A

INTERNSHIP REPORT

On

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

A report submitted in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING

By

MARUBOYINA NAGA PAVAN KUMAR (20AJ1A0491)

Under the esteemed guidance of

Chethan Salunke, SmartBridge Educational Services Pvt. Ltd.

Hyderabad

(DURATION: From Nov 2023 to Apr 2024.)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

AMRITA SAI INSTITUTE OF SCIENCE AND TECHNOLOGY (AUTONOMOUS)

Paritala (V), Kanchikacherla (M), Krishna Dist– 521180.

(Approved by AICTE, NEW DELHI & Permanently affiliated to JNTU, Kakinada)

2020-2024

AMRITA SAI INSTITUTE OF SCIENCE & TECHNOLOGY
(AUTONOMOUS)
Approved by AICTE, New Delhi, Permanently Affiliated to JNTUK, Kakinada,
Recognized by UGC under 2(f) & 12(B) of 1956 Act.,
ISO 9001:2015 Certified Institution, Accredited by NAAC "A" Grade,
Paritala, Kanchikacherla, Krishna Dist, Andhra Pradesh- 521180.
www.amritasai.org.in, Phone: 0866 2428399.

CERTIFICATE

This is to certify that the **internship report** entitled **"ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING"** that is being submitted by **MARUBOYINA NAGA PAVAN KUMAR (20AJ1A0491)** in partial fulfillment for the award of the **Bachelor of Technology** in **ELECTRONICS AND COMMUNICATION ENGINEERING** to the Jawaharlal Nehru Technological University, Kakinada is a record of bonafide work carried by him/her at **SmartBridge Educational Services Pvt. Ltd., Hyderabad.**

Dr. V. Ramesh Babu
M.Tech, Ph.D, FIETE
Professor, Head of Department, ECE

Dr. A. VijayaLakshmi
M.E, Ph.D , FIETE ,MISTE
Professor, ECE
Department Internship Co-ordinator

External Examiner

Google for Developers

Smart Internz

ANDHRA PRADESH STATE COUNCIL OF HIGHER EDUCATION

(A Statutory Body of the Government of A.P)

CERTIFICATE OF COMPLETION

This is to certify that Ms./Mr. **Maruboyina Naga Pavan Kumar** of **Electronics and Communication Engineering (ECE)** with Registered Hall ticket no. **20AJ1A0491** under **Amrita Sai Institute Of Science & Technology** of **JNTUK** has successfully completed Long-Term Internship of 240 hours (6 months) on **Artificial Intelligence & Machine Learning** Organized by **SmartBridge Educational Services Pvt. Ltd.** in collaboration with **Andhra Pradesh State Council of Higher Education.**

Certificate ID: **EXT-APSCHE_AIML-11987**

Date: 20-Apr-2024

Place: Virtual

Amarendar Katkam

Founder & CEO

AMRITA SAI INSTITUTE OF SCIENCE & TECHNOLOGY
(AUTONOMOUS)
Approved by AICTE, New Delhi, Permanently Affiliated to JNTUK, Kakinada,
Recognized by UGC under 2(f) & 12(B) of 1956 Act.,
ISO 9001:2015 Certified Institution, Accredited by NAAC "A" Grade,
Paritala, Kanchikacherla, Krishna Dist, Andhra Pradesh- 521180.
www.amritasai.org.in, Phone: 0866 2428399.

ACKNOWLEDGEMENT

First I would like to thank **Chethan Salunke, SmartBridge Educational Services Pvt. Ltd, Hyderabad** for giving me the opportunity to do an internship within the organization.

I also would like all the people that worked along with me at **Chethan Salunke, SmartBridge Educational Services Pvt. Ltd ,Hyderabad .**with their patience and openness they created an enjoyable working environment.

It is indeed with a great sense of pleasure and immense sense of gratitude that I acknowledge the help of these individuals.

I feel highly obliged to internship co-ordinator, **Dr. A. Vijayalakshmi,** Professor, Department of Electronics and Communication Engineering, for the facilities provided to accomplish this internship.

I would like to thank my Head of the Department **Dr. V. Ramesh Babu** for his constructive criticism throughout my internship.

I am highly indebted to Principal **Dr. M. Sasidhar**, for the facilities provided to accomplish this internship.

I am extremely great full to my department staff members and friends who helped me in successful completion of this internship.

MARUBOYINA NAGA PAVAN KUMAR
(20AJ1A0491)

ABSTRACT

India is an agricultural country and its economy is largely based upon crop productivity and rainfall. For analyzing the crop productivity, rainfall prediction is require and necessary to all farmers. Rainfall Prediction is the application of science and technology to predict the state of the atmosphere. It is important to exactly determine the rainfall for effective use of water resources, crop productivity and pre planning of water structures. Using different data mining techniques it can predict rainfall. Data mining techniques are used to estimate the rainfall numerically. This paper focuses some of the popular data mining algorithms for rainfall prediction. CatBoost classifier, Random Forest, K-Nearest Neighbour algorithm, Logistic Regression , Decision Tree are some of the algorithms compared in this paper. From that comparison, it can analyze which method gives better accuracy for rainfall prediction.

TABLE

CHAPTER NO	TITLE	PAGE
1.	**INTRODUCTION**	1
	1.1 BACKGROUND AND BASICS	1
	1.2 EXISTING SYSTEM	1
	1.3 DISADVANTEGS OF EXISTING SYSTEM	1
	1.4 PROPOSED SYSTEM	2
	1.5 ADVANTAGES OF PROPOSED SYSTEM	2
2.	**LITERATURE REVIEW**	3
3.	**METHODOLOGY**	5
	3.1 SYSTEM REQUIRMENT SPECIFICATION	5
	3.1.1 HARDWARE REQUIREMENTS	5
	3.1.2 SOFTWARE REQUIREMENTS	5
	3.2 SOFTWARE ENVIRONMENT	5
	3.3 ARCHITECTURE	11
	3.4 MODULES	11
	3.5 DATAFLOW DIAGRAM	12
	3.6 UML DIAGRAM	15
	3.7 FEASIBILITY STUDY	18
	3.8 SYSTEM DESIGN AND TESTING PLAN	19
4.	**RESULT AND DISCUSSION**	22
5.	**CONCLUSION**	23
6.	**REFERENCES**	24
7.	**APPENDICES**	25

CHAPTER 1
INTRODUCTION

1.1 BACKGROUND AND BASICS

Rainfall Prediction is one of the most challenging tasks. Though already many algorithms have being proposed but still accurate prediction of rainfall is very difficult. In an agricultural country like India, the success or failure of the crops an d water scarcity in any year is always viewed with greatest concern. A small fluctuation in the seasonal rainfall can have devastating impacts on agriculture sector. Accurate rainfall prediction has a potential benefit of preventing causalities and damages caused by natural disasters. Under certain circumstances such as flood and drought, highly accurate rainfall prediction is useful for agriculture management and disaster prevention. In this paper, various algorithms have been analyzed. Data mining techniques are efficiently used in rainfall prediction.

1.2 EXISTING SYSTEM

Agriculture is the strength of our Indian economy. Farmer only depends upon monsoon to be their cultivation. The good crop productivity needs good soil, fertilizer and also good climate. Weather forecasting is the very important requirement of the each farmer. Due to the sudden changes in climate/weather, The people are suffered economically and physically. Weather prediction is one of the challenging problems in current state. The main motivation of this paper to predict the weather using various data mining techniques. Such as classification, clustering, decision tree and also neural networks. Weather related information is also called the meteorological data. In this paper the most commonly used weather parameters are rainfall, wind speed, temperature and cold.

1.3 DISADVANTAGES OF EXISTING SYSTEM

1.Classification

2.Clustering

3.Decision Tree

1.4 POPOSED SYSTEM

Rainfall is important for food production plan, water resource management and all activity plans in the nature. The occurrence of prolonged dry period or heavy rain at the critical stages of the crop growth and development may lead to significantly reduce crop yield. India is an agricultural country and its economy is largely based upon crop productivity. Thus, rainfall prediction becomes a significant factor in agricultural countries like India. Rainfall forecastinghas been one of the most scientifically and technologically challenging problems around the world in the last century.

1.5 ADVANTAGES OF PROPOSED SYSTEM

1.Numerical Weather Prediction.

2.Statistical Weather Prediction.

3.Synoptic Weather Prediction.

