*Mathematical Biosciences and Engineering*

http://www.aimspress.com/journal/mbe

MBE, 20(5): 8975–9002.
DOI: 10.3934/mbe.2023394
Received: 30 November 2022
Revised: 26 February 2023
Accepted: 01 March 2023
Published: 10 March 2023

*Research article*

# An AI-Enabled ensemble method for rainfall forecasting using Long-Short term memory

**Sarth Kanani[1], Shivam Patel[1], Rajeev Kumar Gupta[1], Arti Jain[2] and Jerry Chun-Wei Lin[3,*]**

[1] Department of Computer Science and Engineering, School of Technology, Pandit Deendayal Energy University, Gandhinagar 382007, Gujarat, India

[2] Department of Computer Science & Engineering and Information Technology, Jaypee Institute of Information Technology, Noida, Uttar Pradesh, India

[3] Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, Bergen, Norway

* **Correspondence:** Email: jerrylin@ieee.org.

**Abstract:** Rainfall prediction includes forecasting the occurrence of rainfall and projecting the amount of rainfall over the modeled area. Rainfall is the result of various natural phenomena such as temperature, humidity, atmospheric pressure, and wind direction, and is therefore composed of various factors that lead to uncertainties in the prediction of the same. In this work, different machine learning and deep learning models are used to (a) predict the occurrence of rainfall, (b) project the amount of rainfall, and (c) compare the results of the different models for classification and regression purposes. The dataset used in this work for rainfall prediction contains data from 49 Australian cities over a 10-year period and contains 23 features, including location, temperature, evaporation, sunshine, wind direction, and many more. The dataset contained numerous uncertainties and anomalies that caused the prediction model to produce erroneous projections. We, therefore, used several data preprocessing techniques, including outlier removal, class balancing for classification tasks using Synthetic Minority Oversampling Technique (SMOTE), and data normalization for regression tasks using Standard Scalar, to remove these uncertainties and clean the data for more accurate predictions. Training classifiers such as XGBoost, Random Forest, Kernel SVM, and Long-Short Term Memory (LSTM) are used for the classification task, while models such as Multiple Linear Regressor, XGBoost, Polynomial Regressor, Random Forest Regressor, and LSTM are used for the regression task. The experiment results show that the proposed approach outperforms several state-of-the-art approaches with an accuracy of 92.2% for the classification task, a mean absolute error of 11.7%, and an R2 score of 76% for the regression task.

**Keywords:** classification; regression; XGBoost classifier; random forest; LSTM; rainfall

## 1. Introduction

Rainfall has played an important role in the development and maintenance of human civilizations. Rain is one of the most important sources of pure water on which humans depend for life. Rain replenishes groundwater, which is the main source of drinking water. Since more than 50% of Australia's land mass is used for agriculture, an accurate rainfall forecasting system can help farmers plan cropping operations, i.e., when to sow seeds, apply fertilizers, and harvest crops. Rainfall prediction [1] can also help farmers decide which crops to plant for maximum harvests and profits. In addition, precipitation plays an important role in the planning and maintenance of water reservoirs, such as dams that generate electricity from hydropower. About half of the renewable energy generated by more than 120 hydropower plants in Australia comes from precipitation. With accurate rainfall forecasts, operators are well informed about when to store water and when to release it to avoid flooding or drought conditions in places with low rainfall. Precipitation forecasts also play a critical role in the aviation industry, from the moment an aircraft starts its engine. An accurate precipitation forecast helps plan flight routes and suggests the right time to take off and land a flight to ensure physical and economic safety. After all, aircraft operations can be seriously affected by lightning, icing, turbulence, thunderstorm activity and more. According to [2], climate is a major factor in aviation accidents, accounting for 23% of accidents worldwide.

Numerous studies have shown that the duration and intensity of rainfall can cause major weather-related disasters such as floods and droughts. AON's annual weather report shows that seasonal flooding in China from June to September 2020 resulted in an estimated economic loss of 35 billion and a large number of deaths [3]. In addition, rainfall also has a negative impact on the mining industry, as heavy and unpredictable rainfall can affect mining activities. For example, the Bowen Basin in Queensland hosts some of Australia's largest coal reserves. The summer rains of 2010–2011 severely impacted mining operations. An estimated 85% of coal mines in Queensland had their operations disrupted as a result (Queensland Flood Commission, 2012) [4, 5]. As of May 2011, the Queensland coal mining sector had recovered only 75% of its pre-flood production and lost 5.7 billion. As a result, rainfall forecasts are becoming increasingly important in developing preventive measures to minimize the impact of such disasters.

Predicting rainfall is challenging because it involves the study of various natural phenomena such as temperature, humidity, wind speed, wind direction, cloud cover, sunlight, and more. Therefore, accurate rainfall forecasts are critical in areas such as energy and agriculture. A report produced by Australia's National Climate Change Adaptation Research Facility examined the impacts of extreme weather events. It states that currently available weather forecasts for industry are inadequate. It lacks location information and other details that enable risk management and targeted planning. Traditional weather forecasts use various hardware parameters to predict parameters and use mathematical calculations to predict heavy rainfall, which are sometimes inaccurate and therefore cannot work effectively. The Australian Bureau of Meteorology currently uses the Australian Predictive Ocean Atmosphere Model (POAMA) forecasts to predict rainfall patterns [6]. POAMA is a standard distribution model used for many weeks to specific seasons to look at weather throughout the year. It uses surveys of ocean, atmospheric, ice, and Earth data to develop ideas for up to nine months. In this work, we use machine learning and deep learning methods [7, 8] based on the analysis of complex patterns based on historical data to effectively and accurately predict the occurrence of rainfall. The

application of this method requires accurate historical data, the presence of patterns that can be detected, and their continuation into the future where predictions are sought.

Several divisive algorithms such as Random Forest [9], Naive Bayes [10], Logistic Regression [11], Decision Tree [12], XGBoost [13], and others have been studied for rainfall prediction. However, the effectiveness of these algorithms varies depending on a combination of preprocessing and data cleaning techniques, feature scaling, data normalization, training parameters, and segmentation testing, leaving room for improvement. The goal of this paper is to provide a customized set of these techniques to train machine learning [14] and deep learning [15] models that provide the most accurate results for rainfall prediction. The models are trained and tested on the Australian rainfall database using the proposed approach. The database contains records from 49 metropolitan areas over a 10-year period starting December 1, 2008. The research contributions of the proposed work are as follows:

1) To remove outliers using Inter Quartile Range (IQR).

2) To balance the data using Synthetic Minority Oversampling Technique (SMOTE) technique.

3) To apply both classification and regression models which at first predict whether it rain or not and if there is rain then find the amount of rain.

4) To apply XGBoost, Random Forest, Kernel SVM, and Long-Short Term Memory (LSTM) for the classification task.

5) To apply Multiple Linear Regressors, XGBoost, Polynomial Regressor, Random Forest Regressor, and LSTM for the regression task.

## 2. Literature review

Luk et al. [16] addressed watershed management and flood control. The goal was to accurately predict the temporal and local distribution of rainfall and the amount of water and quality management. The dataset used to train the ML models was collected from the Upper Parramatta River Catchment Trust (UPRCT), Sydney. Three methods were used for modeling features related to rainfall prediction, namely MultiLayer FeedForward Network (MLFN), Partial Recurrent Neural Network (PRNN) and Time Delay Neural Network (TDNN). The main parameters for the above methods were lag, window size and number of hidden nodes.

Abhishek et al. [17] worked on developing ANN -based effective and nonlinear models for accurate prediction of maximum temperature 365 days a year. The data used were from the Toronto Lester B. Pearson Int'l A station, Ontario, Canada from 1999–2009. They proposed two models and trained them using the Levenberg-Marquardt algorithm with 5 hidden layers and a model with 10 hidden layers. Factors that affected the results were the number of neurons, sampling, hidden layers, transfer function, and overfitting.

Abhishek et al. [18] performed a regression task to predict average rainfall using a feed forward network trained with the back propagation algorithm, the layer recurrent network, and the feed forward network trained with the cascaded back propagation algorithm for a large number of neurons. The data were collected from www.Indiastat.com and the IMD website. The dataset contains records for the months of April to November from 1960 to 2010 in Udupi district of Karnataka.

Saba et al. [19] worked on accurate weather predictions using a hybrid neural network model combining MultiLayer Perceptron (MLP) and Radial Basis Function (RBF). The dataset used was from the weather station in Saudi Arabia. They proposed an extended hybrid neural network approach and compared the results of individual neural networks with those of hybrid neural networks. The results showed that hybrid neural network models have greater learning ability and better generalization ability for certain sets of inputs and nodes.

Biswas et al. [20] focused on the prediction of weather conditions (good or bad) using the classification method. Naive Bayes and chi-square algorithm were used for classification. The main objective was to show that data mining approaches are sufficient for weather prediction. Data was obtained in real time from users and stored in a database. The decision tree generated from the training features is used for classification.

Basha et al. [21] introduced a Machine and Deep Learning-based rain prediction model. This model utilizes the Kaggle dataset to train various models, including the Support Vector Regressor, Autoregressive Integrated Moving Average (ARIMA), and Neural Network. The authors claim that the performance of the model, as measured by the Root Mean Squared Error (RSME), is 72%.

Doroshenko et al. [22] worked on refining numerical weather forecasts using a neural network by error to increase the accuracy of the additional 2m weather forecasts of the regional model COSMO. The dataset was obtained from the Kiev weather station in Ukraine. The authors chose the gated recurrent unit (GRU) approach because the error in the weather forecast is a time series and it also has fewer parameters than LSTM. When a lower error history is chosen, better fitting and refinement of the model is possible.

Appiah-Badu et al. [23] conducted a study to predict the occurrence of rainfall through classification. They employed several classification algorithms, including Decision Tree (DT), Random Forest (RF), Multilayer Perceptron (MLP), Extreme Gradient Boosting (XGB), and K-Nearest Neighbor (KNN). The data for the study was collected from the Ghana Meteorological Agency from 1980 to 2019 and was divided into four ecological zones: Coastal, Forest, Transitional, and Savannah.

