FORECAST OF RAINFALL QUANTITY AND ITS VARIATION USING ENVIRONMENTAL FEATURES

A PROJECT REPORT

Submitted by

CB.EN.U4CSE15417	**Harsha Vardhini V**
CB.EN.U4CSE15435	**Preetham G**
CB.EN.U4CSE15438	**Raghul S**
CB.EN.U4CSE15455	**Vasisht S**

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

AMRITA SCHOOL OF ENGINEERING, COIMBATORE

AMRITA VISHWA VIDYAPEETHAM

COIMBATORE - 641112

April 2019

AMRITA VISHWA VIDYAPEETHAM

AMRITA SCHOOL OF ENGINEERING, COIMBATORE, 641112

BONAFIDE CERTIFICATE

This is to certify that the project report entitled **"Forecast of Rainfal Quantity and its Variation using Environmental Features"** submitted by HARSHA VARDHINI V (CB.EN.U4CSE15417), PREETHAM G (CB.EN.U4CSE15435), RAGHUL S (CB.EN.U4 CSE15438) and VASISHT S (CB.EN.U4CSE15455) in partial fulfillment of the requirements for the award of the Degree **Bachelor of Technology in Computer Science and Engineering** is a bonafide record of the work carried under our guidance and supervision at Department of Computer Science and Engineering, Amrita School of Engineering, Coimbatore.

PROJECT GUIDE
Mr Dayanand Vinod
Assistant Professor
Dept. of Computer Science and Engg.

CHAIRPERSON
Dr Latha Parameswaran
Dept. of Computer Science and Engg.

This project was evaluated by us on : .....

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We express our gratitude to our beloved **Satguru Sri Mata Amritanandamayi Devi** for providing a bright academic climate at this university, which has made this entire task appreciable. This acknowledgement is intended to be a thanksgiving measure to all those people involved directly or indirectly with our project. We want to thank our Vice-Chancellor **Dr Venkat Rangan P** and **Dr Sasangan Ramanathan**, Dean Engineering of Amrita Vishwa Vidyapeetham for providing us with the necessary infrastructure required for completion of the project.

We express our thanks to **Dr Latha Parameswaran**, Chairperson of Department of Computer Science Engineering, **Dr P Bagavathy Sivakumar** and **Prof Prashant R Nair**, Vice-Chairpersons of the Department of Computer Science and Engineering for their valuable help and support during our study. We express our gratitude to our guides, **Mr Dayanand Vinod** and External Guide, for their guidance, support and supervision.

We feel incredibly grateful to **Dr D Venkataraman, Mrs Nalina Devi K, Mrs Manjusha R** and **Mr Arun Kumar C** for their feedback and encouragement which helped us to complete the project. We also thank the staff of the Department of Computer Science Engineering for their support.

We want to extend our sincere thanks to our family and friends for helping and motivating us during the project. Finally, we would like to thank all those who have helped, guided and encouraged us directly or indirectly during the project work. Last but not least, we thank God for His blessings which made our project a success.

ABSTRACT

Rainfall plays a crucial role in the lives of an ordinary man. Developing a prediction model that captures sudden fluctuations in rainfall has always been a challenging task. The project aims at developing three models which predict monthly rainfall for all districts in Tamil Nadu, India and also drawing a district-wise comparison among them to find the best model for prediction. The models developed are District-Specific Model, Cluster-Based Model and Generic-Regression Model. The District-Specific Model trains on data from a particular district, the Cluster-Based Model groups districts based on the climatic conditions and trains on data from a particular cluster and the Generic-Regression Model trains on combined data from all the districts. The project also aims at finding the monthly variation of rainfall across geographical regions. Based on the result of the comparison between the models, the best model is then trained using ensemble regression algorithms such as Random Forest Regression, Extra Trees Regression, Bagging Regression, AdaBoost Regression, Gradient Boosting Regression and Extreme Gradient Boosting Regression. The predicted results of the ensemble regression models mentioned above are then combined using ensemble techniques to such Simple Averaging, Weighted Averaging and Stacking to build a Hybrid Ensemble Regression Model. The rainfall data considered for analysis is a Time-Series data hence Long Short-Term Memory Neural Network was used to draw a comparison between the preliminary regression model, plain ensemble regression model, hybrid ensemble regression model and the LSTM Neural Network.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ACRONYM	MEANING
ADR	AdaBoost Regressor
ANN	Artificial Neural Network
BAR	Bagging Regressor
BPNN	Back Propagation Neural Network
C&RT	Classification and Regression Tree
CBPNN	Cascade-Forward Back Propagation Neural Network
CC	Correlation Coefficient
COE	Coefficient of Efficiency
DTDNN	Distributed Time Delay Neural Network
DTR	Decision Tree Regression
EEMD	Ensemble Empirical Mode Decomposition
ELM	Extreme Learning Machine
ETR	Extra Trees Regressor
GBR	Gradient Boosting Regressor
GDP	Gross Domestic Product
GRNN	Generalised Regression Neural Network
GRU	Gated Recurrent Unit
HNN	Hybrid Neural Network
KNN-R	K-Nearest Neighbours Regression
LSTM	Long Short-Term Memory
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MLP	Multiple Layer Perceptron
MLP-FFN	Multiple Layer Perceptron Feed Forward Network
MLR	Multiple Linear Regression
MSE	Mean Squared Error
NARX	Nonlinear Autoregressive Exogenous Network
NB	Naive Bayes
NMSE	Normalised Mean Squared Error
PR	Polynomial Regression
PRC	Pearson Correlation Coefficient
RBF	Radial Basis Function
RBFNN	Radial Basis Function Neural Network
RFR	Random Forest Regression
RMSE	Root Mean Squared Error

ACRONYM	MEANING
ROS-RVFL	Regularized Online Sequential RVFL
RVFL	Random Vector Functional Link Networks
SD	Standard Deviation
SLFN	Single Layer Feed Forward Neural Network
SVM	Support Vector Machine
SVR	Support Vector Regression
WNN	Wavelet Neural Network
WTTC	World Travel and Tourism Council
XGBR	Extreme Gradient Boosting Regressor

1 INTRODUCTION

1.1 Background

Agriculture is the backbone of India's economy. According to the survey conducted by the WTTC [1], agriculture contributed approximately 500 billion US Dollars to the Indian economy in the year 2016, which is roughly 24% of India's GDP and engages 59% of India's human resources. Indian agriculture is sundry, ranging from poor farm villages to evolved farms using present-day agricultural technologies. Rainfall is the central source of water for the country's agricultural land. It is a boon if the rainfall quantity is in the right amount and a bane if the rainfall is too low or too high where the crops get destroyed. The knowledge about the rainfall quantity and its variation can help the farmers to plan their crops, thus saving time, effort and resources. Predicting rainfall can also help the general public and the government, as they can take precautionary measures in the case of heavy rains which may lead to floods. These preventive measures can not only save human lives but can also minimise the recovery and reconstruction costs for the state.

1.2 Problem Statement

We aim to develop a rainfall prediction model for districts in Tamil Nadu, India using preliminary regression methods, ensemble regression methods and LSTM based Neural Network methods. The model should be able to predict with low prediction error and captures the variation in it. The project also aims at finding the monthly variation of rainfall across geographical regions.

1.3 Specific Objectives

The project aims to develop three models which predict monthly rainfall for all districts in Tamil Nadu, India and also drawing a district-wise comparison among them to find the best model for prediction. Based on the output of the comparison between the models, the best model is then trained with ensemble regression methods to develop different ensemble regression models. The ensemble regression models are then combined to build Hybrid Ensemble Regression Model. Since the rainfall data used in the project is a time-series data, time-series analysis using LSTM Neural Network is performed.

1.4 Findings

Based on analysis it was discovered that:

- Among the District-Specific Model, Cluster-Based Model and Generic-Regression Model, the Generic-Regression Model perform better than the others in all districts by having low prediction errors and having high EVS and R^2 scores.

- On using various advanced Ensemble Regression Algorithms on the entire state's data, all of them performed similarly, and hence based Time-Complexity analysis it was concluded that XGBR performed better than the others.
- Various plain ensemble regression models are combined using ensemble techniques such as Simple Averaging, Weighted Averaging and Stacking, and based on values it was concluded that using Stacking with PR with degree equal to four and combining ETR and BAR performs better than the other Hybrid Ensemble Regression Algorithms.
- Using a non-optimised single layer LSTM to predict rainfall, performed similarly to the optimised version of preliminary regression models and plain ensemble regression models.

2 LITERATURE SURVEY

2.1 Preliminary Regression Analysis

This section reviews in detail about the previous researches conducted in the same territory. The papers are grouped and discussed based on the methods used in them.

