Annual Rainfall Classification Using Machine Learning Techniques

Raksha Kodnad R
Department of MCA
Dayananda Sagar College of Engineering
Bengaluru,India
rakshakodnadr@gmail.com

Chandrika M
Department of MCA
Dayananda Sagar College of Engineering
Bengaluru,India
https://orcid.org/0000-0003-1263-1814

Pavithra B
Department of MCA
Dayananda Sagar College of Engineering
Bengaluru,India
pavithrab-mcavtu@dayanandasagar.edu

2024 1st International Conference on Communications and Computer Science (InCCCS) | 979-8-3503-5885-8/24/$31.00 ©2024 IEEE | DOI: 10.1109/INCCCS60947.2024.10593366

***Abstract*—This research work investigates the classification of 1901-2017 Indian subdivision rainfall data utilizing the Random Forest, Logistic Regression, and Decision Trees machine learning algorithms. The goal is to assess and compare the algorithms' abilities to forecast and categorize rainfall patterns. The dataset includes rainfall information for each month for a complete year. We assess each algorithm's accuracy, efficiency, and interpretability. The findings will provide important information on their relevance for rainfall forecast in India. The study incorporates data visualization to aid understanding of classification results and their consequences for regional weather and agriculture. This research helps to construct more accurate rainfall forecast models, which will aid agriculture, the management of water resources, and disaster mitigation in India**

Keywords—Rainfall Classification, Machine Learning Algorithms, Indian Subdivisions, Prediction Accuracy, Data Visualization.

I. INTRODUCTION

Within artificial intelligence, machine learning is a cutting-edge topic that has increased dramatically in prominence in recent years. It has radically changed the way we approach difficult issues by allowing machines to see patterns, learn from data, and gradually improve their performance. Its applications are many, including natural language processing, autonomous automobiles, healthcare, and finance. This work aims to provide a full understanding of machine learning methodologies as well as applications in research on climate. Machine learning techniques, like how individuals learn through experience, allow computers to identify patterns, predict outcomes, and respond to freshly acquired data without the need for specific programming. The field includes a variety of techniques, ranging from basic statistical methods to advanced deep learning models. Understanding these fundamental concepts is critical for researchers, engineers, and data scientists who want to harness machine learning's possibilities for real-world problem solving. The paper delves into the history, evolution, key components, and the three main categories of machine learning: supervised, unsupervised, and reinforcement learning. Additionally, it explores the pivotal role of data, addresses ethical and societal implications, and highlights notable applications that have profoundly impacted diverse industries, showcasing the versatility and potential of machine learning. (Breiman, Classification and Regression Trees. [3]

II. TYPES OF MACHINE LEARNING

Machine learning encompasses a diverse range of techniques and approaches, each tailored to specific problem-solving scenarios. Understanding the classification of machine learning is crucial for researchers and practitioners as it forms the foundation for selecting the most suitable methodology for a given task. In this section, we will discuss how machine learning techniques can be classed as supervised, unsupervised, and reinforcement learning.[10]

The most common and basic type of machine learning is **supervised learning**, which uses algorithms trained on labelled data to link input points to their matching output labels. The main purpose is to produce input-to-output mapping that makes it easier to make predictions or classify newly discovered data, including regression and classification tasks. The support vector machine, deep neural networks, decision trees, and linear regression algorithms are notable examples of algorithms in this category.

Unsupervised learning, uses data that is unlabelled and tasks algorithms with finding patterns, structures, or correlations on their own. Clustering, assembling related data points, and dimensionality reduction methods are examples of common unsupervised jobs. Principal component analysis (PCA), hierarchical clustering, and k-means clustering are prominent techniques in this context.

Reinforcement learning, distinct from the two former types, trains agents to make decisions to maximize cumulative rewards by interacting with the environment. This involves receiving feedback in the form of rewards or punishments and finds applications in game playing, autonomous robotics, and optimization problems. Key elements include states, actions, rewards, and a policy governing the agent's behaviour, with popular algorithms being Q-learning, deep Q-networks (DQN), and policy gradients.

