**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

**WINTER SEMESTER - 2022**

# Rainfall Prediction using Machine Learning Techniques

A Report

*submitted by*

**Khyati Pareek, Adithi Kusupudi, Srishti Sinha**

**(20BDS0299, 19BCE0588, 20BDS0329)**

*CSE3009 – Internet of Things – J Component*

*B1 Slot*

Faculty Name – Dr.A.Saraswathi Priyadharshini,

Department of Analytics,

SCOPE

**TABLE OF CONTENTS**

# CHAPTER 1

# INTRODUCTION

Rainfall plays important role in forming of fauna and flora of natural life. It is not just significant for the human beings but also for animals, plants and all living things. It plays a significant role in agriculture and farming and undoubtedly; water is one of the most natural resources on earth. The changing climatic conditions and the increasing greenhouse emissions have made it difficult for the human beings and the planet earth to experience the necessary amount of rainfall that is required to satisfy the human needs and its uninterrupted use in everyday life. Therefore, it has become significant to analyze the changing patterns of the rainfall and try to predict the rain not just for the human needs but also to predict for natural disasters that could cause by the unexpected heavy rainfalls. To be more specific and aware of the devastating climatic changing and stay updated; predicting rainfall has been the focus of computer scientist and engineers.

Extreme variations in rainfall have a drastic effect on agriculture. Drought can kill crops while heavy rainfall can increase soil erosion and spoil the plantations. Optimum amount of water is required for survival of crops. Too much or too little water is harmful to the crops and can affect the yield in the long run causing major loss to farmers. Hence rainfall is a major factor affecting the crop yield. Therefore, there is a need to predict rainfall for effective use of water resources for crop productivity to give a better yield and decrease agricultural loss.

Rainfall prediction is significant not only on the micro but also on the macro level. The study is of significance with respect to its vital contribution in the field of agriculture, water reserve management, flood prediction and management with an intention to ease the people by keeping them updated with the weather and rainfall prediction. It is also important to be utilized by the agricultural industries for keeping their crops safe and ensure the production of seasonal fruits and vegetables by updated rainfall prediction. The study will also be significant for the flood management authorities as more precise and accurate prediction for heavy monsoon rains will keep the authorities alert and focused for an upcoming event that of which the destruction could be minimized by taking precautionary measures. The rainfall prediction will impressively help in dealing with the increasing issue of water resource management; as water is a scarce resource and it needs to get saved for the benefit of human beings themselves. Also, it will help the people to manage and plan their social activities accordingly.[1]

Existing rainfall prediction methods are large scale/locality based and don 't sense atmospheric parameters for a specific place which can sometimes be a problem. Our project is essentially a setup which can be implemented in wearable caps as well which can be worn by the farmer while working on the field and this device automatically logs atmospheric parameters to the ThingSpeak cloud and predicts the rainfall for that day using its trained model. This makes it device portable and specific to the farmer ' s location and hence eases the decision of the farmer of how much he has to water the plants and avoid agricultural loss.

Rainfall prediction is one of the challenging and uncertain tasks which has a significant impact on human society. Timely & accurate predictions can help to Proactively reduce human and financial loss. Accuracy of rainfall forecasting is very much important for countries like India whose economy is largely dependent on agriculture.

Due to the dynamic nature of the atmosphere, statistical techniques fail to provide good accuracy for rainfall forecasting. Droughts and Floods have been affecting our farmers since very long. Farming is a long process and depends majorly on the rainfall cycle which makes our focus to be farmers of our nation. It is important to exactly determine the rainfall for effective use of water resources, crop productivity and pre-planning of water structures in this project, we have proposed a code implementation for “SMART CAP” for farmers which records the atmosphere parameters while they are working on the field using node MCU and required sensors, this data is logged in thingspeak[2] via the internet and is then processed by our trained prediction model and the predictions are shown on thingspeak which can be accessed from anywhere in the world.

The linear regression method will be modified to obtain the most optimum error percentage by iterating and adding some percentage of error to the input values. This method provides an estimate of rainfall using different atmospheric parameters like average temperature and cloud cover to predict the rainfall. Further we will also modify our model to make it simple for farmers to understand by implementing a logistic regression model which shows if there is sufficient rainfall on that specific day or does he have to water the crops normally.

# CHAPTER 2

# LITERATURE SURVEY

1. *A Data-Driven Approach for Accurate Rainfall Prediction [2019][3]*

*Methodology:*

This paper proposes a systematic approach to analyze various parameters that affect precipitation in the atmosphere. Based on these findings, an optimum set of features are used in a data-driven machine learning algorithm for rainfall prediction.

*Advantages:*

Makes use of atmospheric parameters for prediction instead of using precipitable water vapor (PWV) derived from global positioning system (GPS) signal - delays to predict rainfall Using a 4-year (2012-2015) database shows a true detection rate of 80.4% and a false alarm rate of 20.3%. Compared to the existing literature, our method significantly reduces the false alarm rates.

*Disadvantages:*

The method derived an overall accuracy of 79.6% which is less compared to other models.

2. *Monthly Rainfall Prediction Using Various Machine Learning Algorithms for Early Warning of Landslide Occurrence [2020][4]*

*Methodology:*

The proposed study involves the development of rainfall forecasting models using four different machine learning algorithms for predicting landslide occurrence well in advance. Normalization is performed to enhance the predictive accuracy of the models. The Gradient Descent optimization algorithm is used to train the models. Mean square error is used as the loss function for training linear regression, BPNN and LSTM models whereas Vapnik's ε insensitive loss function is used for training the SVR models. Evaluation of the models is performed using the mean absolute error and root mean square error (RMSE) metrics.

*Advantages:*

Capable of predicting low as well as medium intensity rainfalls effectively. Neural networks (BPNN and LSTM) are capable enough to forecast rainfall with high accuracy without any prior knowledge on the related meteorological parameters influencing rainfall.

*Disadvantages:*

The developed models are however underperformed in mapping high-intensity rainfalls accurately.

**3.** *Rainfall Prediction using Machine Learning & Deep Learning Techniques [2020][5]*

*Methodology:*

Rainfall has been predicted using deep learning techniques. Two deep learning techniques - Multilayer Perceptron and Autoencoders. MLP is used in prediction and classification tasks. Performance of methodology is also evaluated by using RMSE(Root MSE)

*Advantages:*

Accuracy is good ; Past data is used to predict future data Mean square error is less but when performed validation

*Disadvantages:*

Instead of sensors as input device, they have used CNN to take the input from the past data.

**4.** *Real Time Weather Prediction System Using IOT and Machine Learning [2020][6]*

*Methodology:*

Real time weather prediction system that can be used in number of applications like homes, industries, agriculture, stadiums etc. for predicting the weather information. The system utilizes a temperature and humidity sensor i.e. DHT11 and a light intensity sensor i.e. LDR which upload the sensed data to a ThingSpeak cloud server. Further, the model is run through machine learning techniques.

*Advantages:*

Low cost IoT board and sensors are used Predict the weather parameters in real time environment. The result of the proposed system is slightly better in terms of accuracy.

*Disadvantages:*

Mostly applicable for indoor applications only. Work needs to be done to further bring it into use.

**5.** *Rainfall Prediction using Machine Learning and Neural Network[2020][7]*

*Methodology:*

The proposed system predicts rainfall for the approach which is more accurate. The data set is collected. There are two techniques to predict rainfall. The first one is machine learning approach, which includes LASSO regression. The second one is neural network approach. This system first compares both the process and then accordingly gives result with the best algorithm. Steps associated with the proposed system are input of data, preprocess of data, splitting of data, raining of the algorithm, testing of the dataset, comparing both the algorithm, giving the best algorithm, prediction with the more accurate algorithm and result at the end.

*Advantages:*

One of the major advantages of this system is its ability to increase the quality of algorithm and dataset. The more the data's used for prediction will be efficient, the more will be accuracy of the prediction. Its ability to predict real time rain locally . Its ability to store and retrieve the data's which are stored for longer period of time but along with that it also shows the inability of neural network to do prediction for the same.

*Disadvantages:*

One of the biggest disadvantages of this system is its error finding technique. The model fails when it comes for long term estimation. It is limited to very less area.

**6.** *Rainfall Prediction using Machine Learning Techniques and An Analysis of the Outcomes of these Techniques[2020][8]*

*Methodology:*

Physical processes in rainfall are generally composed of a number of sub-processes. A

accurate modelling of rainfall by a single global model is sometimes not possible (Solomatine and Ostfeld, 2008). To overcome this difficulty, the concept of modular modelling and combining different models has attracted more attention recently in rainfall forecasting. There are two main types of Machine Learning approaches; supervised learning and unsupervised

learning. Supervised learning algorithms are used for building predictive models. The Classification algorithms ANN, Logistic Regression, Naïve Bayes, and RandomForest are experimentally implemented and compared against each other.

*Advantages:*

The main advantage of Neural Networks is its ability to display non-linearity existence between the input and output variables. It was concluded after analyzing various models of supervised learning that the Random Forest classification algorithm has appreciable level of accuracy and acceptance. Random forest has one important advantage that it is fast and is able to handle large

number of input attributes.

*Disadvantages:*

Every algorithm has its advantages and limitations; it is difficult to choose

the best algorithm. The prediction accuracy of the model can be increased by developing a hybrid prediction model where multiple machine learning algorithms are put to work together.

**7.** *Prediction of Rainfall Using Machine Learning Techniques [2021][9]*

*Methodology:*

The proposed method is based on the multiple linear regressions. The data for the prediction is collected from the publicly available sources and the 70 percentages of the data are for training and the 30 percentages of the data is for testing. Multiple regressions are used to predict the values with the help of descriptive variables and is a statistical method. It is having a linear relationship between the descriptive variable and the output values.

*Advantages:*

The error-free prediction provides better planning in agriculture and other industries. The power to work out the relative influence of one or more predictor variables to the criterion value. Ability to spot outliers or anomalies.

*Disadvantages:*

Data must be independent.

**8.** *A rainfall prediction model using artificial neural network[2021][10]*

*Methodology:*

They use multilayered artificial neural network and back-propagation-feed forward neural network by building training and testing data sets and finding the number of hidden neurons in these layers for the best performance.

*Advantages:*

Learning by back-propagation algorithm configuration is the most common in use.

*Disadvantages:*

The dataset used was very limited and small. The models under study are different in the number of hidden neurons.

**9.** *Machine learning techniques to predict daily rainfall amount[2021][11]*

*Methodology:*

For this study, the raw data were collected from the regional meteorological station at Bahir Dar City, Ethiopia. Ten data features such as year, month, date, evaporation, sunshine, maximum temperature, minimum temperature, humidity, wind speed, and rainfall were included. The data preprocessing step included the data conversion, manage missing values, categorical encoding, and splitting dataset for training and testing dataset. The rainfall was predicted using a machine learning technique. Three machine learning algorithms such as Multivariate Linear Regression (MLR), Random Forest (RF), and gradient descent XGBoost were analyzed which took input variables having moderately and strongly related environmental variables with rainfall.

*Advantages:*

A comparison of results among the three algorithms (MLR, RF, and XGBoost) was made and the results showed that the XGBoost was a better-suited machine learning algorithm for daily rainfall amount prediction using selected environmental features. This study aimed to identify the relevant atmospheric features that cause rainfall and predict the intensity of daily rainfall using machine learning techniques.

*Disadvantages:*

This paper took environmental features which had a correlation coefficient greater than 0.2 and analysed the rainfall prediction. The accuracy of the rainfall amount prediction may increase if the sensor data is incorporated for the study. But the sensor data was not considered in this study.

