we have already introduced about Landslides and its causes, some landslide occurring data, Artificial Neural Network, Application of Artificial Neural Network, motivation of the work and objectives of the work. In chapter 2, we will describe Literature Review. It contains description of different types of activation function and related works and research. In chapter 3 we describe my methodology. It contains overview of methodology, flow chart, different types of packages name, description of preparing dataset, preprocessing of data, description of building model. Chapter 4 contains experimental setup, system layout, scraping data, analysis of data. Chapter 5 contains experimental result and performance evaluation and comparison. Chapter 6 contains conclusion and future research.

# Chapter 2

# Literature Review

## 2.1 Artificial Neural Network

Artificial Neural Network (ANN) is an information processing system which is inspired by the biologically human nervous systems as like as human brain process information. Information processing system of this system is the key structure. Artificial Neural Network gather knowledge simultaneously with detecting the patterns and relationships among huge number of data and trained by experience, not from programming. Artificial Neural Network is formed thousands of single neuron or information processing unit. These all are connected with coefficients which is also called weights that build neural structure and organized in layers. The accuracy power of the network comes from the connection of neurons in the network. Every processing element has weighted inputs, transferring function and one output. The behavior of an artificial neural network mainly depends on the learning rule of the network, architecture and activation function. Weights are such type of parameters which can be adjustable. Then we need to multiple input value with weight. We get output of the neuron by passing the weighted sum through an activation function. This produce a output of a neuron. Activation function introduces a way of nonlinearity of in the network. When it is trained then the inter-unit connections are optimized until the error in minimized. Then the network reaches on a good level of accuracy. The network is first trained by some dataset. Then it can give new input information to predict. Neural network has remarkable ability to give a solution from complicated or imprecise data. It also can be used for extracting patterns and detection tendency which are too complicated to notice by human or normal computer techniques. A trained neural network has an expert ability to categorize data and analyze. It can handle new situations arise in predicting. The effective feature of this neural network is backtracking. Using backtracking it can adjust its weights. Which is very important. It adjust the weights to get efficient output. A neural network can create own representation of the data during learning time. It can be used in real time operation. Fault tolerance by redundant information coding is an advantage of neural network.

### 2.2 Activation function

An activation function sets the output behavior of each node, or neuron in an artificial neural network. This output is then used as input for the next node and so on until a desired solution to the original problem is found. Activation functions are crucial basic architectures of artificial neural networks (ANN), since they introduce non-linear properties to the network. [ deepai.org] Activation function allows the module to learn from complicated datasets. If we don’t use activation function nodal activation would be a linear process. Normally any network frameworks starts with computing the weighted sum of the given inputs of the datasets. The weighted sum need to transform into a format applying squashing function.

**2.2.1 Some Activation function**

There are some different types of Activation function. Some Activation functions are: Uni-polar sigmoid, Bi-polar sigmoid, Tanh, Conic Section, Radial Bases Function (RBF). There are also some other Activation function.

**2.2.1.1 Uni-Polar Sigmoid function**

Equation of Uni-Polar sigmoid function is given below:

g (x)=

Uni-Polar sigmoid function is especially useful to use in neural network which is trained by back-propagation. Because we can easily distinguish easily using this function. It also minimize computation capacity during training. Interval of this function is (-∞, ∞) along x-axis and (0,1) along y-axis.

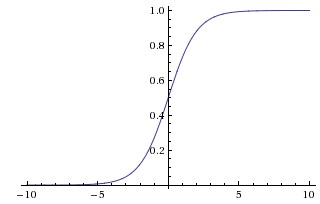


Fig 2.1: Uni-Polar sigmoid function

Because it is easy to distinguish, and this can interestingly minimize the computation

capacity for training. The term sigmoid means ‘S-shaped’, and logistic form of the sigmoid maps

the interval (-∞, ∞) onto (0, 1) as seen in

**2.2.1.2 Bipolar Sigmoid Function**

Equation of Bi-polar sigmoid function is given below:

g (x)=

It is similar to the Uni-polar sigmoid function. It’s interval is (-∞, ∞) along x-axis and (-1,1) along y-axis. A graph of Bi-polar sigmoid function is given below:

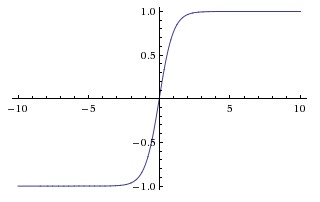


Fig 2.2: Bipolar sigmoid function

**2.2.1.3 Hyperbolic Tangent Function:**

Hyperbolic tangent function can be defined easily as the ratio of hyperbolic sine and cosine function. It also can be defined as ratio of half-difference and half-sum of exponential function of x. The equation is given below:

tanh(x) =

Graph of Hyperbolic tangent function are given below:

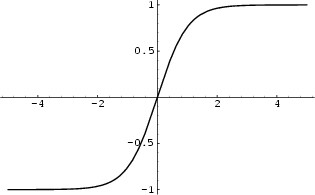


Fig 2.3: hyperbolic tangent function

**2.2.1.4 Radial Basis Function**

It is mainly based on Gaussian Curve. This function takes one parameter which determines the mean value of the function that is used as desired value. A RBF is real valued function. Its value mainly depends only on the distance from the origin. Summation of radial function are typically used to get given functions. This approximate process is interpreted as a simple type of neural network. The equation of RBF is given below:

y(x)=

**2.2.1.5 Binary Step Function with Threshold**

Equation of the function is given below:

y=f(x)= Where ⊖ is a threshold value.

Output will be 0 or 1. That means binary output. Graph of the function I given below:

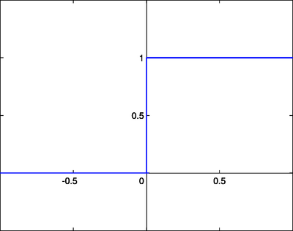


Fig 2.4: Binary step function with threshold

**2.2.1.6 Bipolar Step Function with Threshold**

Equation of Bipolar step function with threshold is given below:

y=f(x)= Where ⊖ is a threshold value.