Niu et al. in [2] proposed the use of classification algorithms such as NB, SVM and BPNN on the open dataset from the China Meteorological Administration. The various features used for forecasting include latitude, longitude, altitude and average temperature. The performance measure used for comparing the models is Accuracy. Based on accuracy, it was concluded that BPNN outperforms SVM and NB.

Tharun et al. in [3] predicted rainfall in the Nilgiris District, Tamil Nadu, India using various regression methods such as SVR, RFR and DTR. The performance measures used to evaluate the regression models are R^2 and Adjusted R^2. Adjusted R^2 is the customised version of R^2 that takes into account the effect of adding an influential weekly predictor. Based on the performance measures RFR outperforms SVR and DTR. Kusiak et al. in [4] used a data mining approach to predict rainfall in Oxford and Iowa. The machine learning models used for prediction are MLP, RFR, C&RT, SVR and KNN-R. The error measures used are MSE, MAE and SD. Smaller values of MAE, MSE and SD indicate that a particular model has an excellent fit to the data. According to the analysis, MLP outperforms the other models.

Lu et al. in [5] investigated the performance of various regression methods to predict average monthly rainfall in Guangxi, China using data from January 1965 to December 2009. The methods used are Simple Averaging Ensemble, MSE Ensemble, Variance Weighed Ensemble and SVR. The error measures used are NMSE, MAPE and PRC. The outcome of the analysis is that SVR performs best. Mohapatra et al. in [6] used MLR to model the rainfall data of Bangalore obtained from the India Water Portal for the years 1901 to 2002 and compare the performance of the validation techniques such as Holdout method and K-Fold Cross-Validation method. The prediction was season-wise (Rainy, Summer and Winter) and the features used are Precipitation and Wet Day Frequency. In all the seasons K-Fold Cross Validation method outperforms the Holdout Method.

Chatterjee et al. in [7] used a combination of clustering and HNN to predict rainfall in the Southern part of West Bengal, India. It is a two-step process where the first step is using the Greedy Forward Selection algorithm to reduce the feature set and find the best possible feature set and then K-Means clustering is applied. The second step is to train each cluster with the Neural Network discreetly. The performance measures such as

Accuracy, Precision and Recall are used to compare the HNN and MLP-FFN. The HNN outperformed MLP-FFN in both feature selected and non-feature selected methods.

R. Venkata Ramana et al. in [8] predicted rainfall in Darjeeling Rain Gauge Station, West Bengal, India using a combination of WNN and ANN. The dataset consisted of average monthly rainfall for 74 years. The performance measures used are RMSE, CC and COE. Using 44 years of data as the training set and the rest of the years as the test set, based on performance measures WNN performed better than ANN. Mislan et al. in [9] proposed two different architectures of Neural Networks, which are 2-50-10-1 and 2-50-20-1. The first digit is the number of neurons in the input layer, the second and the third digits are the number of neurons in the hidden layer, and the last number indicates the number of neurons in the output layer. Architecture 2-50-20-1 outperformed the other.

Manek et al. in [10] compared BPNN, GRNN and RBFNN to predict the rainfall in Thanjavur district of Tamil Nadu, India using the data obtained from the India Water Portal - Met Data Repository. The features used for prediction are Precipitation, Cloud Cover, Vapor Pressure and Average Temperature. RBFNN outperformed GRNN and BPNN. Dash et al. in [11] used SLFN and ELM to predict the rainfall season-wise in the years 1871 to 2014, where the networks were trained with the years 1871 to 2004 and for testing set 2005 to 2014. The performance measures used for evaluating the models are MAE and RMSE. On analysis, SLFN outperformed ELM.

Most of the papers mentioned above use a particular location's data to predict rainfall, but in this paper, the collective knowledge of all the 29 districts data in Tamil Nadu, India is used to predicting rainfall in a particular district. Also, to optimise the result, different parameters for each regression algorithm across all the models are tested. The primary focus is on finding the best model among the District-Specific Model, Generic-Regression Model and the Cluster-Based Model along with the best regression algorithm and the corresponding parameter for each district. Furthermore, Section 6.5 discusses the variation of rainfall across the geographic regions in a detailed manner.

2.2 Ensemble Regression Analysis

This section discusses the works done by researchers in predicting the dependent attribute using ensemble classification or regression methods.

Van Heijst et al. [12] used RFR, GBR, XGBR to predict global and local wind energy production and daily aggregate incoming solar energy. They tuned n_estimators, min_samples_leaf, min_samples_split, max_features in RFR and GBR & n_estimators,

colsample_bylevel, min_child weight, max_depth, eta in XGBR to improve the accuracy of prediction. For prediction of wind energy production two cases were considered, a single wind farm in sotavento (local) and peninsular Spain (global). Prediction of ensemble models was compared to a basic ML model (SVR) and a NN model (MLP) using MAE. The order of best performing models for local model and the global model are RFR > XGB > MLP > SVR > GBR and GBR = XGB > SVR > RFR > MLP respectively. For prediction of the total amount to incoming solar energy per day data from 98 Mesonet weather stations covering the state of Oklahoma having 15 forecast variables from year 1994 to 2007 was used. Similarly, it was compared with SVR and MLP, and the order of best-performing models concerning MAE are GBR = XGBR > SVR = RFR > MLP.

Torres et al. in [13] have built a support system for predicting end prices on eBay. Predictions are made based on the item descriptions and few other numerical variables usually used for this kind of predictions. Text mining was used to find essential terms in the item description, and LSBoosting was used for end price prediction. Although item description was not considered in any price prediction project before It was found that it seems to have more influence on the result than seller feedback rating which is shown to be influential in earlier studies.

Kang et al. in [14] have developed an ensemble of linear regression models so that it achieves high prediction accuracy like non-linear models while maintaining the advantages of linear models. This model was tested with benchmark datasets. The data is divided into several locally linear regions based on an expectation-maximisation procedure, and linear models are built on each subset of the data. Finally, these linear models predictions are combined using ensemble techniques. Performance of LLER was compared with ANN, and linear model (LR) and It was concluded that RMSE of LLER is comparable to ANN & far better than LR. LLER also takes less testing time than ANN. The only limitation of LLER is that it takes more training time than the ANN on most datasets, in particular with more massive datasets.

From the above papers, it can be inferred that Ensemble models always performs comparably to Neural Networks and far better than individual models. It is also evident that ensemble learners take less time for testing compared to complex models like neural networks.

2.3 LSTM based Neural Network Analysis

This section discusses the works done by researchers in predicting rainfall using neural networks and also about the various areas in which LSTM is applied.

Dash et al. in [15] used SLFN, RVFL and ROS-RVFL to predict the Indian Summer Monsoon Rainfall. For this purpose, six sets of forecasts ranging from 5 to 10 years were used. Performance measures such as PRC, RMSE, PP, MAPE and MASE were used to evaluate the developed models. On analysis, it was concluded that ROS-RVFL is more accurate and computationally efficient than SLFN and RVFL. Xiang et al. used KNN, ANN and ELM in [16] to predict daily rainfall in the regions of Kunming, Lincang and Mengzi belonging to Yunnan's Province in the Republic of China. The data has years ranging from January 1951 to August 2007. For this purpose, SVR and ANN were used, and for SVR-ANN hybrid model EEMD was used for combining. RMSE, MAE and PRC are used to evaluate the above-mentioned models. SVR is used for transitory-period module prediction and ANN used for extended-period module prediction. It was concluded that SVR-ANN performed better than individual SVR and ANN.

Kashiwao et al. in [17] predicted rainfall using data obtained from the Japan Meteorological Agency and using models RBFNN and MLP. A 3 layer MLP (3LP) was developed using a hybrid algorithm containing Back Propagation and Random Optimization methods. The evaluation measures used to validate the models are Total Hit Rate, Hit Rate of Precipitation, Hit Rate of Non-Precipitation, Caching Ratem Overlooking Rate and Swing & Miss Rate. Based on the values in the evaluation measures MLP and RBFNN. Dash et al. in [18] for seasonal forecasting of summer monsoon and post-monsoon from the year 2011 to 2016 for the state of Kerala, India. For this purpose KNN, ANN and ELM have been used to predict the rainfall. MAE, RMSE, PP, PRC and MASE are used to evaluate the models. Based on the analysis, it was concluded that ELM performs better than MLP.