Raval et al. [24] worked on a classification task to predict tomorrow's rain using logistic regression, LDA, KNN, and many other models and compared their metrics. They used a dataset containing daily 10-year weather forecasts from most Australian weather stations. It was found that deep learning models produced the best results.

Ridwan et al. [25] proposed a rainfall prediction model for Malaysia. The model was trained using a dataset of ten stations and employs both a Neural Network Regressor and Decision Forest Regression (DFR). The authors claim that the R2 score ranges from 0.5 to 0.9. This approach only predicts rainfall and does not perform any classification tasks.

Adaryani et al. [26] conducted an analysis of short-term rainfall forecasting for applications in hydrologic modeling and flood warning. They compared the performance of PSO Support Vector Regression (PSO-SVR), Long Short-Term Memory (LSTM), and Convolutional Neural Network (CNN). The study considered 5-minute and 15-minute ahead forecast models of rainfall depth based on data from the Niavaran station in Tehran, Iran.

Fahad et al. [27] conducted a study on forecasting rainfall through the use of a deep forecasting model based on Gated Recurrent Unit (GRU) Neural Network. The study analyzed 30 years of climatic data (1991–2020) in Pakistan, considering both positive and negative impacts of temperature and gas

emissions on rainfall. The findings of the study have potential implications for disaster management institutions.

Tables 1 and 2 compare various state-of-the-art approaches for classification as well as regression tasks respectively.

**Table 1.** Literature review for classification.

| Authors | Dataset | Approach used | Best performance |
|---|---|---|---|
| Raval et al. (2021) [24] | Daily weather observations from several Australian weather stations for 10 years | Logistic regression, Linear discriminant analysis, Quadratic discriminant analysis, K-Nearest neighbor, Decision tree, Gradient boosting, Random forest, Bernoulli Naïve Bayes, Deep learning model | Precision = 98.26 F1-Score = 88.61 |
| Appiah-Badu et al. (2021) [23] | Data from the 22 synoptic stations across the four ecological zones of Ghana from 1980 – 2019 | Decision tree, Multilayer perceptron, Random forest, Extreme gradient boosting, K-Nearest neighbor | Precision = 100 Recall = 96.03 F1-Score = 97.98 |

Here, NMSE stands for Normalized Mean Square Error, which allows us to compare the error across sets with different value ranges. Using simple Mean Squared Error (MSE) can result in higher variance for sets with larger values, even if the variance for sets with smaller values is actually greater. For example, if set 1 contains elements with values ranging from 1–100 and set 2 contains elements with values ranging from 1000–10,000, the variance for set 2 will be higher if MSE is used, even if the variance for set 1 is actually greater. NMSE is used to compare the error across sets by dividing the entire set by the maximum value in the range, resulting in a conversion of both sets' ranges to 0-ik -1 for better comparison.

## 3. Data description

The Dataset we used for our study contains data from daily observations over a tenure of 10 years starting from 1/12/2008 up till 26/04/2017 from 49 different locations over Australia [28]. The dataset contains 23 features which are Date, Location, MinTemp, MaxTemp, Rainfall, Evaporation, Sunshine, WindGustDir, WindGustSpeed, WindDir9am, WindDir3pm, WindSpeed9am, WindSpeed3pm, Humidity9am, Humidity3pm, Pressure9am, Pressure3pm, Cloud9am, Cloud3pm, Temp9am, Temp3pm, RainToday and RainTomorrow. The dataset contains around 145 thousand entries.

For the classification task, RainTomorrow is the target variable that predicts the occurrence of rainfall on the next day. Here, 0 indicates no rain, and 1 indicates chance of rainfall. For the regression task, Rainfall is the target variable that forecasts the amount of precipitation in millimeters. We performed exploratory data analysis on the dataset, which is the key to machine learning problems in order to gain maximum confidence in the validity of future results. This analysis helps us to look for anomalies in the data, figure out correlations between features and check for missing values to enhance the outcomes of the machine learning models. Table 3 presents the analysis of null values in

**Table 2.** Literature review for regression.

| Authors | Dataset | Approach used | Limitations | Best performance |
|---|---|---|---|---|
| Luk et al. (2001) [16] | Dataset is collected from the Upper Parramatta River Catchment Trust (UPRCT), Sydney | Multi-Layer Feedforward Network (MLFN), Partial Recurrent Neural Network (PRNN), Time Delay Neural Network | Only used the regression model to predict the amount of rainfall | NMSE = 0.63 |
| Abhishek et al. (2012) [17] | Data available for the station Toronto Lester B. Pearson Int'l A, Ontario, Canada, 1999-2009 | Single layer model, 5 hidden layer model, 10 hidden layer model | Not used any sequential model to capture the time series nature of data. | MSE = 2.75 |
| Saba et al. (2017) [19] | Data used from Saudia Arabian Weather Forecasting Station | Hybrid Model (MLP+RBF) | Not used any time series model, only regression model to predict the amount of rainfall | Correlation coefficient = 0.95, RMSE = 146, Scatter Index = 0.61 |
| Basha et al. (2020) [21] | Dataset chosen is the Kaggle dataset for the rainfall prediction | Support vector regressor, AutoRegressive Integrated Moving Average (ARIMA), and Neural Network | Trained on a small dataset, no oversampling techniques are used to increase the size of dataset | RSME = 0.72 |

the raw dataset. Here, it is visible that most of the attributes contain null values which need to be addressed carefully before passing the data to train the model, otherwise the model will not give accurate predictions.

**Table 3.** Null values of attributes in the dataset.

| Attribute | Null Value | Attribute | Null value | Attribute | Null value |
|---|---|---|---|---|---|
| Date | 0.0% missing values | WindGustSpeed | 7.06% missing values | Pressure3pm | 10.33% missing values |
| Location | 0.0% missing values | WindDir9am | 7.26% missing values | Cloud9am | 38.42% missing values |
| MinTemp | 1.02% missing values | WindDir3pm | 2.91% missing values | Cloud3pm | 40.81% missing values |
| MaxTemp | 0.87% missing values | WindSpeed9am | 1.21% missing values | Temp9am | 1.21% missing values |
| Rainfall | 2.24% missing values | WindSpeed3pm | 2.11% missing values | Temp3pm | 2.48% missing values |
| Evaporation | 43.17% missing values | Humidity9am | 1.82% missing values | RainToday | 2.24% missing values |
| Sunshine | 48.01% missing values | Humidity3pm | 3.1% missing values | RainTomorrow | 2.25% missing values |
| WindGustDir | 7.1% missing values | Pressure9am | 10.36% missing values | | |

Figure 1 presents a correlation matrix that states the correlation coefficient between two features, i.e., how much the features are correlated to each other. The scale of the correlation matrix is from –1 to 1, where 1 represents the perfect positive relationship between the two factors and –1 represents the total negative relationship between the two factors. A correlation coefficient of 0 represents an absence of a relation between the two variables.

From the analysis of the null values, it appears that the attributes Evaporation, Sunshine, Cloud9am, and Cloud3pm contain almost 50% NaN values. Therefore, we have discarded these 4 columns and did not use them for training our model. This is because even if we use one of the available techniques to populate the data, it might differ from the performance of the model. The actual weather data may not match the padded data, which could affect the learning process of the model. Figure 1 presents a correlation matrix that shows the correlation coefficient between two features. As we removed four features from our dataset, they will not be included in our correlation matrix calculation. The Date feature is categorized into its respective months, and the month feature is taken into consideration for better season-wise categorization of data and the study of correlation. The reason for ignoring the Location feature is that the dataset is already classified into various locations. Therefore, finding a correlation between Location and Date is not beneficial in the data analysis. Additionally, the feature RainToday is ignored because the Rainfall feature, which attributes RainToday, is already included in the study of correlation.

Figure 2 displays the distribution of numerical data based on the 0th percentile, 25th percentile (1st quartile), 50th percentile (2nd quartile), 75th percentile (3rd quartile), and 100th percentile. This distribution provides insight into the presence of outliers, which must be eliminated prior to training our predictive model to achieve accurate results. In order to analyze the distribution of data with regards to various quartiles, the data must be continuous. Features such as Location, RainTomorrow, WindDir3pm, WindDir9am, WindGustDir, etc. are categorical features, therefore using a boxplot to remove outliers is only feasible with the use of continuous features for data analysis. As a result, we have used only 10 features that are continuous and have the potential for having outliers.

