JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Comparative Analysis of Machine Learning Algorithms for Rainfall Prediction

Chau Woimalaseng Mein, Richa Singh, * Sonali Mondal, Biswajit Das

Department of Computer Science, Arunachal University of Studies, Namsai, Arunachal Pradesh, India

Abstract: Accurate rainfall prediction is crucial due to its impact on agriculture, a key sector in many countries. Predicting rainfall involves both short-term and long-term forecasts, with short-term generally being more reliable due to fewer variables affecting the outcome. To tackle this, a range of machine learning methods are employed, including Logistic Regression, XGBoost Classifier, Support Vector Classifier (SVC), K-Nearest Neighbors Classifier (KNN), Random Forest Classifier, and Linear SVC. These models offer diverse approaches to evaluating and predicting rainfall, each with unique advantages and features. The goal of this initiative is to make these sophisticated techniques accessible to laypersons, providing a guide to the methodologies used and a comparative study of various machine learning strategies.

Key Words: K-Nearest Neighbors Classifier, Machine Learning, Prediction, Regression, Rainfall, Support Vector Classifier.

I. INTRODUCTION

Rainfall forecasting holds significant importance for a wide range of stakeholders including governments, businesses, and scientific entities. Accurate predictions are crucial for safeguarding lives and property, managing agricultural operations to prevent crop failure, and mitigating property damage caused by unpredictable and heavy rainfall. Rainfall influences numerous aspects of human life, including agriculture, construction, energy production, forestry, and tourism. Accurate rainfall forecasts are especially vital as they are closely linked to natural disasters such as floods, landslides, and avalanches, which have historically affected societies [2]. Therefore, developing reliable methods for predicting rainfall is key to implementing preventative and mitigation strategies against these natural hazards.

In this research, we utilize a range of established machine learning methods, including Logistic Regression, XGBoost Classifier, Support Vector Classifier (SVC), K-Nearest Neighbors (KNN), Random Forest Classifier, and Linear SVC. Each algorithm provides distinct methods for analysing and forecasting rainfall, with their own set of benefits and features.

This paper seeks to cover the entire machine learning lifecycle, from data preprocessing to model implementation and evaluation. The data preprocessing phase involves steps such as imputing missing values, transforming features, encoding categorical features, scaling features, and selecting relevant features [1]. For evaluation, we utilized metrics such as Accuracy, Precision, Recall, and F-Score.

Our dataset comprises a range of meteorological variables that capture the intricate dynamics of weather conditions, including day, atmospheric pressure, maximum and minimum temperatures, temperature, dew point, humidity, cloud cover, rainfall, sunshine duration, wind direction, and wind speed. These features serve as essential indicators of impending weather changes, providing crucial data for predictive modelling.

II. LITERATURE REVIEW

Mohammed et al. (2020) [1] discusses the various machine learning models for predicting rainfall and discovered some techniques that can manages the nonlinear complex patterns related to meteorological information. Their work improves the field by comparing the performance of several algorithms for rainfall prediction. Oswal (2021) [2] The objective is to focus on three key areas: modelling inputs, modelling methods, and preprocessing methods. This study examines the evaluation criteria of algorithms and their ability to calculate rainfall based on weather data. Parmer et al. (2017) [3] The goal is to compare the various approaches and algorithms used by researchers for predict rain in table format. Rahman et al. (2020) [4] Ensemble learning combines different models to increase prediction performance, and their research most likely supports the success of this method in the field of rainfall prediction. Sarasa-Cabezuelo (2022) [5] The research may have modified machine learning algorithms to Australia's distinct climatic situation, so helping to localized weather prediction efforts. Singh et al. (2019) [6] Explore a combined strategy to predict rainfall using machine learning techniques such as Random Forest, Gradient Boosting, AdaBoost, K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Neural Networks (NN). The algorithms were tested on rainfall data from 2007-2017 in North Carolina. Performance determined using metrics such as F-score, precision, accuracy, recall. After analysing eight hybrid models, Gradient boosting-Ada boost emerged as the most effective and yielded positive results. Sudhakar et al. (2022) [7] focuses on improving rainfall prediction by using LLRS (Local Linear Regression Smoothing) algorithm. This study advances meteorology by giving a more accurate approach for predicting rainfall, which is vital for different fields such as agriculture, managing water resources, and preparing for disasters. Patel et al. (2018) [8] The authors emphasized the importance of selecting the optimal