III. TYPES OF MACHINE LEARNING ALGORITHMS

A vast variety of algorithms are used in machine learning, each one intended to address a particular kind of issue. To choose the best approach to take for their particular applications, researchers and practitioners must have a thorough understanding of the different machine learning algorithms. In this section, we will delve into some of the prominent machine learning algorithms within each category of supervised, unsupervised, and reinforcement learning.

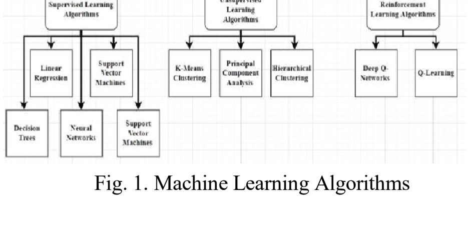

Types of Machine Learning Algorithms
Supervised Learning Algorithms
Unsupervised Learning Algorithms
Reinforcement Learning Algorithms
Linear Regression
Support Vector Machines
K-Means Clustering
Principal Component Analysis
Hierarchical Clustering
Deep Q-Networks
Q-Learning
Decision Trees
Neural Networks
Support Vector Machines

Fig. 1. Machine Learning Algorithms

979-8-3503-5885-8/24/$31.00 ©2024 IEEE

IV. Introduction to the Rainfall Classification

In the realm of environmental science and agriculture, the prediction and classification of rainfall patterns hold critical significance for resource management, crop planning, and disaster preparedness. The availability of comprehensive rainfall data is pivotal for taking informed decisions in these domains. This project centres on a dataset encompassing rainfall measurements for Indian subdivisions during the years 1901-2017. The primary aim is to leverage machine learning techniques, specifically the three techniques: Random Forest, Decision Trees, and Logistic Regression, to classify and predict the rainfall patterns within these subdivisions. Through this comparison, we aim to establish which of the aforementioned algorithms is best suited to this specific task. [2]

The dataset consists of monthly rainfall data, annual and seasonal aggregates. By utilizing these machine learning algorithms, we intend to offer insightful explanations of the efficiency, accuracy, and interpretability of each method when employed to categorise patterns of rainfall. This analysis is pivotal, as it directly impacts critical sectors, such as agriculture, water resource management, and disaster risk reduction in India.

Moreover, in this project, we will not only focus on model performance but also emphasize the importance of data visualization. Effective visualization techniques will aid in comprehending the classification results, in turn making them more accessible to various stakeholders and decision-makers. This project's outcome can facilitate in taking more informed decisions in the management of agricultural resources and disaster response strategies, contributing to the sustainable development of Indian regions and subcontinents.[8]

V. Tools and Technologies Used

To conduct a comprehensive comparison study of the three mentioned machine learning algorithms for the classification of rainfall patterns in Indian subdivisions for the years 1901- 2017, a variety of methods and technologies are used. The following are the key tools and technologies utilized in this study

A. *Python and Jupyter Notebook*

Python serves as the primary programming language for the study, chosen for its versatility, extensive libraries, and frameworks dedicated to machine learning, data analysis, and performing visualization. Jupyter Notebooks play a crucial role in the project, serving as the platform for code development, data exploration, and visualization. They provide an interactive and collaborative environment, facilitating the iterative process of model development and evaluation.

B. *Machine Learning Libraries*

Scikit-Learn: Scikit-Learn provides a wide array of machine learning library functions, making it a central component for implementing Decision Tree model, Random Forest model, and Logistic Regression model.

Pandas: Pandas is used for preprocessing and data modification, enabling effective handling of data and cleansing.

NumPy: NumPy supports numerical operations and array manipulation, which is essential for mathematical computations involved in machine learning.

C. *Data Visualization*

Matplotlib: Matplotlib is popularly used Python library for creating appealing and interactive visualizations to help interpret and communicate results effectively.

Seaborn: Seaborn is used for enhancing the aesthetics of Matplotlib visualizations and simplifying complex data visualization tasks.