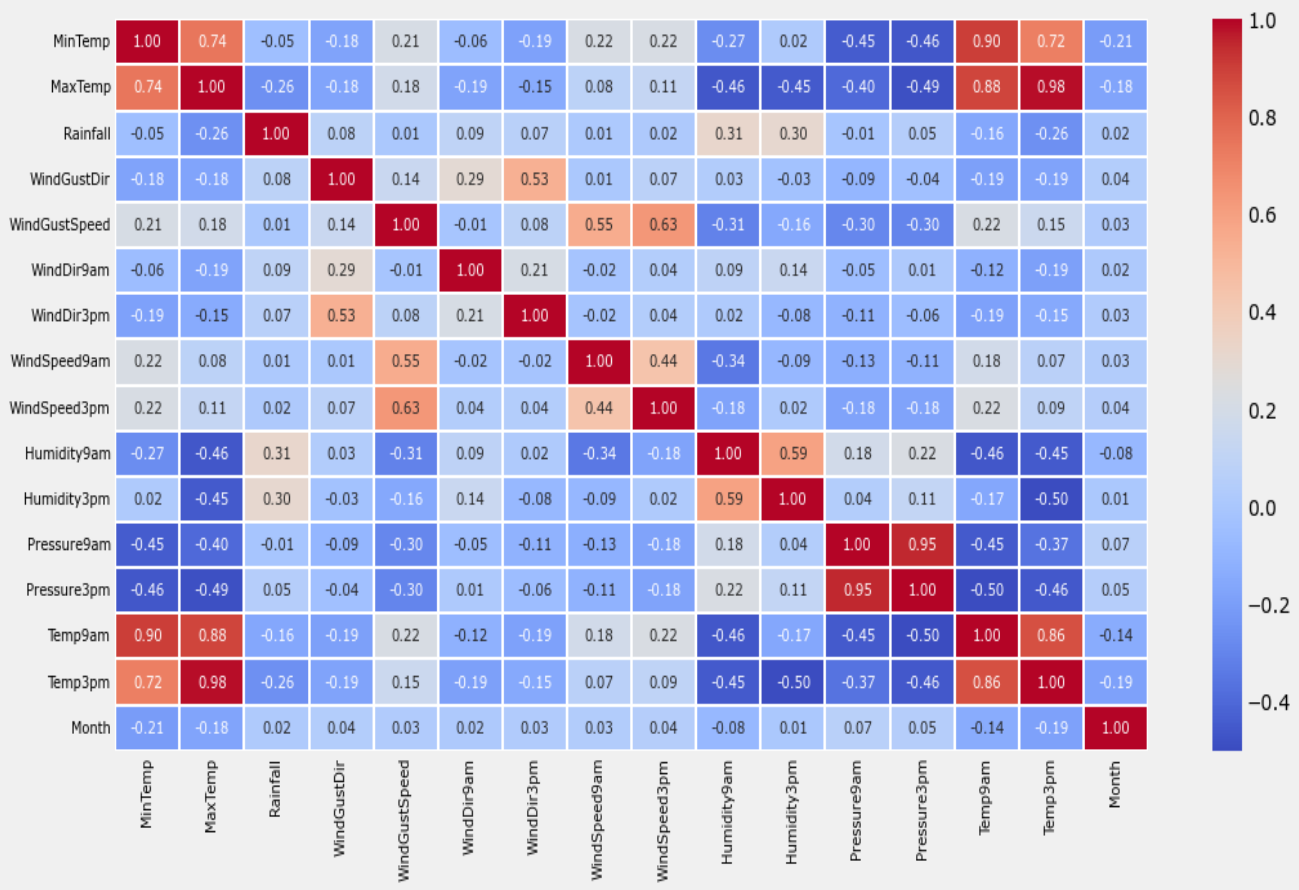


**Figure 1.** Correlation matrix for various features in the dataset.

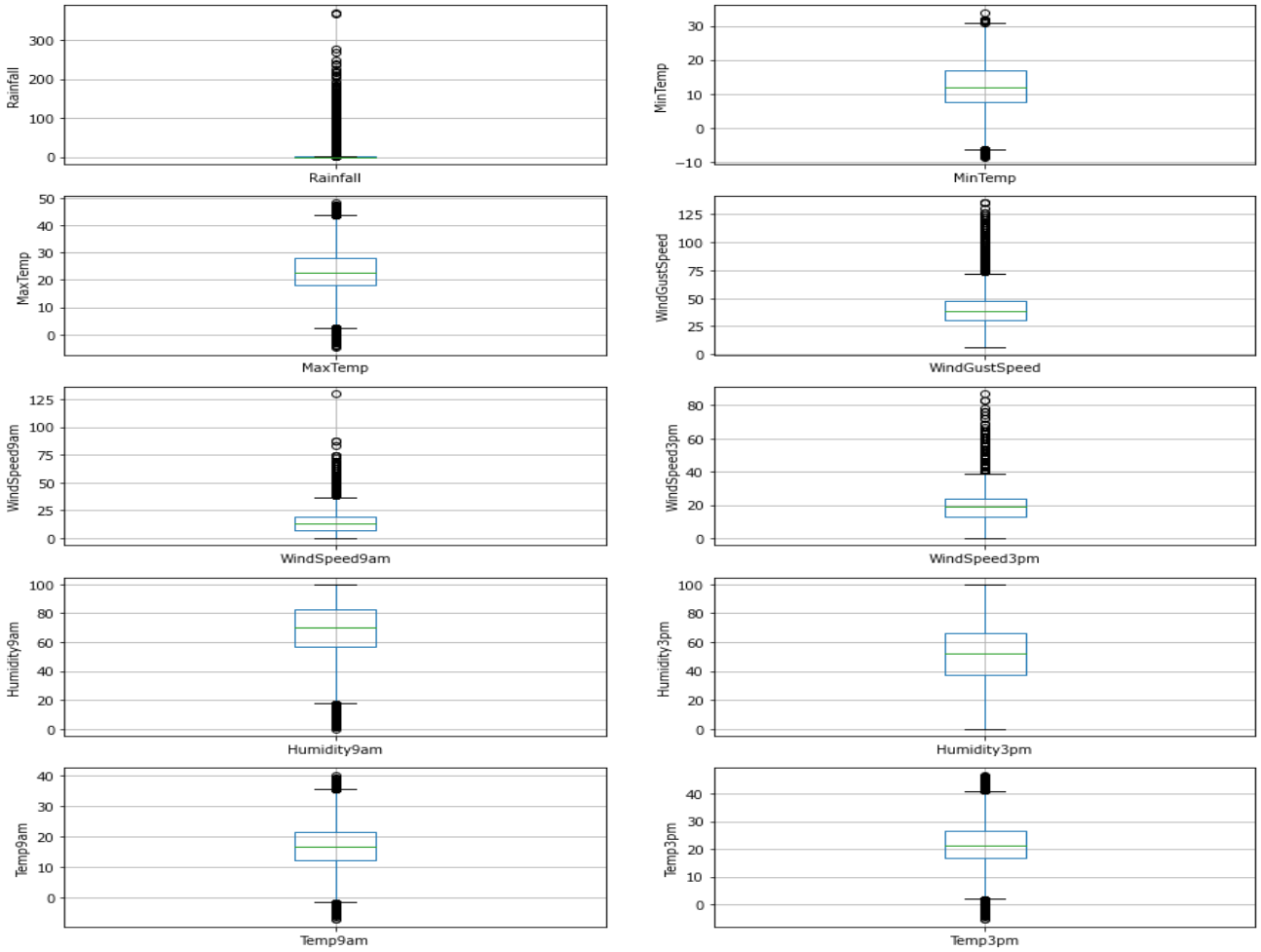


**Figure 2.** Distribution of data points w.r.t. quartile.

## 4. Proposed approach

Due to various factors such as global warming, deforestation, etc., affecting seasonal variables during the year, uncertainty in rainfall has become one of the most discussed topics among researchers. Therefore, the main objective of this work is to apply different techniques of data preprocessing, machine learning and deep learning models: 1) Data pre-processing to remove uncertainties and anomalies in the provided dataset. 2) Forecasting the occurrence of rainfall. 3) Projecting the amount of rainfall in millimeters. 4) Comparing the results of various models used for classification and regression purposes.

The comparison of various algorithms for the same task gives us more insights into the problem statement and helps us make decisions regarding the best model to be used for rainfall forecasting. Figure 3 illustrates the flow diagram of the proposed methodology.

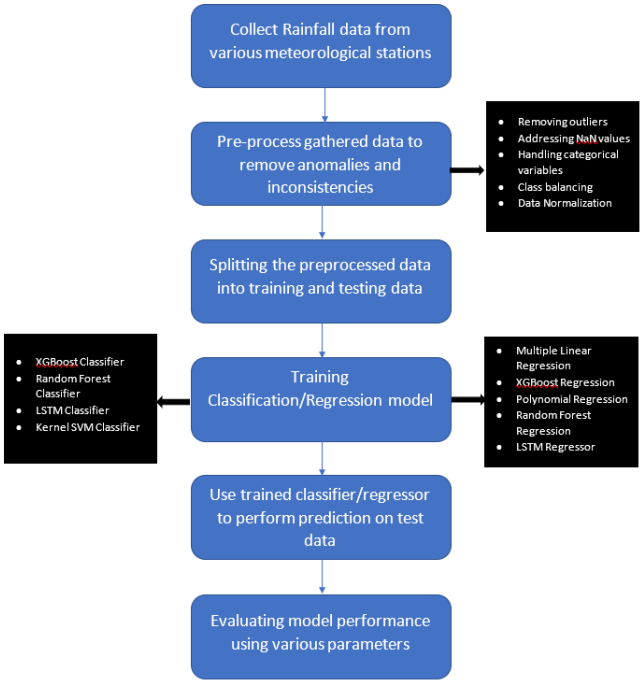


**Figure 3.** Flow diagram of the proposed approach.

### *4.1. Data pre-processing*

Data processing is a method of data mining that refers to the cleaning and modification of raw data collected from various sources that are suitable for the work and provide more favorable results.

#### 4.1.1. Removing outliers

As it is evident from Figure 4 that our data contains several outliers. Thus, we employed the Inter Quartile Range (IQR) approach to remove the outliers. IQR is basically the range between the 1st and the 3rd quartile, i.e., the 25th and 75th percentile. In this approach, the data point which falls below (Q1 – 1.5 * IQR) and above (Q1 + 1.5 * IQR) are considered outliers. After removing all the outliers

approximately 30 thousand rows were removed. Figure 4 represents the IQR approach employed for removing the outliers [29].