CHAPTER 2
LITERATURE REVIEW

- Pritpal Singh et al.[1] Measurable investigation shows the idea of ISMR, which can't be precisely anticipated by insights or factual information. Hence, this review exhibits the utilization of three techniques: object creation, entropy, and artificial neural network (ANN). In view of this innovation, another technique for anticipating ISMR times has been created to address the idea of ISMR. This model has been endorsed and supported by the studio and exploration data. Factual examination of different information and near investigations showing the presentation of the normal technique

- Sam Carmer , Michael Kampouridis, Alex A. Freitas , Antonios Alexandridis et al.[2] The primary impact of this movement is to exhibit the advantages of AI calculations, just as the more prominent degree of clever framework than the advanced rainfall determining methods. We analyze and think about the momentum execution (Markov chain stretched out by rainfall research) with the forecasts of the six most notable AI machines: Genetic programming, Vector relapse support, radio organizations, M5 organizations, M5 models, models - Happy. To work with a more itemized appraisal, we led a rainfall overview utilizing information from 42 metropolitan urban communities.

- Sahar Hadi Poura , Shamsuddin Shahida, Eun-Sung chungb et al. [3] RF wasutilized to anticipate assuming that it would rain in one day, while SVM was utilizedto foresee downpour on a blustery day. The limit of the Hybrid model was fortified by the decrease of day-by-day rainfall in three spots at the rainfall level in the eastern piece of Malaysia. Crossover models have likewise been found to emulate the full change, the quantity of days straight, 95% of the month-to-month rainfall, and the dispersion of the noticed rainfall

- Tanvi Patil, Dr. Kamal Shah et al. [4] The reason for the framework is to anticipate

the climate sooner or later. Climatic still up in the air utilizing various sorts of factors all over the place. Of these, main the main highlights are utilized in climate conjectures. Picking something like this relies a great deal upon the time you pick. Underlying displaying is utilized to incorporate the fate of demonstrating, AI applications, data trade, and character examination.

- N.Divya Prabha, P. Radha et al. [5] Contrasted with different spots where rainfall information isn't accessible, it consumes a large chunk of the day to build up a solid water overview for a long time. Improving complex neural organizations is intended to be a brilliant instrument for anticipating the stormy season. This downpour succession was affirmed utilizing a complex perceptron neural organization. Estimations like MSE (Early Modeling), NMSE (Usually Early Error), and the arrangement of informational collections for transient arranging are clear in the examination of different organizations, like Adanaive. AdaSVM.

- Senthamil Selvi S, Seetha et al. [6] In this paper, Artificial Neural Network (ANN) innovation is utilized to foster a climate anticipating strategy to distinguish rainfall utilizing Indian rainfall information. Along these lines, Feed Forward Neural Network (FFNN) was utilized utilizing the Backpropagation Algorithm. Execution of the two models is assessed dependent on emphasis examination, Mean Square Error (MSE) and Magnitude of Relative Error (MRE). This report likewise gives a future manual for rainfall determining.

- YashasAthreya, VaishaliBV, SagarK and SrinidhiHR, et al.[7] This page features rainfall investigation speculations utilizing Machine Learning. The principle motivation behind utilizing this program is to secure against the impacts of floods. This program can be utilized by conventional residents or the public authority to anticipate what will occur before the flood. The flood card, then, at that point, furnish them with the vital help by moving versatile.

CHAPTER 3

METHODOLOGY

3.1 System Requirement Specification

3.1.1 HARDWARE REQUIREMENTS:

System - Windows7/10

Speed - 2.4GHZ

Hard disk - 40GB

Monitor - 15VGA Color

Ram - 4GB

3.1.2 SOFTWARE REQUIREMENTS:

Coding Language - PYTHON

IDE - PYCHARM

3.2 SOFTWARE ENVIRONMENT

Python:

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

- **Python is Interpreted** − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.
- **Python is Interactive** − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.
- **Python is Object-Oriented** − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.
- **Python is a Beginner's Language** − Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

History of Python

Python was developed by Guido van Rossum in the late eighties and early nineties at the National Research Institute for Mathematics and Computer Science in the Netherlands.

Python is derived from many other languages, including ABC, Modula-3, C, C++, Algol-68, SmallTalk, and Unix shell and other scripting languages.

Python is copyrighted. Like Perl, Python source code is now available under the GNU GeneralPublic License (GPL).

Python is now maintained by a core development team at the institute, although Guido van Rossumstill holds a vital role in directing its progress.

Python Features

Python's features include −

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** − Python's source code is fairly easy-to-maintain.
- **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** − You can add low-level modules to the Python interpreter. Thesemodules enable programmers to add to or customize their tools to be more efficient.
- **Databases** − Python provides interfaces to all major commercial databases.
- **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC

Getting Python

The most up-to-date and current source code, binaries, documentation, news, etc., isavailable on the official website of Python https://www.python.org.

Windows Installation

Here are the steps to install Python on Windows machine.

- Open a Web browser and go to https://www.python.org/downloads/.
- Follow the link for the Windows installer python-XYZ.msifile where XYZ is the version you need to install.
- To use this installer python-XYZ.msi, the Windows system must support Microsoft Installer 2.0. Save the installer file to your local machine and then run it to find out if your machine supports MSI.
- Run the downloaded file. This brings up the Python install wizard, which is really easy to use. Just accept the default settings, wait until the install is finished, and you are done.

The Python language has many similarities to Perl, C, and Java. However, there are somedefinite differences between the languages.

First Python Program

Let us execute programs in different modes of programming.

Interactive Mode Programming

Invoking the interpreter without passing a script file as a parameter brings up the following prompt −

```
$python

Python2.4.3(#1,Nov112010,13:34:43)

[GCC 4.1.220080704(RedHat4.1.2-48)] on linux2

Type"help","copyright","credits"or"license"for more information.

>>>
```

Type the following text at the Python prompt and press the Enter −

```
>>>print"Hello, Python!"
```

If you are running new version of Python, then you would need to use print statement with parenthesis as in **print ("Hello, Python!");**. However in Python version 2.4.3, this produces the following result −

```
Hello, Python!
```

Script Mode Programming

Invoking the interpreter with a script parameter begins execution of the script and continuesuntil the script is finished. When the script is finished, the interpreter is no longer active.

Let us write a simple Python program in a script. Python files have extension **.py**. Type the following source code in a test.py file −

```
print"Hello, Python!"
```

We assume that you have Python interpreter set in PATH variable. Now, try to run thisprogram as follows −

```
$ python test.py
```

This produces the following result −

```
Hello, Python!
```

Python Install

Many PCs and Macs will have python already installed.

To check if you have python installed on a Windows PC, search in the start bar for Pythonor run the following on the Command Line (cmd.exe):

```
C:\Users\Your Name>python --version
```

To check if you have python installed on a Linux or Mac, then on linux open the commandline

or on Mac open the Terminal and type:

```
python --version
```

If you find that you do not have python installed on your computer, then you can downloadit for free from the following website: https://www.python.org/

Python Quickstart

Python is an interpreted programming language, this means that as a developer you writePython (.py) files in a text editor and then put those files into the python interpreter to be executed.

The way to run a python file is like this on the command line:

```
C:\Users\Your Name>python helloworld.py
```

Where "helloworld.py" is the name of your python file.

Let's write our first Python file, called helloworld.py, which can be done in any text editor.

helloworld.py

```
print("Hello, World!")
```

Simple as that. Save your file. Open your command line, navigate to the directory whereyou saved your file, and run:

```
C:\Users\Your Name>python helloworld.py
```

The output should read:

```
Hello, World!
```

Congratulations, you have written and executed your first Python program

The Python Command Line

To test a short amount of code in python sometimes it is quickest and easiest not to writethe code

in a file. This is made possible because Python can be run as a command line itself.

Type the following on the Windows, Mac or Linux command line:

```
C:\Users\Your Name>python
```

Which will write "Hello, World!" in the command line:

```
C:\Users\Your Name>python
Python 3.6.4 (v3.6.4:d48eceb, Dec 19 2017, 06:04:45) [MSC v.1900 32 bit (Intel)] on win32Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello, World!")
Hello, World!
```

Whenever you are done in the python command line, you can simply type the following toquit the python command line interface:

```
exit()
```

Execute Python Syntax

As we learned in the previous page, Python syntax can be executed by writing directly inthe Command Line:

```
>>> print("Hello, World!")
Hello, World!
```

Or by creating a python file on the server, using the .py file extension, and running it in theCommand Line:

```
C:\Users\Your Name>python myfile.py
```

Python Indentations

Where in other programming languages the indentation in code is for readability only, inPython the indentation is very important.

Python uses indentation to indicate a block of code.