D. *Machine Learning Evaluation Metrics*

Accuracy measures the overall correctness of predictions made by machine learning models. Precision evaluates the accuracy of positive predictions, focusing on how many of them are true. Recall assesses the ability of the model to capture all relevant instances within a dataset. F1-score combines precision and recall into a single metric, providing a balanced measure of a model's performance. ROC-AUC (Receiver Operating Characteristic - Area Underneath the Curve) quantifies a model's ability to distinguish between classes, considering the trade-off between rates of false positives and true positives. A confusion matrix is a table that sums up the performance of a classification system by comparing predicted and actual class labels.[3]

E. *Classification Algorithms Employeed*

Decision Trees is a versatile and interpretable algorithm, that find application in classification by partitioning data based on features and constructing a tree-like structure of decision rules. They prove effective in revealing complex decision boundaries and feature importance, making them suitable for a diverse range of applications.

Random Forest, an ensemble learning method, enhances accuracy and addresses overfitting by constructing a several types of Decision Trees and combining their predictions. This approach proves highly effective for implementing both classification and regression tasks, demonstrating robustness and competence in handling large and complex datasets

Logistic Regression, a fundamental classification algorithm, is utilized for binary and multi-class problems. It creates models for the probability of an event based on input features, offering interpretability, computational efficiency, and suitability for scenarios where understanding the influence of selected features on the outcome is crucial. [1]

By leveraging these tools and technologies, the project aims to explore, preprocess, model, and visualize the rainfall data effectively, allowing for a rigorous comparison of Decision Trees algorithm, Random Forest algorithm, Logistic Regression algorithm for the categorization of Indian subdivision rainfall patterns. This comprehensive approach ensures the robustness and reliability of the study's findings and advances comprehension of machine learning algorithms' suitability for environmental data classification.

VI. Implementation

To Implementing a project to classify rainfall patterns in Indian subdivisions for the years 1901-2017 using Logistic Regression, Decision Trees and Random Forest involves several steps. Below is a high-level implementation logic.

A. Data Collection

Gather the rainfall dataset for Indian subdivisions in 1901- 2017(from Data.gov.in), ensuring it contains features like monthly rainfall values, subdivision information, and the target variable (rainfall category).

B. Data Preprocessing

- Load the dataset into Pandas DataFrame.
- Check for missing values and handle them, if necessary.
- Explore the data with summary statistics and visualizations.
- Encode categorical variables if needed (e.g., subdivision names).
- Split the data into training and testing sets. [4]

C. Model Implementation

Features 'JAN' to 'DEC','Annual','Subdivision' The feature 'Annual' will be categorized into "low," "moderate," and "high" based on quantiles or other criteria. The research aims to categorize the 'Annual' precipitation of rain built on the monthly rainfall data ('JAN' to 'DEC') and the 'Annual' along with the 'Subdivision' feature. These attributes reflect input variables that the machine learning algorithms use to make forecasts about variable, 'Annual.'

D. Impact on the Results

Keeping in mind the limitations and advantages of all the three algorithms as mentioned below, the study was carried out.

Decision Tree: Interpretable but prone to overfitting on complex data.

Random Forest: Reduces overfitting and enhances accuracy by aggregating prediction values from multiple Decision Trees.

Logistic Regression: Assumes a linear relationship, providing interpretable results and probabilistic predictions.

In summary, Decision Trees offer transparency, Random Forest improves accuracy, and Logistic Regression provides interpretable, probabilistic outputs. The choice depends on the data characteristics and modelling goals. [7]

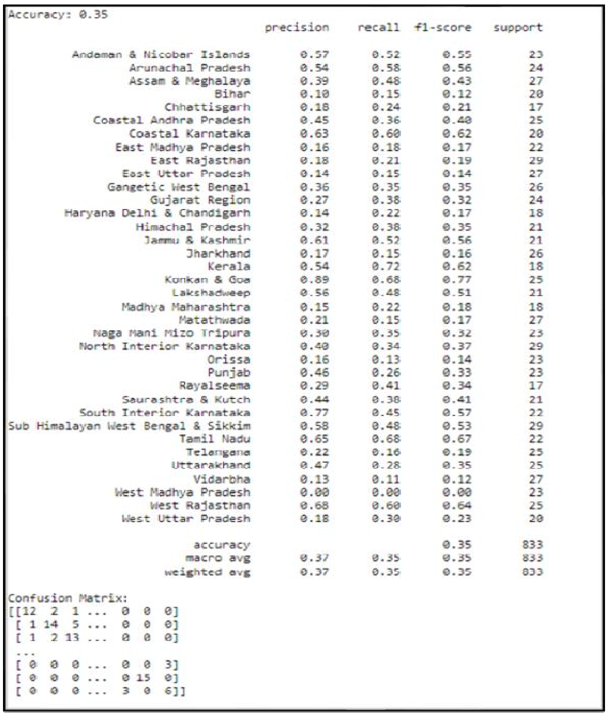
VII. Result and Discussion

A. Model Evaluation

Utilize various evaluation metrics, such as accuracy, confusion matrix, precision, f1-score, and classification report to assess the model performance.