**10.** *Efficient Rainfall Prediction and Analysis using Machine Learning Techniques[2021][12]*

*Methodology:*

The proposed system forecasts rainfall with two machine learning techniques: logistic regression and random forest, for a more precise solution. First, the system compares the procedure and provides the best algorithm to the output. Data entry, pre-processing of data, data division, algorithm training, data set checking, comparisons between both algorithms, prediction of the most reliable algorithm, and results at the end are the steps related to the proposed scheme. The collection of data included in this study consists of many parameters and the known class of output. The output class will be predicated on the other values and perspectives. Using the output class in the dataset compares the reliability and consistency of the Machine Learning techniques. The effects of processing are equal to the known output class, and reliability is calculated by the sum of accuracy, recall, and f test of output.

*Advantages:*

This research aims to find a rainfall prediction system that can solve all issues, find complexity and hidden patterns present, and provide proper and reliable predictions. The confusion matrix layout allows visualization of the performance of both algorithms and offers less inaccurate predictions. The accuracy score for the logistic regression algorithm is slightly more efficient than the random forest algorithm.

*Disadvantages:*

This model has the most significant disadvantage: it is restricted to a smaller region.

# CHAPTER 3

# PROPOSED WORK

Variations in rainfall have a drastic effect on agriculture. Too much or too little rain(water) both can harm the crops and the yields. Hence, predicting the rainfall is essential for the effective use of water resources for crops. It is very important to decrease the agricultural loss.

The existing prediction methods are large and locality based and sometimes they don't sense the atmospheric parameters for specific regions. Whereas, our project focuses on building a system which solves these problems and at the same time be transformed into wearable cap as well which can be worn by the farmers themselves. It automatically logs atmospheric parameters to the system. It is portable and eases the work of the farmer.

A bad rainfall prediction can affect the agriculture mostly farmers as their whole crop is dependent on the rainfall and agriculture is always an important part of every economy. So, making an accurate prediction of the rainfall somewhat good. There are a number of techniques used in machine learning but accuracy is always a matter of concern in prediction made in rainfall.

It will also improve on weather prediction systems and there might be personalized apps to regulate people in day-to-day life. These apps can be integrated in the working of organizations centered around agriculture and other businesses related to it.

Since agriculture is often related to the economy of a country, accurate weather prediction will provide great help to the lives of those directly involved in the processing as well as most other citizens.

## *Tools Required*

*Software:*

- Jupyter Notebook - The Jupyter Notebook is a web-based interactive computing platform. It allows us to create and share the document called the Notebook, containing live codes, documentation, graphs, plots, and visualizations. We'll be using this to post the data on the ThingSpeak Cloud.[13]
- ThingSpeak Cloud - ThingSpeak is an IoT analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak.[14]

*Hardware:*

- DHT 11: The DHT11 is a commonly used Temperature and humidity sensor that comes with a dedicated NTC to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity as serial data. [15]

- Anemometer: An anemometer is a device used for measuring wind speed and direction. It is also a common weather station instrument. The term is derived from the Greek word anemos, which means wind, and is used to describe any wind speed instrument used in meteorology.[16]
- Node MCU: NodeMCU is an open-source firmware for which open-source prototyping board designs are available.[17]

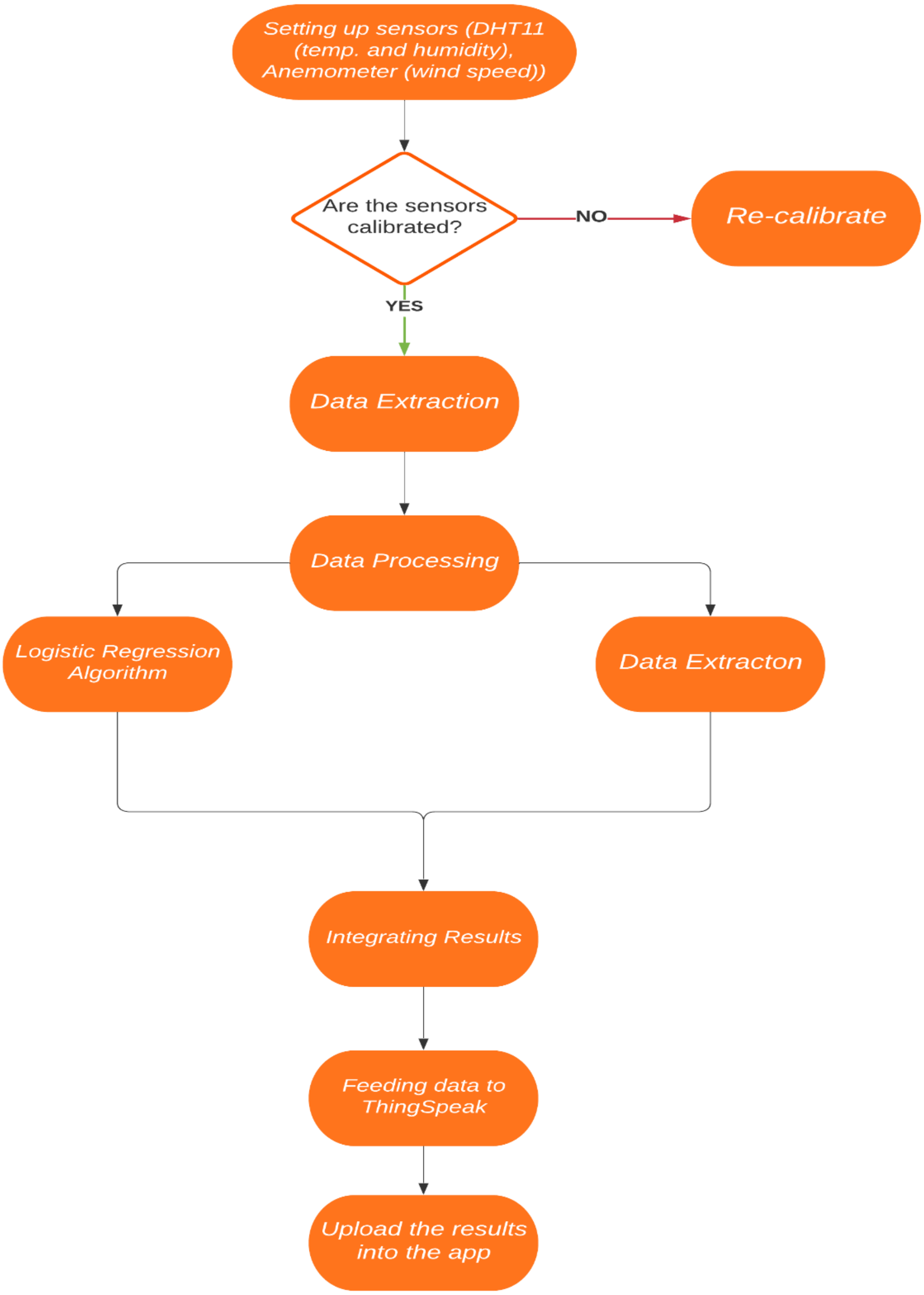



Fig. 1- Proposed Architecture

## *Workflow*

The first step in the work flow is to set up the sensors DHT11 and Anemometer. We then calibrate the sensors. Sensor calibration is an adjustment or set of adjustments performed on a sensor or instrument to make that instrument function as accurately, or error free, as possible.

Data extraction : Disperate types of data is collected and retrieved from a variety of sources.

Data Processing :We then carry out operations on data to retrieve, transform, or classify information.

Linear regression algorithm : Used to describe data and to explain the relationship between data.

Logistic regression algorithm : Used to model the probability of a certain class or event taking place.

Data is logged in thingspeak via the internet and is then processed by our trained prediction model and the Predictions are shown on thingspeak which can be accessed from anywhere in the World.

The final step is to upload the results into the app.

## *Proposed Methodology*

### *Multiple Linear Regression(MLR)*

Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable. For our model the independent variable is precipitation level and dependent variables are temperature, humidity, wind speed, dew point and this model is trained with data of past 3-4 months. This is the training phase, after this our model is ready to take live input data of mentioned dependent variables and predict precipitation level.[18]

### *Logistic Regression*

Multinomial logistic regression is a particular solution to classification problems that use a linear combination of the observed features and some problem-specific parameters to estimate the probability of each particular value of the dependent variable. Hence applying this concept here can give us a categorical prediction depicting rainfall as “No Rain” , “Drizzles” , “Moderate Rains” , “Heavy Rainfall”. This can be easily understandable by farmers as it is in very simple terms. In the same way as linear regression, it is first trained and then ready for use and has the same variables as linear regression but the method is different as it uses a sigmoidal function and a cross entropy function.[19]

*IoT Integration*

This part focuses on sending the real time data to the thingspeak cloud and establishing a connection between our analysis and thingspeak. This is generally easily done using the node MCU and creating a webrequest using thingspeak API. Now as an alternative to this we make the web request to post the data on thingspeak via our python code itself. The “urllib” library helps us do this with the thing speak API of our channel. We have created a new field in thingspeak that represents our output(precipitation level in inches). And the write api key of our channel can be taken from thingspeak to create a POST request from python.

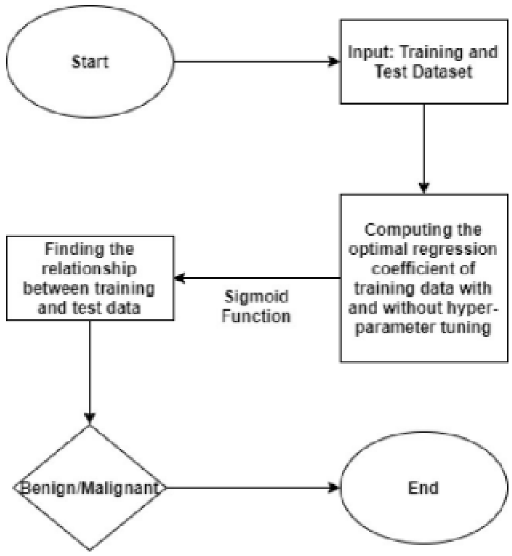



Fig. 2- Implementation Diagram for Multiple Linear Regression

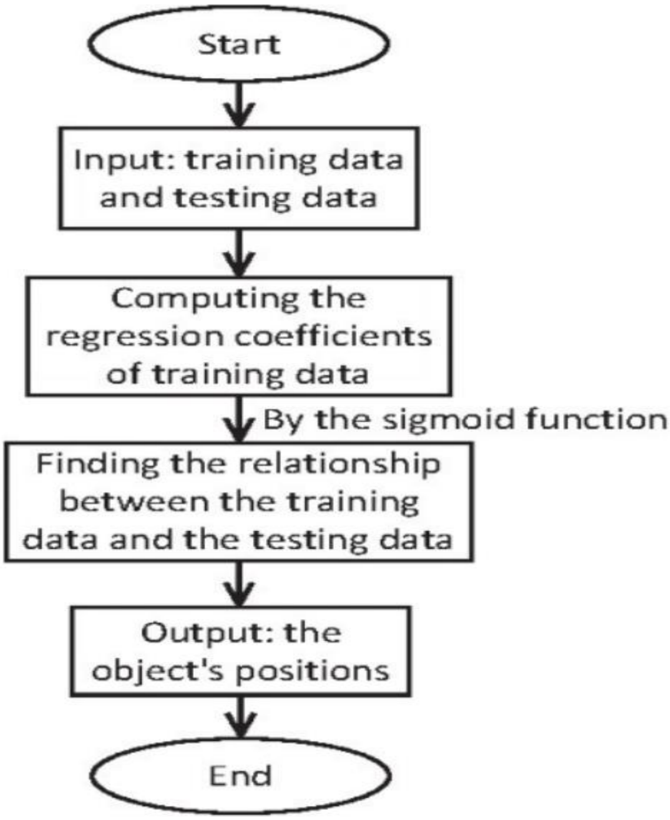



Fig. 3- Implementation Diagram for Logistic Regression

## *Outcome*

A bad rainfall prediction can affect the agriculture mostly farmers as their whole crop is dependent on the rainfall and agriculture is always an important part of every economy. So, making an accurate prediction of the rainfall somewhat good. Hence, we are proposing after a number of techniques used in machine learning and will try to propose accurate results as accuracy is always a matter of concern in prediction made in rainfall.

