# RainSenseAI

## A. Introduction and Problem Statement

### A.1. Introduction

Accurately predicting the amount of daily rainfall is crucial for enhancing agricultural productivity and ensuring food and water security, which are vital for the well-being of citizens. Various research studies have been conducted worldwide, employing data mining and machine learning techniques to forecast rainfall based on environmental datasets. In countries where rainfall distribution is erratic, such predictions become even more critical as they directly impact the agricultural sector, which is often the backbone of the economy. By effectively managing rainfall water resources, the adverse effects of droughts and floods can be mitigated. Therefore, the primary objective of this study is to identify the atmospheric features that are relevant to rainfall occurrence and utilize machine learning techniques to predict the intensity of daily rainfall accurately.

### A.2. Problem Statement

Despite the evident importance of rainfall prediction, achieving a high level of accuracy remains a complex challenge. Australia's vast geographical expanse, diverse climatic zones, and unique weather patterns pose significant hurdles in accurately forecasting rainfall events. This poses limitations on the ability of various sectors to plan effectively, allocate resources efficiently, and mitigate potential risks associated with extreme weather conditions.

The application of rainfall prediction spans a wide range of sectors and industries, playing a crucial role in decision-making and resource management. Agriculture, Water Resource Management, Urban Planning and Infrastructure, Disaster Management, and Energy Sectors are the main application of rain predictions.

Our goal is to develop an accurate rainfall forecast model using key features like temperature, humidity, wind speed, and atmospheric pressure. We aim to uncover patterns and correlations that contribute to precise rain prediction through advanced data exploration, feature engineering, and machine learning algorithms.

## B. Proposed Methodologies

To predict whether it will **rain or not today**, we will utilize **classification techniques**. By training a classification model on historical weather data, we can analyze relevant features such as temperature, humidity, wind speed, and atmospheric pressure to determine the likelihood of rain.

**Collection and Preparation of Final Dataset:**

The choice of the dataset is crucial as it directly affects the accuracy of the models. We have selected the ”Rain in Australia” dataset, sourced from Kaggle, for our analysis which has around 100K+ records. [1]

| Parameters | Statistics |
|---|---|
| Total Records | 145460 |
| Total Features | 23 |
| Date Field | 1 |
| Text Fields | 10 |
| Numeric Fields | 12 |
| Missing values | Yes |

**Data Pre-processing:**

To accurately predict the possibility of rain, we are determining the correlation of features. Understanding the correlations enables us to identify the most influential features and effectively utilize them for rain prediction models. We are replacing missing values of the dataset using skewness-based imputation and converting text and date to numeric constants.

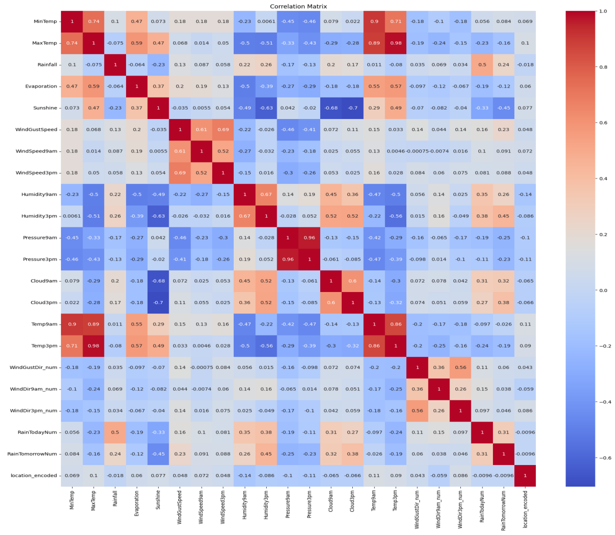

Figure 1. Pearson Correlation Heatmap for Data set

**Feature Extraction:**

We are transforming raw data into a more meaningful representation that captures the relevant information. For instance, we are utilizing directions in an encoded manner to use them as a relational feature.