B. Result

The following figures describe the results found by the model.



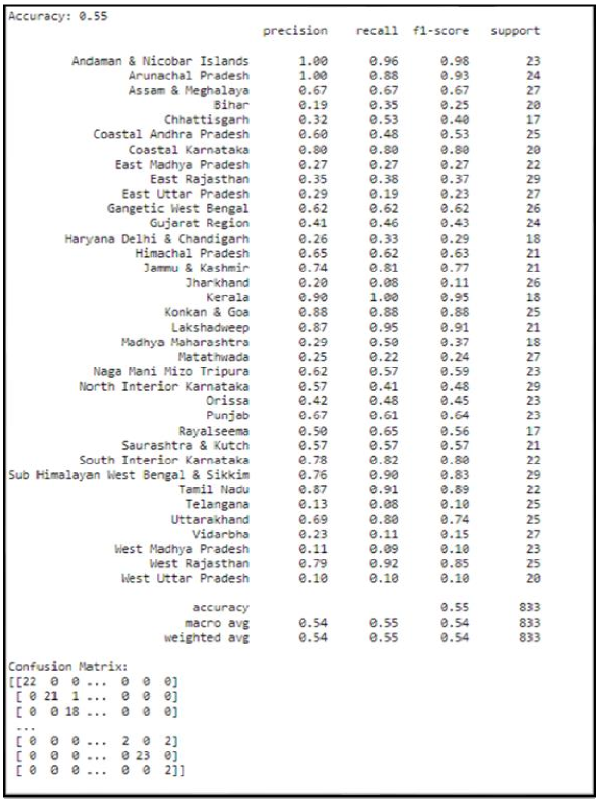
```
Accuracy: 0.35
                                    precision    recall  f1-score   support

         Andaman & Nicobar Islands       0.57      0.52      0.55        23
                 Arunachal Pradesh       0.54      0.58      0.56        24
                 Assam & Meghalaya       0.39      0.48      0.43        27
                             Bihar       0.10      0.15      0.12        20
                      Chhattisgarh       0.18      0.24      0.21        17
            Coastal Andhra Pradesh       0.45      0.36      0.40        25
                 Coastal Karnataka       0.63      0.60      0.62        20
               East Madhya Pradesh       0.16      0.18      0.17        22
                    East Rajasthan       0.18      0.21      0.19        29
                East Uttar Pradesh       0.14      0.15      0.14        27
               Gangetic West Bengal       0.36      0.35      0.35        26
                    Gujarat Region       0.27      0.38      0.32        24
        Haryana Delhi & Chandigarh       0.14      0.22      0.17        18
                  Himachal Pradesh       0.32      0.38      0.35        21
                   Jammu & Kashmir       0.61      0.52      0.56        21
                         Jharkhand       0.17      0.15      0.16        26
                            Kerala       0.54      0.72      0.62        18
                      Konkan & Goa       0.89      0.68      0.77        25
                       Lakshadweep       0.56      0.48      0.51        21
                Madhya Maharashtra       0.15      0.22      0.18        18
                        Matathwada       0.21      0.15      0.17        27
            Naga Mani Mizo Tripura       0.30      0.35      0.32        23
          North Interior Karnataka       0.40      0.34      0.37        29
                            Orissa       0.16      0.13      0.14        23
                            Punjab       0.46      0.26      0.33        23
                        Rayalseema       0.29      0.41      0.34        17
                 Saurashtra & Kutch       0.44      0.38      0.41        21
          South Interior Karnataka       0.77      0.45      0.57        22
Sub Himalayan West Bengal & Sikkim       0.58      0.48      0.53        29
                        Tamil Nadu       0.65      0.68      0.67        22
                         Telangana       0.22      0.16      0.19        25
                       Uttarakhand       0.47      0.28      0.35        25
                          Vidarbha       0.13      0.11      0.12        27
               West Madhya Pradesh       0.00      0.00      0.00        23
                    West Rajasthan       0.68      0.60      0.64        25
                West Uttar Pradesh       0.18      0.30      0.23        20

                          accuracy                           0.35       833
                         macro avg       0.37      0.35      0.35       833
                      weighted avg       0.37      0.35      0.35       833

Confusion Matrix:
[[12  2  1 ...  0  0  0]
 [ 1 14  5 ...  0  0  0]
 [ 1  2 13 ...  0  0  0]
 ...
 [ 0  0  0 ...  0  0  3]
 [ 0  0  0 ...  0 15  0]
 [ 0  0  0 ...  3  0  6]]
```