Devi et al. used BPNN, CBPNN, DTDNN and NARX in [19] predicting rainfall using two different datasets one with daily rainfall, temperature and humidity of Nilgiris, Tamil Nadu, India and the other having daily rainfall of 14 rain gauge stations in and around Coonoor, Niligirs. MSE and CC are used to evaluate the models and on analysis NARX outperformed the other networks. The performance of BPNN, CBPNN and DTDNN are same, but the prediction capabilities of BPNN is better than CBPN and DTDNN. Salman et al. in [20] used LSTM to predict Weather using the data collected by Weather Underground at Hang Nadim Indonesia Airport. For this purpose, they compared the performance Single Layer LSTM and 4-Layer LSTM and also found the effect of the intermediate variable in the weather prediction. Based on the values of Validation Accuracy and RMSE it was concluded that 4-layer LSTM outperforms the single layer LSTM.

Xu et al. in [21] used LSTM, SVR, BPNN and Elman Network for displacement prediction of the landslide in Baijiabao, China. The factors considered for predicting periodic

term are PRC and Mutual information. Based on RMSE values it was concluded that the dynamic models LSTM and Elman Network perform better than the static models SVR and BPNN. On further analysis between the dynamic models, it was deduced that LSTM performs better than Elman Network. Zhang et al. in [22] used MLP, WNN, LSTM and GRU to model the data collected using IoT combined Sewer Overflow structure. Based on the analysis, it was concluded that LSTM and GRU are multi-step-ahead in time series prediction, but GRU predicts similar to LSTM in short period.

Pak et al. in [23] used MLP and hybrid neural network combining CNN and LSTM for the prediction of ozone concentration in a region. CNN was used to extract the features of tremendous air quality and meteorological data, and LSTM was used to Ozone concentration prediction in Beijing City. The RMSE, MAE and MAPE were used as error measures and based on the analysis it was concluded that CNN-LSTM outperforms MLP. In [24] Xiao et al. used LSTM to predict the occurrence of pests and diseases in cotton fields. Using AUC as the performance, it corroborated the weather factors that influence the occurrence of the same.

Neural Networks has been severely used by researchers to predict rainfall all over the world. However, the rainfall prediction in most cases is time-series analysis and in all the papers mentioned above that use LSTM to predict dependent attribute, has always performed better than other models. It can be concluded that using LSTM to predict rainfall will produce better results than the other neural networks.

3 SYSTEM SPECIFICATIONS

- **System**: Lenovo yoga 530
- **Processor**: Intel 8^{th} Gen i5
- **RAM**: 8 GB DDR4
- **ROM**: 256 GB SSD
- **Graphics Card**: Nvidea MX130s 2GB
- **Operating System**: Windows 10 Home

4 METHODOLOGY

This section describes the dataset used for investigation and defines all the regression algorithms, clustering algorithms and performance measures used in this paper.

4.1 Dataset Description

The India Water Portal - Met Data Repository is used to collect the data. The data collected for a particular district comprises of the dependent attribute 'Rainfall' and eight independent attributes namely 'Average Temperature', 'Cloud Cover', 'Maximum Temperature', 'Minimum Temperature', 'Crop Evapotranspiration', 'Potential Evapotranspiration', 'Vapor Pressure' and 'Wet Day Frequency'. The dataset of each feature contains 102 records and 12 columns where each row contains data of a particular year, and each column contains data of a particular month across the years.

4.2 Regression Algorithms

4.2.1 Multiple Linear Regression

It is a straight line approach to model the correlation between the dependent variable and multiple independent variables using single-dimensional predictor functions. The model parameters depend on the dataset and is not standard for all the datasets.

4.2.2 Support Vector Regression

It is a supervised learning method which builds hyper-plane(s) in a dimensional space used for regression and classification examination or detecting outliers. The various kernels used to transform the data for prediction are Linear, Non-Linear, Polynomial, Sigmoid and RBF.

4.2.3 Polynomial Regression

It is similar to MLR where the relationship between the dependent and the independent variable modelled as n^{th} order polynomial on the independent variables.

4.2.4 Decision Tree Regression

Decision tree uses supervised learning to build regression or classification models in the form of a tree structure. The tree has three different nodes, namely the root node, decision nodes and leaf nodes. The root node is the primary node, decision node has branches (two or more), and the leaf node is a node at the end of the tree.

4.3 Clustering Model

4.3.1 K-Means Clustering

It focuses on dividing N points into K clusters with the closest mean, serving as the centre of the cluster. The Euclidean Distance is used to allocate a data point to a particular cluster centre.

4.3.2 Elbow Method

This method focuses on finding the optimal number of clusters. Sum Square Error (SSE) is the sum of the mean Euclidean Distance of all the points against the centroid. SSE is computed for every increment in the number of clusters (K). When the SSE starts dropping by decidedly smaller angles, then that K value is the optimal number of clusters.

4.4 Ensemble Techniques

4.4.1 Simple Averaging

Various prediction models are used to make predictions to a data point. The average of the predicted values of the models becomes the new predicted value.

4.4.2 Weighted Averaging

It is a modification to the simple averaging method, where the predicted values of each model are averaged based on their performance and importance to the end predicted value.

4.4.3 Stacking

Stacking is also an ensemble learning technique, but it not only uses the testing set predictions but also uses training set predictions. The prediction made on the training set by multiple models are used as input to a regression algorithm to predict the actual dependent attribute.

4.5 Advanced Ensemble Algorithms

4.5.1 Random Forest Regression

Also known as Random Decision Forests is an ensemble learning method to perform regression and classification tasks. It constructs multiple decision trees during the training period and outputs the mean prediction of the individual trees.

4.5.2 Extra Trees Regression

It fits multiple completely randomised decision trees (Extra Trees) on multiple samples of the dataset and averages the result to decrease the predictive error and controls over-fitting.

4.5.3 Bagging Regression

Bootstrap Aggregating (Bagging) is an ensemble learning algorithm which designed to improve variance and to avoid over-fitting. The base-estimator can be changed for better results. It is a modified case of the Model Averaging method.

4.5.4 AdaBoost Regression

It is also known as Adaptive Boosting is similar to the Bootstrap Aggregating where the base estimator can be changed for improved performance. The output of the weak learners is combined into a weighted sum that represents the final output of the boosted regression. It is delicate to noisy data and outliers.

4.5.5 Gradient Boosting Regression

It produces an ensemble weak prediction models in a hierarchy fashion like the other boosting methods. It generalises them by allowing modifying of the arbitrary differentiable loss.

4.5.6 Extreme Gradient Boosting

It is the advanced implementation of Gradient Boosting Regression which includes a variety of regularisation techniques and reduces overfitting and increases the performance.

4.6 Long Short-Term Memory Neural Network

It is an artificial recurrent neural network, where each LSTM unit consists of an input gate, output gate and a forget gate. To optimise the functioning of a neural network the epoch and batch size can be fixed. An epoch is the forward or backward pass of all the training examples, and batch size is the number of training examples in the epochs.

4.7 Evaluation Measures

4.7.1 Mean Squared Error

It measures the quality of a model where the value is the mean squared difference between the actual and predicted value as given in (1).

$$MSE = \frac{\sum_{i=1}^{N}(Y_t - Y_p)^2}{N} \tag{1}$$

Where Y_t is the actual value, Y_p is the predicted value and N is the number of observations in the dataset.

4.7.2 Root Mean Squared Error

It is the square root of the mean squared error as given in (2).

$$RMSE = \sqrt[2]{MSE} \tag{2}$$

4.7.3 Mean Absolute Error

It measures the mean absolute difference between the actual and the predicted value in a set of predictions as given in (3).

$$MAE = \frac{\sum_{i=1}^{N}|Y_t - Y_p|}{N} \tag{3}$$

4.7.4 Median Absolute Error

It measures the variance of a uni-variate sample of quantitative data. It is defined as the median of the absolute residuals between the original and the predicted value as given in (4).

$$MDAE = median(|Y_t - Y_p|) \tag{4}$$

4.7.5 Explained Variance Score

It measures the ratio to which a regression model is capturing the dispersion in the dataset. It is the mean squared difference between the predicted value and the mean of the actual values in the dataset as given in (5).

$$EVS = \frac{\sum_{i=1}^{N}(Y_p - Y_m)^2}{N} \tag{5}$$

Where Y_m is the mean of Y_t in the dataset.

4.7.6 R^2 Score

It is commonly known as the coefficient of determination which is the ratio of dispersion in the predictive variable from independent variables as given in (6).

$$R^2 = \frac{EVS}{TV} \tag{6}$$

Where EVS is the Explained Variance Score and TV is Total Variation. If the value of R^2 is 0%, then none of the variability of the response data is around the mean, and if it is 100%, then all the variability of the response data is around the mean.