# CHAPTER 4

# RESULTS AND DISCUSSION

## *Linear Regression Results*

After the training phase is done, we take a few values of temperature, humidity, dew point, wind speed and input these to our model. And now our model predicts the following output. Linear regression analysis of our dataset shows the below trends in each parameter.

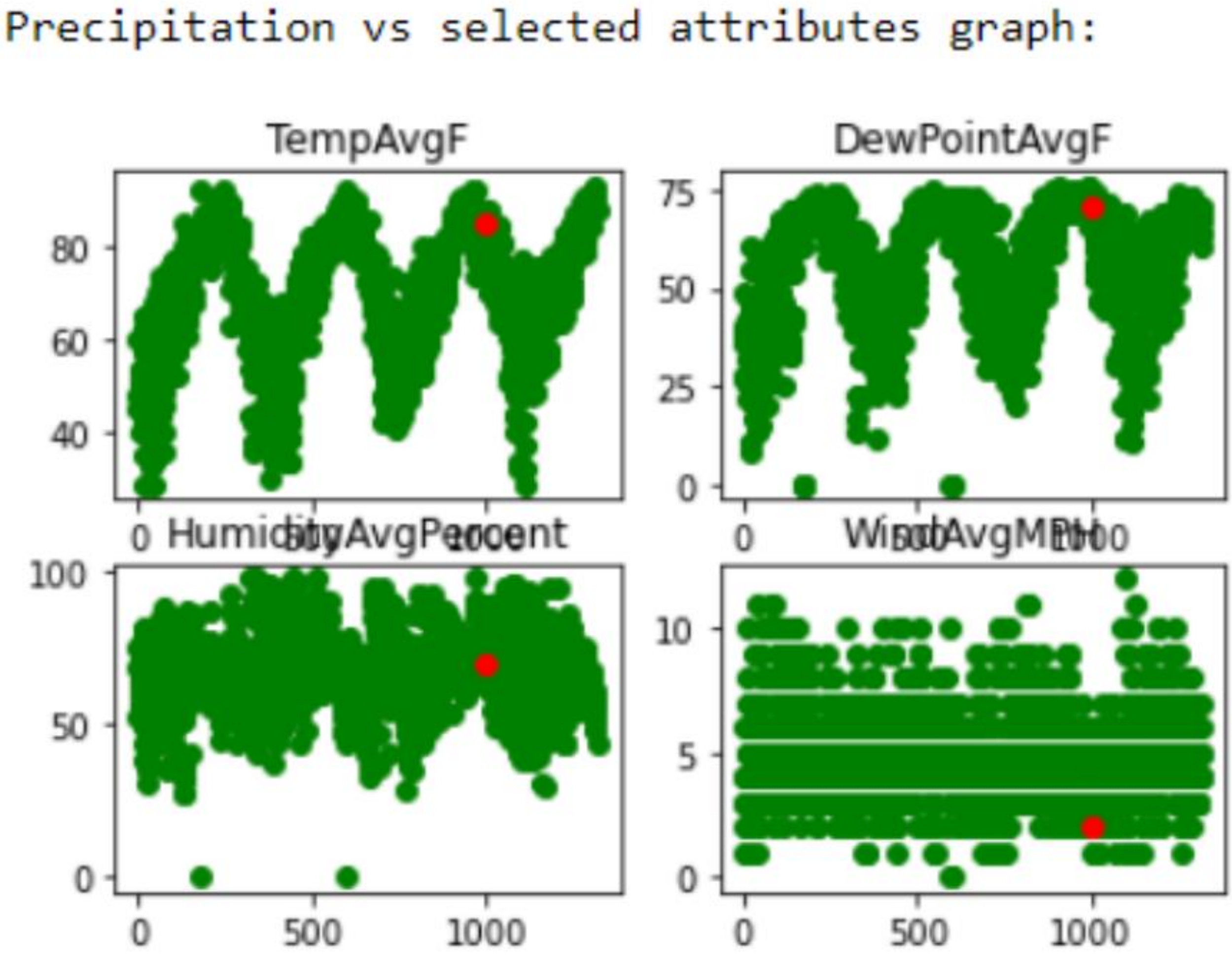



Fig. 4- Linear Regression Analysis

## *Logistic Regression Results*

After the training phase is done, we take a few values of temperature, humidity, dew point, wind speed and input these to our model. And now our model predicts the following categorical output. Logistic regression analysis of our dataset shows the below trends in each parameter.

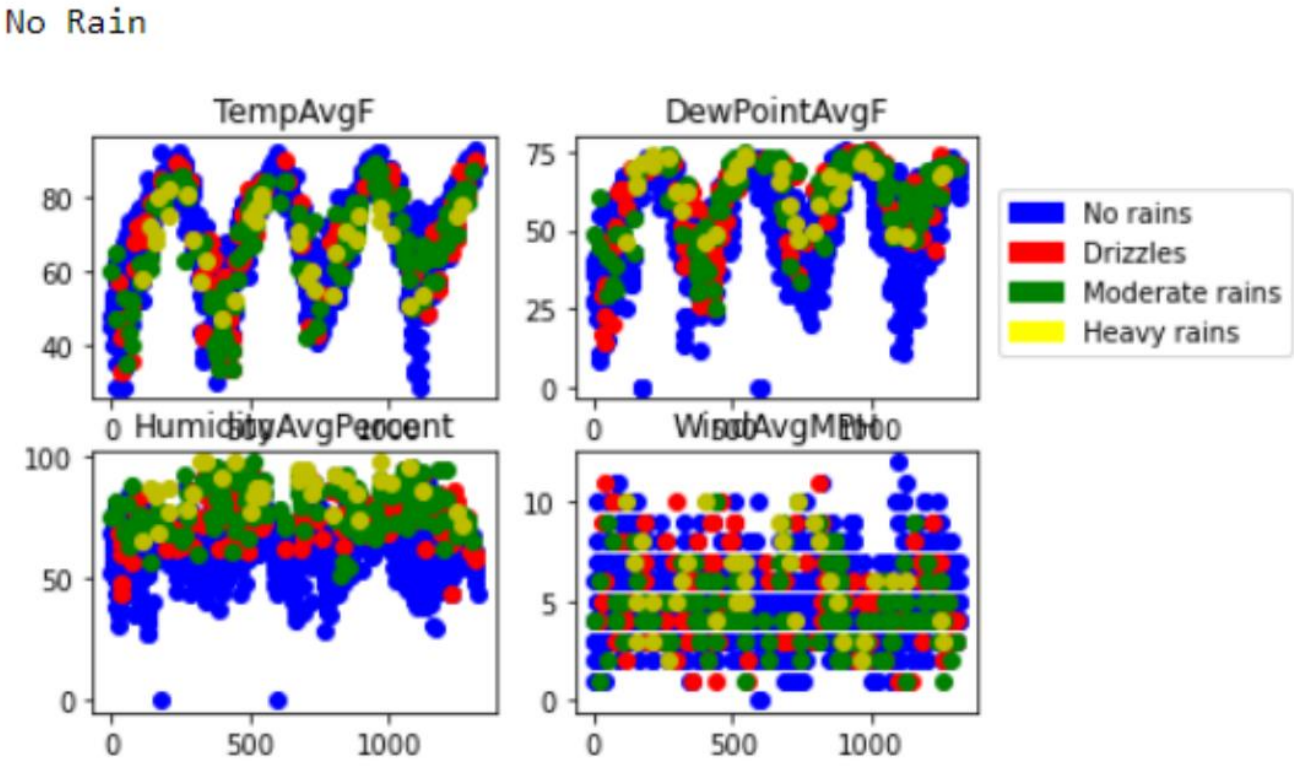



Fig. 5- Logistic Regression Analysis

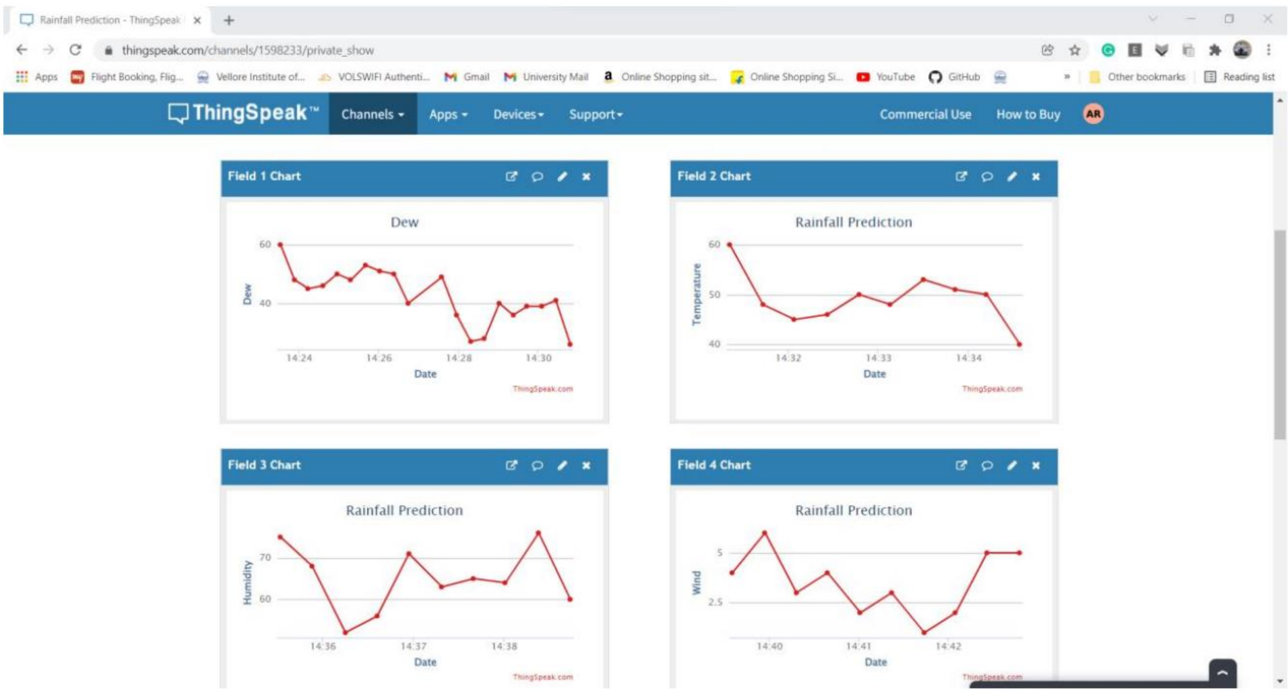



Fig. 6- IoT Integration Analysis

# CHAPTER 5

# CONCLUSION AND FUTURE WORK

Rainfall is one the most significant natural phenomenon that is not only important for the human beings only but the living beings. Due to the changing climatic conditions, rainfall cycles are also changing and the temperature of the earth is rising. The changing temperature is also affecting the agriculture, industry and sometimes may cause flooding and land slide. Therefore, it is essential for the human beings to keep a check upon this natural phenomenon in order to survive. The water is a scarce natural resource without which human life is impossible and also there is no substitute to this natural resource. Thus, predicting the rainfall for agriculture and water reserves, also it also good for keeping human beings alert of natural disasters like flood and landslide. However, to overcome these issues and meet the demands, a system to forecast rainfall is essential using artificial intelligence of neural that is popular within the modern technology.

The choice of algorithm basically depends on the nature of prediction. For the above kind of input data and required output, linear regression gives more accurate results than logistic regression. A bad rainfall prediction can affect the agriculture mostly farmers as their whole crop is dependent on the rainfall and agriculture is always an important part of every economy. So, making an accurate prediction of the rainfall somewhat good. There are a number of techniques used in machine learning but accuracy is always a matter of concern in prediction made in rainfall. In the future, this system can be used to help in the agriculture and food industry as it will help farmers predict the outcome of their plot. It will also improve on weather prediction systems and there might be personalized apps to regulate people in day-to-day life. These apps can be integrated in the working of organizations centered around agriculture and other businesses related to it. Since agriculture is often related to the economy of a country, accurate weather prediction will provide great help to the lives of those directly involved in the processing as well as most other citizens.