algorithm for the decision-making situation, which is key for applications such as rainfall prediction, where accurate and cost-effectiveness are essential. Their research enhances the field by offering a thorough comparison of algorithm performances, enabling the identification of the optimal tree method for forecasting rainfall patterns. Schonlua et al. (2020) [9] found the random forest algorithm, an ensemble learning approach for processing large datasets and dealing with nonlinearities in the data. They developed random forest, a new command that makes it easier to utilize random forests in statistical programs. The article proved the algorithm's superior predictive performance over linear regression, especially in complex applications such as rainfall prediction, where the relationships between variables are not strictly linear. The authors' contribution is to make the random forest technique more accessible and understandable, resulting in improved applicability in statistical learning applications such as rainfall forecasting. Cheng et al. (2018) [10] introduced a unique kNN algorithm that dynamically computes the appropriate value of k for each test sample, solving limitations of fixed k values and ignoring sample connection. The concept is especially relevant for rainfall prediction, as the perfect number of neighbors (k) might vary significantly across data points. Their technique increases the kNN algorithm's ability to predict rainfall by enhancing accuracy and adaptability to diverse datasets.

III. Methodology

The model used for prediction anticipates rainfall. First, we input data that we have in correct form to conduct experiments, analyse our dataset, and detect variations of rain patterns. We predict rainfall by splitting the dataset into training and testing or validation sets, then applying multiple algorithms, such as (SVC, linear-SVC, KNN, etc.) and statistical techniques, compare and drawing conclusions about the ways utilized.

3.1 Dataset Description:

- Day: Date of observation.
- Maxtemp: The highest temperatures are typically reduced during rain.
- Dewpoint: Rainy days often see increased dewpoints.
- Humidity: Expect higher humidity when rainfall is forecast, usually with cloud presence.
- Sunshine: Rainy conditions are associated with diminished sunshine.
- Windspeed: Wind speeds tend to escalate on rainy days.
- Winddirection: Wind direction often shifts and intensifies during rainfall.
- Cloud: Extensive cloudiness is prevalent on days with rainfall.
- Mintemp: The lowest temperatures rise on rainy days.
- Temperature: Temperatures generally decrease when it rains.
- Pressure: Barometric pressure tends to fall on days with rainfall.

Large datasets require feature reduction to increase accuracy, reduce computing time, and improve storage.

3.2 Models

We selected classifiers from different categories, including linear, tree-based, distance-based, rule-based, and ensemble models. All classifiers were developed using scikit-learn, except for the Decision table, which was built with Weka [2].

The following categorization techniques were used to predict the models of the experiments:

Logistic Regression is a technique used to forecast binary outcomes (such as 1/0, Yes/No, True/False) based on several independent variables. It employs dummy variables to encode binary and categorical outcomes. This method can be seen as a special case of linear regression, where the outcome is categorical and the dependent variable is the logarithm of the odds. It models data using a logistic function to estimate the likelihood of an event's occurrence. Thus, Logistic Regression is particularly well-suited for binary classification tasks.[2]

Decision Tree is an algorithm that employs a hierarchical tree structure to illustrate the interactions among various variables. It initiates from a base point known as the root and splits into progressively finer branches. Each split represents a decision point, indicated by decision nodes. The algorithm terminates at the leaf nodes, where the data segments are homogeneous and cannot be subdivided any further. This characteristic of the Decision Tree makes it particularly advantageous for our case, given that our target variable is binary categorical [5,8].