5 PROCESS FLOW

The process flow for the proposed architecture is given in Fig. 1.

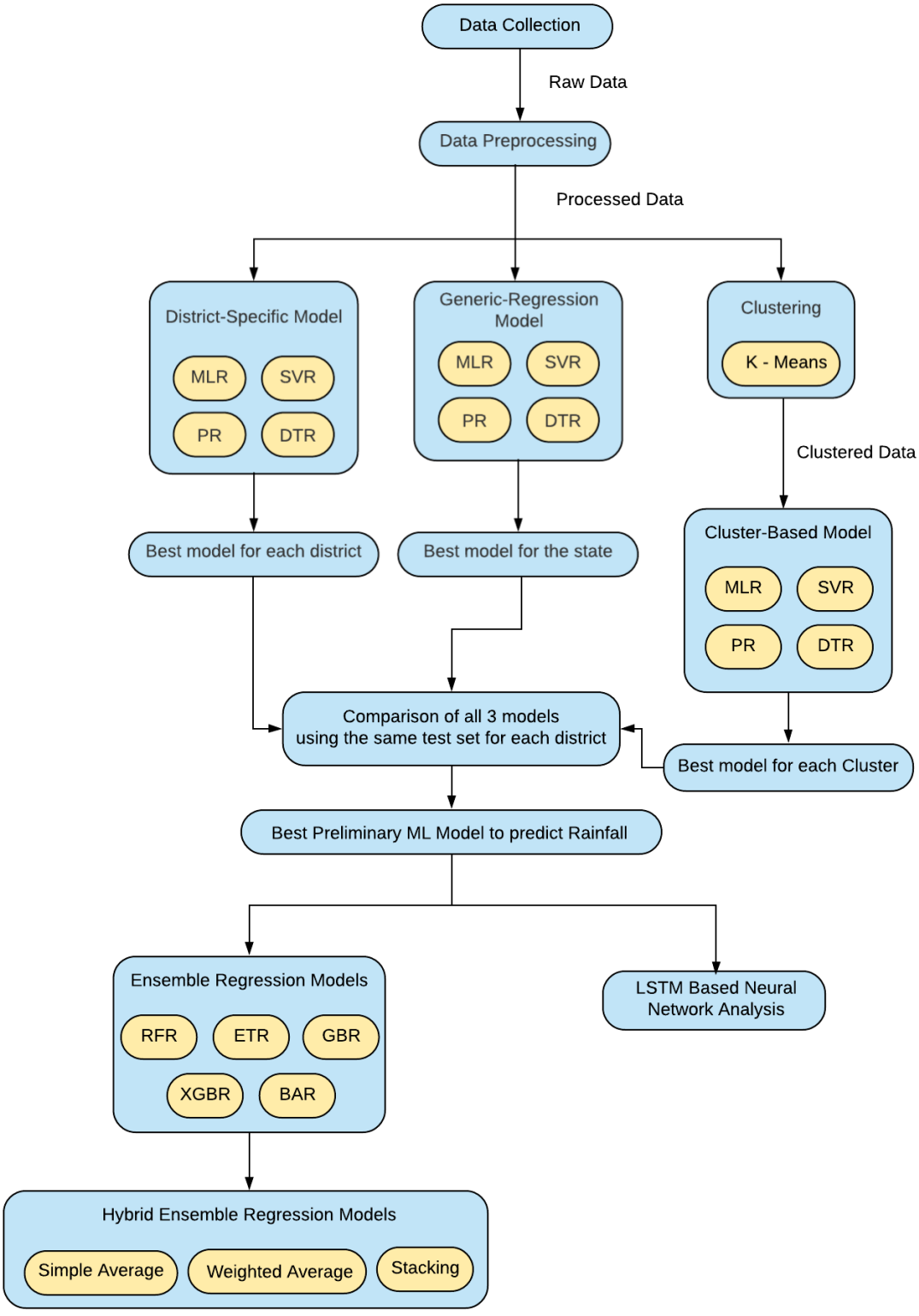

Figure 1: Process Flow

5.1 Data Pre-processing

5.1.1 Data Transformation

The dataset obtained from the source had separate files for each feature in all the districts. Combining the datasets of features into a single dataset makes computation far easier where each column contains data of a particular feature and is arranged sequentially from 1901 January to 2002 December.

5.1.2 Data Normalisation

All the attributes used are numerical and have different ranges. For the regression algorithm to work with high efficiency and accuracy, the attributes have to be normalised. Using Min-Max Normalisation for this purpose brings the range of all the features from 0 to 1, thereby reducing the chance of having different weights for the features. The formula for the Min-Max Normalisation is given in (7).

$$X'_{\mathrm{i}} = \frac{X_{\mathrm{i}} - X_{\mathrm{min}}}{X_{\mathrm{max}} - X_{\mathrm{min}}} \tag{7}$$

Where X_i is the i^{th} element in the feature, X_{min} is the minimum value of the feature, X_{max} is the maximum value of the feature and X'_i is the normalised value of the i^{th} element in the feature.

5.2 District-Specific Model

For each district, the rainfall has been predicted using four regression algorithms with different parameter values. To predict rainfall for a particular district, only the data collected from that district is used to train the model. Repeated K-Fold Cross Validation method has been used to validate the model with ten splits and ten repetitions, and the evaluation measures have been used to find the best model for each district.

5.3 Generic-Regression Model

The data of all the districts have been combined into one single dataset (Generic Dataset) to build the Generic-Regression Model. The generic dataset has 35496 tuples (29 districts * 1224 tuples per district) to which the same process as in the District-Specific Model has been used for prediction.

5.4 Cluster-Based Model

K-Means clustering has been used to find the districts with similar climatic conditions. The datasets are combined based on the clusters formed, and rainfall is predicted for a

district by training the model with data of the cluster to which the district belongs. The rest remains the same as the District-Specific Model.

5.5 Hybrid Ensemble Regression Model

Hybrid Ensemble Regression Model uses the prediction made by combinations of ensemble algorithms in Section 6.6 using Simple Averaging, Weighted Averaging and Stacking to make better predictions.

6 RESULTS AND DISCUSSIONS

This section discusses in detail the results obtained on using the machine learning regression algorithms, ensemble regression algorithms and LSTM Neural Network to model the data for all the districts in Tamil Nadu, India.

6.1 Correlation between the attributes

The Pearson Correlation Coefficient finds the linear relationship between any two continuous variables. It helps in finding the right set of features for predicting the target variable. The formula for calculating the correlation between any two attributes is given in (8).

$$\rho_{x,\,y} = \frac{\sum(X_i - X_m)(Y_i - Y_m)}{\sqrt{\sum(X_i - X_m)^2 \sum(Y_i - Y_m)^2}} \tag{8}$$

Where X and Y are the continuous variables. X_i and Y_i represents the i^{th} element in the vectors and X_m and Y_m are the mean values of the corresponding vectors. In this process, the actual values are re-scaled, and the Standard Deviation is computed. If ρ is closer to 1 then the variables are positively correlated, if ρ is closer to -1, then the variables are negatively correlated, and if the variables are independent of each other, then ρ is closer to 0. The correlation heat-map of the attributes is given in Fig. 2.

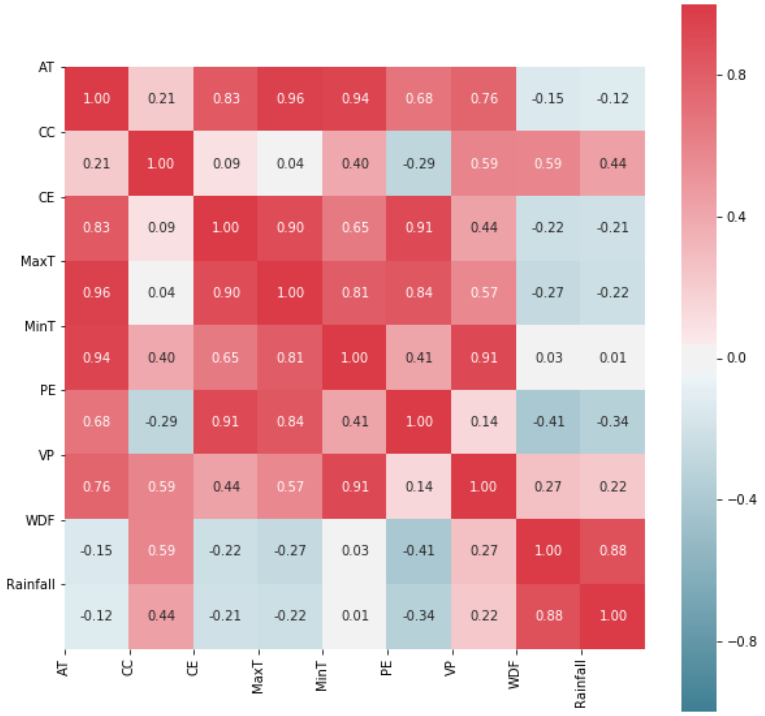

Figure 2: Correlation Heat Map

Cloud Cover, Vapour Pressure and Wet Day Frequency are positively correlated with rainfall, and Average Temperature, Crop Evapotranspiration, Maximum Temperature and

Potential Evapotranspiration are negatively correlated with rainfall as shown in Fig. 2. Also, Minimum Temperature has only a slight impact on the amount of rainfall, so it is excluded from the prediction process.