# CHAPTER 6

# REFERENCES

[1] http://docs.neu.edu.tr/library/6689487225.pdf

[2] https://thingspeak.com/

[3] S. Manandhar, S. Dev, Y. H. Lee, Y. S. Meng and S. Winkler, "A Data-Driven Approach for Accurate Rainfall Prediction," in IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 11, https://ieeexplore.ieee.org/document/8789447

[4] S. Srivastava, N. Anand, S. Sharma, S. Dhar, L. Kumar Sinha, "Monthly Rainfall Prediction Using Various Machine Learning Algorithms for Early Warning of Landslide Occurrence," 2020 International Conference for Emerging Technology (INCET), https://ieeexplore.ieee.org/document/9154184

[5] Z. Basha, N. Bhavana, P. Bhavya, V. Sowmya, "Rainfall Prediction using Machine Learning and Deep Learning Techniques," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), https://www.researchgate.net/publication/343438919_Rainfall_Prediction_using_Machine_Learning_Deep_Learning_Techniques

[6] G. Verma, P. Mittal and S. Farheen, "Real Time Weather Prediction System Using IOT and Machine Learning," 2020 6th International Conference on Signal Processing and Communication (ICSC), https://ieeexplore.ieee.org/document/9182766

[7] K. Dutta, P. Gouthaman, "Rainfall Prediction using Machine Learning and Neural Network," *International Journal of Recent Technology and Engineering (IJRTE), https://www.ijrte.org/wp-content/uploads/papers/v9i1/A2747059120.pdf*

[8] G. Sethupathi M., Y. Sai Ganesh, M. Mansoor Ali, "Efficient Rainfall Prediction and Analysis using Machine Learning Techniques and Outcomes of these Techniques," *Turkish Journal of Computer and Mathematics Education, https://turcomat.org/index.php/turkbilmat/article/download/7135/5809*

[9] G. Naik, "Prediction of Rainfall using Machine Learning Techniques," *International Journal for Research in Engineering Application & Management (IJREAM),* http://ijream.org/papers/IJREAMV07I02SJ008.pdf

[10] K. Abhishek, A. Kumar, R. Ranjan, S. Kumar, "A Rainfall Prediction Model using Artificial Neural Network," 2021 IEEE Control and System Graduate Research Colloquium (ICSGRC 2021), https://ieeexplore.ieee.org/abstract/document/6287140

[11] Liyew, C.M., Melese, H.A., "Machine learning techniques to predict daily rainfall amount," *J Big*

*Data* 8, https://journalofbigdata.springeropen.com/articles/10.1186/s40537-021-00545-4#citeas

[12] G. Sethupathi M., Y. Sai Ganesh, M. Mansoor Ali, “Efficient Rainfall Prediction and Analysis using Machine Learning Techniques and Outcomes of these Techniques,” *Turkish Journal of Computer and Mathematics Education, https://turcomat.org/index.php/turkbilmat/article/download/7135/5809*

[13] https://jupyter.org/, https://en.wikipedia.org/wiki/Project_Jupyter

[14] https://en.wikipedia.org/wiki/ThingSpeak

[15] http://wiki.sunfounder.cc/index.php?title=Humiture_Sensor_Module

[16] https://en.wikipedia.org/wiki/Anemometer

[17] https://en.wikipedia.org/wiki/NodeMCU

[18]https://www.investopedia.com/terms/m/mlr.asp#:~:text=Key%20Takeaways-,Multiple%20linear%20regression%20(MLR)%2C%20also%20known%20simply%20as%20multiple,uses%20just%20one%20explanatory%20variable.

[19]https://www.sciencedirect.com/topics/computer-science/logistic-regression#:~:text=Logistic%20regression%20is%20a%20process,%2Fno%2C%20and%20so%20on.

[20] S. Prabhakaran, P. Naveen Kumar, P. Sai Mani Tarun, “Rainfall Prediction using Modified Linear Regression,” https://www.researchgate.net/publication/318268105_Rainfall_prediction_using_modified_linear_regression

[21]“Understanding how to simulate and analyze data from ThingSpeak” https://www.mathworks.com/products/thingspeak.html

## ANNEXURE

*Code for predicting rainfall:*

```
#importing libraries
import pandas as pd
import numpy as np
# read the data in a pandas dataframe
data = pd.read_csv(r'C:\Users\HP\Downloads\vellore_weather.csv')
data.head()
data = data.drop(columns=['Events', 'Date','SeaLevelPressureHighInches'])
data.head()
data = data.replace('T', 0.0)
data = data.replace('-', 0.0)
data.describe()
data.to_csv('vellore_final.csv')
import sklearn as sk
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as plt
data = pd.read_csv("vellore_final.csv")
X = data.drop(['PrecipitationSumInches'], axis = 1)
# the output or the label.
Y = data['PrecipitationSumInches']
# reshaping it into a 2-D vector
Y = Y.values.reshape(-1, 1)
day_index = 1000
days = [i for i in range(Y.size)]
# initialize a linear regression classifier
clf = LinearRegression()
# train the classifier with our
# input data.
clf.fit(X, Y)
inp = np.array([[74], [60], [45], [67], [49], [43], [33], [45],
[57], [29.68], [10], [7], [2], [0], [20], [4], [31]])
inp = inp.reshape(1, -1)
print('The precipitation in inches for the input is:', clf.predict(inp))
```

```
print("the precipitation trend graph: ")
plt.scatter(days, Y, color = 'g')
plt.scatter(days[day_index], Y[day_index], color ='r')
plt.title("Precipitation level")
plt.xlabel("Days")
plt.ylabel("Precipitation in inches")
plt.show()
x_vis = X.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
print("Precipitation vs selected attributes graph: ")
for i in range(x_vis.columns.size):
    plt.subplot(2, 2, i + 1)
    plt.scatter(days, x_vis[x_vis.columns.values[i][:100]], color = 'g')
    plt.scatter(days[day_index],x_vis[x_vis.columns.values[i]][day_index],color ='r')
    plt.title(x_vis.columns.values[i])
plt.show()
import pandas as pd
import numpy as np

import sklearn as sk
from sklearn import metrics, datasets
from sklearn.linear_model import LogisticRegression
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
data = pd.read_csv("vellore_final.csv")
X = data.drop(['PrecipitationSumInches'], axis=1)
Y_temp = data['PrecipitationSumInches']
Y_temp = Y_temp.values.reshape(-1, 1)
Y = pd.DataFrame(columns=['No Rains', 'Drizzle', 'Moderate Rains', 'Heavy Rains'])
for i in range(Y_temp.size):
    if(Y_temp[i]<0.001):
        Y.loc[i] = [1, 0, 0, 0]
    elif(Y_temp[i]>=0.001 and Y_temp[i]<0.1):
        Y.loc[i] = [0, 0, 1, 0]
    else:
```

```
        Y.loc[i] = [0, 0, 0, 1]
Y = []
x1 = pd.DataFrame(columns=X.columns.values)
x2 = pd.DataFrame(columns=X.columns.values)
x3 = pd.DataFrame(columns=X.columns.values)
x4 = pd.DataFrame(columns=X.columns.values)
for i in range(Y_temp.size):
    if(Y_temp[i]<0.001):
        Y.append(1)
        x1.loc[i] = X.loc[i]
    elif(Y_temp[i]>=0.001 and Y_temp[i]<0.1):
        Y.append(2)
        x2.loc[i] = X.loc[i]
    elif(Y_temp[i]>=0.1 and Y_temp[i]<1.2):
        Y.append(3)
        x3.loc[i] = X.loc[i]
    else:
        Y.append(4)
        x4.loc[i] = X.loc[i]
Y = np.array(Y).reshape(len(Y), )
logr = LogisticRegression(multi_class='ovr', solver='liblinear').fit(X, Y)
input = np.array([[74], [60], [45], [67], [49], [43], [93], [75], [57], [29.68],[29.59],
[10],[7],[2],[20],[4],[31],[0.46]])
input = input.reshape(1, -1)
classes = ['None', 'No Rain', 'Drizzles', 'Moderate Rains', 'Heavy Rains']
print(classes[int(logr.predict(input))])
x1 = x1.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
x2 = x2.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
x3 = x3.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
x4 = x4.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
for i in range(4):
            plt.subplot(2,2,i+1)
            plt.scatter(x1.index.values, x1[x1.columns.values[i]], color='b')
            plt.scatter(x2.index.values, x2[x2.columns.values[i]], color='r')
```

```
            plt.scatter(x3.index.values, x3[x3.columns.values[i]], color='g')
            plt.scatter(x4.index.values, x4[x4.columns.values[i]], color='y')
            plt.title(x1.columns.values[i])
blue_patch = mpatches.Patch(color='blue', label='No rains')
red_patch = mpatches.Patch(color='red', label='Drizzles')
green_patch = mpatches.Patch(color='green', label='Moderate rains')
yellow_patch = mpatches.Patch(color='yellow', label='Heavy rains')
plt.legend(handles=[blue_patch, red_patch, green_patch, yellow_patch],borderaxespad=0.)
plt.show()
Y = np.array(Y).reshape(len(Y), )
logr = LogisticRegression(multi_class='ovr', solver='liblinear').fit(X, Y)
input = np.array([[58], [43], [28], [37], [22], [18], [75], [49], [22], [30.35], [30.14],
[10],[10],[10],[14],[4],[21],[0]])
input = input.reshape(1, -1)
classes = ['None', 'No Rain', 'Drizzles', 'Moderate Rains', 'Heavy Rains']
print(classes[int(logr.predict(input))])
x1 = x1.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
x2 = x2.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
x3 = x3.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
x4 = x4.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
for i in range(4):
    plt.subplot(2,2,i+1)
    plt.scatter(x1.index.values, x1[x1.columns.values[i]], color='b')
    plt.scatter(x2.index.values, x2[x2.columns.values[i]], color='r')
    plt.scatter(x3.index.values, x3[x3.columns.values[i]], color='g')
    plt.scatter(x4.index.values, x4[x4.columns.values[i]], color='y')
    plt.title(x1.columns.values[i])
blue_patch = mpatches.Patch(color='blue', label='No rains')
red_patch = mpatches.Patch(color='red', label='Drizzles')
green_patch = mpatches.Patch(color='green', label='Moderate rains')
yellow_patch = mpatches.Patch(color='yellow', label='Heavy rains')
plt.legend(handles=[blue_patch, red_patch, green_patch, yellow_patch],borderaxespad=0.)
plt.show()
Y = np.array(Y).reshape(len(Y), )
```

```
logr = LogisticRegression(multi_class='ovr', solver='liblinear').fit(X, Y)
input =
np.array([[56],[42],[27],[42],[29],[20],[78],[64],[50],[30.47],[30.06],[10],[8],[2],[20],[9],[31],[0.06]])
input = input.reshape(1, -1)
classes = ['None', 'No Rain', 'Drizzles', 'Moderate Rains', 'Heavy Rains']
print(classes[int(logr.predict(input))])
x1 = x1.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
x2 = x2.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
x3 = x3.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
x4 = x4.filter(['TempAvgF', 'DewPointAvgF', 'HumidityAvgPercent', 'WindAvgMPH'], axis=1)
for i in range(4):
    plt.subplot(2,2,i+1)
    plt.scatter(x1.index.values, x1[x1.columns.values[i]], color='b')
    plt.scatter(x2.index.values, x2[x2.columns.values[i]], color='r')
    plt.scatter(x3.index.values, x3[x3.columns.values[i]], color='g')
    plt.scatter(x4.index.values, x4[x4.columns.values[i]], color='y')
    plt.title(x1.columns.values[i])
blue_patch = mpatches.Patch(color='blue', label='No rains')
red_patch = mpatches.Patch(color='red', label='Drizzles')
green_patch = mpatches.Patch(color='green', label='Moderate rains')
yellow_patch = mpatches.Patch(color='yellow', label='Heavy rains')
plt.legend(handles=[blue_patch, red_patch, green_patch, yellow_patch],borderaxespad=0.)
plt.show()
```