K-Nearest Neighbor (KNN) algorithm is straightforward in its approach, learning directly from the dataset without assuming any specific data distribution. It functions by identifying the K nearest data points to a new point and then assigning to this new point the most common class observed among these neighbors. This method requires a system to measure distances between points. KNN works better with fewer features because having more features may need more data and cause the model to overfit.[10]

Random Forest is an ensemble learning method used in supervised learning. It involves combining multiple weak learners to create a single, strong predictor. This ensemble approach enhances the overall prediction accuracy and robustness. [9]

3.3 Evaluation

For the evaluation of our classifier, we utilized the following metrics:[2]

Accuracy measures the proportion of correct predictions against the total samples tested. It is most effective when the dataset is balanced across classes. Given our data's imbalance, we will consider additional metrics.

Precision calculates the ratio of true positive predictions to the total predicted positives. It is defined for Rainfall as:

Recall quantifies the proportion of true positives out of all relevant samples (those that should have been correctly identified as positive). It is defined for rainfall as:

The F1 Score is the Harmonic Mean of precision and recall, ranging between 0 and 1. It assesses both the accuracy (correct classifications) and the robustness (not overlooking significant instances) of the classifier. A higher F1 Score indicates better overall performance. It is calculated for Rainfall as:

3.4 Support Vector Machine (SVM)

Support Vector Machine, a type of feed forward network, is suitable for various tasks such as pattern classification and nonlinear regression. [3,4,7]

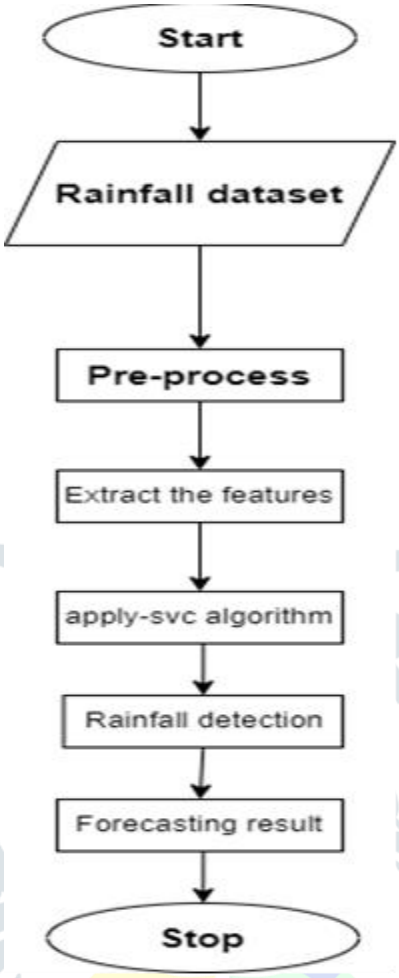

Fig 1: Model for predict rainfall

Flowchart:
Predict Rainfall
Input: Dataset of Rainfall
Output: Accuracy of Prediction
Step 1: Import the dataset containing rainfall data from a CSV file.
Step 2: Impute missing data with the average value of the dataset.
Step 3: Feature Scaling - Adjust the data features to a uniform scale.
Step 4: Reduce the dataset size using SVC for feature reduction.
Step 5: Divide the data into an 80% training set and a 20% testing or validation set.
Step 6: Utilize various algorithms including Logistic Regression, SVCs, KNN, and Random Forest Classifier.
Step 7: Create bar charts to visualize the accuracy comparisons among the algorithms.
Step 8: Display the analytical results.

IV. Experimental Results

Dataset of rain during a day is collected which based on some atmospheric factors of that place. Below is the pie chart of our target, whether rainfall or not.

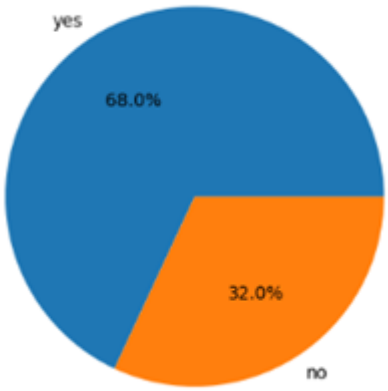

Fig 2: pic chart of the number data for the target.