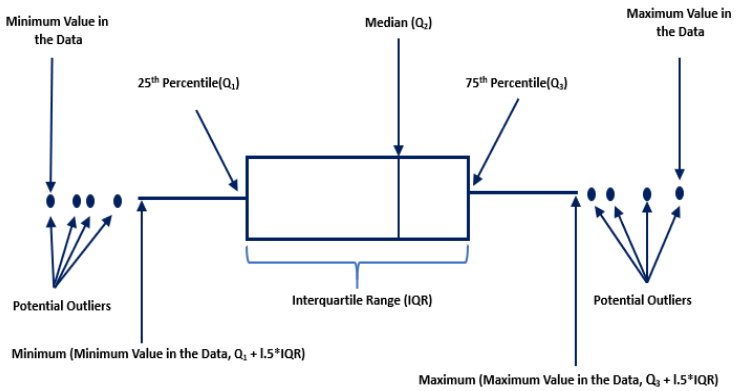


**Figure 4.** IQR approach for data cleaning.

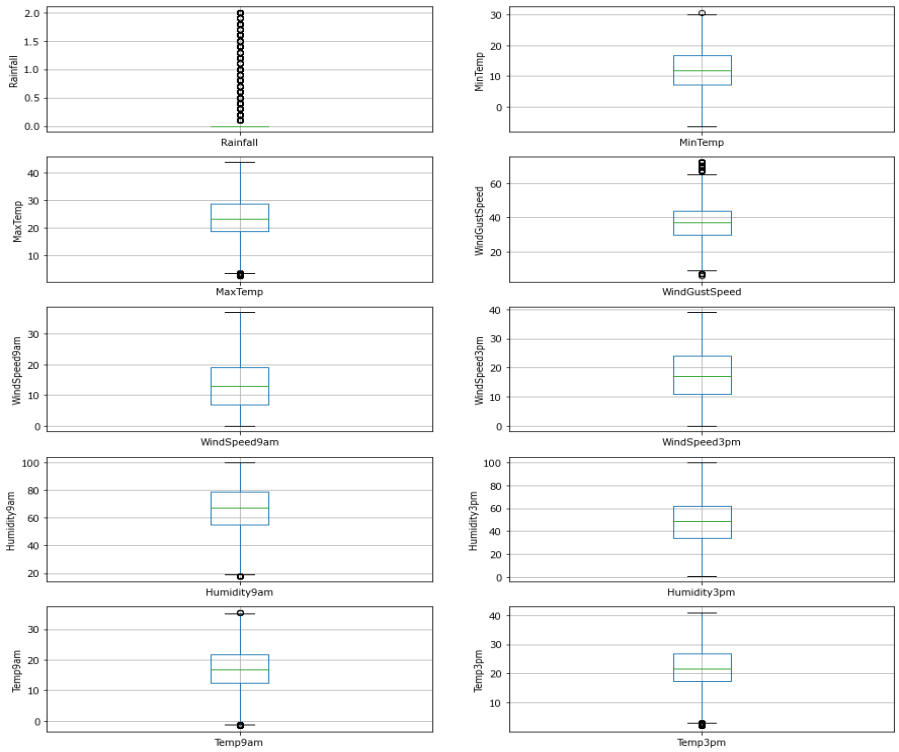


**Figure 5.** Distribution of normalized cleaned data points w.r.t. quartiles.

Figures 5 and 6 show the distribution of normalized values and IQR range plot after removing the outliers from the dataset.

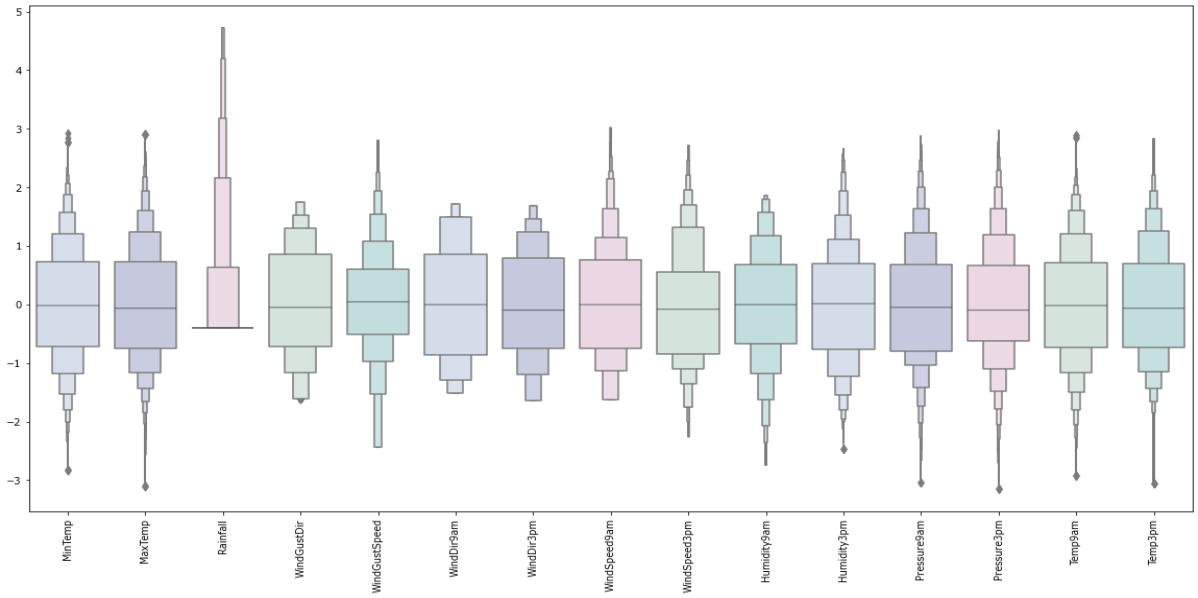


**Figure 6.** Distribution of cleaned data points w.r.t. quartiles.

#### 4.1.2. Addressing NaN values

From the analysis of the null values, it appears that the attributes evaporation, sunshine, cloud9am and cloud3pm contain almost 50% NaN values. Therefore, we discarded these columns and did not use them for training our model. This is because even if we use one of the available techniques to populate the data, it might differ from the performance of the model. The actual weather data may not match that of the padded data, which could affect the learning process of the model.

For the remaining attributes, we filled the numeric features with the mean value of the attribute and the categorical values with the mode of each feature. However, because location and seasons also play an important role in measuring the attributes, we divided the data set into 4 seasons, namely summer (January to March), fall (March to June), winter (June to September), and spring (September to December). We then averaged the values using the month and location attributes to group the data by season and location and populated the NaN values using a similar pairwise approach. Similarly, for categorical data, the maximum occurrence of the location-season pair value is used to populate the NaN values.

For the Rainfall, RainToday, and RainTomorrow attributes, we took a different approach to handling NaN values. For the Rainfall attribute, we replaced the NaN values with 0. If NaN values were filled with mean values, the model could not generalize better. For the RainToday and RainTomorrow features, we omitted the rows with NaN values because if we fill them with the most common class values, it could affect classification precision.

#### 4.1.3. Handling categorical variables

To train the model containing categorical features, it needs to be converted into a numerical format. For features RainToday and RainTomorrow, we used LabelEncoder that replaced the values Yes and No with 1 and 0 respectively. We could use LabelEncoder to convert directional features into numerical

format but for better generalization, we replaced direction with their respective degree value. {'N': 0, 'NNE': 22.5, 'NE': 45.0, 'ENE': 67.5, 'E': 90.0, 'ESE': 112.5, 'SE': 135.0, 'SSE': 157.5, 'S': 180.0, 'SSW': 202.5, 'SW': 225.0, 'WSW': 247.5, 'W': 270.0, 'WNW': 292.5, 'NW': 315.0, 'NNW': 337.5}

#### 4.1.4. Class balancing

An important factor affecting the performance of the model is the imbalance of the output classes. If the ratio of the values of the two classes is not close to 1, the model will be biased in favor of the class whose values matter more than those of the others. One of the simplest and most effective solutions is to oversample for class imbalance using SMOTE (Synthetic Minority Oversampling Technique) [30]. Originally, the ratio of class 0 to class 1 frequencies was about 5:1. Because of this, the performance of the model was not better with unseen data. Figure 7(a),(b) show the bar graph of the number of values of both classes before and after balancing the classes. Class equalization is performed only for training classification models.

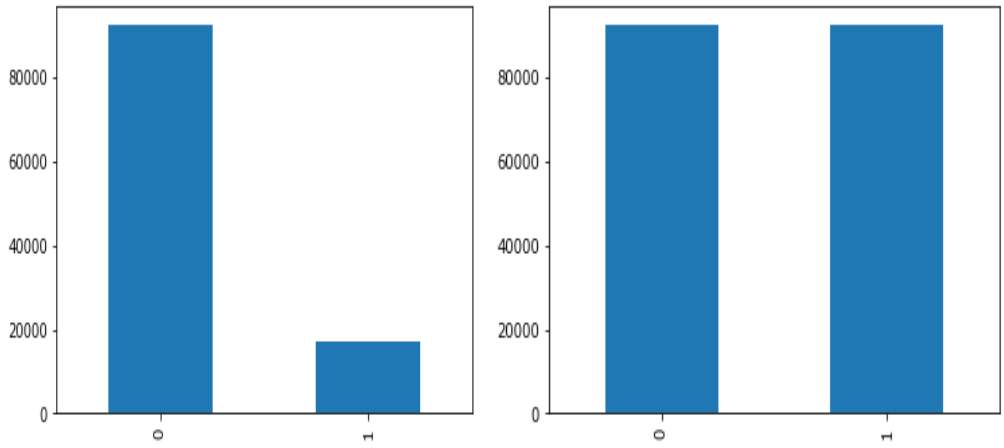


**Figure 7.** Target variable distribution (a) Before class balancing (b) After class balancing.

#### 4.1.5. Data normalization

Scaling the data is very important in performing a regression task. By scaling our variables, we can compare different variables with the same calibration. We used a standard scaler to normalize our characteristics. The standard scaler converts the feature values to a range of –3 to 3. Equation (1) represents the equation that the standard scaler uses to scale the values.

$$z = \frac{(xi - \mu)}{\sigma} \tag{1}$$

#### 4.1.6. Dataset capacity

After conducting exploratory data analysis and data cleaning, the dataset comprises 110 k rows and 19 features. The approach is divided into two parts: 1) forecasting the occurrence of rainfall and 2) estimating the amount of rainfall. For forecasting the occurrence of rainfall, which is a classification task, the training data consists of approximately 147 k rows and the testing data consists of 36 k rows.