*Code for NodeMCU:*

```
#include <ESP8266WiFi.h>                                          // NodeMCU Library
#include <DHT.h>                                                 // DHT Sensor Library
#include <ThingSpeak.h>                                          // ThingSpeak Library

#define dhtpin 4

DHT dht(dhtpin, DHT11);                                         // Declaring D3 as dht pin
WiFiClient client;                                     // WiFi client object for ThingSpeak
long myChannelNumber = 1715895 ;                       // ThingSpeak channel ID
const char myWriteAPIKey[] = "IZMCMM7I2YRAMX19";                 // ThingSpeak write key

 void setup() {
    Serial.begin(115200);                                      // Initializing baud rate
    WiFi.begin("OnePlus Nord", "12345678"); // Connect to WiFi using given id and pass
    while (WiFi.status() != WL_CONNECTED)          // WiFi getting connected
    {
      delay(500);
      Serial.print("..");
    }
    Serial.println();
    Serial.println("NodeMCU is connected!");
    Serial.println(WiFi.localIP());                          // Print IP connected
    dht.begin();                                             // Connect to dht sensor
    ThingSpeak.begin(client);                                // Connect to ThingSpeak
 }

void loop() {
  float h = dht.readHumidity();                    // Reading and saving Humidity value
  float t = dht.readTemperature();                  // Reading and saving Temperature value
  Serial.println("Temperature: "+(String)t); // Printing these values on serial monito
  Serial.println("Humidity: " + (String) h);
  Serial.println();

 //                        SENDING                        DATA                       TO
THINGSPEAK
// Send humidity value to ThingSpeak in channel 2
    ThingSpeak.writeField(myChannelNumber, 2, h, myWriteAPIKey);
    //delay(100);
);         // Send temperature value to ThingSpeak in channel 1
    ThingSpeak.writeField(myChannelNumber, 1, t, myWriteAPIKey
 delay(1000);
 }
```

*Tinkercad Codes:*

*Code for circuit-1:*

```
String ssid     = "Simulator Wifi";  // SSID to connect to
String password = ""; // Our virtual wifi has no password
String host     = "api.thingspeak.com"; // Open Weather Map API
const int httpPort   = 80;

String url = " ";
String url1     = "/update?api_key=IZMCMM7I2YRAMX19&field1=";

int l = 0;


int setupESP8266(void) {
  // Start our ESP8266 Serial Communication
  Serial.begin(115200);   // Serial connection over USB to computer
  Serial.println("AT");   // Serial connection on Tx / Rx port to ESP8266
  delay(10);        // Wait a little for the ESP to respond
  if (!Serial.find("OK")) return 1;

  // Connect to 123D Circuits Simulator Wifi
  Serial.println("AT+CWJAP=\"" + ssid + "\",\"" + password + "\"");
  delay(10);        // Wait a little for the ESP to respond
  if (!Serial.find("OK")) return 2;

  // Open TCP connection to the host:
  Serial.println("AT+CIPSTART=\"TCP\",\"" + host + "\"," + httpPort);
  delay(50);        // Wait a little for the ESP to respond
  if (!Serial.find("OK")) return 3;

  return 0;
}

void anydata(int l, String url) {



  // Construct our HTTP call
   String httpPacket = "GET " + url + String(l) + " HTTP/1.1\r\nHost: " + host +
"\r\n\r\n";
  int length = httpPacket.length();

  // Send our message length
  Serial.print("AT+CIPSEND=");
  Serial.println(length);
  delay(10); // Wait a little for the ESP to respond if (!Serial.find(">")) return -1;

  // Send our http request
```

```
  Serial.print(httpPacket);
  delay(10); // Wait a little for the ESP to respond
  if (!Serial.find("SEND OK\r\n")) return;


}


void setup() {

  setupESP8266();

}

void loop() {
 int light = map(analogRead(A1),1023,0,0,100);
  url = url1;
 anydata(light, url);
  Serial.println(light);
  Serial.println(url1);

  delay(10000);


}
```

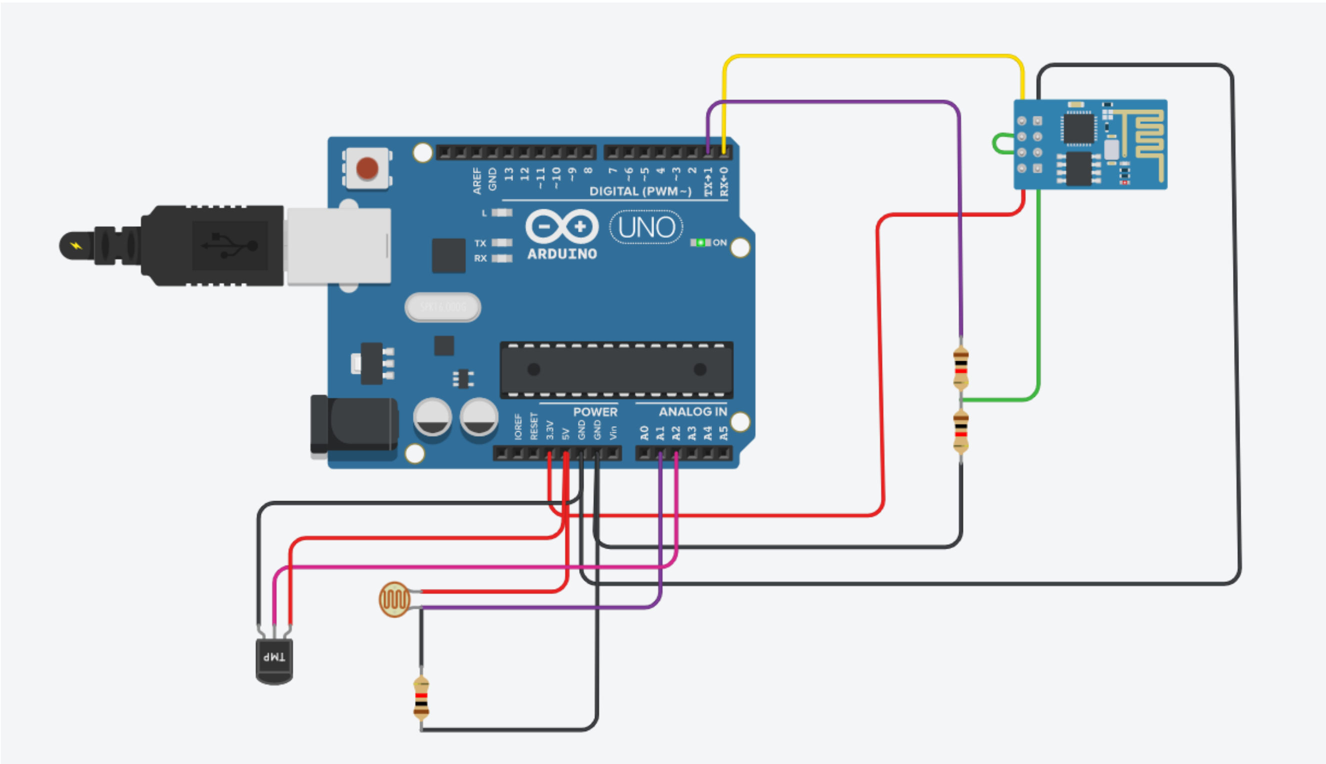



Fig. 7 - Circuit-1 Snippet

*Code for circuit-2:*

```
String ssid     = "Simulator Wifi";  // SSID to connect to
String password = ""; // Our virtual wifi has no password
String host     = "api.thingspeak.com"; // Open Weather Map API
const int httpPort   = 80;

String url = " ";
String url1     = "/update?api_key=IZMCMM7I2YRAMX19&field1=";
String url2     = "/update?api_key=IZMCMM7I2YRAMX19&field2=";
int l = 0;


int setupESP8266(void) {
  // Start our ESP8266 Serial Communication
  Serial.begin(115200);   // Serial connection over USB to computer
  Serial.println("AT");   // Serial connection on Tx / Rx port to ESP8266
  delay(10);        // Wait a little for the ESP to respond
  if (!Serial.find("OK")) return 1;

  // Connect to 123D Circuits Simulator Wifi
  Serial.println("AT+CWJAP=\"" + ssid + "\",\"" + password + "\"");
  delay(10);        // Wait a little for the ESP to respond
  if (!Serial.find("OK")) return 2;

  // Open TCP connection to the host:
  Serial.println("AT+CIPSTART=\"TCP\",\"" + host + "\"," + httpPort);
  delay(50);        // Wait a little for the ESP to respond
  if (!Serial.find("OK")) return 3;

  return 0;
}

void anydata(int l, String url) {



  // Construct our HTTP call
   String httpPacket = "GET " + url + String(l) + " HTTP/1.1\r\nHost: " + host +
"\r\n\r\n";
  int length = httpPacket.length();

  // Send our message length
  Serial.print("AT+CIPSEND=");
  Serial.println(length);
  delay(10); // Wait a little for the ESP to respond if (!Serial.find(">")) return -1;

  // Send our http request
```

```
  Serial.print(httpPacket);
  delay(10); // Wait a little for the ESP to respond
  if (!Serial.find("SEND OK\r\n")) return;


}


void setup() {

  setupESP8266();

}

void loop() {
 int light = map(analogRead(A1),1023,0,0,100);
  url = url1;
 anydata(light, url);
  Serial.println(light);
  Serial.println(url1);

  delay(10000);

 int temp = map(analogRead(A2),1023,0,100,0);
 url = url2;
  anydata(temp, url);
  Serial.println(temp);
  Serial.println(url2);

  delay(10000);
}
```