Example

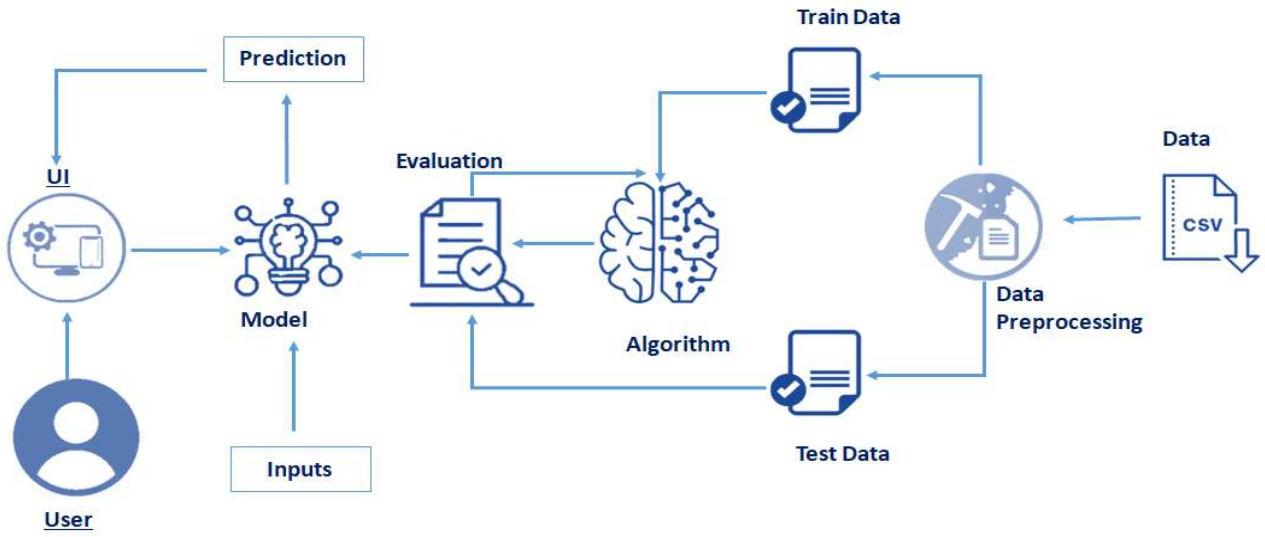
```
if 5 > 2:
  print("Five is greater than two!")
```

Python will give you an error if you skip the indentation:

Example

```
if 5 > 2:
print("Five is greater than two!")
```

3.3 ARCHITECTURE



3.4 MODULES

- Data Collection
- Data Cleaning
- Data Selection
- Data Transformation
- Data Mining Stage

Data Collection

The data used for this work was collected from meteorologist‟s centre. The case data covered the period of 2012 to 2015. The following procedures were adopted at this stage ofthe research: Data Cleaning, Data Selection, Data Transformation and Data Mining.

Data Cleaning

In this stage, a consistent format for the data model was developed which is searchmissing

data, finding duplicated data, and weeding out of bad data. Finally system cleaned data were transformed into a format suitable for data mining.

Data Selection

At this stage, data relevant to the analysis like decision tree was decided on and retrieved from the dataset. The Meteorological dataset had ten attributes in that were using two attributes for future prediction. Due to the nature of the Cloud Form data where all the values are the same and the high percentage of missing values in the sunshine data both were not used in the analysis.

Data Transformation

This is also known as data consolidation". It is the stage in which the selected data is transformed into forms appropriate for data mining. The data file was saved in Comma Separated Value (CVS) file format and the datasets were normalized to reduce the effect ofscaling on the data.

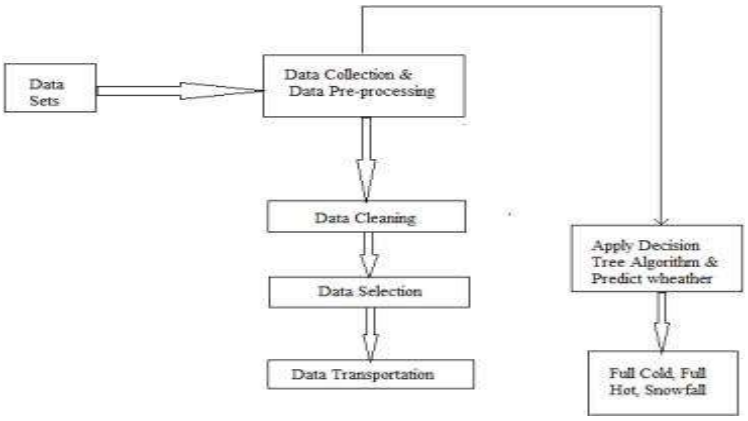
Data Mining Stage

The data mining stage was divided into three phases. At each phase all the algorithmswere used to analyse the meteorological datasets. The testing method adopted for this research was percentage split that train on a percentage of the dataset, cross validate on it and test on the remaining percentage. There after interesting patterns representing knowledge were identified.

3.5 DATA FLOW DIAGRAM

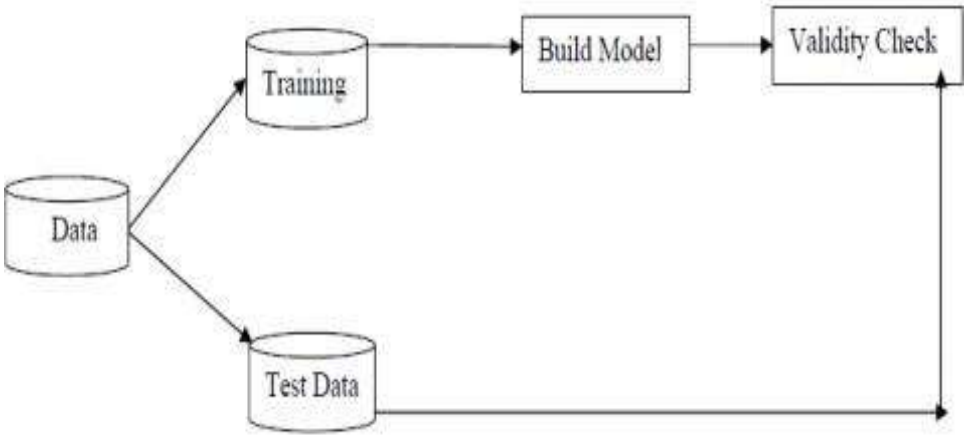
1. The DFD is also called as bubble chart. It is a simple graphical formalism that canbe used to represent a system in terms of input data to the system, various processing carried out on this data, and the output data is generated by this system.
2. The data flow diagram (DFD) is one of the most important modeling tools. It is used to model the system components. These components are the system process, the data used by the process, an external entity that interacts with the system and the information flows in the system.
3. DFD shows how the information moves through the system and how it is modifiedby a series of transformations. It is a graphical technique that depicts informationflow