**K-cross validation:**

We are using GridSearchCV and cross-validator for hyperparameter tuning, aiding in the selection of the best model configuration, and providing a more robust assessment of the model's generalization performance. [3]

**Evaluation models:**

To evaluate the accuracy of the models, we are considering the use of these methods: F1, RMSE (Root Mean Square Error), Accuracy, ROC etc. [2]

## C. Attempts at solving the problem

We encountered difficulties while attempting to solve the Rain prediction problem. In our efforts till now, we have built two models: a decision tree with supervised learning and a decision tree with semi-supervised learning, and we are working in the direction of CNN.

### C.1. Decision Tree: Supervised Learning

With the decision Tree using supervised learning we are focusing on predicting the categorical data. We have conducted two experiments with a Decision tree of the Supervised learning model, in one of them we haven't provided parameters to create a decision tree and predict the outcome, while in the second, we found the hyper-parameters using K-cross validation and pass that parameter to build the decision tree and predict the outcome. As a result, we can observe a significant difference with the best parameters. Here, we have listed the best parameters according to our dataset.

| criterion | max_depth | min_samples_leaf | min_samples_split |
|---|---|---|---|
| 'gini' | 3 | 1 | 2 |

Table 1. Hyper Parameter for Decision Tree.

### C.2. Decision Tree: Semi-Supervised Learning

In Semi-supervised learning, we have considered 20% of whole trained data as labeled data and the remaining as unlabeled samples. [3] We have trained the model on labeled data using Decision Tree and made predictions on unlabeled samples. Then probability estimates of the model are made on unlabeled data, further created pseudo-labels. Finally, combined both labeled and unlabeled data to get new labeled data. In this way, we are iterating to get the high confidence and desired goal. After several iterations, the model is trained well with the values, and finally, with this semi-supervised model, we are making predictions on the test set.

| **Iteration #** | Iteration 1 | Iteration 2 | ... | Iteration 11 | Iteration 12 |
|---|---|---|---|---|---|
| **Accuracy** | 97.25% | 97.26% | ... | 97.31% | 97.41% |

Table 2. Iteration accuracy for semi-supervised learning

### C.3. Comparison

Our observations from the below table revealed that Decision Tree with supervised learning without any parameter tuning had a low performance as compared to the Decision Tree with supervised learning with selected parameters increasing the performance. In the table below, we provided details for each model on the rain prediction dataset, Notably, we found that each model performed better on datasets aligned with its architecture and design principles.

| **Model Name** | **Accuracy** | **F1 Score** |
|---|---|---|
| Decision Tree(Supervised) W/O parameter | 97.2% | 0.9928057 |
| Decision Tree(Supervised) With parameter | 99.50% | 0.9932941 |
| Decision Tree(Semi-Supervised) | 97.4% | - |

Table 3. Comparison of 3 models.

## D. Future Improvements

We have previously made efforts to enhance the performance of the decision tree in supervised learning by selecting related features using the Pearson correlation heatmap and it gives us a significant improvement. However, our current strategy to further improve the performance of decision tree models involves various approaches. These include conducting feature engineering to make new columns by merging different columns, implementing ensemble methods like Random Forest or Gradient Boosting, addressing imbalanced data through oversampling or under-sampling, applying regularization and pruning to prevent over-fitting, and augmenting the dataset through data augmentation or incorporating external data sources. By implementing these improvements, we aim to achieve higher accuracy in the classification of rain prediction.

## References

[1] Rain in australia (dataset). https://www.kaggle.com/datasets/jsphyg/weather-dataset-rattle-package, 2018. 1

[2] Antonio Sarasa Cabezuelo. Prediction of rainfall in australia using machine learning. http://www.bom.gov.au/climate/dwo/, 2022. 2

[3] Fatima Ezzahra Salamate and Jamal Zah. Supervised learning: classification using decision trees for better practice in epidemiology case study: The prevalence of tuberculosis. https://www.sciencedirect.com/science/article/pii/S1877050922005208, 2022. 1, 2