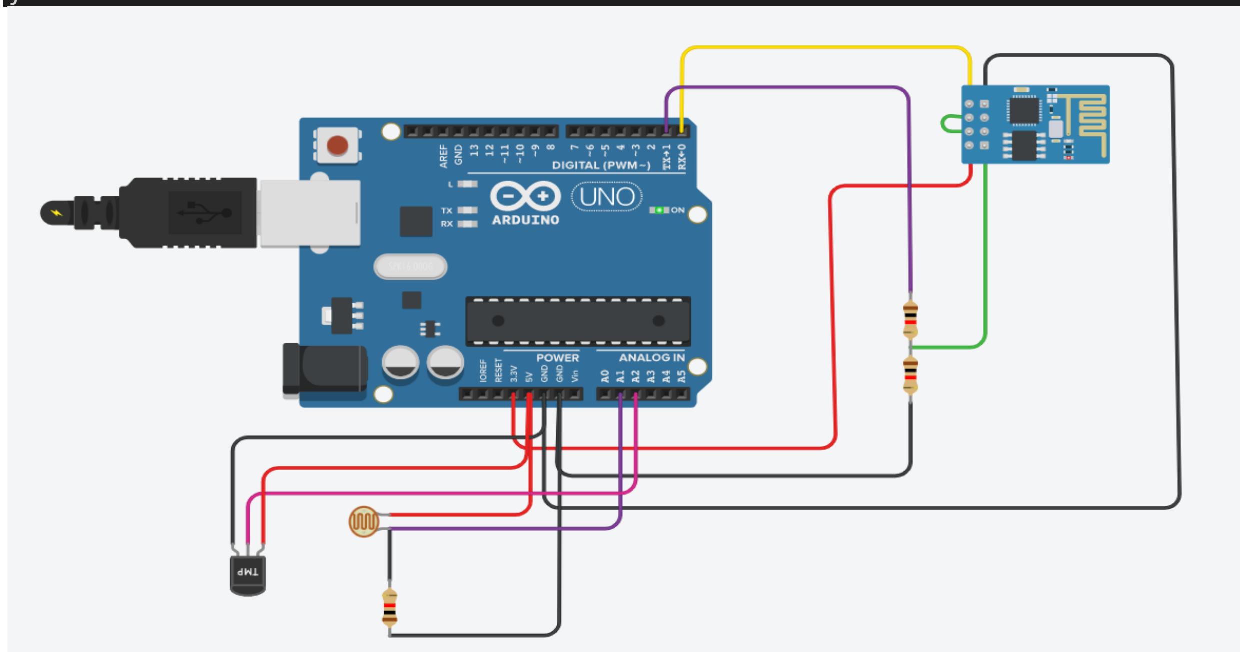



Fig. 8 - Circuit-2 Snippet

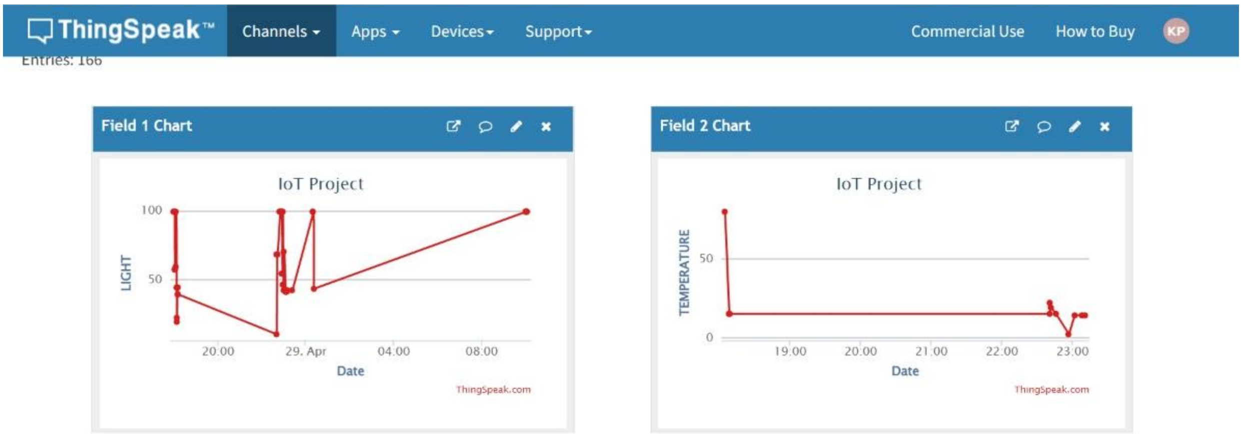



Fig. 9 – Graph Variations based on Tinkercad circuits

# IOT INTEGRATED RAINFALL PREDICTION USING ML TECHNIQUES

PRESENTED BY:
ADITHI KUSUPUDI - 19BCE0588
KHYATI PAREEK - 20BDS0299
SRISHTI SINHA - 20BDS0329

# INTRODUCTION

- Extreme variations in rainfall has a drastic effect on agriculture. Drought can kill crops while heavy rainfall can increase soil erosion and spoil the plantations. Optimum amount of water is required for survival of crops. Too much or too little water is harmful to the crops and can affect the yield in the long run causing major loss to farmers. Hence rainfall is a major factor affecting the crop yield. Therefore, there is a need to predict rainfall for effective use of water resources for crop productivity to give a better yield and decrease agricultural loss.

- Existing rainfall prediction methods are large scale/locality based and don't sense atmospheric parameters for a specific place which can sometimes be a problem. Our project is essentially a wearable cap which can be worn by the farmer while working on the field and this device automatically logs atmospheric parameters to the ThingSpeak cloud and predicts the rainfall for that day using its trained model. This makes our device portable and specific to the farmer's location and hence eases the decision of the farmer of how much he has to water the plants and avoid agricultural loss.

# Monthly Rainfall Prediction Using Various Machine Learning Algorithms for Early Warning of Landslide Occurrence [2020]

## METHODOLOGY

- The proposed study involves the development of rainfall forecasting models using four different machine learning algorithms for predicting landslide occurrence well in advance. The models are calibrated and validated on independent data.
- Normalization is performed to enhance the predictive accuracy of the models. The Gradient Descent optimization algorithm is used to train the models. Mean square error (MSE) is used as the loss function for training linear regression, BPNN and LSTM models whereas Vapnik's ε-insensitive loss function is used for training the SVR models.
- Evaluation of the models is performed using the mean absolute error (MAE) and root mean square error (RMSE) metrics.

## ADVANTAGES

- Capable of predicting low as well as medium intensity rainfalls effectively.
- Neural networks (BPNN and LSTM) are capable enough to forecast rainfall with high accuracy without any prior knowledge on the related meteorological parameters influencing rainfall

## DISADVANTAGES

- The developed models are however underperformed in mapping high-intensity rainfalls accurately.

# Prediction of Rainfall Using Machine Learning Techniques [2021]

## METHODOLOGY

- The proposed method is based on the multiple linear regressions.
- The data for the prediction is collected from the publically available sources and the 70 percentages of the data are for training and the 30 percentages of the data is for testing.
- Multiple regressions are used to predict the values with the help of descriptive variables and is a statistical method. It is having a linear relationship between the descriptive variable and the output values.

## ADVANTAGES

- The error-free prediction provides better planning in agriculture and other industries.
- The power to work out the relative influence of one or more predictor variables to the criterion value.
- Ability to spot outliers or anomalies.

## DISADVANTAGES

- Data must be independent.

# Real Time Weather Prediction System Using IOT and Machine Learning [2020]

## METHODOLOGY

- Real time weather prediction system that can be used in number of applications like homes, industries, agriculture, stadiums etc. for predicting the weather information.
- The system utilizes a temperature and humidity sensor i.e. DHT11 and a light intensity sensor i.e. LDR which upload the sensed data to a ThingSpeak cloud server.
- Further, the model is run through machine learning techniques.

## ADVANTAGES

- Low cost IoT board and sensors are used
- Predict the weather parameters in real time environment.
- The result of the proposed system is slightly better in terms of accuracy.

## DISADVANTAGES

- Mostly applicable for indoor applications only.
- Work needs to be done to further bring it into use

# Rainfall Prediction using Machine Learning & Deep Learning Techniques [2020]

| METHODOLOGY | ADVANTAGES | DISADVANTAGES |
| --- | --- | --- |
| • Rainfall has been predicted using deep learning techniques.<br>• Two deep learning techniques - Multilayer Perceptron and Auto-Encoders.<br>• MLP is used in prediction and classification tasks.<br>• Performance of methodology is also evaluated by using RMSE(Root MSE) | • Accuracy is good ; Past data is used to predict future data<br>• Mean square error is less but when performed validation | • Instead of sensors as input device, they have used CNN to take the input from the past data. |

# A Data-Driven Approach for Accurate Rainfall Prediction [2019]

| METHODOLOGY | ADVANTAGES | DISADVANTAGES |
| --- | --- | --- |
| • This paper proposes a systematic approach to analyze various parameters that affect precipitation in the atmosphere.<br><br>• Based on these findings, an optimum set of features are used in a data-driven machine learning algorithm for rainfall prediction. | • Makes use of atmospheric parameters for prediction instead of using precipitable water vapor (PWV) derived from global positioning system (GPS) signal - delays to predict rainfall<br><br>• Using a 4-year (2012-2015) database shows a true detection rate of 80.4% and a false alarm rate of 20.3%.<br><br>• Compared to the existing literature, our method significantly reduces the false alarm rates. | • The method derived an overall accuracy of 79.6% which is less compared to other models. |

# Problem Statement

- Variations in rainfall have a drastic effect on agriculture.
- Too much or too little rain(water) both can harm the crops and the yields.
- Hence, predicting the rainfall is essential for the effective use of water resources for crops
- Important to decrease agricultural loss
- Existing prediction methods :
  - large/ locality-based
  - sometimes don't sense atmospheric parameters for specific regions
- Our project :
- Focuses on caps which can be worn by the farmers themselves
- Automatically logs atmospheric parameters to the system

# THANK YOU!

# IOT INTEGRATED RAINFALL PREDICTION USING ML TECHNIQUES

PRESENTED BY:
ADITHI KUSUPUDI - 19BCE0588
KHYATI PAREEK - 20BDS0299
SRISHTI SINHA - 20BDS0329

# PROPOSED IDEA

- Extreme variations in rainfall has a drastic effect on agriculture.
- Hence rainfall is a major factor affecting the crop yield. Therefore, there is a need to predict rainfall for effective use of water resources for crop productivity to give a better yield and decrease agricultural loss.
- Existing rainfall prediction methods are large scale/locality-based and don't sense atmospheric parameters for a specific place which can sometimes be a problem.
- Our project is essentially a setup for a wearable cap which can be worn by the farmer while working on the field and this device automatically logs atmospheric parameters to the ThingSpeak cloud and predicts the rainfall for that day using its trained model.
- This will make the device portable and specific to the farmer's location and hence eases the decision of the farmer of how much he has to water the plants and avoid agricultural loss.

# BENEFITS

- A bad rainfall prediction can affect the agriculture mostly farmers as their whole crop is dependent on the rainfall and agriculture is always an important part of every economy. So, making an accurate prediction of the rainfall somewhat good. There are a number of techniques used in machine learning but accuracy is always a matter of concern in prediction made in rainfall.

- It will also improve on weather prediction systems and there might be personalized apps to regulate people in day to day life. These apps can be integrated in the working of organizations centered around agriculture and other businesses related to it.

- Since agriculture is often related to the economy of a country, accurate weather prediction will provide great help to the lives of those directly involved in the processing as well as most other citizens.

# TOOLS REQUIRED (Software)

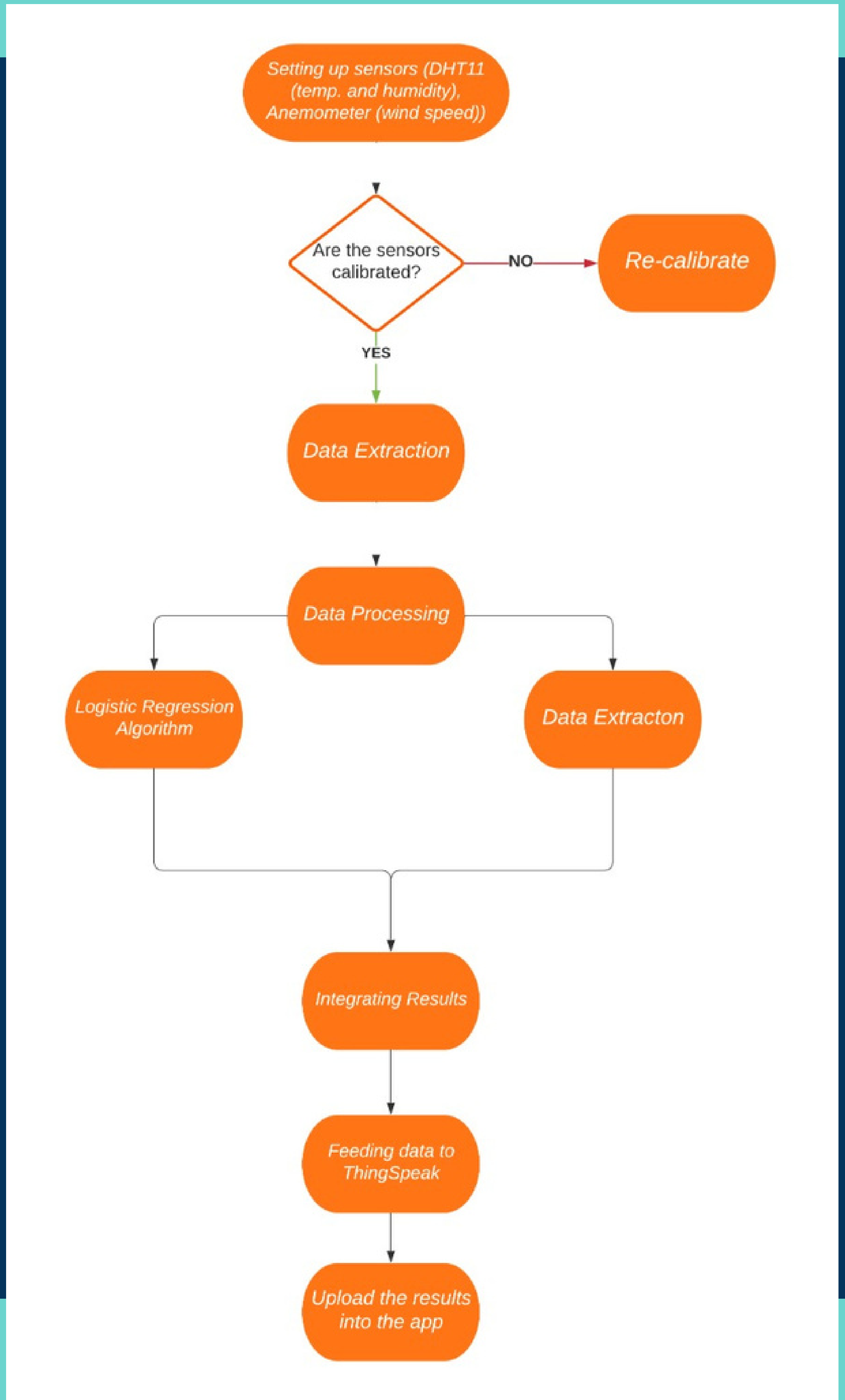
- Jupyter Notebook -  The Jupyter Notebook is a web-based interactive computing platform. It allows us to create and share the document called the Notebook, containing live codes, documentation, graphs, plots, and visualizations. We'll be using this to post the data on the ThingSpeak Cloud.