6.2 Parameter Selection for the Regression Algorithms

For the regression algorithms to predict more accurately, their corresponding parameter values have to be tuned. For SVR, different kernels like Linear, Polynomial (Degree = 3), Sigmoid and RBF are tested. Similarly for DTR, maximum depths ranging from two to seven and for PR, degrees ranging from two to five are tested.

6.3 Performance Analysis of the Models

6.3.1 District-Specific Model

The regression algorithms and the parameter required to build the best model for a district is chosen based on the models' performance measures. A good model should have low MSE, RMSE, MAE and MDAE and high EVS and R^2 values. The performance of all the regression algorithms with different parameters for the Chennai District is in Table 1.

Method	Parameter	MSE	RMSE	MAE	MDAE	EVS	R^2
MLR	-	0.004	0.0635	0.0442	0.0324	0.8257	0.8241
PR	**Degree = 2**	**0.0039**	**0.0615**	0.0423	0.0282	**0.8305**	**0.8289**
	Degree = 3	0.004	0.0629	**0.0402**	**0.0239**	0.8273	0.8253
	Degree = 4	0.0558	0.1933	0.0846	0.0408	-1.5956	-1.6172
	Degree = 5	4149.7	50.1	15.3	2.7	-187415	-188520
DTR	Max Depth = 2	0.0057	0.0747	0.0484	0.0252	0.7569	0.7552
	Max Depth = 3	0.0043	0.0646	0.0393	0.0198	0.817	0.8156
	Max Depth = 4	0.004	0.0628	0.0371	0.0183	0.8278	0.8264
	Max Depth = 5	**0.0039**	**0.0616**	**0.036**	**0.0181**	**0.8342**	**0.833**
	Max Depth = 6	0.0042	0.0639	0.0368	0.0184	0.8211	0.8199
	Max Depth = 7	0.0044	0.0653	0.0374	0.0184	0.8132	0.812
SVR	Kernel = Linear	0.0046	0.0674	0.05	0.0406	0.8053	0.8002
	Kernel = Poly	0.0103	0.1004	0.0727	0.0592	0.5758	0.5637
	Kernel = RBF	**0.0038**	**0.0609**	**0.0424**	**0.031**	**0.8395**	**0.8372**
	Kernel = Sigmoid	0.2638	0.5119	0.3532	0.2414	-10.41	-10.93

Table 1: Comparison on performance of the regression algorithms for the Chennai District

On observing the values in Table 1, for PR, degree two is an excellent choice, as the degree rises the MSE, RMSE, MAE and MDAE values tend to increase, and the EVS and R^2 values tend to decrease. For degree four and five the EVS and R^2, scores are negative which indicates that the models are unstable and do not capture the variation well. Also, DTR with a maximum depth of five outperforms the others, and SVR with RBF kernel outperforms SVR with other kernels.

Extending the analysis done in Table 1 to the other districts, it was found that MLR performs better for the districts Dharmapuri, Dindigul, Madurai, Ramanathapuram, Theni, Tirunelveli and Virudhunagar. Likewise, SVR with RBF kernel for Kancheepuram, Tiruvannamalai and Vellore and PR with degree two performs better for the other districts.

6.3.2 Generic-Regression Model

The generic data is used for training the model, where the performance measures of all the regression algorithms along with their parameters are given in Table 2.

Method	Parameter	MSE	RMSE	MAE	MDAE	EVS	R^2
MLR	-	0.0006	0.0254	0.0156	0.0101	0.7845	0.7844
PR	Degree = 2	0.00057	0.0239	0.0145	0.0089	0.8081	0.8081
	Degree = 3	0.00054	0.0231	0.0137	0.0079	0.8207	0.8206
	Degree = 4	**0.00052**	**0.0227**	**0.0134**	**0.0076**	**0.8268**	**0.8267**
	Degree = 5	0.00053	0.0229	0.0135	0.0078	0.8236	0.8235
DTR	Max Depth = 2	0.00098	0.0313	0.0192	0.0111	0.6731	0.673
	Max Depth = 3	0.00074	0.0272	0.016	0.0091	0.7518	0.7518
	Max Depth = 4	0.00067	0.0259	0.0149	0.0083	0.7759	0.7758
	Max Depth = 5	0.00064	0.0253	0.0145	0.0081	0.7862	0.7862
	Max Depth = 6	**0.00063**	**0.0252**	**0.0142**	**0.0079**	**0.7878**	**0.7877**
	Max Depth = 7	0.00065	0.0254	**0.0142**	**0.0079**	0.7833	0.7833
SVR	**Kernel = Linear**	**0.0015**	**0.0388**	**0.0311**	**0.0279**	0.6963	**0.494**
	Kernel = Poly	0.0041	0.0641	0.0574	0.0577	0.5824	-0.3829
	Kernel = RBF	0.0027	0.0523	0.0463	0.0466	0.6845	0.0814
	Kernel = Sigmoid	0.0016	0.0394	0.033	0.0318	**0.7475**	0.4795

Table 2: Comparison on performance of the regression algorithms for the Generic Data

From Table 2, it can be inferred that PR with degree four fits the data better and outperforms the other degrees. DTR with a maximum depths of six outperforms the other depth values, and SVR with RBF kernel outperforms the other kernels.

6.3.3 Cluster-Based Model

For each district, the median of all the features across 102 years has been considered as input for clustering using K-Means, and Elbow Method has been used to find the optimal number of clusters. The graph of the number of cluster centres versus the sum of squared distances is shown in Fig. 3.

It can be clearly observed in Fig. 3 that six is the optimal number of clusters as the sum of squared distances drops by minimal angles from that point. The clusters formed as a result of K-Means clustering with K as six is shown in Fig. 4 and that is cross-verified with works done by Palanisami et al. in Diversification of Agriculture in Coastal Districts of Tamil Nadu- a Spatio- Temporal Analysis [25].

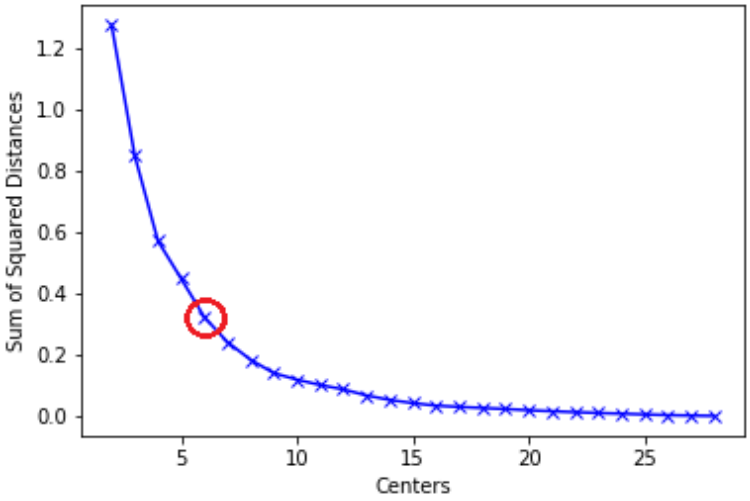

Figure 3: Elbow Method

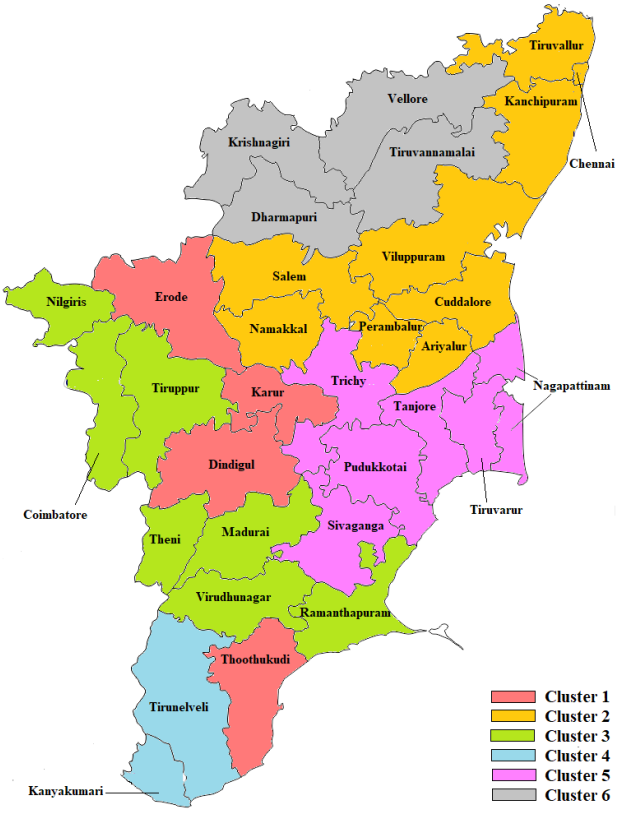

Figure 4: Graphical Representation of formed Clusters

Based on the results obtained after performing clustering, the districts were grouped, and all the chosen regression algorithms with different parameter values have been applied to each of the grouped data. The performance measures of all the regression algorithms along with different parameter values for Cluster 1 are shown in Table 3.