Fig. 2. Decision Tree



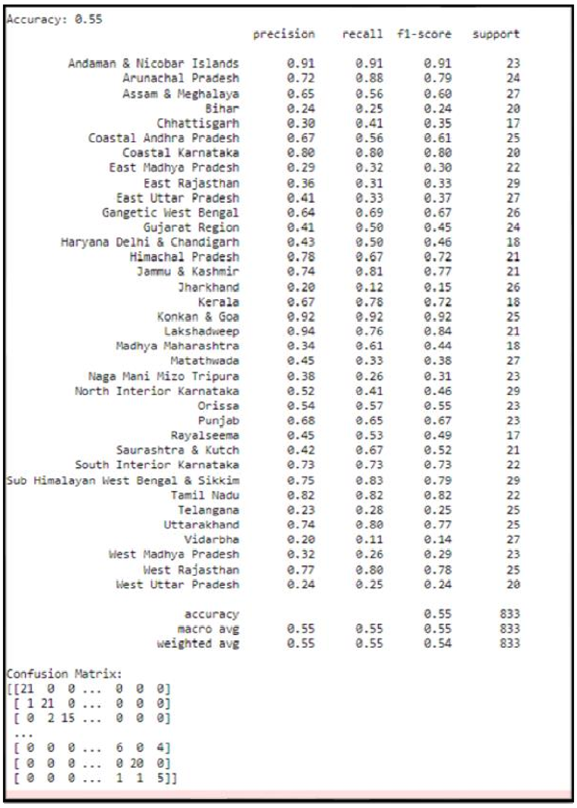
```
Accuracy: 0.55
                                    precision    recall  f1-score   support

         Andaman & Nicobar Islands       1.00      0.96      0.98        23
                 Arunachal Pradesh       1.00      0.88      0.93        24
                 Assam & Meghalaya       0.67      0.67      0.67        27
                             Bihar       0.19      0.35      0.25        20
                      Chhattisgarh       0.32      0.53      0.40        17
            Coastal Andhra Pradesh       0.60      0.48      0.53        25
                 Coastal Karnataka       0.80      0.80      0.80        20
               East Madhya Pradesh       0.27      0.27      0.27        22
                    East Rajasthan       0.35      0.38      0.37        29
                East Uttar Pradesh       0.29      0.19      0.23        27
              Gangetic West Bengal       0.62      0.62      0.62        26
                    Gujarat Region       0.41      0.46      0.43        24
        Haryana Delhi & Chandigarh       0.26      0.33      0.29        18
                  Himachal Pradesh       0.65      0.62      0.63        21
                   Jammu & Kashmir       0.74      0.81      0.77        21
                         Jharkhand       0.20      0.08      0.11        26
                            Kerala       0.90      1.00      0.95        18
                      Konkan & Goa       0.88      0.88      0.88        25
                       Lakshadweep       0.87      0.95      0.91        21
                Madhya Maharashtra       0.29      0.50      0.37        18
                        Matathwada       0.25      0.22      0.24        27
            Naga Mani Mizo Tripura       0.62      0.57      0.59        23
          North Interior Karnataka       0.57      0.41      0.48        29
                            Orissa       0.42      0.48      0.45        23
                            Punjab       0.67      0.61      0.64        23
                        Rayalseema       0.50      0.65      0.56        17
                 Saurashtra & Kutch       0.57      0.57      0.57        21
          South Interior Karnataka       0.78      0.82      0.80        22
Sub Himalayan West Bengal & Sikkim       0.76      0.90      0.83        29
                        Tamil Nadu       0.87      0.91      0.89        22
                         Telangana       0.13      0.08      0.10        25
                       Uttarakhand       0.69      0.80      0.74        25
                          Vidarbha       0.23      0.11      0.15        27
               West Madhya Pradesh       0.11      0.09      0.10        23
                    West Rajasthan       0.79      0.92      0.85        25
                West Uttar Pradesh       0.10      0.10      0.10        20

                          accuracy                           0.55       833
                         macro avg       0.54      0.55      0.54       833
                      weighted avg       0.54      0.55      0.54       833

Confusion Matrix:
[[22  0  0 ...  0  0  0]
 [ 0 21  1 ...  0  0  0]
 [ 0  0 18 ...  0  0  0]
 ...
 [ 0  0  0 ...  2  0  2]
 [ 0  0  0 ...  0 23  0]
 [ 0  0  0 ...  0  0  2]]
```