LEVEL - 0



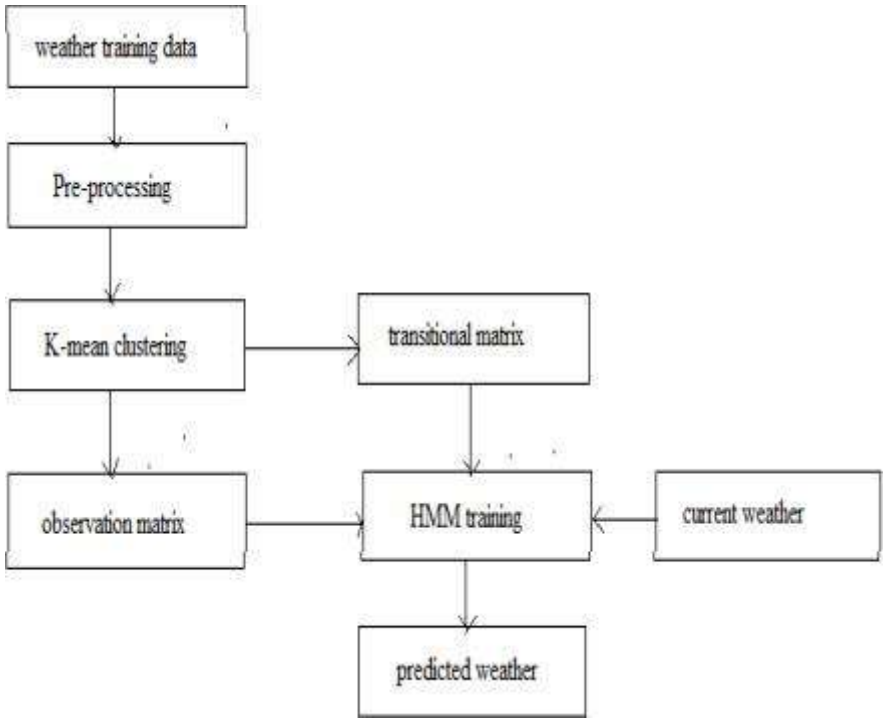
- **DATAFLOW DIAGRAM 1**

LEVEL - 1



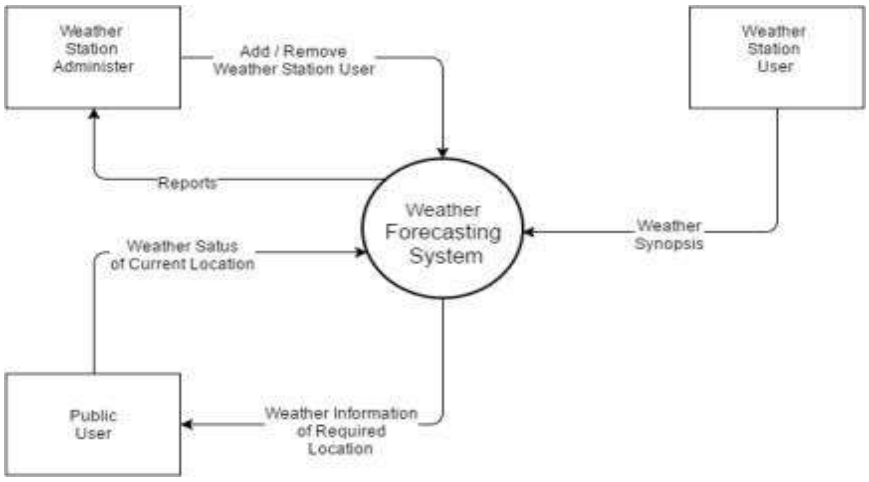
- **DATAFLOW DIAGRAM 2**

LEVEL - 2



- **DATAFLOW DIAGRAM 3**

LEVEL – 3



- **DATAFLOW DIAGRAM 4**

3.6 UML DIAGRAMS

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

GOALS:

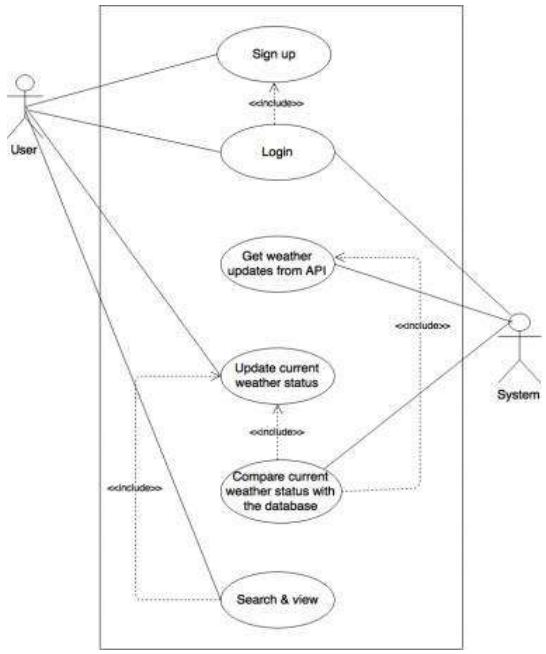
The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modeling Language so that they candevelop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patternsand components.
7. Integrate best practices.

USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a

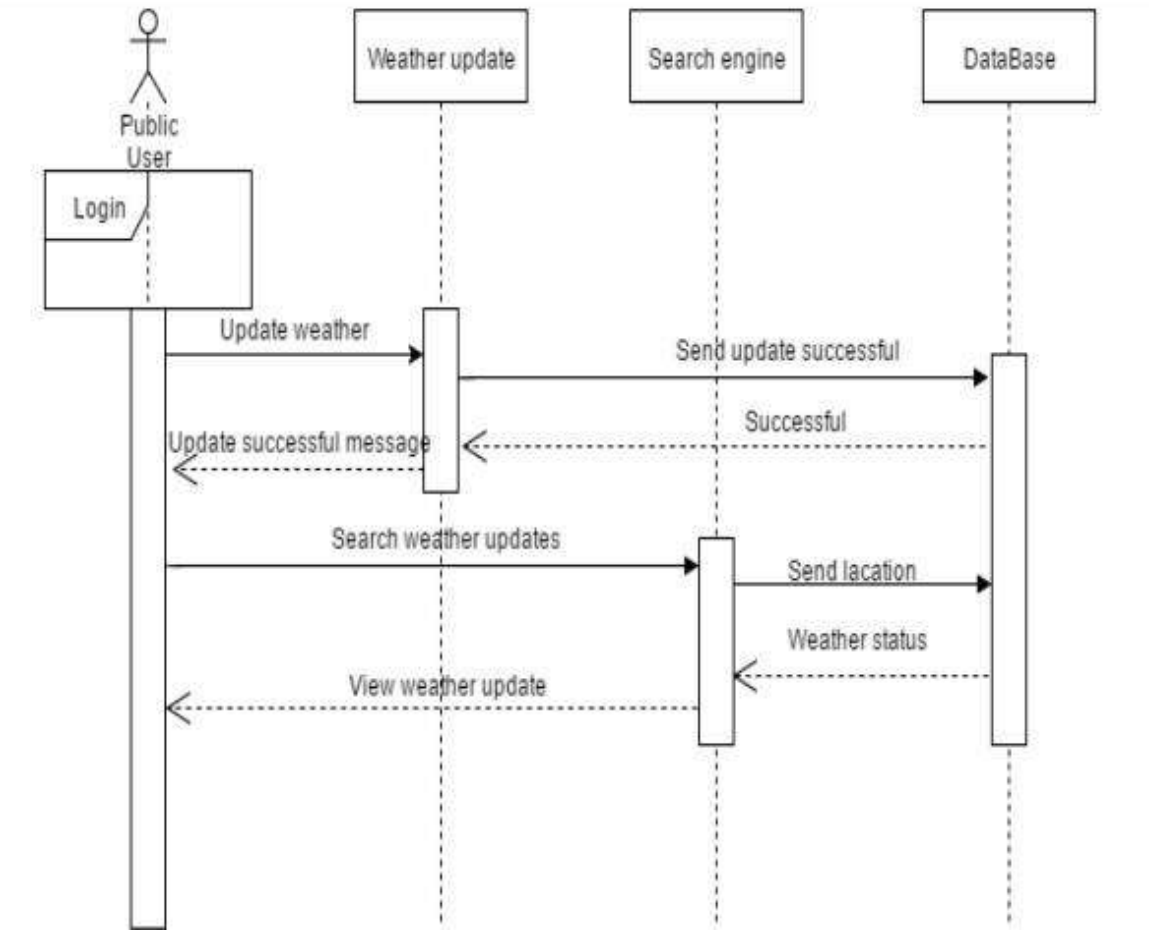
graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



USECASE DIAGRAM

SEQUENCE DIAGRAM:

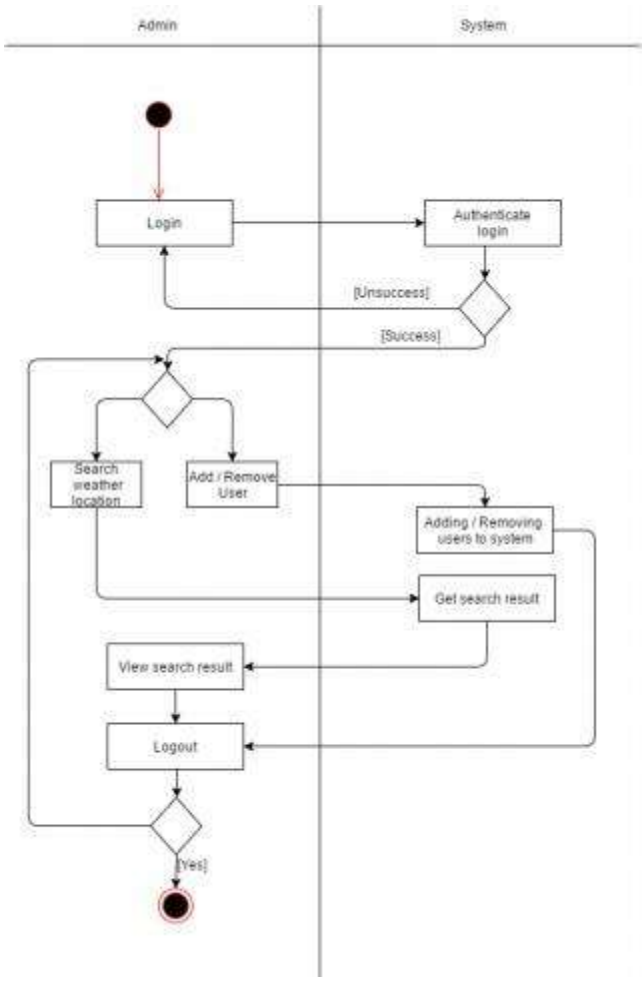
A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of aMessage Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



3.6.1 SEQUENCE DIAGRAM

ACTIVITY DIAGRAM:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.