As shown in Table 3, PR with degree three, DTR with a maximum depth of five

and SVR with linear kernel outperforms the other regression models for the cluster 1. The same has been extended to the other clusters, and it was found that PR is the best regression algorithm where the best degree for cluster 4 is two, cluster 2 is four and for the other clusters is three.

Method	Parameter	MSE	RMSE	MAE	MDAE	EVS	R^2
MLR	-	0.0044	0.0663	0.0462	0.0309	0.7412	0.7406
PR	Degree = 2	0.0043	0.0654	0.0453	0.0307	0.748	0.7475
	Degree = 3	**0.0041**	**0.0635**	**0.0437**	**0.029**	**0.7624**	**0.7619**
	Degree = 4	0.0044	0.0659	0.0454	0.0304	0.7444	0.7438
	Degree = 5	0.0106	0.1005	0.0604	0.0.0384	0.379	0.3777
DTR	Max Depth = 2	0.0056	0.0745	0.0534	0.0363	0.6729	0.6722
	Max Depth = 3	0.0048	0.069	0.0478	0.0325	0.7193	0.7187
	Max Depth = 4	0.0047	0.0682	0.0463	0.0309	0.7257	0.7251
	Max Depth = 5	**0.0046**	**0.0678**	**0.0454**	0.0299	**0.7287**	**0.7282**
	Max Depth = 6	0.0048	0.0688	0.0456	**0.0295**	0.7204	0.7198
	Max Depth = 7	0.005	0.0705	0.0463	0.0296	0.7062	0.7056
SVR	**Kernel = Linear**	**0.0048**	**0.0694**	**0.0526**	**0.0419**	**0.7343**	**0.7162**
	Kernel = Poly	0.0068	0.0824	0.0675	0.0641	0.6359	0.6001
	Kernel = RBF	0.005	0.071	0.0557	0.0469	0.7292	0.7031
	Kernel = Sigmoid	0.7071	0.8392	0.5115	0.3084	-38.35	-40.84

Table 3: Comparison on performance of the regression algorithms for the Cluster 1

6.4 Comparison on performance of District Specific Model, Cluster-Based Model and Generic Regression Model

A comparison was drawn between the performance of the District-Specific Model, Cluster-Based Model and the Generic-Regression Model by testing them on the same test data. At a time only one district is considered for comparison. Repeated K-Fold Cross-Validation with ten folds and ten repeats, has been applied a district's data, where the test set obtained in each iteration has been removed from the respective clustered data and the generic data using a customised index. The same set of record has been removed from the generic dataset and the clustered dataset that contains that district, for testing. Then the remaining records have been used for training the respective models. The comparison between the performance of the three models is shown in Table 4 and Table 5.

Table 4: Comparison between the Models using the Performance Measures MSE, RMSE and MAE

Cluster	District Name	MSE			RMSE			MAE		
		District	**Cluster**	**Generic**	**District**	**Cluster**	**Generic**	**District**	**Cluster**	**Generic**
Cluster 1	Dindigul	0.0064	0.0055	**0.0006**	0.0796	0.0734	**0.0245**	0.0559	0.0505	**0.0166**
	Erode	0.0042	0.0026	**0.0003**	0.064	0.0503	**0.0165**	0.0435	0.0338	**0.0109**
	Karur	0.0114	0.0031	**0.0003**	0.1064	0.0555	**0.0184**	0.0778	0.0398	**0.0131**
	Thoothukkudi	0.0074	0.0029	**0.0003**	0.0857	0.0533	**0.0177**	0.0593	0.0369	**0.0121**
Cluster 2	Ariyalur	0.003	0.0008	**0.0001**	0.0539	0.0275	**0.0107**	0.0356	0.0179	**0.0069**
	Chennai	0.0031	0.0022	**0.0003**	0.055	0.0466	**0.0183**	0.0353	0.0287	**0.0112**
	Cuddalore	0.002	0.0005	**0.0001**	0.0441	0.0231	**0.0093**	0.0286	0.0146	**0.0059**
	Kancheepuram	0.0041	0.0017	**0.0003**	0.0634	0.0413	**0.0165**	0.0456	0.0261	**0.0103**
	Namakkal	0.008	0.0012	**0.0002**	0.0891	0.0346	**0.0132**	0.0646	0.0247	**0.0094**
	Perambalur	0.0045	0.0013	**0.0002**	0.0669	0.0352	**0.0134**	0.0453	0.0239	**0.0091**
	Salem	0.0071	0.0011	**0.0002**	0.084	0.0333	**0.0127**	0.0593	0.0229	**0.0087**
	Thiruvallur	0.0046	0.0016	**0.0002**	0.0674	0.0396	**0.0155**	0.0486	0.0253	**0.0098**
	Viluppuram	0.002	0.0005	**0.0001**	0.0441	0.0223	**0.0086**	0.0298	0.0148	**0.0057**
Cluster 3	Coimbatore	0.0035	**0.0009**	**0.0009**	0.059	0.0301	**0.0295**	0.0385	0.0193	**0.0187**
	Madurai	0.007	**0.0008**	**0.0008**	0.0827	**0.028**	**0.028**	0.0546	0.0183	**0.018**
	Ramanathapuram	0.0061	**0.0003**	**0.0003**	0.0776	**0.018**	**0.018**	0.0542	0.0122	**0.012**
	Theni	0.0051	0.0018	**0.0017**	0.0703	0.0412	**0.0411**	0.0433	0.0248	**0.0245**
	The Nilgiris	0.0025	0.002	**0.0019**	0.0486	0.0436	**0.0428**	0.0253	0.0224	**0.0214**
	Virudhunagar	0.0088	**0.0007**	**0.0007**	0.0927	**0.0266**	0.0269	0.0621	0.0176	**0.0175**
Cluster 4	Tirunelveli	0.0082	0.0075	**0.0005**	0.09	0.0856	**0.0228**	0.0611	0.0577	**0.0154**
Cluster 5	Nagapattinam	0.0032	0.0024	**0.0002**	0.0563	0.0489	**0.0151**	0.0378	0.0324	**0.0101**
	Pudukkottai	0.004	0.0019	**0.0002**	0.0625	0.0434	**0.0137**	0.0425	0.0292	**0.0092**
	Sivaganga	0.0048	0.0024	**0.0002**	0.0686	0.0485	**0.0152**	0.0483	0.0337	**0.0105**
	Thanjavur	0.0044	0.0024	**0.0002**	0.0656	0.0483	**0.0149**	0.044	0.032	**0.0098**
	Thiruvarur	0.0054	0.0041	**0.0004**	0.0733	0.0636	**0.0198**	0.0514	0.0426	**0.0132**
	Tiruchirapalli	0.0063	0.0021	**0.0002**	0.0787	0.0458	**0.0143**	0.0576	0.0326	**0.0101**
Cluster 6	Dharmapuri	0.0056	0.0032	**0.0002**	0.0744	0.0566	**0.0125**	0.053	0.0382	**0.0084**
	Tiruvannamalai	0.0054	0.004	**0.0002**	0.0735	0.0628	**0.014**	0.0523	0.0416	**0.0091**
	Vellore	0.0074	0.0049	**0.0002**	0.0854	0.0696	**0.0153**	0.0612	0.0467	**0.0102**