The number of rows is higher than the actual dataset because the SMOTE technique was applied to oversample the data and balance both classes for classification tasks. For predicting the amount of rainfall, which is a regression task, the training data consists of 87 k rows and the testing data consists of 22 k rows. In this approach, we did not use any oversampling technique for regression.

### *4.2. Forecasting occurrence of rainfall*

The task here is to predict the occurrence of precipitation in two classes, i.e., whether it will rain tomorrow or not. The above task is to implement a classification approach using various features and their corresponding target values from a given dataset. The classification approach is divided into three parts as: 1) Preparing data for classification. 2) Fitting the training data to train a classification model, and 3) Evaluating the model performance. Figures 8 and 9 show flow of the overall implementation of the classification approach. The flow of the overall implementation of the classification approach.

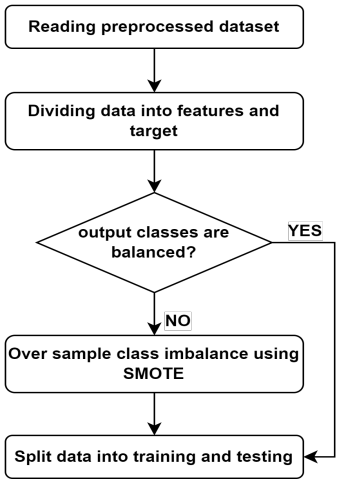


**Figure 8.** Preparing data for classification.

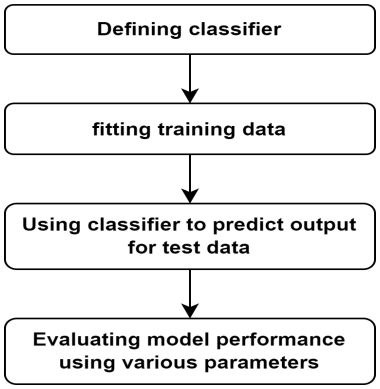


**Figure 9.** Fitting data for classification.

For forecasting the occurrence of rainfall, we have implemented four classification models as follows:

**XGBoost Classifier:** XGBoost [31] stands for eXtreme Gradient Boosting which is a fast and effective boosting algorithm based on gradient boosted decision tree algorithm. XGBoost uses a finer regularization technique, Shrinkage, and Subsampling column to prevent over-submerging, and this is one of the differences in gradient development. The XGBoost classifier takes a 2-dimensional array of training features and target variables as input for training the classification model.

**Random Forest Classifier:** Random forest or random decision forest [32] is an integrated learning method for classification, regression, and other activities that work by building a pile of decision trees during training. It basically creates a set of decision trees from a randomly selected small set of the training set and then collects votes from different decision trees to determine the final forecast. For classification tasks, random forest clearing is a class selected by many trees. Random Forest classifier also takes a 2-dimensional array of training features and target variables as input for training the classification model.

**Kernel SVM Classifier:** Support Vector Machines [33] are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis. The main motivation of SVM is to create the best decision-making limit that can divide two or more classes so that we can accurately place data points in the correct class for which various kernels are used. We have chosen the Gaussian Radial Basis Function (RBF) as the kernel for training our SVM model as rainfall data is non-linear data. Equation (2) represents the gaussian radial basis function used by the support vector machine.

$$F(x, x_j) = exp(-y * ||x - x_j||^2 \tag{2}$$

**LSTM Classifier:** An LSTM or Long-Short-Term-Memory classifier [34] is an artificial recurrent neural network that has both feedforward communication and feedback, and is often used to classify and make predictions over time-series data. For training the LSTM classifiers a different data format needs to be supplied in which the data must first be converted to a 3-D array according to the provided format: (number of samples, time steps, number of features). To prevent overfitting and visualize the training progress, callbacks are passed as parameters while training the prediction model. In our approach we have used two callbacks that are:

- **ReduceLRonPlateau:** It reduces the learning rate by 'factor' times passed as an argument if the metric has stopped improving for the 'patience' number of epochs. Reducing the learning rate often benefits the model.

- **EarlyStopping:** This will stop training if the monitored metric has stopped improving for the 'patience' number of epochs.

Figure 10(a),(b) show the layout of the LSTM neural network trained and unrolled RNN for a timestamp of 15 days respectively.

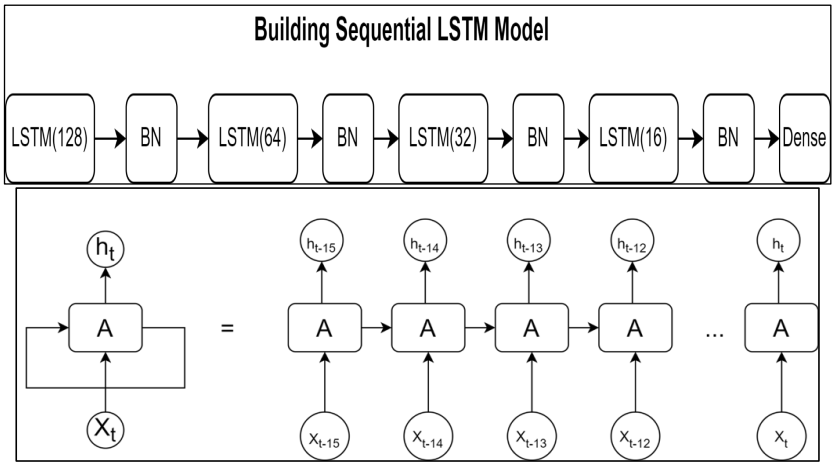


**Figure 10.** The layout of the LSTM neural network trained and unrolled RNN for a timestamp. (a) Sequential LSTM model, and (b) Unrolled RNN for the timestamp of 15 days.

To predict the occurrence of rainfall tomorrow, the designed algorithm is then shown in Algorithm 1.

**Algorithm 1:** Algorithm for Classification

**Input :** Rainfall Forecasting Dataset
I = [‘MinTemp’, ‘MaxTemp’, ‘Rainfall’, ‘WindGustDir’, ‘WindGustSpeed’, ‘WindDir9am’, ‘WindDir3pm’, ‘WindSpeed9am’, ‘WindSpeed3pm’, ‘Humidity9am, Humidity3pm’, ‘Pressure9am’, ‘Pressure3pm’, ‘Temp9am’, ‘Temp3pm’, ‘RainToday’, ’RainTomorrow’]
**Output:** Yes, No

1 Preprocess the input data and divide it into features and targets.
2 Balance output classes using SMOTE.
3 Scale the data using Standard Scaler.
4 Define classification model.
5 Train the classifier according to the defined parameters.

### *4.3. Projecting the amount of rainfall in millimeters*

Classification is one of the steps in precipitation forecasting. It tells you whether it will be a sunny day or whether you will need an umbrella throughout the day, since the forecast only says that it will rain at any time. However, another important aspect is that predicting the amount of rainfall based on features of the flow is a good prediction because it helps us make decisions such as whether to leave the station or stay home because it will rain heavily, whether to hold back the water stored in the reservoir or release some water from the reservoir because heavy rain is predicted in the watershed, and more. Therefore, this part deals with different regression techniques that are used to forecast the amount of rainfall in millimeters.

The regression task is also divided into three parts:

1) Preparing data for the regression model.

2) Training a regression model.

3) Evaluating the regression model.

For projecting the amount of rainfall, we have implemented various regression algorithms including:

- Multiple Linear Regression
- XGBoostRegressor
- Polynomial Regression
- Random Forest Regression
- LSTM-based Deep Learning Model

**Multiple Linear Regression:** Linear Regression [35] is a supervised learning to perform a regression task. This regression method detects the linear relationship between the input features and the target variable.

$$\hat{y} = b_0 + b_1 x_1 + b_2 x_2 + ... + b_k x_k \tag{3}$$

Equation (3) is the statistical equation used for prediction by the Multiple Linear regression algorithm. In order to achieve the best fit line the model aims to predict $\hat{y}$ so that the difference in error between the predicted value and the real value is as small as shown in Eq (4).

$$mininmize \frac{1}{n} \sum_{i=1}^{n} (pred_i - y_i)^2 \tag{4}$$

**XGBoostRegressor:** XGBoost [36] stands for eXtreme Gradient Boosting which is a fast and effective boosting algorithm based on gradient boosted decision tree algorithm. XGBoost's objective function consists of a loss function and a regularization term. It tells us about the difference between real values and predicted values, i.e, how far the model results are from the real values. We used reg: linear as the XGBoost loss functions to perform the regression task.

**Polynomial Regression:** Polynomial regression [37] is a special case of linear regression in which the correlation between the independent variable x and the target variable y is modeled as an nth polynomial degree of x. This regression technique is used to identify a curvilinear relationship between independent and dependent variables.

$$\hat{y} = b_0 + b_1 x_1 + b_2 x_1^2 + b_3 x_1^3 + ... + b_n x_1^n \tag{5}$$

Equation (5) represents the statistical equation used for prediction by the Polynomial regression algorithm. In order to achieve the best fit line, the model aims to predict $\hat{y}$ so that the difference in error between the predicted value and the real value is as small as shown in Eq (4).

**Random Forest Regression:** Random Forest or random decision forest [38] is an integrated learning method for classification, regression and other activities that work by building a pile of decision trees during training. Random Forest has many decision trees as base learning models. The end results of the random forest is the mean of all the results of decision trees.

**LSTM based Deep Learning Model:** An LSTM or Long-Short-Term-Memory classifier [39] is an artificial recurrent neural network that has both feedforward communication and feedback. LSTM for regression is typically a time series problem. For training the LSTM regression model data must first be converted to 3-D array according to the provided format: (number of samples, time steps, number of features). To prevent overfitting and visualize [40] the training progress, callbacks are passed as parameters while training the prediction model. In our approach we have used two callback that are:

- **ReduceLRonPlateau:** It reduces learning rate by "factor" times passed as argument if metric has stopped improving for "patience" number of epochs. Reducing learning rate oftens benefits the model.
- **EarlyStopping:** This will stop training if the monitored metric has stopped improving for "patience" number of epochs.