3.6.2 ACTIVITY DIAGRAM

3.7 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

The feasibility study investigates the problem and the information needs of thestakeholders. It seeks to determine the resources required to provide an information systems solution, the cost and benefits of such a solution, and the feasibility of such a solution.

The goal of the feasibility study is to consider alternative information systems solutions, evaluate their feasibility, and propose the alternative most suitable to the organization. The feasibility of a proposed solution is evaluated in terms of its components.

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirementsof the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system musthave a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity.

3.8 SYSTEM DESIGN AND TESTING PLAN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input

focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the informationis to be displaced for immediate need and also the hard copy output. It is the mostimportant and direct source information to the user. Efficient and intelligent output design improves the system‟s relationship to help user decision-making.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

TEST PLAN

Software testing is the process of evaluation a software item to detect differences between given input and expected output. Also to assess the feature of a software item. Testing assesses the quality of the product. Software testing is a process that should be done during the development process. In other words software testing is a verification and validation process.

Verification

Verification is the process to make sure the product satisfies the conditions imposed at the start of the development phase. In other words, to make sure the product behaves

Validation

Validation is the process to make sure the product satisfies the specified requirements at the end of the development phase. In other words, to make sure the product is built as per customer requirements.

Basics of software testing

There are two basics of software testing: black box testing and white box testing.

Black box Testing

Black box testing is a testing technique that ignores the internal mechanism of the system and focuses on the output generated against any input and execution of the system. It is also called functional testing.

White box Testing

White box testing is a testing technique that takes into account the internal mechanism of a system. It is also called structural testing and glass box testing. Black box testing is often used for validation and white box testing is often used for verification.

CHAPTER 4

RESULTS AND DISCUSSION

CatBoost Classifier ,RandomForest Classifier ,Logistic Regression ,KNeighbours Classifier are the classification method used for time series predict in this research work out of above algorithms the catboost classifier gives the accuracy of 86% which is high. Two group are separated from the data set for training and for testing the algorithms of classification. To execute the classification algorithms, the tool used is flask webapp data examination. For classification procedure no more than a separation of data is particular from the loaded data. To choose a subset from innovative data, "Select attribute" are utilised by the operative. The preferred subset is then subjected to "X-Validation" operator. It develop the classification representation which is validated by the test data.

CHAPTER 5
CONCLUSION

In conclusion, the machine learning model developed in this project demonstrates promising results in predicting rain based on meteorological features. Further optimizations and feature engineering could potentially improve the model's performance. Additionally, the model could be integrated into a web application or API for real-time rainfall prediction and decision-making in various sectors such as agriculture, water resource management, and disaster preparedness.

FUTURE WORK

The future work of the project would be the improvement of architecture for light and other weather scenarios. Also, can develop a model for small changes in climate in future. An algorithm for testing daily basis dataset instead of accumulated dataset could be of paramount Importance for further research.

CHAPTER 6
REFERENCES

[1] Xiong, Lihua, and Kieran M. OConnor. “An empirical method to improve the prediction limits of the GLUE methodology in rainfallrunoff modeling.” Journal of Hydrology 349.1-2 (2008): 115-124.

[2] Schmitz, G. H., and J. Cullmann. “PAI-OFF: A new proposal for online flood forecasting in flash flood prone catchments.” Journal of hydrology 360.1-4 (2008): 1-14.

[3] Riordan, Denis, and Bjarne K. Hansen. “A fuzzy casebased system for weather prediction.” Engineering Intelligent Systems for Electrical Engineering and Communications10.3 (2002): 139-146.

[4] Guhathakurta, P. “Long-range monsoon rainfall prediction of 2005 for the districts and sub-division Kerala with artificial neural network.” Current Science 90.6 (2006): 773-779.

[5] Pilgrim, D. H., T. G. Chapman, and D. G. Doran. “Problems of rainfall-runoff modelling in arid and semiarid regions.” Hydrological Sciences Journal 33.4 (1988): 379-400.

[6] Lee, Sunyoung, Sungzoon Cho, and Patrick M. Wong. “Rainfall prediction using artificialneural networks.” journal of geographic information and Decision Analysis 2.2 (1998): 233- 242..

[7] French, Mark N., Witold F. Krajewski, and Robert R. Cuykendall. “Rainfall forecasting in space and time using a neural network.” Journal of hydrology 137.1-4 (1992): 1-31.

[8] Charaniya, Nizar Ali, and Sanjay V. Dudul. “Committee of artificial neural networks for monthly rainfall prediction using wavelet transform.” Business, Engineering and Industrial Applications (ICBEIA), 2011 International Conference on. IEEE, 2011.

[9] Noone, David, and Harvey Stern. “Verification of rainfall forecasts from the Australian Bureau of Meteorology‟s Global Assimilation and Prognosis(GASP) system.” Australian Meteorological Magazine 44.4 (1995): 275-286.

[10] Hornik, Kurt, Maxwell Stinchcombe, and Halbert White. “Multilayer feedforward networks are universal approximators.” Neural networks 2.5 (1989): 359-366.

[11] Haykin, Simon. Neural networks: a comprehensive foundation. Prentice Hall PTR, 1994.

[12] Rajeevan, M., Pulak Guhathakurta, and V. Thapliyal. “New models for long range forecasts of summer monsoon rainfall over North West and Peninsular India.” Meteorology and Atmospheric Physics 73.3-4 (2000): 211-225.