Output will be 1 or -1 only. Graph of the function is given below:

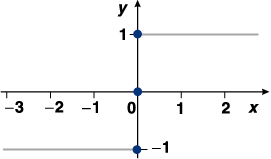


Fig 2.5: Bipolar step threshold function

**2.3 Different types of packages used in coding**

We use python a programming language for coding. Python is resourceful language. It has a lot of useful packages to use. We need to use complex computation for data analysis. In python there are some special packages for data analysis. In our project we use these packages. The used packages are given below:

**2.3.1 Tensorflow**

Tensorflow is an open source software library for data analysis and numerical computation. It is a symbolic math library. In machine learning there are a lot of complex computation which is really very difficult for in normal computation. Tensorflow is specialized package in data science. So, tensorflow is used in machine learning application. Artificial neural network is an algorithm of machine learning. So, tensorflow is used in neural network. Tensorlow allows easy deployment of computation across variety of platforms (CPU, GPUs, TPUs) for its flexible architecture. It also allows deploy computation from desktops to clusters of servers. Engineers and researchers of Google Brain team developed with Google’s AI Organization. It gives a strong support for machine learning and deep learning. It use many other scientific domains.

We use tensor flow in our project. As, Tensorflow is a library for data analysis. So, get a lot of advantage by using tensorflow.

**2.3.2 Keras**

Keras is a Python library. Which is especially for neural network. It is mainly design for fast computation. For data analysis sometimes we need to analyze millions of data for experiment .So system becomes very slow. It takes a lot of time. In that case we can use this package to do fast our system. Keras is mainly design to fast the system. It is capable to run on top of Tensorflow, Microsoft Cognitive Toolkit or Theano. Keras focuses on user-friendliness, modular and extensible. It also developed by Google. It is a part of the research effort of project ONEIROS. Though tensorflow is a famous library used for deep learning, it is not easy to use. On the other hand Keras is user friendly.

**2.3.3 Theano**

Theano is a python library. It optimizes the compiler which manipulate and evaluate mathematical expressions. Theano is especially useful to matrix value. In theano computation it use NumPy syntax. It helps to run the computation on either CPU or GPU architecture. It is developed by University of Montreal. We use this package theano.

**2.4 Related works and research**

C. Lian, Z. Zeng, W. Yao, H. Tang [1] proposed Ensemble of extreme learning machine for landslide prediction based on time series analysis. Landslide depends on many geographical parameters. They divided the parameters into four types of component. These four components are: trend component, periodic component, impulse component and random component. Trend components are like Long-term factor such as slope. Periodic components are like rainfall temperature. Impulse components is fluctuation of water reservoir. Random components are like earthquakes. They use Extreme learning machine for their model. Extreme learning machine is a single-hidden –layer feed-forward neural network.

Ahmed Mohamed Youssef, Hamid Reza Pourghasemi, Zohre Sadat Pourtaghi, Mohamed M. Al-Katheeri [2] used RF, BRT, CART, and GLM for predicting landslide. They Also compare these four algorithm. RF is an ensemble-learning technique. It generates many classification trees that are aggregated to compute a classification. RF increases the diversity among all classification trees by resampling the data. BRT is a combination of statistical and machine learning techniques. CART is a rule-based algorithm. It generates a binary tree using “binary recursive partitioning”. Which is a process that divides a node into yes/no answers as predictor values. GLM can be obtained depending on extension of general linear models.

X. Cui, X. Zhao, M. Ji, S. Wang, P. Zhang [3] used Support vector machine (SVM) to predict the occurrence of landslide. Support vector machine is a machine learning algorithm.

Dieu Tien Bui, Tran Anh Tuan, Harald Klempe, Biswajeet Pradhan, Inge Revhaug [4] used Support Vector Machine (SVM), Artificial neural network, Kernal logistic regression, Logistic model tree for predicting landslide. They also compare the result between these algorithms.

Milos Marjanovic, Milos Kovacevic, Branislav Bajat, Vit Vozenilek [5] used support vector machine for predicting landslide susceptibility. In this paper they introduces support vector machine for solving spatial modeling problems.

Binh Thai Pham, Biswajeet Pradhan, Dieu Tien Bui, Indra Prakash, M.B. Dholakia[6] implemented Support vector machine, Linear Regression, FLDA, BN and NB algorithms to predict landslide susceptibility assessment. They also compare performance of these algorithm.

Biswajeet Pradhan [7] used back propagation artificial neural network for landslide susceptibility. They also compare the frequency ratio and bivariate logistic regression modelling. They use slope angle, slope aspect, curvature, altitude soil type as parameter. They get best accuracy 94% and worst accuracy 91%.

# Chapter 3

# Methodology

**3.1 Overview of Methodology:**

In this project we will predict the probability of happening landslide. One need to select date and identify the latitude and longitude of the desired place. In this system we use Artificial Neural Network for prediction. We use multiple linear regression. Multiple linear regression is a classifier which is used to explain the relationship between one continuous dependent variable with two or more independent variables. First of all import dataset. Then we need to find the dependent and independent variable. For this we need to know the relationship of dependent and independent variable. We need to split variables into dependent and independent variables. Then we need to scale the feature. For feature scaling we have used fit\_transform or transform method which need to import from sklearn.preprocessing. We use Tensorflow, Keras, Theano three software library of Python. Tensorflow is specialized package in data science. It allows easy deployment of computation. Keras is designed to fast the system. It is capable to run on the top of Tensorflow, Theano. Then we build the model. We use keras to build the model. We run tensorflow on the backend. We import Sequential from keras.model and Dense from keras.layer. Dense is used for building the layers. Sequential is used for building the model. We use here activation function. We use sigmoid and relu as activation function. We use relu activation function on input and hidden layer. We use sigmoid function as activation function on output layer. Then we train our module with our train dataset. We use adam as optimizer. After training our system we test the system with test dataset. Then we calculate the accuracy of the system. If the accuracy is satisfied then we can use the model for prediction. If accuracy is not satisfied then we need to train the system with train dataset.