Distribution plot is a separated of data by presenting the frequency or probability of different values. The graph below is the spread of continuous features provided by the datasets.

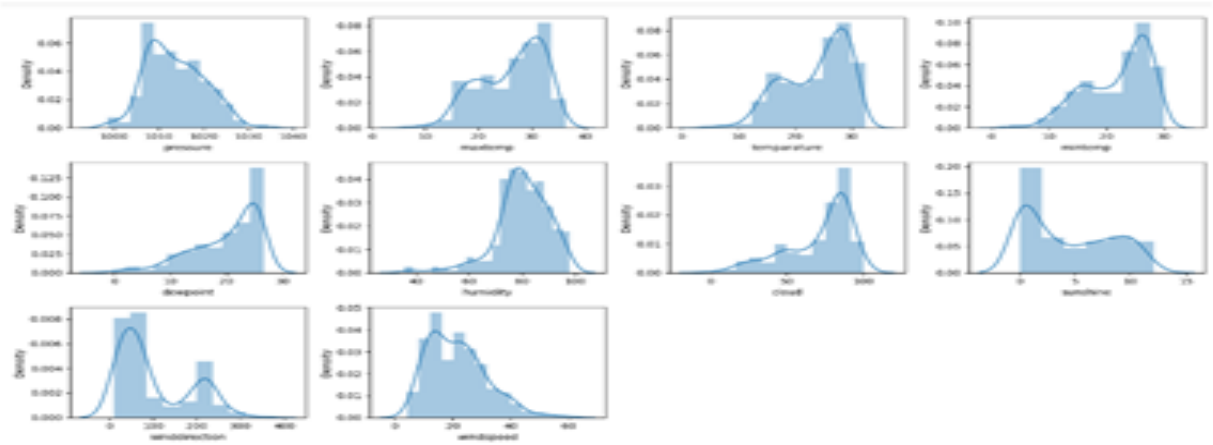

Fig 3: Spread plot of continuous data of each target.

Highly correlated variables can enlarge the feature space and potentially degrade model efficacy. Thus, it is essential to assess if the dataset contains strongly linked attributes. The following heatmap will help identify significant correlation among n features.

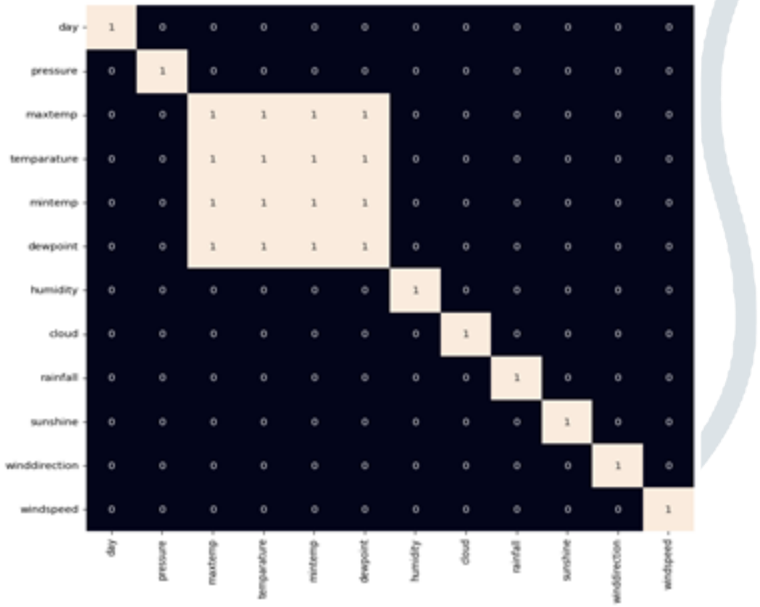

Fig 4: Heat map detect highly correlated

The below is the chart plotted for the rainfall of a day with their compare of training and validation accuracy values.