Fig. 3. Random Forest



```
Accuracy: 0.55
                                  precision    recall  f1-score   support

      Andaman & Nicobar Islands       0.91      0.91      0.91        23
              Arunachal Pradesh       0.72      0.88      0.79        24
              Assam & Meghalaya       0.65      0.56      0.60        27
                          Bihar       0.24      0.25      0.24        20
                   Chhattisgarh       0.30      0.41      0.35        17
         Coastal Andhra Pradesh       0.67      0.56      0.61        25
              Coastal Karnataka       0.80      0.80      0.80        20
             East Madhya Pradesh       0.29      0.32      0.30        22
                 East Rajasthan       0.36      0.31      0.33        29
             East Uttar Pradesh       0.41      0.33      0.37        27
           Gangetic West Bengal       0.64      0.69      0.67        26
                 Gujarat Region       0.41      0.50      0.45        24
     Haryana Delhi & Chandigarh       0.43      0.50      0.46        18
               Himachal Pradesh       0.78      0.67      0.72        21
                Jammu & Kashmir       0.74      0.81      0.77        21
                      Jharkhand       0.20      0.12      0.15        26
                         Kerala       0.67      0.78      0.72        18
                   Konkan & Goa       0.92      0.92      0.92        25
                    Lakshadweep       0.94      0.76      0.84        21
             Madhya Maharashtra       0.34      0.61      0.44        18
                     Matathwada       0.45      0.33      0.38        27
         Naga Mani Mizo Tripura       0.38      0.26      0.31        23
       North Interior Karnataka       0.52      0.41      0.46        29
                         Orissa       0.54      0.57      0.55        23
                         Punjab       0.68      0.65      0.67        23
                     Rayalseema       0.45      0.53      0.49        17
             Saurashtra & Kutch       0.42      0.67      0.52        21
       South Interior Karnataka       0.73      0.73      0.73        22
Sub Himalayan West Bengal & Sikkim       0.75      0.83      0.79        29
                     Tamil Nadu       0.82      0.82      0.82        22
                      Telangana       0.23      0.28      0.25        25
                    Uttarakhand       0.74      0.80      0.77        25
                       Vidarbha       0.20      0.11      0.14        27
            West Madhya Pradesh       0.32      0.26      0.29        23
                West Rajasthan       0.77      0.80      0.78        25
             West Uttar Pradesh       0.24      0.25      0.24        20

                       accuracy                           0.55       833
                      macro avg       0.55      0.55      0.55       833
                   weighted avg       0.55      0.55      0.54       833

Confusion Matrix:
[[21  0  0 ...  0  0  0]
 [ 1 21  0 ...  0  0  0]
 [ 0  2 15 ...  0  0  0]
 ...
 [ 0  0  0 ...  6  0  4]
 [ 0  0  0 ...  0 20  0]
 [ 0  0  0 ...  1  1  5]]
```