Cluster	District Name	MDAE			EVS			R^2		
		District	**Cluster**	**Generic**	**District**	**Cluster**	**Generic**	**District**	**Cluster**	**Generic**
Cluster 1	Dindigul	0.0379	0.0336	**0.0112**	0.7174	**0.7589**	0.7527	0.7146	**0.7571**	0.7508
	Erode	0.0279	0.0212	**0.0069**	0.8407	**0.8654**	**0.8654**	0.8395	0.8643	**0.8646**
	Karur	0.057	0.0268	**0.009**	0.6979	**0.7445**	0.7383	0.6945	**0.7424**	0.7355
	Thoothukkudi	0.0421	0.0251	**0.0081**	0.7018	**0.75**	0.7446	0.6987	**0.7478**	0.7423
Cluster 2	Ariyalur	0.0237	0.0109	**0.0042**	0.8882	**0.9187**	0.9069	0.8873	**0.9181**	0.9062
	Chennai	0.0206	0.0155	**0.0058**	0.8663	**0.9033**	0.8894	0.865	**0.9024**	0.8884
	Cuddalore	0.0186	0.0087	**0.0035**	0.9374	**0.9573**	0.948	0.9369	**0.9569**	0.9475
	Kancheepuram	0.0351	0.0146	**0.0055**	0.8517	**0.916**	0.9016	0.8495	**0.9153**	0.9006
	Namakkal	0.0448	0.0166	**0.0062**	0.7828	**0.8286**	0.8135	0.7803	**0.8273**	0.8117
	Perambalur	0.0283	0.0151	**0.0058**	0.8059	**0.8409**	0.8273	0.8045	**0.8398**	0.8258
	Salem	0.0394	0.0149	**0.0055**	0.7982	**0.8406**	0.8289	0.7963	**0.8391**	0.8274
	Thiruvallur	0.0357	0.0145	**0.0056**	0.8243	**0.9043**	0.8909	0.8227	**0.9035**	0.8901
	Viluppuram	0.0193	0.009	**0.0034**	0.9386	**0.9568**	0.9521	0.9381	**0.9565**	0.9517
Cluster 3	Coimbatore	0.0232	0.0112	**0.0107**	0.8636	0.8847	**0.8888**	0.8625	0.8839	**0.888**
	Madurai	0.0348	0.0115	**0.011**	0.6421	**0.6847**	0.6822	0.639	**0.6814**	0.6797
	Ramanathapuram	0.038	0.0081	**0.0076**	0.7631	**0.8117**	0.8103	0.761	**0.81**	0.8092
	Theni	0.026	0.0145	**0.0138**	0.7353	0.7668	**0.7679**	0.7329	0.7648	**0.7664**
	The Nilgiris	0.0116	0.0099	**0.0085**	0.8494	0.8785	**0.8817**	0.8483	0.8774	**0.8809**
	Virudhunagar	0.0417	0.0115	**0.0112**	0.6459	0.6991	**0.6934**	0.6423	**0.6968**	0.6907
Cluster 4	Tirunelveli	0.0407	0.0388	**0.01**	0.6777	0.7094	**0.7374**	0.6736	0.7069	**0.7348**
Cluster 5	Nagapattinam	0.0247	0.0204	**0.0065**	0.8671	0.8862	**0.8875**	0.8658	0.8853	**0.8866**
	Pudukkottai	0.0275	0.018	**0.0057**	0.8381	**0.8579**	0.8532	0.8368	**0.8567**	0.8523
	Sivaganga	0.0333	0.022	**0.0069**	0.8038	**0.8321**	0.8254	0.8021	**0.8306**	0.8239
	Thanjavur	0.028	0.0198	**0.006**	0.8167	0.8446	**0.8453**	0.8152	0.8434	**0.8442**
	Thiruvarur	0.0359	0.0266	**0.0081**	0.7617	**0.8242**	0.8196	0.7598	**0.8224**	0.8181
	Tiruchirapalli	0.0416	0.0227	**0.0069**	0.7559	**0.8034**	0.8022	0.7528	**0.8019**	0.8
Cluster 6	Dharmapuri	0.0363	0.023	**0.005**	0.7909	**0.8504**	0.8448	0.7894	**0.8493**	0.8439
	Tiruvannamalai	0.0379	0.0251	**0.0053**	0.8366	**0.879**	0.873	0.8352	**0.878**	0.872
	Vellore	0.0448	0.0272	**0.0059**	0.7833	**0.8396**	0.8343	0.7817	**0.8385**	0.8332

Table 5: Comparison between the Models using the Performance Measures MDAE, EVS and R^2

Based on all six performance measures used, Cluster-Based Model performs better than the District-Specific Model across all the districts as shown in Table 4 and Table 5. The Generic-Regression Model has the least MSE, RMSE, MAE and MDAE values for all the districts and the Cluster-Based Model has the highest EVS and R^2 scores for a maximum number of districts, which implies that Cluster-Based Model captures the variation well compared to the other models. However, the difference in value between the Cluster-Based Model and the Generic-Regression Model is negligible.

6.5 Variation in Rainfall Distribution across the Geographical Regions and Time

To visualise the variation of rainfall across months, the median of the rainfall values recorded for a particular month across years for all the districts in a cluster has been calculated. The continuous lines plots in Fig. 5 and 6 is the line connecting the median rainfall across months for each cluster and dotted lines are the average rainfall for each cluster.

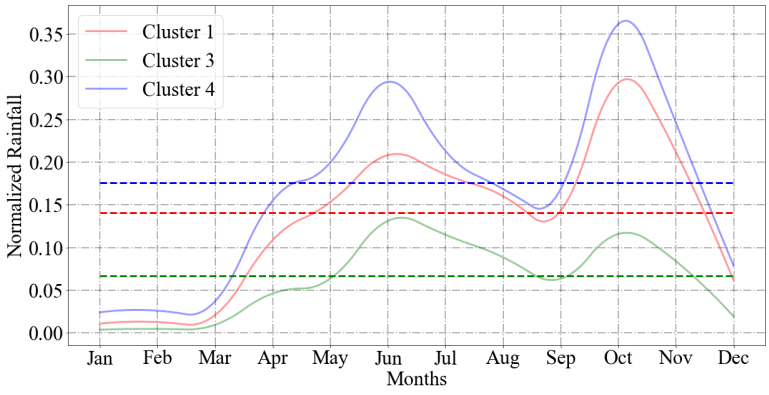

Figure 5: Variation of Rainfall across months for Clusters 1, 3 and 4

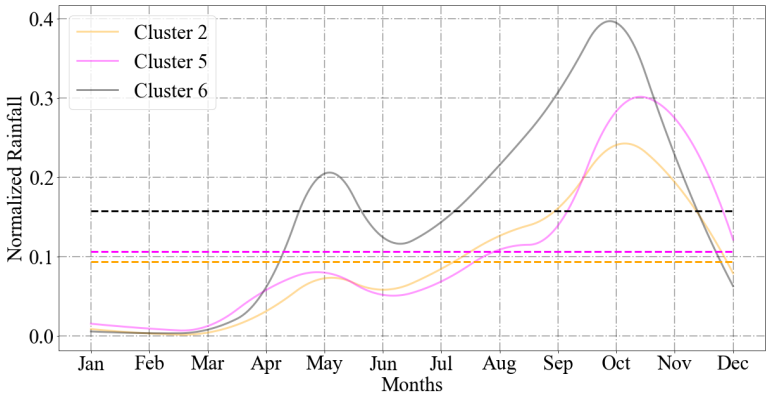

Figure 6: Variation of Rainfall across months for Clusters 2, 5 and 6

For every cluster, the rainfall is deficient in the first three months of the year and is maximum in October. All the districts in Tamil Nadu, India receives a high amount of rainfall twice in a year. The first time it is caused by South-West Monsoon, and the second time it is caused by North-East Monsoon. Two patterns are observed in the variation of rainfall among the clusters which is displayed in Fig. 5 and Fig. 6. Clusters 1, 3 and 4 receive high rainfall in June and October, these are clusters of districts which lies on the western half of the state whose rainfall is influenced by the Western Ghats whereas clusters 2, 5 and 6 receives high rainfall in May and October, these are clusters of districts which lies on the eastern half of the state near the coastal regions. The dotted lines in Fig. 5 and Fig. 6 shows that cluster 4 has the maximum rainfall followed by clusters 6, 1, 5, 2 and 3 across the months in the respective order.