Algorithm 2 is used for forecasting the amount of precipitation, as is given here.

**Algorithm 2:** Algorithm for Regression

**Input :** Rainfall Amount Prediction Dataset
I = ['MinTemp', 'MaxTemp', 'Rainfall', 'WindGustDir', 'WindGustSpeed', 'WindDir9am', 'WindDir3pm', 'WindSpeed9am', 'WindSpeed3pm', 'Humidity9am, Humidity3pm', 'Pressure9am', 'Pressure3pm', 'Temp9am', 'Temp3pm', 'RainToday']
**Output:** Amount of precipitation in millimeters

1 Preprocess the input data and divide into features and target.
2 Scale the data using Standard Scaler.
3 Scale the data using Standard Scaler.
4 Define regression model.
5 Train regressor according to defined parameters.

## 5. Evaluation results

There are numerous evaluation metrics that can be used for measuring model performance. In our paper we have evaluated our machine learning and deep learning models on confusion matrix, accuracy, precision, recall and F1-score for classification models and mean absolute error, mean squared error and r2 score for regression models.

- **For Classification**

**Confusion Matrix:** Confusion matrix yields the output of a classification model in a matrix format. The matrix is defined as shown in Table 4.

**Table 4.** Confusion matrix.

| | | Predicted Values | |
|---|---|---|---|
| | | Positive | Negative |
| Actual Values | Positive | TP | FN |
| | Negative | FP | TN |

A list of evaluation metrics used for evaluating trained classifiers is stated in Eqs (6)–(9).

$$Accuracy = \frac{No.of correct predictions}{Totalno.of predictions} \tag{6}$$

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{7}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{8}$$

$$F1 = \frac{2 * (Precision * Recall)}{(Precision + Recall)} \tag{9}$$

A) **XGBoost Classifier:** Figure 11 and Table 5 represent the confusion matrix and the classification report for the XGBoost classifier. The classification report shows that the precision, recall, and f1-score for both the classes, i.e., rain and no rain are 96, 89, 92%, and 89, 96, 93% respectively.

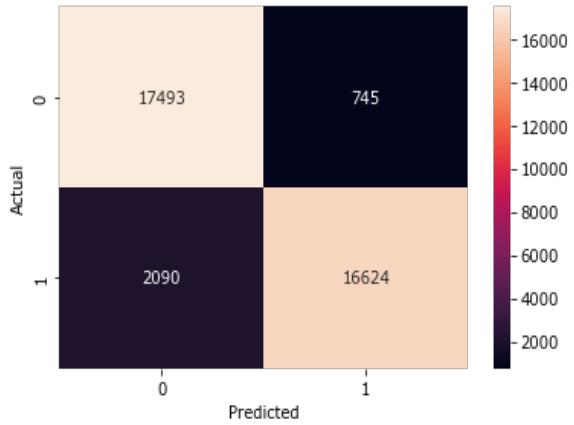


**Figure 11.** Confusion matrix for XGBoost Classifier.

**Table 5.** Classification report for XGBoost Classifier.

| | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| 0 | 89% | 96% | 93% | 18238 |
| 1 | 96% | 89% | 92% | 18714 |
| Accuracy | | | 92% | 36952 |
| Macro Avg | 93% | 92% | 92% | 36952 |
| Weighted Avg | 93% | 92% | 92% | 36952 |

B) **Kernel SVM:** Figure 12 and Table 6 represent the confusion matrix and the classification report for the Support Vector Machine Classifier with a radial basis kernel function. The classification report shows that the precision, recall, and f1-score for both the classes, i.e., rain and no rain are 81, 82, 92%, and 82, 81, 81% respectively.

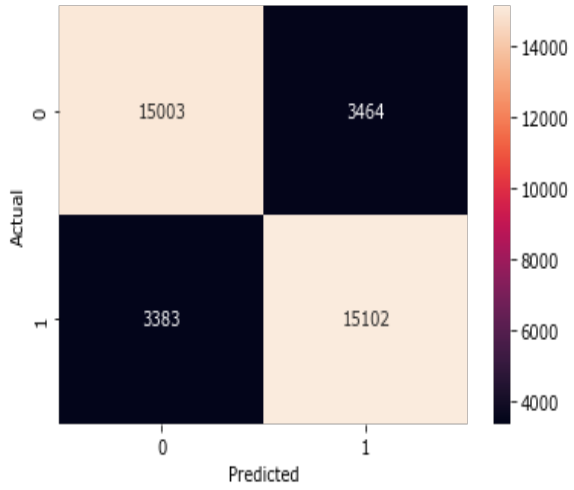


**Figure 12.** Confusion matrix for SVM Classifier.

**Table 6.** Classification report for SVM Classifier.

| | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| 0 | 82% | 81% | 81% | 18467 |
| 1 | 81% | 82% | 82% | 18485 |
| Accuracy | | | 81% | 36952 |
| Macro Avg | 81% | 81% | 81% | 36952 |
| Weighted Avg | 81% | 81% | 81% | 36952 |

C) **LSTM Classifier:** Figure 13 and Table 7 represent the confusion matrix and the classification report for the LSTM classifier. The classification report shows that the precision, recall, and f1-score for both the classes, i.e., rain and no rain are 86, 87, 86%, and 87, 85, 86% respectively.

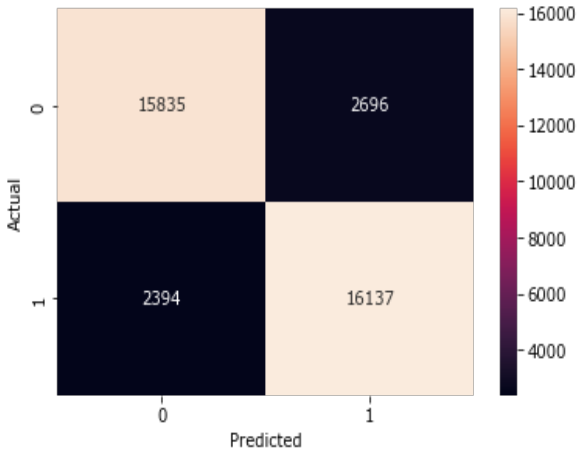


**Figure 13.** Confusion matrix for LSTM Classifier.

**Table 7.** Classification report for LSTM Classifier.

| | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| 0 | 87% | 85% | 86% | 18531 |
| 1 | 86% | 87% | 86% | 18531 |
| Accuracy | | | 86% | 37062 |
| Macro Avg | 86% | 86% | 86% | 37062 |
| Weighted Avg | 86% | 86% | 86% | 37062 |

D) **Random Forest Classifier:** Figure 14 and Table 8 represent the confusion matrix and the classification report for the Random Forest classifier. The classification report shows that the precision, recall, and f1-score for both the classes, i.e., rain and no rain are 92, 91, 91%, and 91, 92, 91% respectively.

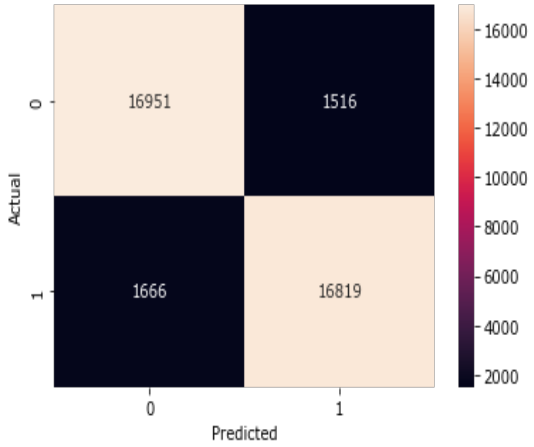


**Figure 14.** Confusion matrix for Random Forest Classifier.

**Table 8.** Classification report for Random Forest Classifier.

| | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| 0 | 91% | 92% | 91% | 18467 |
| 1 | 92% | 91% | 91% | 18485 |
| Accuracy | | | 91% | 36952 |
| Macro Avg | 91% | 91% | 91% | 36952 |
| Weighted Avg | 91% | 91% | 91% | 36952 |

Table 9 and Figure 15 represent the comparison of the evaluation results of the employed classification models. It is visible that the XGBoost classifier surpasses all the other classifiers with an accuracy (92.2%), precision (95.6%), and F1-Score (91.9%). However, Random Forest Classifier provided the best recall (91.2%) over the other classifiers. On the other hand, Kernel SVM with Radial Basis Function performed the worst among the four classifiers with an accuracy (81.4%), precision (80.9%), recall (82.1%), and F1-Score (81.5%), respectively.