CHAPTER 7

APPENDICIES

Sample code

```
# %%
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import preprocessing
import scipy.stats as stats
from sklearn.model_selection import train_test_split
from collections import Counter
from imblearn.over_sampling import SMOTE
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report
from sklearn import metrics
from sklearn.ensemble import RandomForestClassifier
from catboost import CatBoostClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
import joblib

# %%
df = pd.read_csv("weatherAUS.csv")
pd.set_option("display.max_columns", None)
df

# %%
numerical_feature = [feature for feature in df.columns if df[feature].dtypes != 'O']
discrete_feature=[feature for feature in numerical_feature if len(df[feature].unique())<25]
continuous_feature = [feature for feature in numerical_feature if feature not in discrete_feature]
```

```
categorical_feature = [feature for feature in df.columns if feature not in numerical_feature]
print("Numerical Features Count {}".format(len(numerical_feature)))
print("Discrete feature Count {}".format(len(discrete_feature)))
print("Continuous feature Count {}".format(len(continuous_feature)))
print("Categorical feature Count {}".format(len(categorical_feature)))

# %%
# Handle Missing Values
df.isnull().sum()*100/len(df)

# %%
print(numerical_feature)

# %%
print(discrete_feature)

# %%
def randomsampleimputation(df, variable):
    df[variable]=df[variable]
    random_sample=df[variable].dropna().sample(df[variable].isnull().sum(),random_state=0)
    random_sample.index=df[df[variable].isnull()].index
    df.loc[df[variable].isnull(),variable]=random_sample

# %%
randomsampleimputation(df, "Cloud9am")
randomsampleimputation(df, "Cloud3pm")
randomsampleimputation(df, "Evaporation")
randomsampleimputation(df, "Sunshine")

# %%
df
```

```
# %%
corrmat = df.corr(method = "spearman")
plt.figure(figsize=(20,20))
#plot heat map
g=sns.heatmap(corrmat,annot=True)

# %%
for feature in continuous_feature:
    data=df.copy()
    sns.distplot(df[feature])
    plt.xlabel(feature)
    plt.ylabel("Count")
    plt.title(feature)
    plt.figure(figsize=(15,15))
    plt.show()

# %%
#A for loop is used to plot a boxplot for all the continuous features to see the outliers
for feature in continuous_feature:
    data=df.copy()
    sns.boxplot(data[feature])
    plt.title(feature)
    plt.figure(figsize=(15,15))

# %%
for feature in continuous_feature:
    if(df[feature].isnull().sum()*100/len(df))>0:
        df[feature] = df[feature].fillna(df[feature].median())

# %%
df.isnull().sum()*100/len(df)
```

```
# %%
discrete_feature

# %%
def mode_nan(df,variable):
    mode=df[variable].value_counts().index[0]
    df[variable].fillna(mode,inplace=True)
mode_nan(df,"Cloud9am")
mode_nan(df,"Cloud3pm")

# %%
df["RainToday"] = pd.get_dummies(df["RainToday"], drop_first = True)
df["RainTomorrow"] = pd.get_dummies(df["RainTomorrow"], drop_first = True)
df

# %%
for feature in categorical_feature:
        print(feature,    (df.groupby([feature])["RainTomorrow"].mean().sort_values(ascending    =
False)).index)

# %%
windgustdir = {'NNW':0, 'NW':1, 'WNW':2, 'N':3, 'W':4, 'WSW':5, 'NNE':6, 'S':7, 'SSW':8, 'SW':9,
'SSE':10,
     'NE':11, 'SE':12, 'ESE':13, 'ENE':14, 'E':15}
winddir9am = {'NNW':0, 'N':1, 'NW':2, 'NNE':3, 'WNW':4, 'W':5, 'WSW':6, 'SW':7, 'SSW':8,
'NE':9, 'S':10,
     'SSE':11, 'ENE':12, 'SE':13, 'ESE':14, 'E':15}
winddir3pm = {'NW':0, 'NNW':1, 'N':2, 'WNW':3, 'W':4, 'NNE':5, 'WSW':6, 'SSW':7, 'S':8, 'SW':9,
'SE':10,
     'NE':11, 'SSE':12, 'ENE':13, 'E':14, 'ESE':15}
df["WindGustDir"] = df["WindGustDir"].map(windgustdir)
df["WindDir9am"] = df["WindDir9am"].map(winddir9am)
```

```
df["WindDir3pm"] = df["WindDir3pm"].map(winddir3pm)

# %%
df["WindGustDir"] = df["WindGustDir"].fillna(df["WindGustDir"].value_counts().index[0])
df["WindDir9am"] = df["WindDir9am"].fillna(df["WindDir9am"].value_counts().index[0])
df["WindDir3pm"] = df["WindDir3pm"].fillna(df["WindDir3pm"].value_counts().index[0])

# %%
df.isnull().sum()*100/len(df)

# %%
df1 = df.groupby(["Location"])["RainTomorrow"].value_counts().sort_values().unstack()

# %%
df1

# %%
df1[1].sort_values(ascending = False)

# %%
df1[1].sort_values(ascending = False).index

# %%
len(df1[1].sort_values(ascending = False).index)

# %%
location = {'Portland':1, 'Cairns':2, 'Walpole':3, 'Dartmoor':4, 'MountGambier':5,
    'NorfolkIsland':6, 'Albany':7, 'Witchcliffe':8, 'CoffsHarbour':9, 'Sydney':10,
    'Darwin':11, 'MountGinini':12, 'NorahHead':13, 'Ballarat':14, 'GoldCoast':15,
    'SydneyAirport':16, 'Hobart':17, 'Watsonia':18, 'Newcastle':19, 'Wollongong':20,
    'Brisbane':21, 'Williamtown':22, 'Launceston':23, 'Adelaide':24, 'MelbourneAirport':25,
    'Perth':26, 'Sale':27, 'Melbourne':28, 'Canberra':29, 'Albury':30, 'Penrith':31,
```

```
    'Nuriootpa':32, 'BadgerysCreek':33, 'Tuggeranong':34, 'PerthAirport':35, 'Bendigo':36,
    'Richmond':37, 'WaggaWagga':38, 'Townsville':39, 'PearceRAAF':40, 'SalmonGums':41,
    'Moree':42, 'Cobar':43, 'Mildura':44, 'Katherine':45, 'AliceSprings':46, 'Nhil':47,
    'Woomera':48, 'Uluru':49}
df["Location"] = df["Location"].map(location)

# %%
df["Date"] = pd.to_datetime(df["Date"], format = "%Y-%m-%dT", errors = "coerce")

# %%
df["Date_month"] = df["Date"].dt.month
df["Date_day"] = df["Date"].dt.day

# %%
df

# %%
corrmat = df.corr()
plt.figure(figsize=(20,20))
#plot heat map
g=sns.heatmap(corrmat,annot=True)

# %%
sns.countplot(df["RainTomorrow"])

# %%
df

# %%
for feature in continuous_feature:
    print(feature)
```

```
# %%
IQR=df.MinTemp.quantile(0.75)-df.MinTemp.quantile(0.25)
lower_bridge=df.MinTemp.quantile(0.25)-(IQR*1.5)
upper_bridge=df.MinTemp.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['MinTemp']>=30.45,'MinTemp']=30.45
df.loc[df['MinTemp']<=-5.95,'MinTemp']=-5.95

# %%
IQR=df.MaxTemp.quantile(0.75)-df.MaxTemp.quantile(0.25)
lower_bridge=df.MaxTemp.quantile(0.25)-(IQR*1.5)
upper_bridge=df.MaxTemp.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['MaxTemp']>=43.5,'MaxTemp']=43.5
df.loc[df['MaxTemp']<=2.7,'MaxTemp']=2.7

# %%
IQR=df.Rainfall.quantile(0.75)-df.Rainfall.quantile(0.25)
lower_bridge=df.Rainfall.quantile(0.25)-(IQR*1.5)
upper_bridge=df.Rainfall.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['Rainfall']>=1.5,'Rainfall']=1.5
df.loc[df['Rainfall']<=-0.89,'Rainfall']=-0.89

# %%
IQR=df.Evaporation.quantile(0.75)-df.Evaporation.quantile(0.25)
```

```
lower_bridge=df.Evaporation.quantile(0.25)-(IQR*1.5)
upper_bridge=df.Evaporation.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['Evaporation']>=14.6,'Evaporation']=14.6
df.loc[df['Evaporation']<=-4.6,'Evaporation']=-4.6

# %%
IQR=df.WindGustSpeed.quantile(0.75)-df.WindGustSpeed.quantile(0.25)
lower_bridge=df.WindGustSpeed.quantile(0.25)-(IQR*1.5)
upper_bridge=df.WindGustSpeed.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['WindGustSpeed']>=68.5,'WindGustSpeed']=68.5
df.loc[df['WindGustSpeed']<=8.5,'WindGustSpeed']=8.5

# %%
IQR=df.WindSpeed9am.quantile(0.75)-df.WindSpeed9am.quantile(0.25)
lower_bridge=df.WindSpeed9am.quantile(0.25)-(IQR*1.5)
upper_bridge=df.WindSpeed9am.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['WindSpeed9am']>=37,'WindSpeed9am']=37
df.loc[df['WindSpeed9am']<=-11,'WindSpeed9am']=-11

# %%
IQR=df.WindSpeed3pm.quantile(0.75)-df.WindSpeed3pm.quantile(0.25)
lower_bridge=df.WindSpeed3pm.quantile(0.25)-(IQR*1.5)
upper_bridge=df.WindSpeed3pm.quantile(0.75)+(IQR*1.5)
```

```
print(lower_bridge, upper_bridge)

# %%
df.loc[df['WindSpeed3pm']>40.5,'WindSpeed3pm']=40.5
df.loc[df['WindSpeed3pm']<=-3.5,'WindSpeed3pm']=-3.5

# %%
IQR=df.Humidity9am.quantile(0.75)-df.Humidity9am.quantile(0.25)
lower_bridge=df.Humidity9am.quantile(0.25)-(IQR*1.5)
upper_bridge=df.Humidity9am.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['Humidity9am']>=122,'Humidity9am']=122
df.loc[df['Humidity9am']<=18,'Humidity9am']=18

# %%
IQR=df.Pressure9am.quantile(0.75)-df.Pressure9am.quantile(0.25)
lower_bridge=df.Pressure9am.quantile(0.25)-(IQR*1.5)
upper_bridge=df.Pressure9am.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['Pressure9am']>=1034.25,'Pressure9am']=1034.25
df.loc[df['Pressure9am']<=1001.05,'Pressure9am']=1001.05

# %%
IQR=df.Pressure3pm.quantile(0.75)-df.Pressure3pm.quantile(0.25)
lower_bridge=df.Pressure3pm.quantile(0.25)-(IQR*1.5)
upper_bridge=df.Pressure3pm.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)
```