- ThingSpeak Cloud - ThingSpeak is an IoT analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak.

# TOOLS REQUIRED (Hardware)

- Node MCU
- DHT 11
- Anemometer

# Proposed Architecture



# Work Flow

- The first step in the work flow is to set up the sensors DHT11 and Anenometer.
- We then calibrate the sensors. Sensor calibration is an adjustment or set of adjustments performed on a sensor or instrument to make that instrument function as accurately, or error free, as possible.
- Data extraction : Disperate types of data is collected and retrieved from a variety of sources.
- Data Processing :We then carry out operations on data to retrieve, transform, or classify information.
- Linear regression algorithm : Used to describe data and to explain the relationship between data.
- Logistic regression algorithm : Used to model the probability of a certain class or event taking place.
- Data is logged in thingspeak via The internet and is then processed by our trained prediction model and the Predictions are shown on thingspeak which can be accessed from anywhere in the World.
- The final step is to upload the results into the app.

# Methodology/Algorithm

- ***MULTIPLE LINEAR REGRESSION(MLR)_***

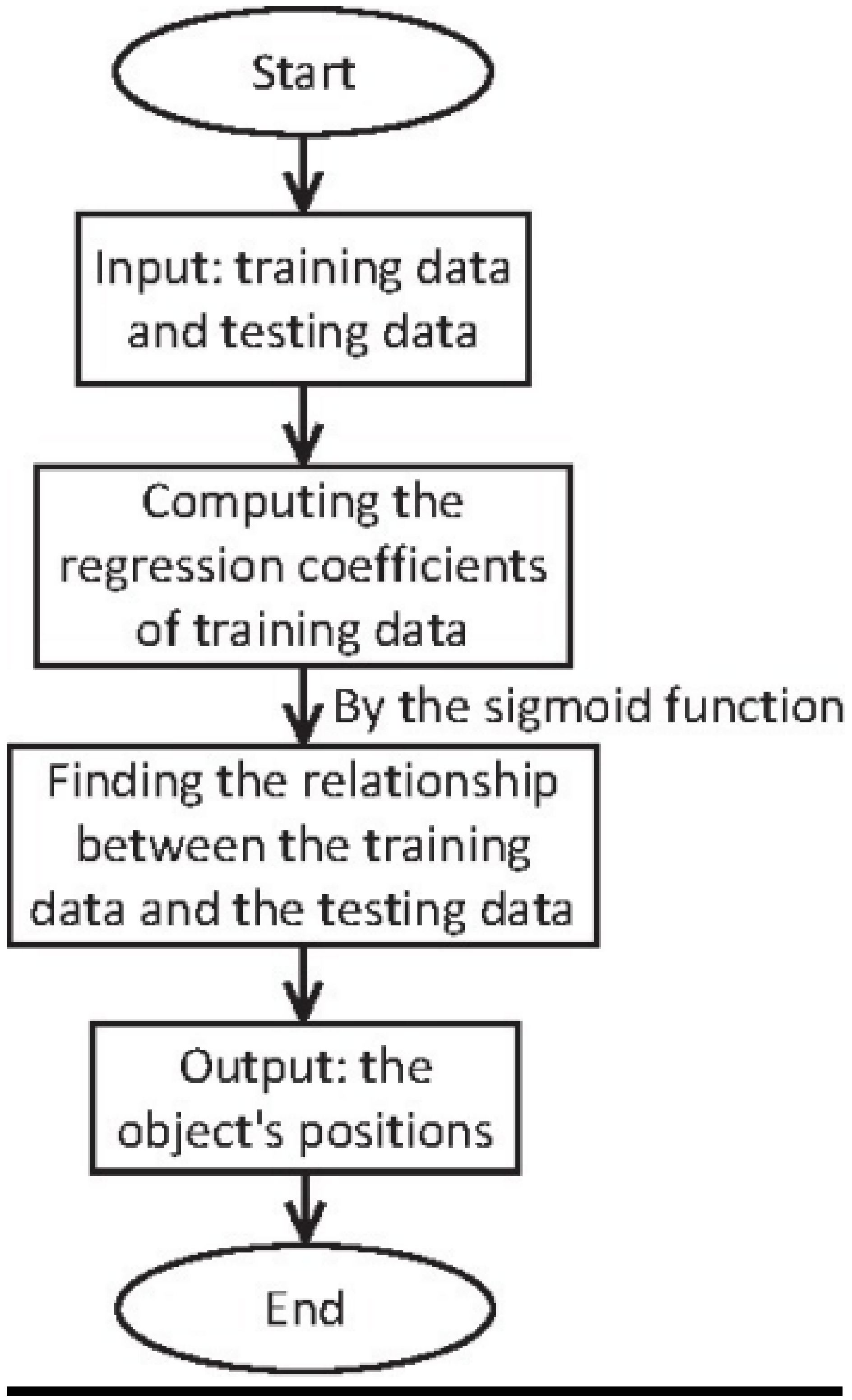
- It uses several explanatory variables to predict the outcome of a response variable. The goal is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable.
- Independent variable - Precipitation level ; Dependent variables - temperature, humidity, wind speed, dew point
- This model is trained with data of past 3-4 months. This is the training phase, after this our model is ready to take live input data of mentioned dependent variables and predict precipitation level.

# Methodology/Algorithm

- ***LOGISTIC REGRESSION***

- Multinomial logistic regression is a particular solution to classification problems that use a linear combination of the observed features and some problem-specific parameters to estimate the probability of each particular value of the dependent variable.
- Hence applying this concept here can give us a categorical prediction depicting rainfall as “No Rain” , “Drizzles” , “Moderate Rains” , “Heavy Rainfall”. This can be easily understandable by farmers as it is in very simple terms.
- In the same way as linear regression, it is first trained and then ready for use and has the same variables as linear regression but the method is different as it uses a sigmoidal function and a cross entropy function.