**3.2 Flowchart**

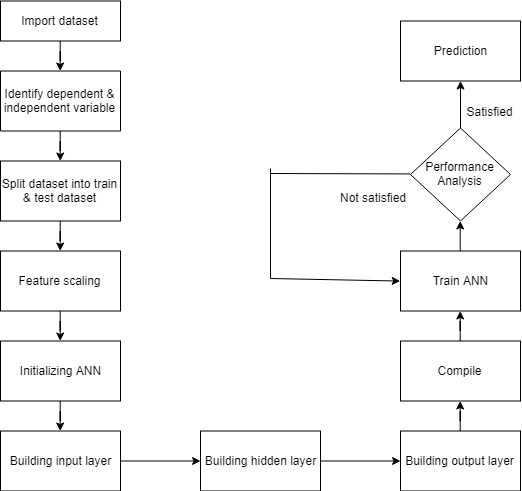


Fig 3.1: Flowchart of methodology

**3.4 Preparing dataset**

Dataset plays an important role in Machine learning or any prediction model. We are predicting the probability of occurring landslide. So, we need a suitable dataset for using in our model. For prediction we need four types of data. One parameter is height, second one is slope. Third parameter is rainfall data. Forth parameter is rainfall data of five days. We don’t found suitable dataset. So, we prepare our dataset ourselves.

**3.4.1 Landslide occurring data**

First of all we collect landslide occurring data. We collect landslide occurring data of Cox’s Bazar district from a government landslide report named. We collect latitude and longitude of landslide affected area.

**3.4.2 Collecting height, slope and soil data**

We also get some height and slope data from there. Rest of height and slope data we collect from a software named GIS. Full form of GIS is Geographic Information system. A geographic information system is a system which is designed to capture, store, manipulate, analyze, manage, and present spatial or geographic data. Using this software we can get height and slope of a place using latitude and longitude. For getting height or elevation data we need a raster image of the place. From this raster image we get elevation. For slope data we need DEM data. So, we need to collect DEM of the place. GIS has a tool named slope analysis. We apply the tool on DEM of that place. We will get slope of that place. Thus we get slope and height or elevation of landslide affected place.

**3.4.3 Collecting Rainfall data**

Our third parameter is rainfall data. We collect this data from weather forecast website. For getting rainfall data or precipitation data we need to use web scraping. We get rainfall data of previous rain data also using web scraping. We use web scraping algorithm for getting pracipitationor rainfall data.

**3.4.3.1 Web scraping algorithm**

Input: User’s day

Require: Write data into a CSV file.

1. Begin

2. Access into using url

3. Fetch expected portion using attribute using user’s input

4. Prepare the data for your dataset

5. Write on csv file

6. End

**3.4.4 Sample dataset**

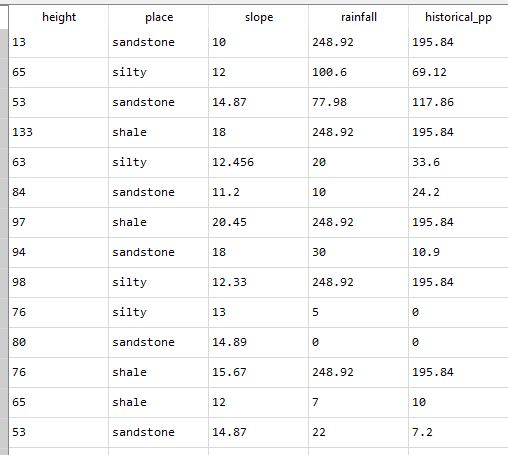
****

Fig 3.2: Sample dataset

**3.5 Preprocessing of dataset**

For using the data of our dataset we need to preprocess the input data. Preprocessing process are given below:

**3.5.1 Importing dataset**

First of all we need to import our data. For doing this we need to import panda library of python. It has a method to read csv file. Using this we import our dataset.

**3.5.2 Split variables into dependent and independent**

Then we need to find out dependent and independent variable. Then split variables into dependent and independent variables.

**3.5.3 Split train and test dataset**

Then we need to split dataset into train dataset and test dataset. For doing this we can use sklearn.model\_selsction a python library.

**3.5.4 Feature scaling**

We need to scale our input. If we use feature scaling our system will be fast. For our project we use Standard Scaler which assumes that our data is normally distributed within each feature. The mean and standard deviation are separately calculated first. Then the feature is based on:

By using this all features are now on same scales. If we don’t transform all feature into same scale, system will not work properly. So, we need to scale all features on same scale. We use fit\_transform. Transform performs standardization by centering and scaling. Fit computes the mean and std to be used for later scaling.

**3.6 Building Model**

Then we need to build the model. We use Keras and Tensorflow to build the model. Keras is a python library. It is design for fast computation. It has library for building model. It has also library for adding layer. So, for building model we need to add layer. For our project we use three layer. First layer or input layer. Second layer hidden layer. Next output layer. For first and second layer we use ReLU activation function. ReLU function is called rectifier function. A unit employing the rectifier is also called a rectified linear unit (ReLU). We use sigmoid function in output layer.

Then we give train dataset to our model. By this train dataset we train our system. Then we use test data set to test and calculate the accuracy. If accuracy is satisfied. Then we can use it as a prediction model. We use Adam for Optimizing algorithm. The short explanation on Adam is given below:

**3.6.1 Adam (Optimizer Algorithm)**

Adam is an Optimization algorithm. Which is first-order gradient-based optimization of stochastic objective functions. It is based on adaptive estimates of lower-order moments. The method is easy to implement. It is computationally efficient and has little memory requirements. Adam is invariant to askew rescaling of the gradients, and is appropriate for issues that are substantial as far as information as well as parameters. The technique is additionally suitable for non-stationary targets and issues with extremely uproarious or potentially scanty inclinations. The hyper-parameters have instinctive understandings and normally require small tuning.

Adam optimizer has six parameters: learning rate, beta\_1, beta\_2, epsilon, decay, amsgrad.