Classifier Names	Training accuracies	Validating accuracies
LogisticRegression	0.88939673	0.896666667
XGBClassifier	0.99999999	0.839166666
SVC	0.9.0267922	0.885833333
KNeighborsClassifier	0.946718517	0.844583333

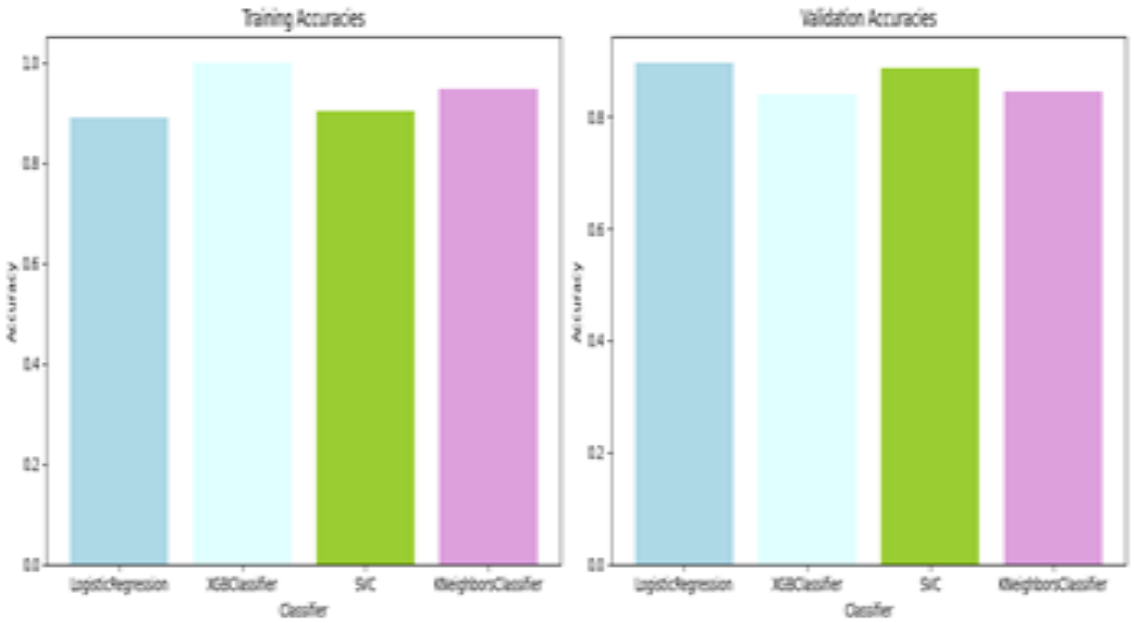

Fig 5: Bar graph of the rainfall given data.

The below bar graphs show all the accuracy of various algorithms used for validating and testing the values of the data.

Classifier names	**Accuracies**
LogisticRegression	0.8966667
XBGClassifier	0.8391666
SVC	0.8858333
KNeighborsClassifier	0.8445833
LinearSVC	0.8
RandomForestClassifier	0.8375

