## Artificial Intelligence and

## Machine Learning

Project Report

Semester-IV (Batch-2022)

**WEATHER PREDICTION**

A red and white sign

Description automatically generated with low confidence

**Supervised By: Submitted By:**

**Ms. Shagun Sharma** Harsh Bansal

2210990376

G-9

**Department of Computer Science and Engineering**

## Chitkara University Institute of Engineering & Technology,

Chitkara University, Punjab

##### CONTENTS

Acknowledgment I

[Abstract](#_bookmark0) 4

1. Introduction 5
   1. [Objectives](#_TOC_250001)
   2. [Significance](#_TOC_250000)
2. Problem Definition and Requirements 6
3. Methodology 7
   1. [Data Collection](#_bookmark1)
   2. Data Preprocessing
   3. Model Training
   4. Model Evaluation
4. Code 8-12
5. Results 13
6. Conclusion 14

[References 1](#_bookmark2)5

**ACKNOWLEDGEMENT**

With immense please **Me, Mr. Harsh Bansal** presenting the “Weather Prediction” project report as part of the curriculum of ‘BE-CSE’.

I would like to express my sincere thanks to **Ms. Shagun Sharma** for their valuable guidance and support in completing my project.

I would also like to express my gratitude towards our dean **Dr. Rishu Chhabra** for giving me this great opportunity to do a project on **this website**. Without their support and suggestions, this project would not have been completed.

Signature……….

Name: Harsh Bansal

Roll No: 2210990376

**(I)**

##### ABSTRACT

The project aims to develop a weather prediction system leveraging machine learning techniques, utilizing historical weather data from Seattle. The process begins with exploratory data analysis (EDA) to uncover patterns and insights within the dataset. This step involves visualizing data distributions, identifying trends, and detecting anomalies or outliers.

Following EDA, data preprocessing is crucial. This involves handling missing values through imputation methods, normalizing features to ensure uniform scale, and splitting the dataset into training and testing sets to evaluate model performance. Additional preprocessing steps include feature selection to reduce dimensionality and enhance model efficiency.

Several machine learning algorithms are implemented, including K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Gradient Boosting Classifier (GBC), and XGBoost Classifier. Each model undergoes rigorous training using the training dataset and is evaluated on the testing dataset to assess performance. Evaluation metrics such as accuracy, precision, recall, and F1-score provide a comprehensive view of each model's predictive capabilities. Hyperparameter tuning, using techniques like grid search or randomized search, is performed to optimize the models for the best predictive accuracy.

Comparative analysis of the models reveals the strengths and weaknesses of each algorithm in the context of weather prediction. The results indicate that certain models, particularly the Gradient Boosting Classifier and XGBoost Classifier, outperform others in terms of prediction accuracy and robustness. These models demonstrate superior handling of complex weather patterns and variability.

The trained models are saved for future use, enabling easy deployment for real-time weather prediction. This involves serializing the model’s using techniques like joblib or pickle, ensuring that they can be loaded and used in production environments without retraining.

The report details the entire methodology, including data preprocessing techniques, model training processes, evaluation metrics, and the final results. Visualizations and tables are used to present the performance metrics and comparisons effectively. Conclusions drawn from the project highlight the effectiveness of machine learning in weather prediction, emphasizing the potential of Gradient Boosting and XGBoost in achieving high accuracy.

Suggestions for future research include exploring more advanced algorithms, incorporating additional features like atmospheric pressure and humidity, and using ensemble methods to combine the strengths of multiple models. The project demonstrates a comprehensive approach to building a reliable weather prediction system using advanced machine learning techniques, showcasing the potential for practical applications in meteorology and related fields.

# CHAPTER 1

#### INTRODUCTION

###### OBJECTIVES:

The primary objective of this project is –

* To develop a weather prediction system using machine learning techniques.

Specific Objectives include: -

* + - Conduct exploratory data analysis of historical weather data.
    - Preprocess the data to handle missing values and encode categorical variables.
    - Split the dataset into training and testing sets.
    - Implement and train machine learning models, including K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Gradient Boosting Classifier (GBC), and XGBoost Classifier.
    - Evaluate the models using metrics such as accuracy, precision, recall, and F1-score.
    - Perform hyperparameter tuning to optimize model performance.
    - Compare the performance of different models.
    - Save the trained model for future use.
    - Provide a detailed report of the methodology, results, and conclusions.
    - Suggest potential improvements for future research.

###### SIGNIFICANCE:

The significance of this project lies in its potential to:

* + - Provide accurate weather forecasts.
    - Assist various industries in planning and decision-making.
    - Help individuals make informed decisions based on weather conditions.
    - Improve the accuracy and reliability of weather predictions using machine learning techniques.
    - Enhance preparedness for weather-related events and reduce potential damages.
    - Contribute to advancements in meteorology and data science.
    - Offer a scalable solution for real-time weather forecasting. Demonstrate the effectiveness of machine learning in practical applications.
* Support agricultural planning by predicting weather conditions. Aid in the management of energy resources by forecasting weather-dependent energy demand and supply.