**3.7 Back propagation**

Neural networks can learn their weights and biases. Backpropagation is used for updating their weights and biases. Backpropagation works far faster than earlier approaches to learning, making it possible to use neural nets to solve problems which had previously been insoluble.  Backpropagation algorithm is the workhorse of learning in neural networks. Backpropagation is an expression for the partial derivative ∂C/∂w of the cost function CC with respect to any weight w (or bias b) in the network.  The expression tells us how quickly the cost changes when we change the weights and biases. Let's begin with a notation which lets us refer to weights in the network in an unambiguous way. We'll use wljk to denote the weight for the connection from the k th neuron in the (l−1)th layer to the jth neuron in the lth layer. We use a similar notation for the network's biases and activations. Explicitly, we use blj for the bias of the jth neuron in the lth layer. And we use alj for the activation of the jth neuron in the lth layer.

where the sum is over all neurons k in the (l−1)th layer. To rewrite this expression in a matrix form we define a weight matrix wl for each layer l. The entries of the weight matrix wl are just the weights connecting to the lth layer of neurons, that is, the entry in the jth row and kth column is wljk. Similarly, for each layer l we define a bias vector bl. We can update weight value using backpropagation.

# Chapter 4

# Implementation and Experimental result

**4.1 Experimental setup**

For running this project we need to have django in our PC. Django is a framework of python. We also need to import some python packages. These packages are: HttpResponse, render, tensorflow, keras, Sequential, Dense, numpy, matplotlib.pyplot, seaborn, urlopen, BeautifulSoup, pandas, StandardScaler. We need to put csv file and python code in same folder.

**4.2 System layout**

We have an User Interface for getting input from users.

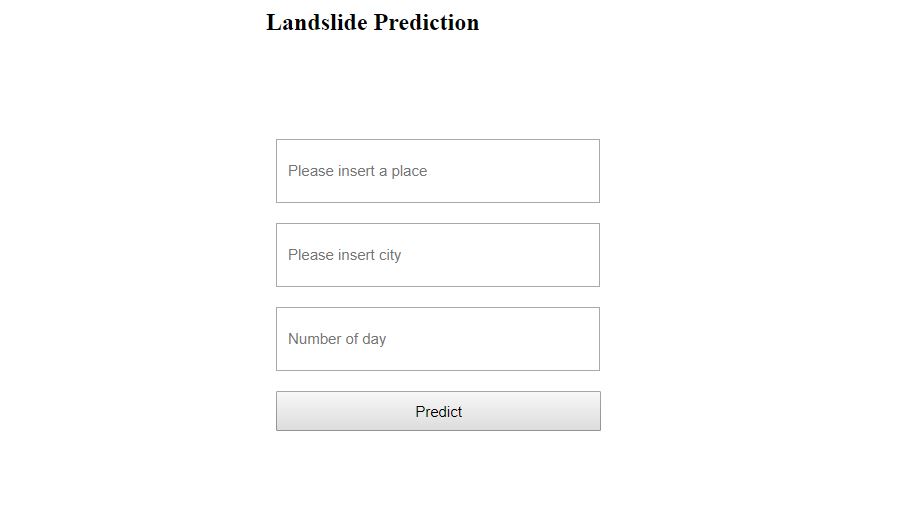


Fig 4.1: User Interface

In this page we take name of the place where user want to go for tour. We also take city name and number of day when user want to go for the tour. We use number of day and city name for getting rainfall and historical rain data. We use place name for getting height and slope data.

## 4.3 Scrape rainfall data from weather forecast website

We get day value. We use the day value to scrape rainfall data from website. Source code is fiven below:

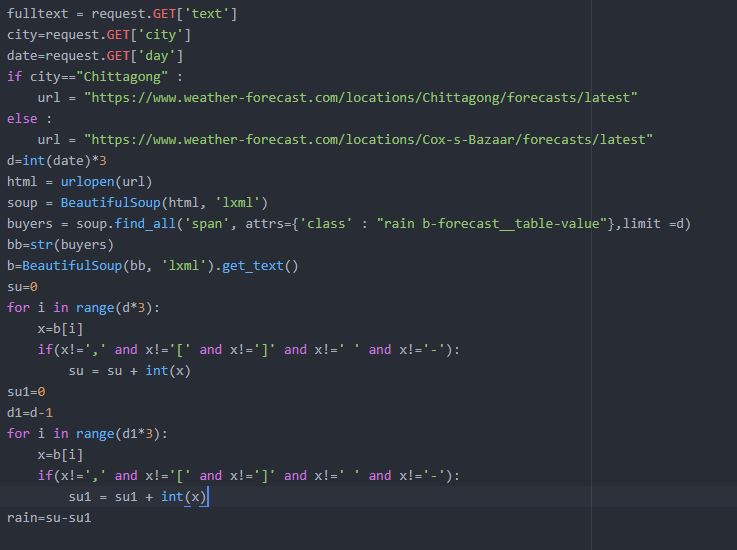


Fig 4.2: Source code for webscraping

Here we use BeautifulSoap for scraping from the url. We use urlopen function to get the HTML of the page. When we get html of that url we can split our expected portion from html. We get expected portion using attribute name. Suppose we split all span class name “rain b-forecast\_table-value”.

Output of web scraping:

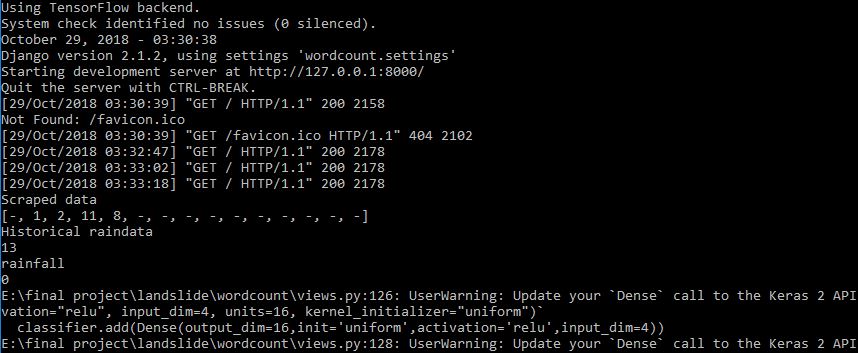


Fig 4.3: Output of web scraping

**4.4 Pre-processing data for ANN model**

We need to preprocess data for using in any model. So, we need to import data set first.