# Methodology/Algorithm

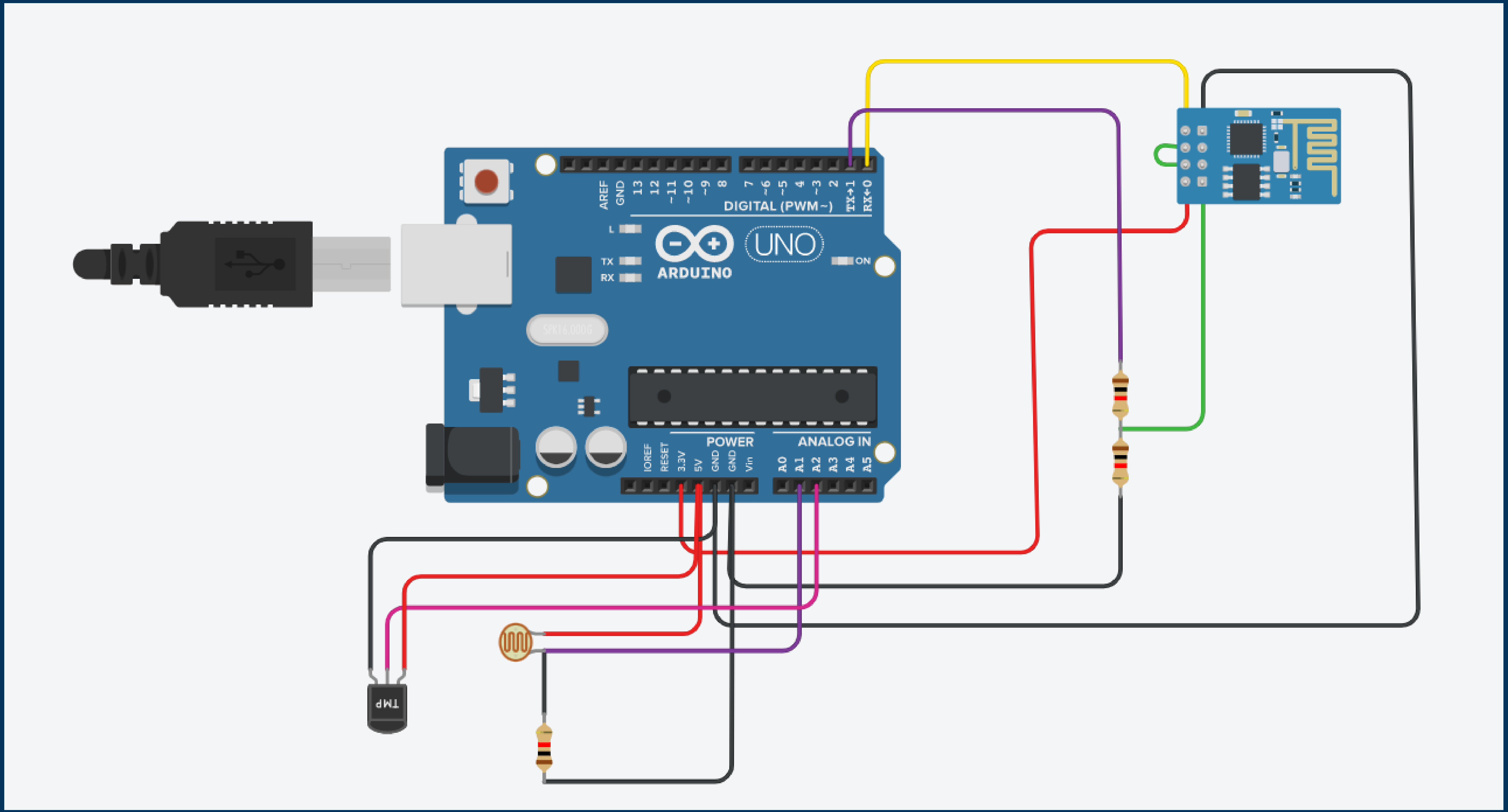


# Methodology/Algorithm

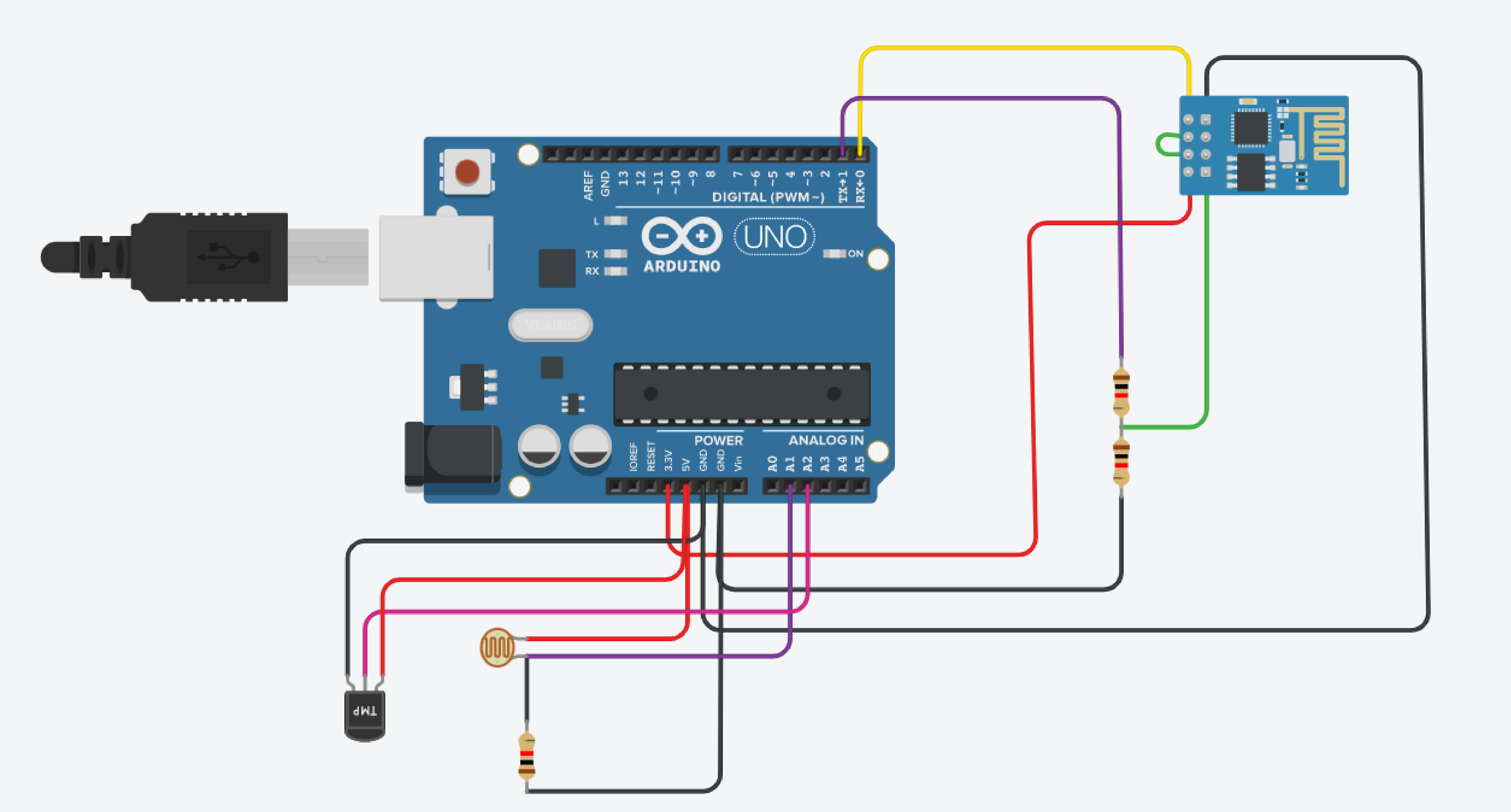
- IOT INTEGRATION

- This part focuses on sending the real time data to the thingspeak cloud and establishing a connection between our analysis and thingspeak.
- This is generally easily done using the node MCU and creating a webrequest using thingspeak API.
- Now as an alternative to this we make the web request to post the data on thingspeak via our python code itself.
- The “urllib” library helps us do this with the thing speak API of our channel. We will create a new field in thingspeak that represents our output(precipitation level in inches).

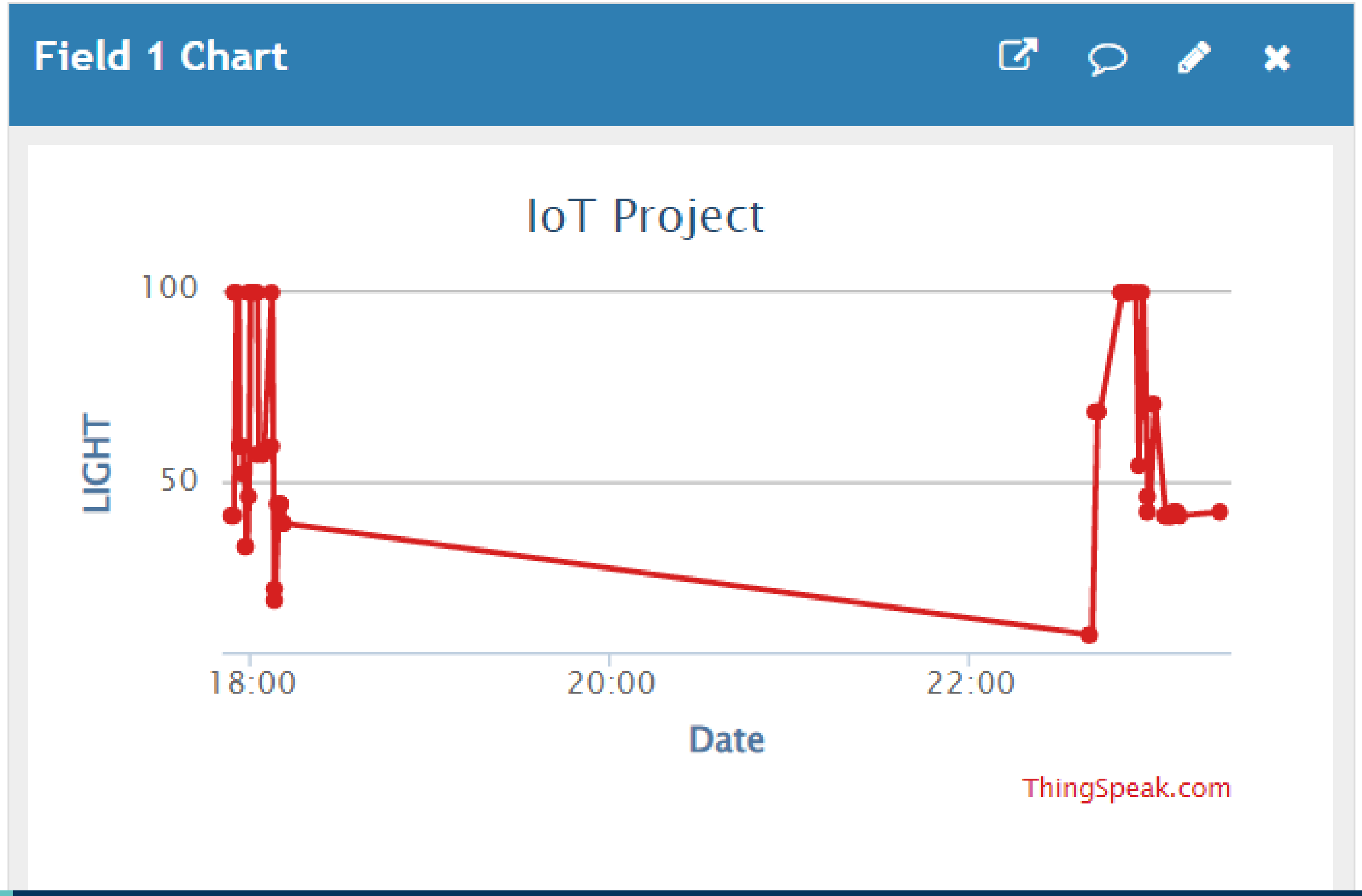
# Circuit Diagram

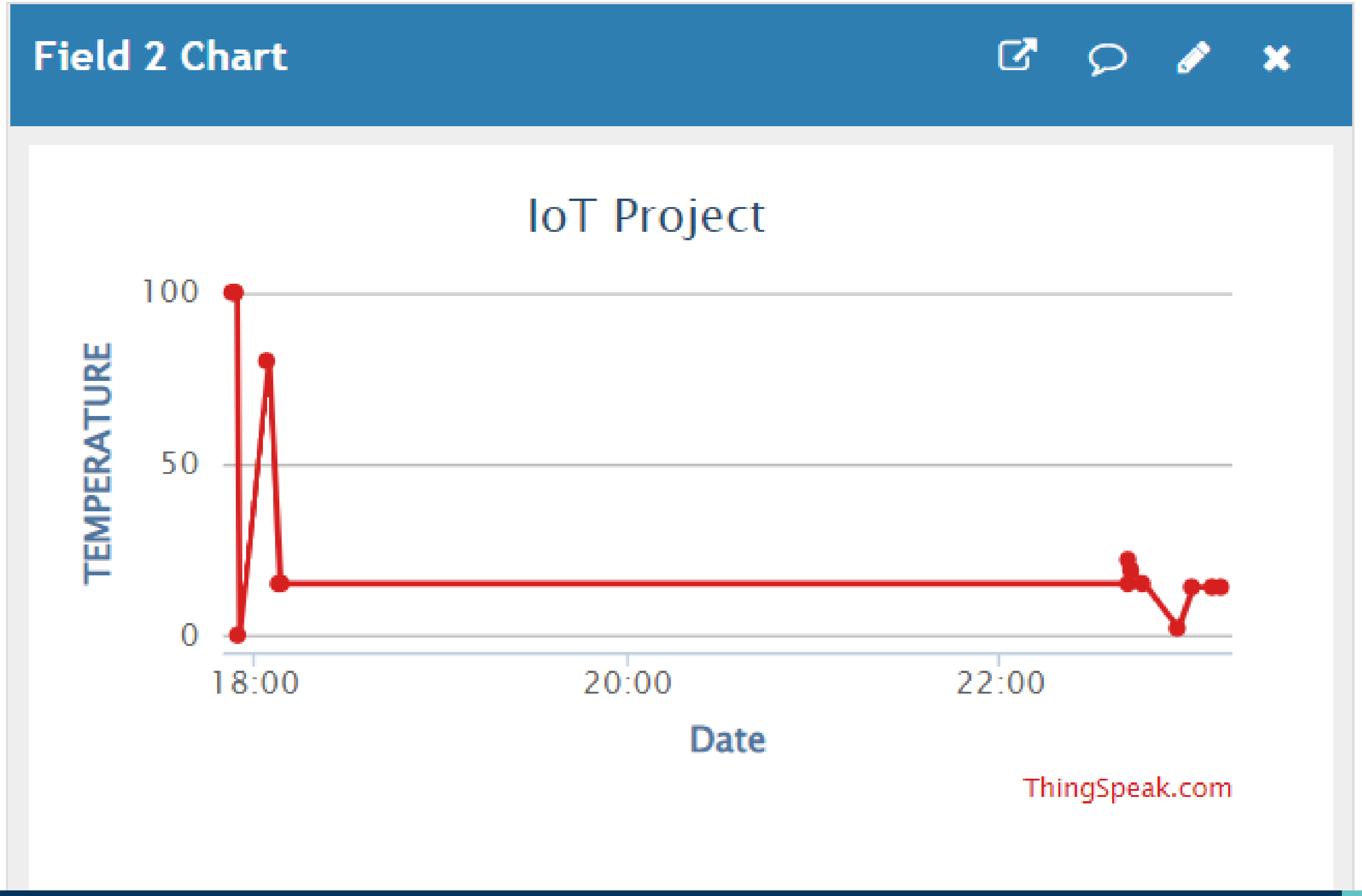


# Circuit Diagram



# ThingsSpeak Graph



# THANK YOU!