Fig. 4. Logistic Regression

C. Visualization

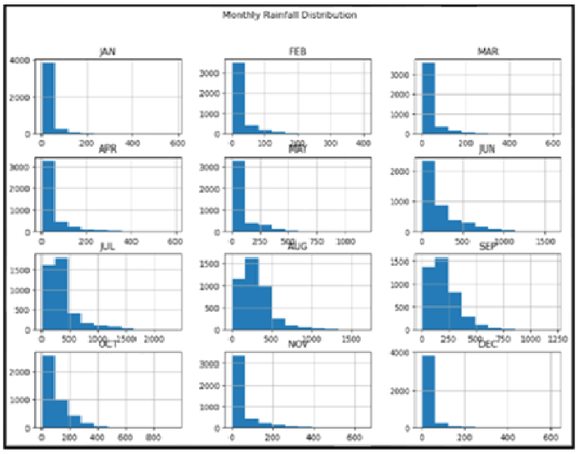


Fig. 5. Monthly Rainfall Distribution

The provided data visualization code generates histograms to illustrate the distribution of monthly rainfall across the twelve months of the years. Each histogram represents a different month, and the bins parameter is set to 10, This indicates that the monthly rainfall range is divided into ten intervals per month. The histograms are displayed in a single figure, making it easy to compare the rainfall distribution over several months. The resulting visualization provides insights about the central tendency, variability, and identify patterns in rainfall for each month, helping in the comprehension of the data's seasonal patterns and variations. The 'Monthly Rainfall Distribution' title encapsulates the goal of the visualisation, letting viewers to immediately understand both the subject and the setting of the plots.[5]

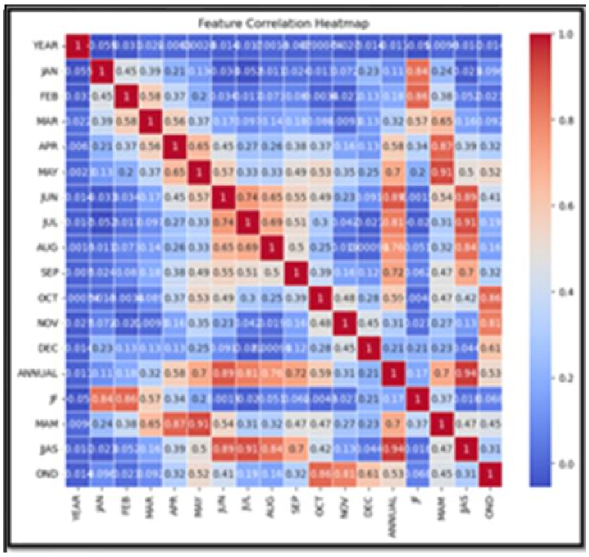


Fig. 6. Feature Correlation Heatmap

The feature correlation heatmap provides an appealing visual representation of pairwise correlations between features (in this case, monthly rainfall data from 'JAN' to 'DEC') in the 'data' dataset. The color-coded map, generated using Seaborn and Matplotlib, helps to quickly identify relationships between features. Positive correlations appear in warmer colours (e.g., red and orange), while negative correlations appear in cooler colours (e.g., blue). The numerical values within each cell of the heatmap indicate the strength and direction of the correlation. This visualization is invaluable for assessing how features interact and can assist in feature selection, identifying redundant variables, and understanding potential dependencies within the dataset.[5]