**4.4.1 Import dataset**

For analyzing any dataset we need to import it first. For importing need a python library named panda. Using read\_csv function of this library we can any dataset.

Source code for importing dataset:

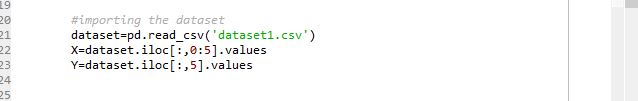


Fig 4.4: Importing dataset

Output:

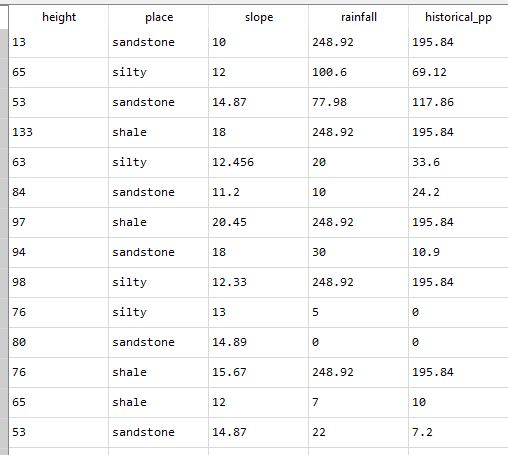


Fig 4.5: Dataset

**4.4.2 Preprocess the dataset value**

We need to scale our input. If we use feature scaling our system will be fast. For our project we use Standard Scaler which assumes that our data is normally distributed within each feature. The mean and standard deviation are separately calculated first. Then the feature is based on:

By using this all features are now on same scales. If we don’t transform all feature into same scale, system will not work properly. So, we need to scale all features on same scale. We use fit\_transform. Transform performs standardization by centering and scaling. Fit computes the mean and std to be used for later scaling.

Source code:

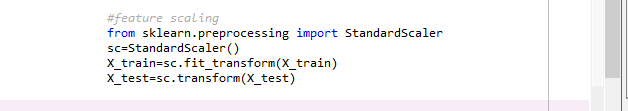


Fig 4.6: Preprocess dataset

Output:

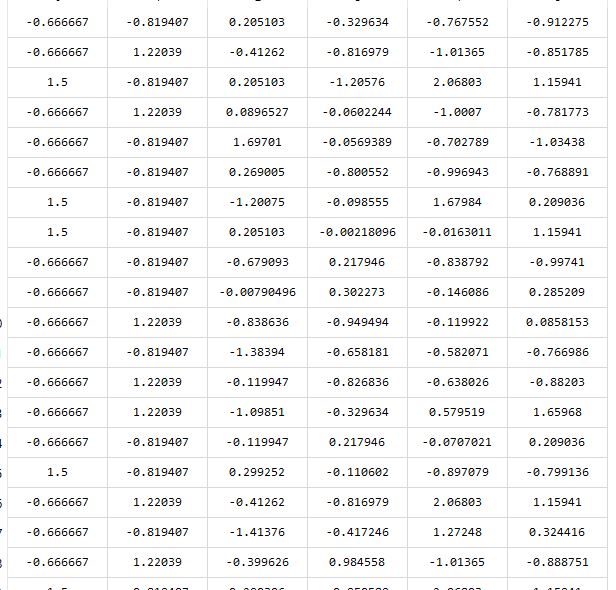


Fig 4.7: Scaled data

**4.5 Analyzing data**

Now we have all the dataset value in standardized form. So, we can use it for model. Now apply it on our model. For that we need to write the following command :

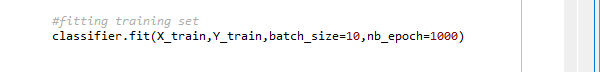
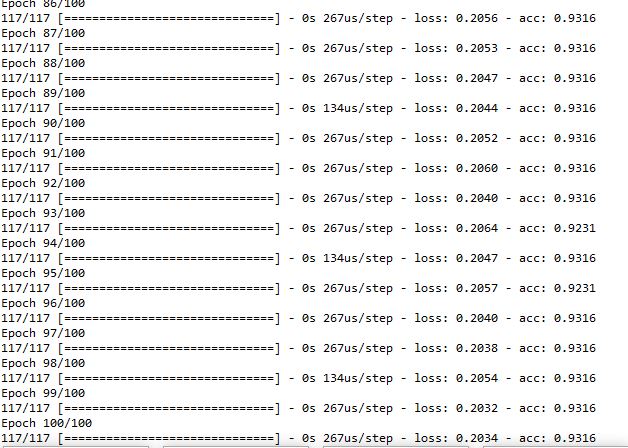


Fig 4.8: Train dataset

We give X\_train as input and Y\_train as output to the ANN model. Then it start our inner calculation of our model.

Here is our model output:

 Fig 4.9: Training dataset

So, we get our final accuracy 93.16%. We think this is much better accuracy.

# Chapter 5

# Experimental Results and Performance Evaluation

## 5.1 Experimental Results

## 5.1.1 Analyzing test dataset

For analyzing any dataset we need to import it first. For importing need a python library named panda. Using read\_csv function of this library we can any dataset. We need to do feature scaling. If we don’t transform all feature into same scale, system will not work properly. So, we need to scale all features on same scale. We use transform. Transform performs standardization by centering and scaling. Suppose we have a following test dataset:

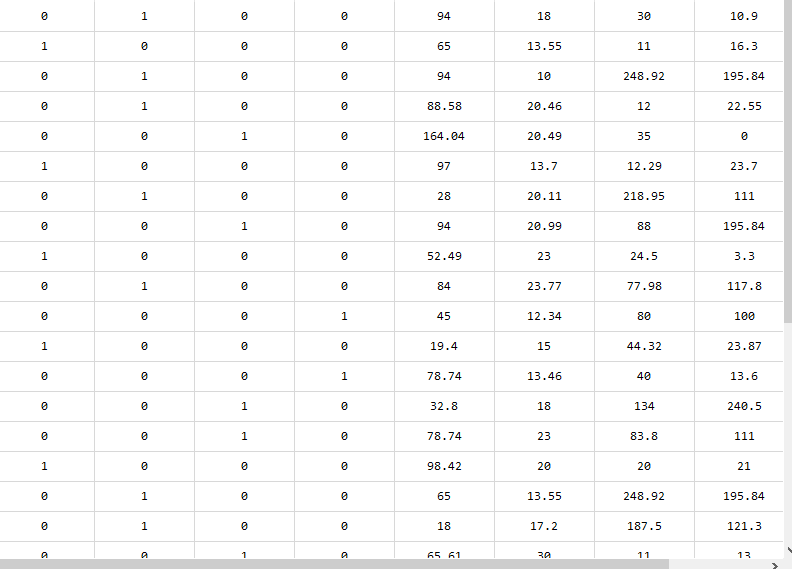


Fig 5.1: Test dataset

Now applying transform on this dataset we get:

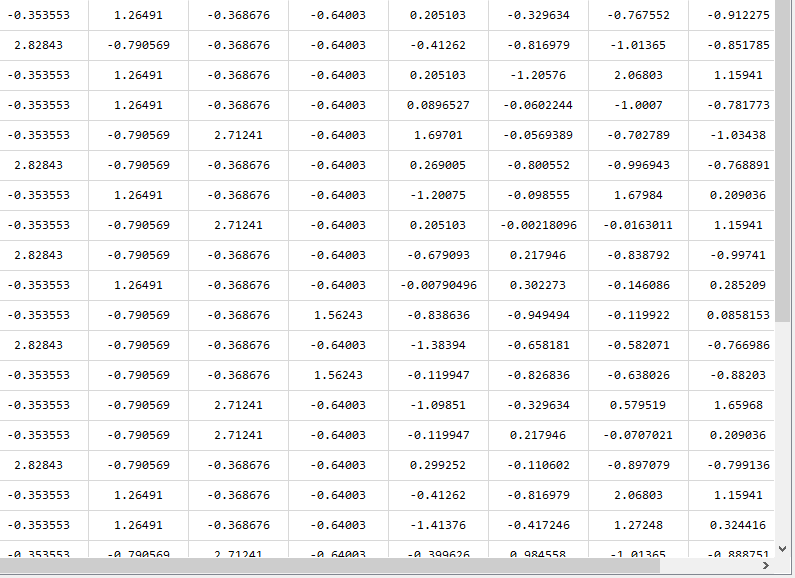


Fig 5.2: Transformed test data

Now data is ready for testing. Apply it on our model.

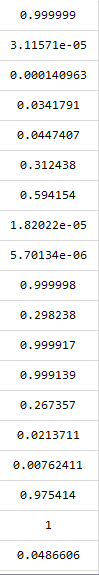
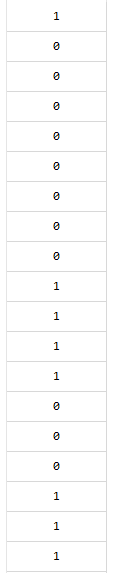
 

Fig 5.3: Predicted value Fig 5.4: Actual value

So, we get predicted value from our model and actual value from test dataset. As, Accuracy is 94.16%. So we get predicted value near to the actual value.

If we assume a threshold value 0.8. That means if predicted value is greater 0.8 we

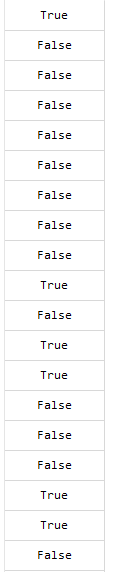
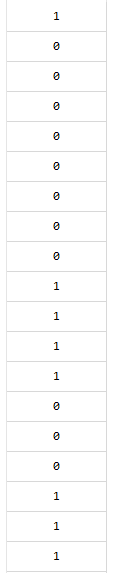
 

Fig 5.6: Value after thresholding Fig 5.7: Actual value

We test our system using 117 data. From them 66 was actually true we got 58 true and 8 false. Rest of 51 was false we got 51 as a false output.

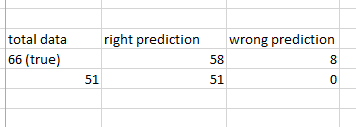


Fig 5.8: Experiment result

Showing result in Bar graph:

Fig 5.9: Bar graph of Experimental result

Showing comparison in graph:

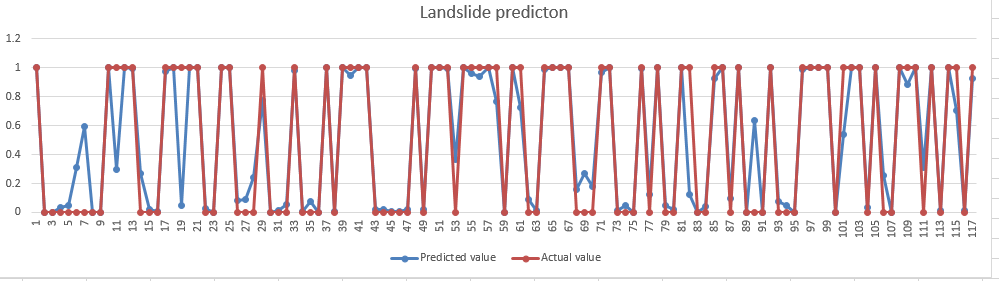


Fig 5.10: Graph of our result

Total data=117

Right prediction=111

Wrong prediction=8

Accuracy = 111/117\*100 %

=92.66 %

So we got an accuracy 92.66%.