**Table 9.** Evaluation results for Classification.

| Approach | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| XGBoost Classifier | 92.2% | 95.61% | 88.4% | 91.87% |
| Random Forest Classifier | 91.3% | 91.42% | 91.16% | 91.29% |
| LSTM Classifier | 85.84% | 85.81% | 85.88% | 85.85% |
| Kernel SVM Classifier (rbf kernel) | 81.44% | 80.86% | 82.06% | 81.45% |

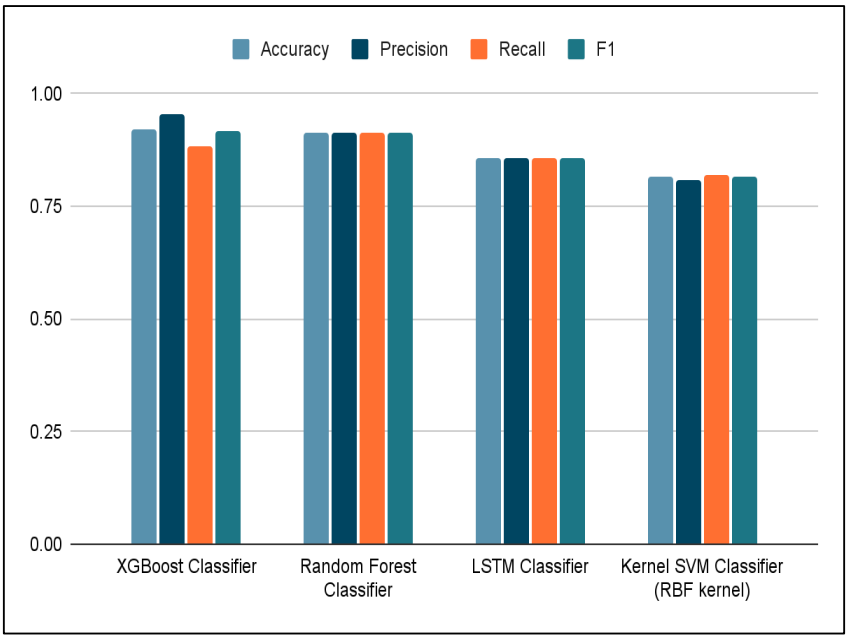


**Figure 15.** Comparing Evaluation Results for Classification.

**Table 10.** State-of-the-art Results.

| State-of-the-Art Approach | Best Accuracy |
|---|---|
| Oswal (2019) [41] | 84% |
| He (2021) [42] | 82% |

From Table 10, we can confirm that our classification method is significantly superior to the various modern methods that use the Australian Kaggle rain dataset in terms of accuracy.

After undergoing several data processing techniques, a cleaned dataset with approximately 110 thousand rows is utilized for the classification task. Upon analyzing the class imbalance in the "RainTomorrow" feature, it was found to be highly skewed towards the "No" rainfall class, with a 9:1 ratio of "No" to "Yes" rainfall values. Models trained on this highly skewed data produced precision and accuracy values between 0.80 to 0.85. To address this imbalance, we used the SMOTE (Synthetic Minority Oversampling Technique) to balance both classes. This increased the data from 110 k rows to 183 k rows. The data was then divided into training and testing sets accordingly. The precision was improved to 95% and accuracy increased to 92% after using a balanced data set. This improvement in

accuracy is attributed to the use of optimized data preprocessing and cleaning techniques, feature scaling techniques, data normalization techniques, training parameters, and train-test split ratios.

- **For Regression**

The evaluation metrics employed for evaluating the trained regression models are Mean Absolute Error (MAE), Mean Squared Error (MSE), and R2 Score ($R^2$) as stated in Eqs (10)–(12).

$$MAE = (\frac{1}{n}) \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{10}$$

$$MSE = (\frac{1}{n}) \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{11}$$

$$R^2 = 1 - \frac{SS_{Res}}{SS_{TOT}} \tag{12}$$

Table 11 and Figure 16(a),(b) present the evaluation results of Multiple Linear regression, XGBoostRegressor, Polynomial regression, Random Forest regression, and LSTM-based deep learning model.

**Table 11.** Evaluation Results for Regression.

| Approach | MAE | MSE | R2 Score |
|---|---|---|---|
| Multiple Linear Regression | 0.124 | 0.039 | 0.754 |
| XGBoost Regression | 0.121 | 0.04 | 0.737 |
| Polynomial Regression | 0.120 | 0.039 | 0.746 |
| Random Forest Regression | 0.117 | 0.036 | 0.760 |
| LSTM based deep learning model | 0.104 | 19.83 | 0.70 |

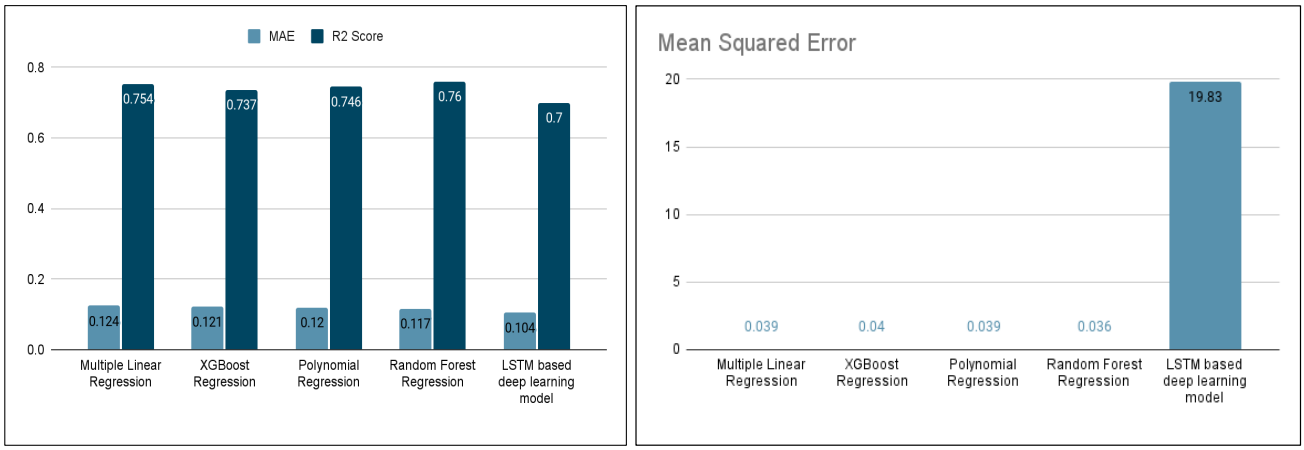


**Figure 16.** Comparing (a) MAE, R2 Score (b) MSE for Regression.

Here, it is observed that the Random Forest regressor outperformed all the other regression models with a mean absolute error of 0.117, mean squared error of 0.036, and R2 score of 0.76. On the other hand, the LSTM based deep learning model performed the worst among the five regression models with a mean absolute error of 0.104, mean squared error of 19.83, and R2 score of 0.70.

- **Novelty and discussions:** Data processing is a critical aspect of building a machine learning or deep learning model. Instead of filling missing values with the mean or mode of the entire dataset, the proposed solution uses seasonal and location-based data filling to fill numeric values with the mean and categorical values with the mode. LSTM-based models are often considered to be the best for modeling relationships in time series data. However, in the proposed method, the ensemble learning-based random forest model outperforms the LSTM model in both the classification and regression tasks. Random forest leverages class votes from each decision tree it grows, making it less susceptible to the impact of an inconsistent dataset and less prone to overfitting. In contrast, neural network models require more consistent data to make accurate predictions and may not perform well with inconsistent datasets.

- **Limitations:** Data collection is a major obstacle to accurate rainfall forecasting. Real-time weather data is hard to obtain and must be gathered from multiple meteorological stations, resulting in inconsistent and abnormal data due to incompatible data types and measurements. Therefore, we dropped the "Evaporation", "Sunshine", "Cloud9am", and "Cloud3pm" columns while handling NaN values as they each had 50% NaN values. Although these features can have a strong correlation with the rainfall value.

## 6. Conclusions and future works

In this work, we implemented different machine learning and deep learning models for predicting the occurrence of rainfall the next day and for predicting the amount of rainfall in millimeters. We used the Australian rainfall dataset in this work. The dataset contains weather data from 49 locations on the Australian continent. In this work, we managed to obtain more accurate results than various state-of-the-art approaches. We achieved an accuracy of 92.2%, precision of 95.6%, F1 score of 91.9%, and recall of 91.1% for next day rainfall prediction. For the prediction of the amount of precipitation in millimeters, we obtained a mean absolute error of 0.117, a mean square error of 0.036, and an R2 value of 0.76. To obtain the above results, we applied several data preprocessing techniques, such as. analyzing null values, populating null values with seasonal and location-specific values, removing outliers using the interquartile range approach, selecting features by analyzing the correlation matrix, converting categorical values to numerical values to use the data for training the predictive model, balancing the class of target variables for the classification task, and normalizing the data using standard scalars for the regression task. We also compared different statistical machine learning and deep learning models for both the classification and regression tasks. This work uses publicly available datasets for training classification and regression models. Satellite and radar data can be used for training models and predicting rainfall in real time.

In the future, further robustness can be achieved with the use of more recent and accurate data collected from meteorological departments. Incorporating additional features, such as "time of rainfall", "time of strongest wind gusts", "relative humidity at two points of time in a day" and

"atmospheric pressure at sea level", could greatly enhance the model. These additional features are highly correlated with the "RainTomorrow" and "Rainfall" features. If the proposed model is trained with more features, it could lead to an increase in model performance. We would also like to work on transfer learning models to get better results.

## Acknowledgments

This work is partially supported by Western Norway University of Applied Sciences, Bergen, Norway.

## Conflict of interest

The authors declare there is no conflict of interest.