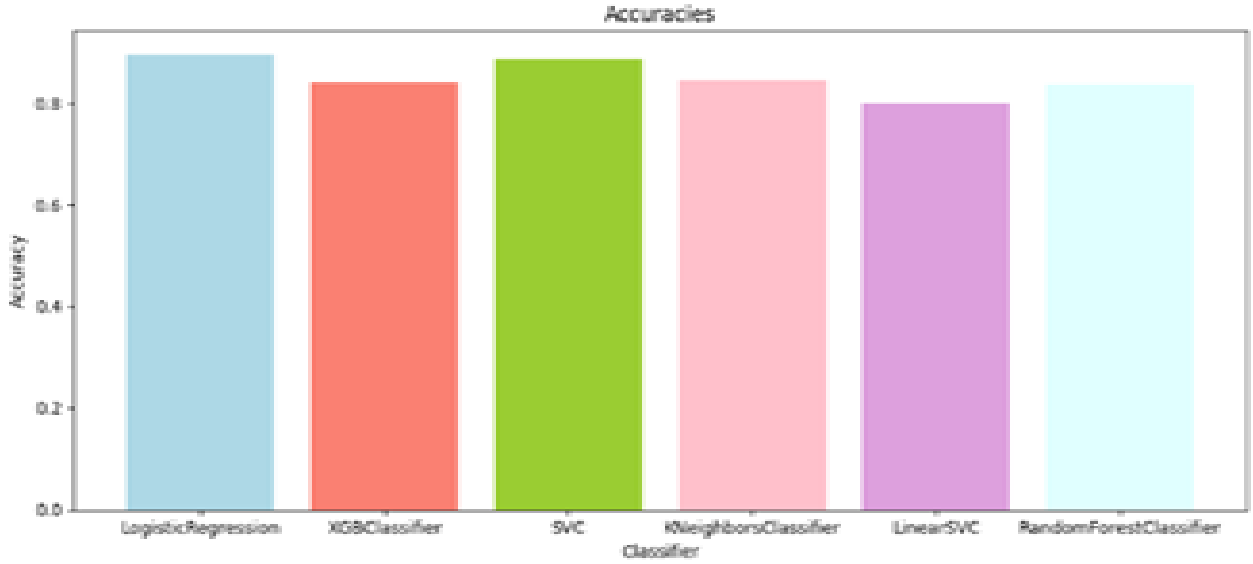

Fig 6: Bar graphs for all the accuracy values

V. CONCLUSIONS

This study shows the essential necessity of precise rainfall forecast in a variety of areas. We used a set of machines learning methods, including Support Vector Classifier (SVC), to handle the complexity of rainfall prediction with accuracy. Through extensive data preprocessing and model validation, we discovered SVC's ability to capture subtle rainfall patterns. Furthermore, our investigation of feature reduction strategies and the identification of strongly associated features streamlined predictive models, resulting in improved accuracy. This study highlights machine learning's revolutionary potential in improving our ability to adapt to the effects of irregular rainfall, clearing the way for more informed decision-making and active methods for reducing risks.

REFERENCES

1) Mohammed, M., Kolapalli, R., Golla, N. and Maturi, S.S., 2020. Prediction of rainfall using machine learning techniques. International Journal of Scientific and Technology Research, 9(01), pp.3236-3240.
2) Oswal, N. (2021) 'Predicting Rainfall using Machine Learning Techniques,' Literature Review [Preprint]. https://doi.org/10.36227/techrxiv.14398304.v1.
3) Parmar, A., Mistree, K. and Sompura, M., 2017, March. Machine learning techniques for rainfall prediction: A review. In International conference on innovations in information embedded and communication systems (Vol. 3).
4) Sani, N.S., Abd Rahman, A.H., Adam, A., Shlash, I. and Aliff, M., 2020. Ensemble learning for rainfall prediction. International Journal of Advanced Computer Science and Applications, 11(11).
5) Sarasa-Cabezuelo, A., 2022. Prediction of rainfall in Australia using machine learning. Information, 13(4), p.163.
6) Singh, G. and Kumar, D., 2019, January. Hybrid prediction models for rainfall forecasting. In 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 392-396). IEEE.
7) SUDHAKAR, K., RENUKA, B.V., JYOTHI, P., MONICA, S. and AHAMMAD, S.S., Rainfall prediction using LLRS Algorithm.
8) Patel, H.H. and Prajapati, P., 2018. Study and analysis of decision tree-based classification algorithms. International Journal of Computer Sciences and Engineering, 6(10), pp.74-78.
9) Schonlau, M. and Zou, R.Y., 2020. The random forest algorithm for statistical learning. The Stata Journal, 20(1), pp.3-29.
10) Zhang, S., Cheng, D., Deng, Z., Zong, M. and Deng, X., 2018. A novel kNN algorithm with data-driven k parameter computation. Pattern Recognition Letters, 109, pp.44-54.