**5.2 Comparison**

Bangladesh is a landslide prone country. So, almost every year Bangladesh is effected by landslide. It is a hot topic for research. So, there are a lot of research work on this topic. But there is no research work on landslide prediction using artificial neural network or any other machine learning algorithm in Bangladesh. There are a lot of landslide research for hill track area such as Chittagong, Rangamati, Bandorban, Kagrachori, Cox’s Bazar. They analyze data using GIS (Geographic Information System) which is a system designed to capture, store, manipulate, analyze, manage, and present spatial or geographic data. But they don’t use machine learning algorithm. In our project we use neural network for landslide prediction.

There have some research work on landslide using Machine learning algorithm in outside of the Bangladesh. Some comparisons are given below:

B. T. Pham, B. Pradhan, D. T. Bui [6] use Fisher’s Linear Discriminant Analysis (FLDA) Logistic regression, Bayesian Network, Naïve Bayes (NB). They got accuracy 92.4% for Logistic regression, 92.2% for Fisher’s Linear Discriminant Analysis (FLDA). 91% for BN and NB. In our project we get an accuracy about 92.66%. We use Multi-layer regression.

A. M. Youssef, H. R. Pourghasemi, Z. S. Pourtaghi, M. M. Al-Katheeri [2] used RF, BRT, CART, and GLM for predicting landslide. They also compare these four algorithm. RF is an ensemble-learning technique. It generates many classification trees that are aggregated to compute a classification. BRT is a combination of statistical and machine learning techniques. CART is a rule-based algorithm. It generates a binary tree using “binary recursive partitioning”. Which is a process that divides a node into yes/no answers as predictor values. GLM can be obtained depending on extension of general linear models. They perform their experiment at Soudi-Arabia. They use slope, rainfall data, land use, curvature, altitude. They got accuracy 81.2% for Random forest, 85.6% for BRT, 86.2% for CART, 76.9%for GLM. We got accuracy 92.66% for our proposed methodology.

D. Kawabata, J. Bandibas used geological data, DEM from Aster images and artificial neural network (ANN). This method contains two major phases. First one is the data integration and analysis. Second one is artificial neural network training and mapping. Used parameters are slope, elevation, geology, density of geological boundaries. They got 90% accuracy of predicting landslide. We use five parameters height, slope, soil type, rainfall data, historical rainfall data. In this project we use Multi-layer regression. In this system we got 92.66%.

**Chapter 6**

**Conclusion and Future Research**

## 6.1 Conclusion:

Every year a lot of people died due to landslide. Bangladesh is a small south-asian country. Every year many natural calamities effect Bangladesh. Landslide is one of them. In Bangladesh landslides mainly occur due to heavy rain, heavy pressure on earth surface, deforestation, cultivation, construction, vibration from traffic, earthquake etc. Most of the landslide occur in Bangladesh at hill track area. Mainly in Chittagong, Rangamati, Bandorbon, Kagrachori and Cox’s Bazar. Some people and some tourist also died due to landslide. These motivated us to do this project. Our project will help especially tourist. In our project we use artificial neural network which is a Machine learning algorithm. We use Multi-layer regression. We use five parameters height, slope, soil type, rainfall data, historical rainfall data. We use python a programming language, django which a framework of python. We use HTML, CSS in frontend. We build up our system using python. We got an accuracy of 94%.

## 6.2 Suggestions for Further Research:

Landslide is a hot topic for research. Every year a lot of landslide occur around the world. So, anyone can it as a research topic. But there is a problem here. There is not enough data available on this topic. In our project we face a lot of problem in collecting data. We use five factor height, slope, soil type, rainfall data, historical rain data. But there are some other factors on which landslide depends. So, anyone can add some other parameters. If anyone add some other parameters accuracy will be increased. For our online system we take height, slope and soil type information form a file. This file contains these data of some tourist place of Bangladesh. Anyone can build a system which is fully automated. All data will be taken from online sources.

**References**

[1] C. Lian, Z. Zeng, W. Yao, H. Tang “*Ensemble of extreme learning machine for landslide displacement prediction based on time series analysis.*” Neural Computing and Applications 24, no. 1 (2014): 99-107.

[2] A. M. Youssef, H. R. Pourghasemi, Z. S. Pourtaghi, M. Al-Katheeri “*Landslide susceptibility mapping using random forest, boosted regression tree, classification and regression tree, and general linear models and comparison of their performance at Wadi Tayyah Basin, Asir Region, Saudi Arabia*” Landslides DOI 10.1007/s10346-015-0614-1.

[3] X. Cui, X. Zhao, M. Ji, S. Wang, P. Zhang “*Research on landslide prediction based on support*

*vector machine.*" In Computer Design and Applications (ICCDA), 2010 International Conference on,vol. 3, pp. V3-540. IEEE, 2010.

[4] D. T. Bui, T. A. Tuan, H. Klempe, B. Pradhan, I. Revhaug “*Spatial prediction models for shallow landslide hazards: a comparative assessment of the efficacy of support vector machines, artificial neural networks, kernel logistic regression, and logistic model tree*” Landslides (2016) 13:361–378 DOI 10.1007/s10346-015-0557-6.

# [5] M. Marjanovic, M. Kovacevic, B. Bajat, V. Vozenilek “*Landslide susceptibility assessmentusing SVM machine learning algorithm*”https://doi.org/10.1016/j.enggeo.2011.09.006.

[6] Binh Thai Pham, Biswajeet Pradhan, Dieu Tien Bui, Indra Prakash, M.B. Dholakia “*A comp-*

*arative study of different machine learning methods for landslide susceptibility assessment: A case*

*study of Uttarakhand area (India)”* © 2016 Elsevier Ltd.

[7] Biswajeet Pradhan “*Landslide susceptibility assessment and factor effect analysis: backpropagation artificial neural networks and their comparison with frequency ratio and bivariate logistic regression modelling*” © 2016 Elsevier Ltd.

[8] Micheletti, Natan, L. Foresti, S. Robert, M. Leuenberger, A. Pedrazzini, M. Jaboyedoff, and M. Kanevski."*Machine learning feature selection methods for landslide susceptibility mapping.*" Mathematical Geosciences 46, no. 1 (2014): 33-57.

[9] http://www.cbc.ca/beta/news/landslideprediction.