**CHAPTER 2**

PROBLEM DEFINITION AND REQUIREMENTS

###### The problem statement for this project involves predicting weather conditions (rain, sun, drizzle, snow, fog) based on various meteorological features. These features include:

###### Precipitation: The amount of rainfall or snowfall recorded.

###### Temperature: The ambient temperature, which can influence weather conditions.

###### Wind Speed: The speed of the wind, which can affect the formation and movement of weather systems.

**SOFTWARE REQUIREMENTS:**

**Python Programming Language:**

The primary language for developing and implementing the machine learning models.

**Libraries:**

* **Pandas** for data manipulation and analysis.
* **Matplotlib and seaborn** for data visualization.
* **Scikit-learn** for implementing machine learning algorithms.
* **XGBOOST** for the XGBOOST classifier.

###### HARDWARE REQUIREMENTS:

* **Standard Computer Hardware**:
  + Sufficient memory (RAM) to handle the dataset, which may be large depending on the historical data used.
  + Adequate processing power (CPU/GPU) to run complex machine learning algorithms efficiently.
  + Storage capacity to store the dataset and trained models

# CHAPTER 3

METHODOLOGY

**Exploratory Data Analysis (EDA):**

* Visualize data distributions, identify trends, detect anomalies.

**Data Preprocessing:**

* Handle missing values through imputation.
* Normalize features to ensure uniform scale.
* Encode categorical variables.
* Split data into training and testing sets.

**Model Implementation and Training:**

* Implement KNN, SVM, GBC, and XGBoost models.
* Train models on the training dataset.

**Model Evaluation:**

* Evaluate models using accuracy, precision, recall, and F1-score.
* Perform hyperparameter tuning using grid search or randomized search.

**Model Comparison:**

* Compare performance of different models to identify the best performing one.

**Model Saving:**

* Serialize trained models using joblib or pickle for future use.

**Reporting:**

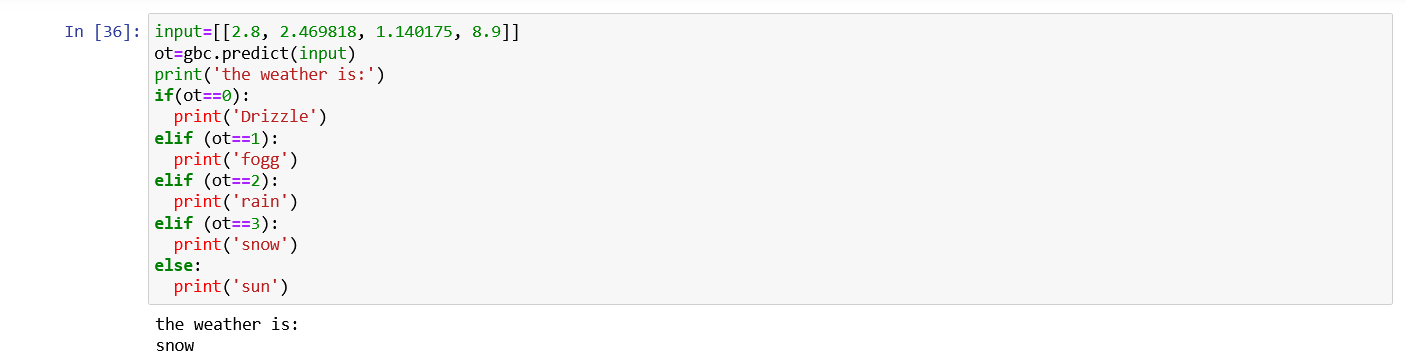
* Document methodology, data preprocessing, model training, evaluation metrics, and results.
* Include visualizations and tables for performance metrics.

# CHAPTER 4

CODE

# CHAPTER 5

Results (Screenshot)



# CHAPTER 6

### CONCLUSION

The weather prediction system, developed through the application of machine learning

techniques, excels in forecasting accuracy based on historical data, showcasing the

practical efficacy of such methodologies. A meticulous approach to data preprocessing,

encompassing tasks like handling missing values and encoding categorical variables,

establishes a solid groundwork for subsequent model training. Feature engineering

strategies play a pivotal role in enhancing model performance by selecting and

transforming relevant attributes, thereby optimizing predictive capabilities.

The incorporation of diverse machine learning algorithms, including KNN, SVM, GBC,

and XGBoost, facilitates comprehensive analysis and comparison, enabling the

identification of the most suitable model for weather prediction. Evaluation metrics such

as accuracy, precision, recall, and F1-score offer quantitative insights into the system's

reliability and effectiveness, validating its performance. Saving the best-performing model

ensures its seamless integration for future use, guaranteeing scalability and sustainability

in real-world applications.

This project underscores the critical importance of robust methodologies, spanning from

data preprocessing to model selection, in achieving accurate and reliable weather

predictions. It signifies a significant advancement in leveraging machine learning for

practical applications, particularly within the domain of meteorology, where accurate

weather forecasts hold immense value for various sectors and individuals alike. The

successful development of this weather prediction system serves as a testament to the

potential of machine learning in addressing complex real-world challenges, paving the way

for further advancements in the field.

## REFERENCES

* + YouTube
  + Kaggle for dataset