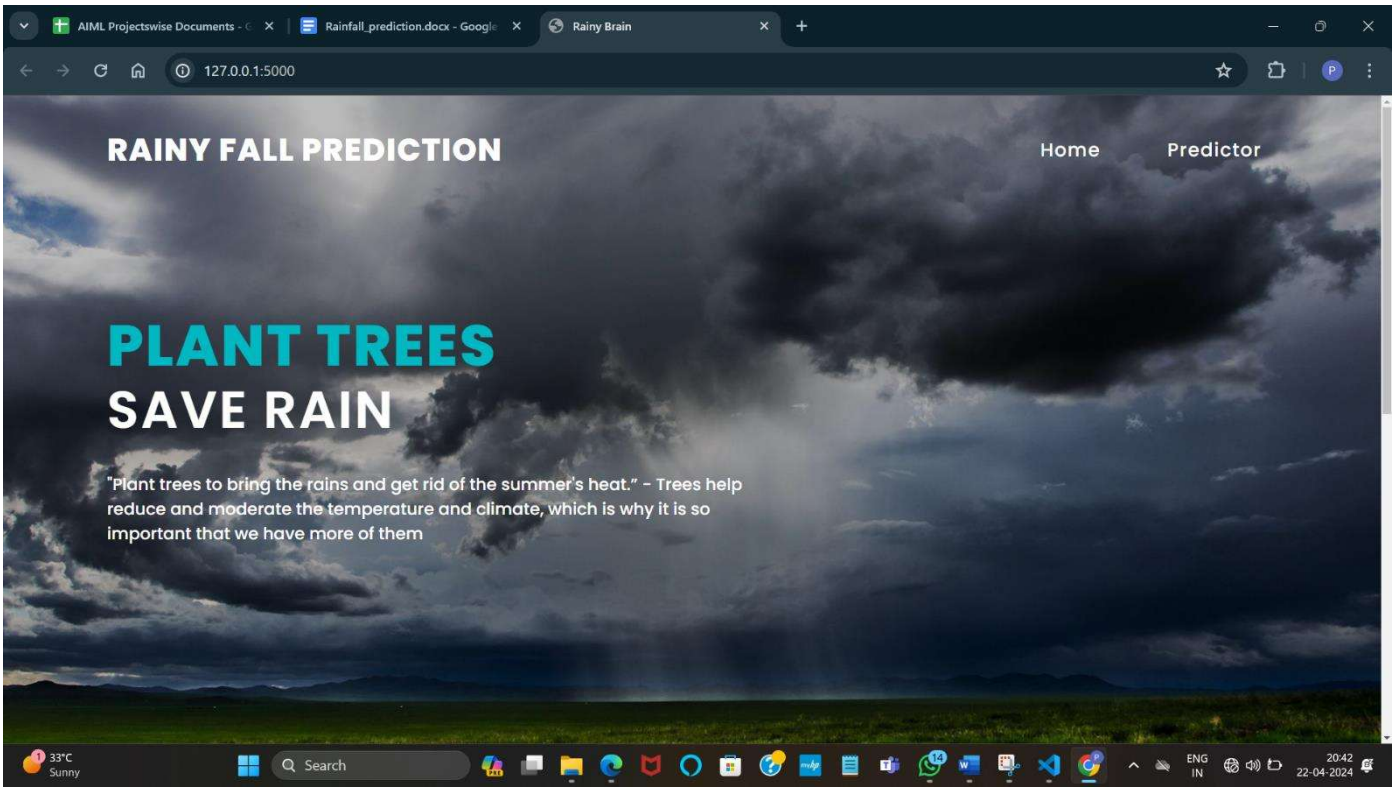
```
# %%
df.loc[df['Pressure3pm']>=1031.85,'Pressure3pm']=1031.85
df.loc[df['Pressure3pm']<=998.65,'Pressure3pm']=998.65

# %%
IQR=df.Temp9am.quantile(0.75)-df.Temp9am.quantile(0.25)
lower_bridge=df.Temp9am.quantile(0.25)-(IQR*1.5)
upper_bridge=df.Temp9am.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['Temp9am']>=35.3,'Temp9am']=35.3
df.loc[df['Temp9am']<=-1.49,'Temp9am']=-1.49

# %%
IQR=df.Temp3pm.quantile(0.75)-df.Temp3pm.quantile(0.25)
lower_bridge=df.Temp3pm.quantile(0.25)-(IQR*1.5)
upper_bridge=df.Temp3pm.quantile(0.75)+(IQR*1.5)
print(lower_bridge, upper_bridge)

# %%
df.loc[df['Temp3pm']>=40.45,'Temp3pm']=40.45
df.loc[df['Temp3pm']<=2.45,'Temp3pm']=2.45

# %%
for feature in continuous_feature:
    data=df.copy()
    sns.boxplot(data[feature])
    plt.title(feature)
    plt.figure(figsize=(15,15))

# %%
```

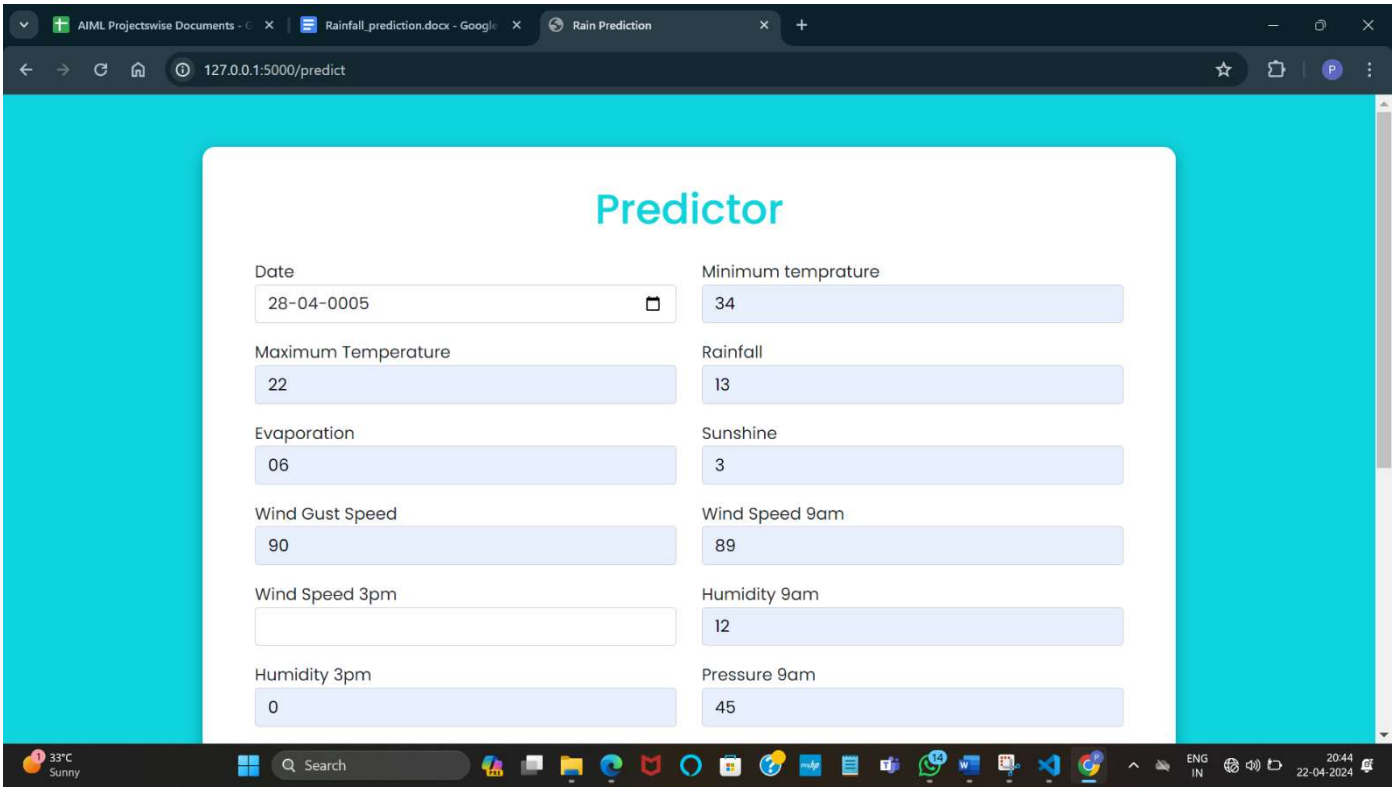
```
def qq_plots(df, variable):
    plt.figure(figsize=(15,6))
    plt.subplot(1, 2, 1)
    df[variable].hist()
    plt.subplot(1, 2, 2)
    stats.probplot(df[variable], dist="norm", plot=plt)
    plt.show()

# %%
for feature in continuous_feature:
    print(feature)
    plt.figure(figsize=(15,6))
    plt.subplot(1, 2, 1)
    df[feature].hist()
    plt.subplot(1, 2, 2)
    stats.probplot(df[feature], dist="norm", plot=plt)
    plt.show()

# %%
df.to_csv("preprocessed_3.csv", index=False)

# %%
X = df.drop(["RainTomorrow", "Date"], axis=1)
Y = df["RainTomorrow"]

# %%
X_train, X_test, y_train, y_test = train_test_split(X,Y, test_size =0.2, stratify = Y, random_state =
0)

# %%
y_train
```