[10] Kitutu, M. G., A. Muwanga, J. Poesen, and J.A. Deckers. "*Influence of soil properties on landslide occurrences in Bududa district, Eastern Uganda.*" African journal of agricultural research 4, no. 7 (2009): 611-620.

[11] htttps://en.wikipedia.org/wiki/Artificial\_Neural\_Network.

[12] Krkač, Martin, D. Špoljarić, Sanja Bernat, and Snježana Mihalić Arbanas. "*Method for prediction of landslide movements based on random forests.*" Landslides 14, no. 3 (2017): 947- 960.

# [13] D. T. Bui, T. A. Tuan, H. Klempe” *Spatial prediction models for shallow landslide hazards: a comparative assessment of the efficacy of support vector machines, artificial neural networks, kernel logistic regression, and logistic model tree”* DOI 10.1007/s10346-015-0557-6.

**Appendix**

Code of web scraping:

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from urllib.request import urlopen

from bs4 import BeautifulSoup

url = "https://www.weather-forecast.com/locations/Chittagong/forecasts/latest"

d=int(date)\*3

html = urlopen(url)

soup = BeautifulSoup(html, 'lxml')

buyers = soup.find\_all('span', attrs={'class' : "rain b-forecast\_\_table-value"},limit =d)

bb=str(buyers)

b=BeautifulSoup(bb, 'lxml').get\_text()

print("Scraped data")

print(b)

su=0

for i in range(d\*3):

x=b[i]

if(x!=',' and x!='[' and x!=']' and x!=' ' and x!='-'):

su = su + int(x)

print("Historical raindata")

print(su)

su1=0

d1=d-1

for i in range(d1\*3):

x=b[i]

if(x!=',' and x!='[' and x!=']' and x!=' ' and x!='-'):

su1 = su1 + int(x)

rain=su-su1

Code of neural network:

import tensorflow

import keras

from keras.models import Sequential

from keras.layers import Dense

import numpy as np

import matplotlib.pyplot as plt

import pandas as pd

#importing the dataset

dataset=pd.read\_csv('E:\dataset1.csv')

X=dataset.iloc[:,0:5].values

Y=dataset.iloc[:,5].values

#Encoding catagorial data

from sklearn.preprocessing import LabelEncoder, OneHotEncoder

labelencoder\_X\_1=LabelEncoder()

X[:, 1]=labelencoder\_X\_1.fit\_transform(X[:,1])

Onehotencoder=OneHotEncoder(categorical\_features=[1])

X= Onehotencoder.fit\_transform(X).toarray()

X=X[:,1:]

#splitting train and test dataset

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test,Y\_train,Y\_test=train\_test\_split(X,Y,test\_size=0.2,random\_state=0)

#feature scaling

from sklearn.preprocessing import StandardScaler

sc=StandardScaler()

X\_train=sc.fit\_transform(X\_train)

X\_test=sc.transform(X\_test)

#importing keras lib and packages

#initializing ANN

classifier =Sequential()

#Adding input and first hidden layer

classifier.add(Dense(output\_dim=16,init='uniform',activation='relu',input\_dim=8))

#Adding second hidden layer

classifier.add(Dense(output\_dim=32,init='uniform',activation='relu'))

#Adding output layer

classifier.add(Dense(output\_dim=1,init='uniform',activation='sigmoid'))

#compiling ANN

classifier.compile(optimizer='adam',loss='binary\_crossentropy',metrics=['accuracy'])

#fitting training set

classifier.fit(X\_train,Y\_train,batch\_size=10,nb\_epoch=100)

#predicting test set

y\_pred=classifier.predict(X\_test)

Code of html, css:

Homepage:

<!DOCTYPE html>

<html lang="en-US">

<head>

<meta charset="UTF-8">

</head>

<body style="padding-left:100px;">

<div class="header" >

<div class="header-1" >

<h2 style="height:100px;weight:100% background-color:#150F6F; padding-left:450px;">Landslide Prediction</h2>

</div>

<div >

</div

</div>

<div style="padding-left:450px;">

<form class="" action="{% url 'count' %}" method=""> <!-- url info matches name in urls.py

so if you change name of file, it still loads because urls.py has name= to refer to -->

<input type="text" name="text" style="padding:10px;height:40px;font-size:15px; font:Arial; margin:10px; width:300px;" placeholder="Please insert a place" required/>

<br />

<input type="text" name="city" style="padding:10px;height:40px;font-size:15px; font:Arial; margin:10px; width:300px;" placeholder="Please insert city" required/>

<br />

<input type="int" name="day" style="padding:10px;height:40px;font-size:15px; font:Arial; margin:10px; width:300px;" placeholder="Number of day" required/>

<br />

<input type="submit" name="" value="Predict" style="padding:10px;height:40px;font-size:15px; font:Arial; margin:10px; width:325px;"/>

</form>

</div>

</body>

</html>

Code of result page:

<body>

<a href="{% url 'home' %}">Return to home page</a>

<!-- <h1>{{ count }} words were sent to Terminal</h1> -->

<!-- 'count' is key of len(wordlist) in views.py count()'s dictionary -->

<div style="padding-left:200px;">

<p>Our prediction vaue:</p>

{{ prediction }}%

</div>

<h1>Place Location in Map:</h1>

<div id="map" style="width:80%;height:500px; padding-left:200px;"></div>

<script>

function myMap() {

var myCenter = new google.maps.LatLng({{lat}},{{lon}});

var mapCanvas = document.getElementById("map");

var mapOptions = {center: myCenter, zoom: 5};

var map = new google.maps.Map(mapCanvas, mapOptions);

var marker = new google.maps.Marker({position:myCenter});

marker.setMap(map);

map.setZoom(15);

// Zoom to 9 when clicking on marker

google.maps.event.addListener(marker,'click',function() {

map.setZoom(9);

map.setCenter(marker.getPosition());

});

}

</script>

</script>

<script async defer

src="https://maps.googleapis.com/maps/api/js?key=AIzaSyC5Ed9wVC6\_VoDRhIJoEfqNSJtPU4VnC08&callback=myMap">

</script> -->

</body>