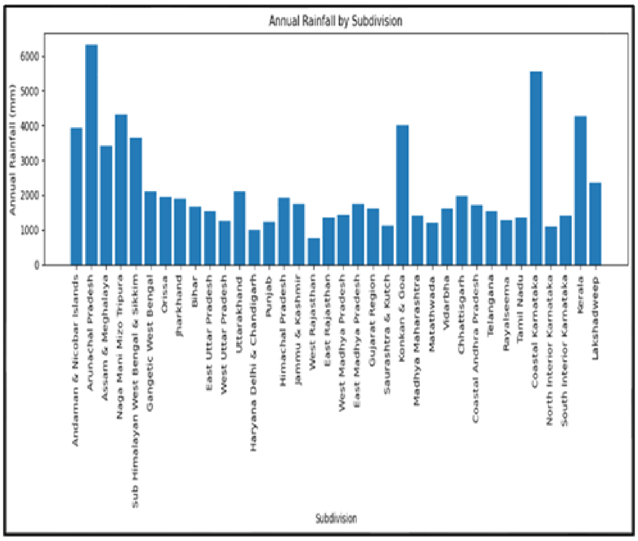


Fig. 7. Annual Rainfall by Sub-Division

The provided data visualization is a bar plot that illustrates the annual rainfall for different subdivisions. Each bar in the plot represents a specific subdivision, and the height of the bar corresponds to the annual rainfall (in millimetres) for that particular subdivision. This visualization makes possible for a quick comparison of annual rainfall across various geographical regions, making it easy to identify variations and patterns. The application of rotation on the x- axis labels enhances readability, by ensuring that the subdivision names do not overlap. The plot provides a clear overview of how annual rainfall is distributed in different subdivisions, which is also valuable for understanding regional climate patterns and their possible influence on agriculture, water resource management, and environmentalplanning. [9]

In the current research work, we conducted a comprehensive analysis of rainfall data for Indian subdivisions in the years 1901-2017, focusing on the classification of annual rainfall patterns into 'low,' 'moderate,' and 'high' categories adopting machine learning algorithms. Our comparative study included Decision Trees, Logistic Regression, and Random Forest. After a thorough evaluation, the Logistic Regression model and Random Forest model demonstrated higher accuracy compared to the Decision Tree model. Alongside accuracy, we considered various evaluation metrics and potential real- world implications. The choice of the best model depends on the specific objectives and cost considerations of the application. This study acts as a valuable foundation for many research avenues in rainfall analysis and contributes to informed decision-making in many domains such as agriculture and environmental planning. [6]

VIII. Conclusion

In this classification problem of categorizing annual rainfall patterns into "low," "moderate," and "high," in terms of accuracy, both logistic regression and random forest outperform the decision tree model. However, It is crucial to emphasize that accuracy alone should not be used as the main criterion for model selection because it does not provide a whole view of a model's performance, particularly in data sets with imbalances or when multiple kinds of mistakes have varied outcomes.

Additional considerations for model selection may include precision, recall, F1-score, and the specific problem objectives. On the basis of the application, false positives or false negatives may carry different costs or implications. Therefore, to arrive at an informed selection, it is best to consider a variety of evaluation indicators.

In this case, you have a tie between Random Forrest and Logistic Regression in regards to accuracy. To choose the best model, consider the following steps.

- Analyse Additional Metrics: Examine evaluation metrics like recall, precision, and F1-score to have a better understanding of the models' performance.
- Domain Knowledge: Consider the practical implications of misclassifications in your specific application. Is one type of error more critical than the other?
- Model Complexity: Assess the complexity of the models. Random Forest tends to be more complex and may overfit on small datasets, while Logistic Regression is simpler and more interpretable.
- Cross-Validation: Perform cross-validation to guarantee that the model's performance is stable across various data splits. [6]

References

[1] L. Breiman, Classification and Regression Trees (1st ed.)., Routledge, 1984.

[2] L. Breiman, "Random forests," *Machine learning,* vol. 45, pp. 5-32, 2001.

[3] P. Domingos, "A Few Useful Things to Know about Machine Learning. Commun," *ACM,* vol. 55(10), 2012.

[4] K. C. a. M. C. Gouda, "Data mining for weather and climate studies," *Int. J. Eng. Trends Technol,* vol. 32 , pp. 29-32, 2016.

[5] T. T. S. M. B. a. M. W. Nocke, "Visualization of climate and climate change data: An overview," *Digital earth summit on geoinformatics,* pp. 226-232, 2008.

[6] D. a. U. G. Gupta, "A comparative study of classification algorithms for forecasting rainfall," *IEEE,* pp. 1-6, 2015 .

[7] T. a. V. M. Pranckevičius, "Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification," *Baltic Journal of Modern Computing 5,* vol. 2, 2017.

[8] W. X. X. J. W. B. P. H. H. D. T. B. Z. D. a. J. M. Chen, "A comparative study of logistic model tree, random forest, and classification and regression tree models for spatial prediction of landslide susceptibility.," *Catena ,* vol. 151, pp. 147- 160., 2017.

[9] S. K. a. V. K. Jain, "Trend analysis of rainfall and temperature data for India," *Current Science,* pp. 37-49., 2012.

[10] E. Alpaydin., Machine Learning: The New AI, 2nd ed., MIT Press, 2021.