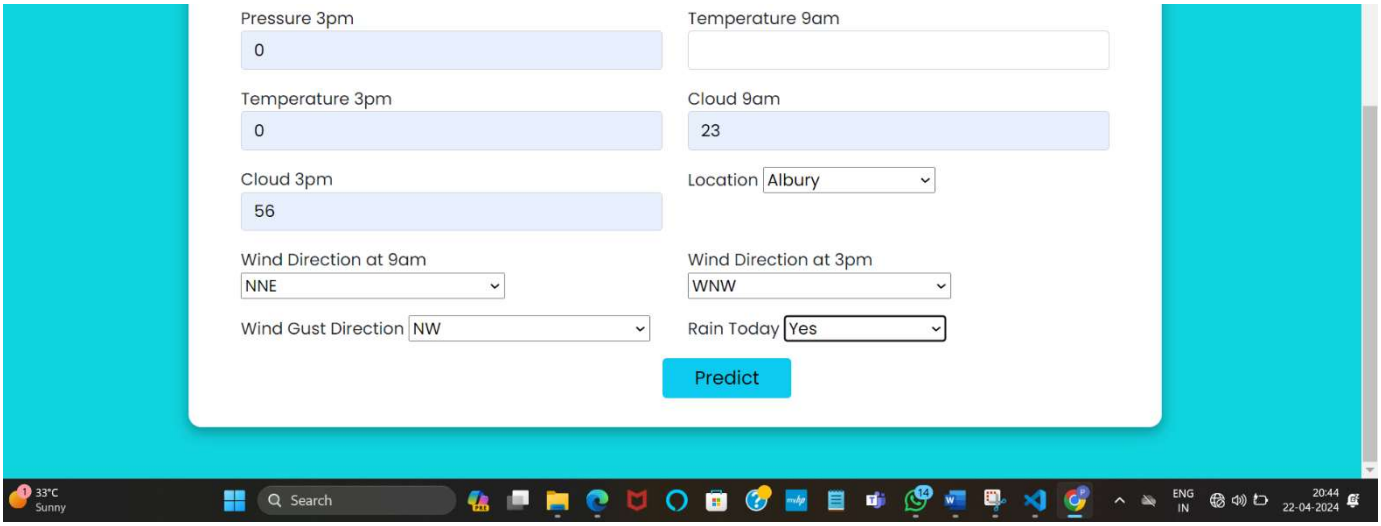
```
# %%
sm=SMOTE(random_state=0)
X_train_res, y_train_res = sm.fit_resample(X_train, y_train)
print("The number of classes before fit {}".format(Counter(y_train)))
print("The number of classes after fit {}".format(Counter(y_train_res)))

# %%
cat = CatBoostClassifier(iterations=2000, eval_metric = "AUC")
cat.fit(X_train_res, y_train_res)

# %%
y_pred = cat.predict(X_test)
print(confusion_matrix(y_test,y_pred))
print(accuracy_score(y_test,y_pred))
print(classification_report(y_test,y_pred))

# %%
metrics.plot_roc_curve(cat, X_test, y_test)
metrics.roc_auc_score(y_test, y_pred, average=None)

# %%
rf=RandomForestClassifier()
rf.fit(X_train_res,y_train_res)

# %%
y_pred1 = rf.predict(X_test)
print(confusion_matrix(y_test,y_pred1))
print(accuracy_score(y_test,y_pred1))
print(classification_report(y_test,y_pred1))

# %%
metrics.plot_roc_curve(rf, X_test, y_test)
```

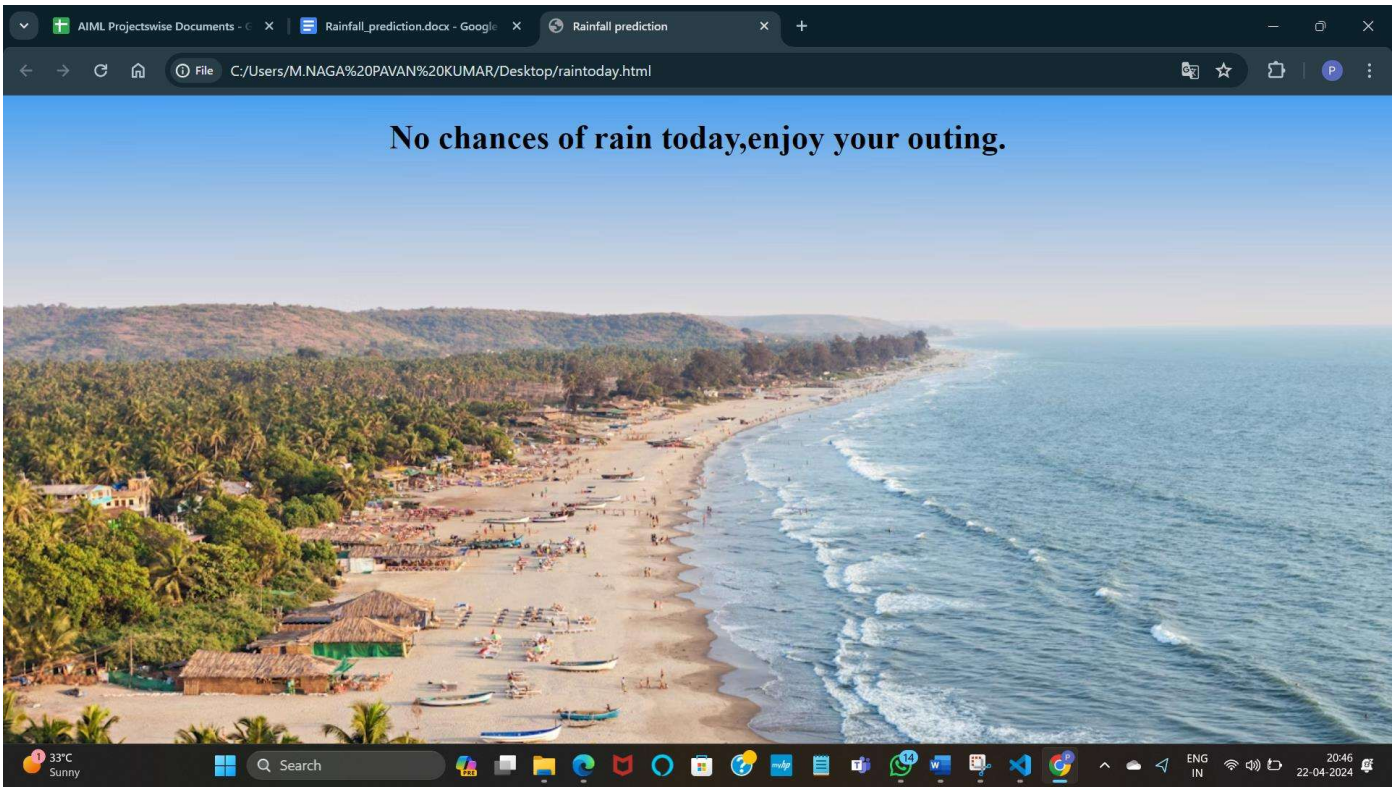
```
metrics.roc_auc_score(y_test, y_pred1, average=None)

# %%
logreg = LogisticRegression()
logreg.fit(X_train_res, y_train_res)

# %%
y_pred2 = logreg.predict(X_test)
print(confusion_matrix(y_test,y_pred2))
print(accuracy_score(y_test,y_pred2))
print(classification_report(y_test,y_pred2))

# %%
metrics.plot_roc_curve(logreg, X_test, y_test)
metrics.roc_auc_score(y_test, y_pred2, average=None)

# %%
knn = KNeighborsClassifier(n_neighbors=3)
knn.fit(X_train_res, y_train_res)

# %%
y_pred4 = knn.predict(X_test)
print(confusion_matrix(y_test,y_pred4))
print(accuracy_score(y_test,y_pred4))
print(classification_report(y_test,y_pred4))

# %%
metrics.plot_roc_curve(knn, X_test, y_test)
metrics.roc_auc_score(y_test, y_pred4, average=None)

# %%
joblib.dump(cat, "cat.pkl")
```

SCREENSHOTS






Pressure 3pm
0
Temperature 9am
Temperature 3pm
0
Cloud 9am
23
Cloud 3pm
56
Location Albury
Wind Direction at 9am
NNE
Wind Direction at 3pm
WNW
Wind Gust Direction NW
Rain Today Yes
Predict
33°C
Sunny
Search
ENG
IN
20:44
22-04-2024


AIML Projectswise Documents -
Rainfall_prediction.docx - Google
Rainfall prediction
File
C:/Users/M.NAGA%20PAVAN%20KUMAR/Desktop/raintoday.html
No chances of rain today,enjoy your outing.
33°C
Sunny
Search
ENG
IN
20:46
22-04-2024