## References

1. R. K. Gupta, A. Jain, J. Wang, V. P. Singh, S. Bharti, *Artificial intelligence of things for weather forecasting and climatic behavioral analysis*, IGI Global, (2022), 1–277. https://doi.org/10.4018/978-1-6684-3981-4
2. G. Kulesa, Weather and aviation: How does weather affect the safety and operations of airports and aviation, and how does FAA work to manage weather-related effects?, in *The Potential Impacts of Climate Change on Transportation Workshop*, (2002), 1–10. https://doi.org/10.1016/j.media.2013.04.012
3. *Economic Losses Due to Climatic Changes*, 2022. Available from: https://www.mnw.cn/news/fj/.
4. V. Sharma, S. van de Graaff, B. Loechel, D. Franks, Extractive resource development in a changing climate: learning the lessons from extreme weather events in Queensland, Australia: Final report, 2012. Available from: http://hdl.handle.net/102.100.100/101882?index=1.
5. J. Abbot, J. Marohasy, Using artificial intelligence to forecast monthly rainfall under present and future climates for the Bowen Basin, Queensland, Australia, *Int. J. Sustainable Dev. Plann.*, **10** (2015), 66–75.
6. A. Zhong, D. Hudson, O. Alves, G. Wang, H. Hendon, Predictive Ocean Atmosphere Model for Australia (POAMA), in *10th EMS Annual Meeting*, (2010), 2010–2016.
7. E. Vamsidhar, K. V. S. R. P. Varma, P. S. Rao, R. Satapati, Prediction of rainfall using backpropagation neural network model, *Int. J. Comput. Sci. Eng.*, **2** (2010), 1119–1121.
8. A. Y. Barrera-Animas, L. O. Oyedele, M. Bilal, T. D. Akinosho, J. M. D. Delgado, L. A. Akanbi, Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting, *Mach. Learn. Appl.*, **7** (2022), 100204. https://doi.org/10.1016/j.mlwa.2021.100204
9. A. J. Hill, R. S. Schumacher, Forecasting excessive rainfall with random forests and a deterministic convection-allowing model, *Weather Forecast.*, **36** (2021) 1693–1711. https://doi.org/10.1175/WAF-D-21-0026.1

10. S. A. Fayaz, M. Zaman, M. A. Butt, Knowledge discovery in geographical sciences—A systematic survey of various machine learning algorithms for rainfall prediction, in *International Conference on Innovative Computing and Communications*, Springer, **1388** (2022), 593–608. https://doi.org/10.1007/978-981-16-2597-8_51

11. X. Xing, C. Wu, J. Li, X. Li, L. Zhang, R. He, Susceptibility assessment for rainfall-induced landslides using a revised logistic regression method, *Nat. Hazards*, **106** (2021), 97–117. https://doi.org/10.1007/s11069-020-04452-4

12. M. Marjanovic, M. Krautblatter, B. Abolmasov, U. Duric, C. Sandic, V. Nikolic, The rainfall-induced landsliding in Western Serbia: A temporal prediction approach using Decision Tree technique, *Eng. Geol.*, **232** (2018), 147–159. https://doi.org/10.1016/j.enggeo.2017.11.021

13. X. Zhou, H. Wen, Z. Li, H. Zhang, W. Zhang, An interpretable model for the susceptibility of rainfall-induced shallow landslides based on SHAP and XGBoost, *Geocarto Int.*, (2022), 1–27. doilinkhttps://doi.org/10.1080/10106049.2022.2076928

14. A. Jain, R. Gairola, S. Jain, A. Arora, Thwarting spam on Facebook: Identifying spam posts using machine learning techniques, *Res. Anthol. Mach. Learn. Tech. Methods Appl.*, (2022), 693–713. https://doi.org/10.4018/978-1-6684-6291-1.ch037

15. N. T. Jani, R. K. Gupta, S. K. Bharti, A. Jain, Advancements in weather forecasting with deep learning, *Artif. Intell. Things Weather Forecast. Clim. Behav. Anal.*, (2022), 75–86. https://doi.org/10.4018/978-1-6684-3981-4.ch006

16. K. C. Luk, J. E. Ball, A. Sharma, An application of artificial neural networks for rainfall forecasting, *Math. Comput. model.*, **33** (2001), 683-693. https://doi.org/10.1016/S0895-7177(00)00272-7

17. K. Abhishek, M. P. Singh, S. Ghosh, A. Anand, Weather forecasting model using artificial neural network, *Procedia Technol.*, **4** (2012), 311–318. https://doi.org/10.1016/j.protcy.2012.05.047

18. K. Abhishek, R. Ranjan, S. Kumar, A rainfall prediction model using artificial neural network, in *2012 IEEE Control and System Graduate Research Colloquium*, (2012), 82–87. doilinkhttps://doi.org/10.1109/ICSGRC.2012.6287140

19. T. Saba, A. Rehman, J. S. AlGhamdi, Weather forecasting based on hybrid neural model, *Appl. Water Sci.*, **7** (2017), 3869–3874. https://doi.org/10.1007/s13201-017-0538-0

20. M. Biswas, T. Dhoom, S. Barua, Weather forecast prediction: An integrated approach for analyzing and measuring weather data, *Int. J. Comput. Appl.*, **975** (2018), 8887. https://doi.org/10.5120/ijca2018918265

21. C. Z. Basha, N. Bhavana, P. Bhavya, V. Sowmya, Rainfall prediction using machine learning & deep learning techniques, in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, (2020), 92–97. https://doi.org/10.1109/ICESC48915.2020.9155896

22. A. Doroshenko, V. Shpyg, R. Kushnirenko, Machine learning to improve numerical weather forecasting, in *2020 IEEE 2nd International Conference on Advanced Trends in Information Theory (ATIT)*, (2020), 353–356. https://doi.org/10.1109/ATIT50783.2020.9349325

23. N. K. A. Appiah-Badu, Y. M. Missah, L. K. Amekudzi, N. Ussiph, T. Frimpong, E. Ahene, Rainfall prediction using machine learning algorithms for the various ecological zones of Ghana, *IEEE Access*, **10** (2021), 5069–5082. https://doi.org/10.1109/ACCESS.2021.3139312
24. M. Raval, P. Sivashanmugam, V. Pham, H. Gohel, A. Kaushik, Y. Wan, Automated predictive analytics tool for rainfall forecasting, *Sci. Rep.*, **11** (2021), 1–13. https://doi.org/10.1038/s41598-021-95735-8
25. W. M. Ridwan, M. Sapitang, A. Aziz, K. F. Kushiar, A. N. Ahmed, A. El-Shafie, Rainfall forecasting model using machine learning methods: Case study Terengganu, Malaysia, *Ain Shams Eng. J.*, **12** (2021), 1651–1663. https://doi.org/10.1016/j.asej.2020.09.011
26. F. R. Adaryani, S. J. Mousavi, F. Jafari, Short-term rainfall forecasting using machine learning-based approaches of PSO-SVR, LSTM and CNN, *J. Hydrol.*, **614** (2022), 128463. https://doi.org/10.1016/j.jhydrol.2022.128463
27. S. Fahad, F. Su, S. U. Khan, M. R. Naeem, K. Wei, Implementing a novel deep learning technique for rainfall forecasting via climatic variables: An approach via hierarchical clustering analysis, *Sci. Total Environ.*, **854** (2023), 158760. https://doi.org/10.1016/j.scitotenv.2022.158760
28. *Kaggle Dataset*, Rain in Australia, 2022. Available from: https://www.kaggle.com/datasets/jsphyg/weather-dataset-rattle-package.
29. P. Pedamkar, Statistics for machine learning, 2022. Available from: https://www.educba.com/statistics-for-machine-learning/.
30. N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, SMOTE: synthetic minority over-sampling technique, *J. Artif. Intell. Res.*, **16** (2002), 321–357. https://doi.org/10.1613/jair.953
31. Y. Han, J. Kim, D. Enke, A machine learning trading system for the stock market based on N-period Min-Max labeling using XGBoost, *Expert Syst. Appl.*, **211** (2023), 118581. https://doi.org/10.1016/j.eswa.2022.118581
32. M. Esteve, J. Aparicio, J. J. Rodriguez-Sala, J. Zhu, Random Forests and the measurement of super-efficiency in the context of Free Disposal Hull, *Eur. J. Oper. Res.*, **304** (2023), 729–744. https://doi.org/10.1016/j.ejor.2022.04.024
33. X. Xie, Y. Li, S. Sun, Deep multi-view multiclass twin support vector machines, *Informa. Fusion*, **91** (2023), 80–92. https://doi.org/10.1016/j.inffus.2022.10.005
34. V. H. Pereira-Ferrero, L. P. Valem, D. C. G. Pedronette, Feature augmentation based on manifold ranking and LSTM for image classification, *Expert Syst. Appl.*, **213** (2023), 118995. https://doi.org/10.1016/j.eswa.2022.118995
35. A. Banik, T. K. Bandyopadhyay, S. K. Biswal, V. Panchenko, S. Garhwal, Comparative performance assessment of multi-linear regression and artificial neural network for prediction of permeate flux of disc-shaped membrane, *Intelligent Computing and Optimization, Lecture Notes in Networks and Systems*, Springer, **569** (2023), 24–33. https://doi.org/10.1007/978-3-031-19958-5_3

36. J. Dong, W. Zeng, L. Wu, J. Huang, T. Gaiser, A. K. Srivastava, Enhancing short-term forecasting of daily precipitation using numerical weather prediction bias correcting with XGBoost in different regions of China, *Eng. Appl. Artif. Intell.*, **117** (2023), 105579. https://doi.org/10.1016/j.engappai.2022.105579

37. K. Sarkodie, A. Fergusson-Rees, M. Abdulkadir, N. Y. Asiedu, Gas-liquid flow regime identification via a non-intrusive optical sensor combined with polynomial regression and linear discriminant analysis, *Ann. Nucl. Energy*, **180** (2023), 109424. https://doi.org/10.1016/j.anucene.2022.109424

38. F. Ricardo, P. Ruiz-Puentes, L. H. Reyes, J. C. Cruz, O. Alvarez, D. Pradilla, Estimation and prediction of the air-water interfacial tension in conventional and peptide surface-active agents by random forest regression, *Chem. Eng. Sci.*, **265** (2023), 118208. https://doi.org/10.1016/j.ces.2022.118208

39. J. Chen, Y. Zhang, J. Wu, W. Cheng, Q. Zhu, SOC estimation for lithium-ion battery using the LSTM-RNN with extended input and constrained output, *Energy*, **262** (2023), 125375. https://doi.org/10.1016/j.energy.2022.125375

40. S. Iyer, A. Jain, J. Wang, *Handbook of research on lifestyle sustainability and management solutions using AI, big data analytics, and visualization*, IGI Global, (2022), 1–411. https://doi.org/10.4018/978-1-7998-8786-7

41. N. Oswal, Predicting rainfall using machine learning techniques, preprint, arXiv:1910.13827.

42. Z. He, Rain prediction in Australia with active learning algorithm, in *2021 International Conference on Computers and Automation (CompAuto)*, (2021), 14–18. https://doi.org/10.1109/CompAuto54408.2021.00010


AIMS Press