6.6 Performance Analysis of the Ensemble Regression Algorithms on the Generic-Regression Model

The parameters tested for optimized fitting of the dataset in the RFR, ETR, GBR and XGBR are **n_estimators** (Number of trees in the forest), **max_depth** (Maximum Depth of the tree), **min_samples_split** (The minimum number of samples required to split an internal node) and **min_samples_leaf** (The minimum number of samples required to be at the leaf node). However, the performance analysis of min_samples_split and min_samples_leaf are not included in the report because their tuning did not produce a noticeable difference in the performance measures. Likewise, for ABR and BAR the parameters tuned are **n_estimators** and **base_estimator** (It is the base learner on which the boosting or bagging ensemble is constructed).

6.6.1 Random Forest Regression

The MSE and R^2 values for the RFR with respective number of estimators is shown in Fig. 7 and Fig. 8.

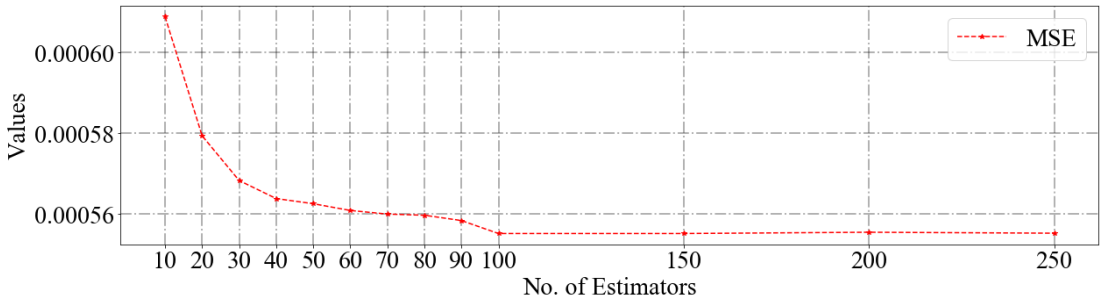

Figure 7: RFR with different number of estimators versus their corresponding MSE values

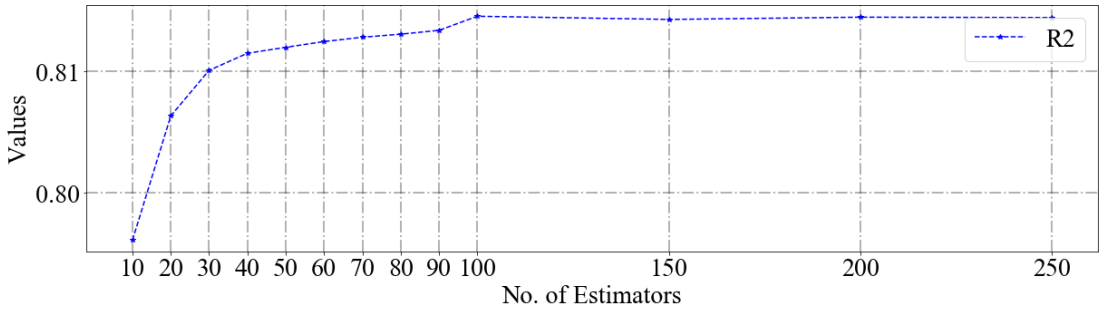

Figure 8: RFR with different number of estimators versus their corresponding R^2 values

We can infer from Fig. 7 and Fig. 8 that the value of MSE and R^2 becomes stagnant after 100 estimators; therefore 100 can be fixed as the number of estimators. To further optimise the result, the parameter values of Max Depth is changed from 2 to 20 as shown in Table 6.

Max Depth	MSE	RMSE	MAE	MDAE	EVS	R^2
2	0.000901	0.02998	0.0184	0.01133	0.699	0.699
3	0.000680	0.02604	0.0154	0.00872	0.773	0.773
4	0.000633	0.02512	0.0146	0.00809	0.789	0.789
5	0.000593	0.02433	0.0141	0.00793	0.802	0.802
6	0.000576	0.02397	0.0138	0.00775	0.807	0.807
7	0.000562	0.02367	0.0136	0.00768	0.812	0.812
8	0.000553	0.02348	0.0135	0.00758	0.815	0.815
9	0.000549	0.02341	0.0134	0.00749	0.816	0.816
10	**0.000546**	**0.02333**	0.0133	0.00743	**0.818**	**0.818**
15	0.000548	0.02338	**0.0132**	**0.00739**	0.817	0.817
20	0.000555	0.02353	0.0134	0.00746	0.814	0.814

Table 6: Performance of different Maximum Depths in RFR

Table 6 shows that as the value of maximum depths increases the values of MSE, RMSE, MAE and MDAE tends to decrease and value of EVS and R^2 tends to increase. Furthermore, it can also be inferred that RFR with number of estimators equal to 100 and maximum depth equal to 10 gives the best results.

6.6.2 Extra Trees Regression

Fig. 9 and Fig. 10 shows the impact of number of estimators on MSE and R^2 values for ETR.

It is evident from Fig. 9 and Fig. 10 that the MSE and R^2 values becomes inert as

the number of estimators goes beyond 80. Now fixing the number of estimators as 80, the maximum depth is changed from 2 to 20 for which the performance of it is shown in Table 7.

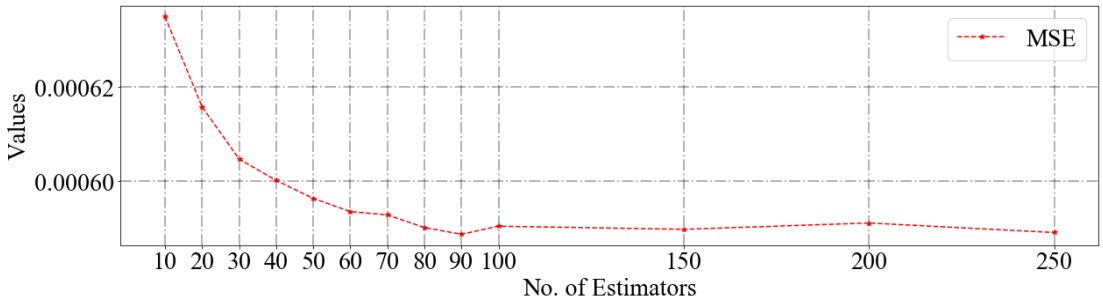

Figure 9: ETR with different number of estimators versus their corresponding MSE values

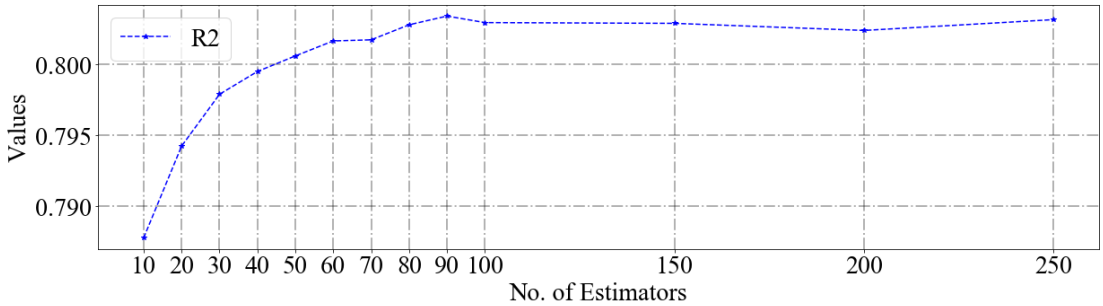

Figure 10: ETR with different number of estimators versus their corresponding R^2 values

Max Depth	**MSE**	**RMSE**	**MAE**	**MDAE**	**EVS**	**R^2**
2	0.001091	0.0329	0.0221	0.0177	0.636	0.636
3	0.000796	0.0281	0.0180	0.0121	0.733	0.733
4	0.000684	0.0261	0.0160	0.0099	0.771	0.771
5	0.000633	0.0251	0.0151	0.0089	0.788	0.788
6	0.000606	0.0246	0.0145	0.0084	0.797	0.797
7	0.000592	0.0242	0.0142	0.0081	0.802	0.802
8	0.000580	0.0241	0.0140	0.0079	0.806	0.806
9	0.000573	0.0238	0.0138	0.0078	0.809	0.809
10	0.000565	0.0237	0.0136	0.0076	0.811	0.811
15	**0.000561**	**0.0236**	**0.0132**	**0.0074**	**0.813**	**0.813**
20	0.000575	0.0239	0.0134	0.0075	0.808	0.808

Table 7: Performance of different Maximum Depths in ETR

From Table 7 it can be observed that maximum depth 15 performs better compared to

the other maximum depths. Similarly to RFR, the performance of ETR does not change much as the maximum depth increases.

6.6.3 Bagging Regression

The MSE and R^2 values for the BAR with respective number of estimators is shown in Fig. 11 and Fig. 12.

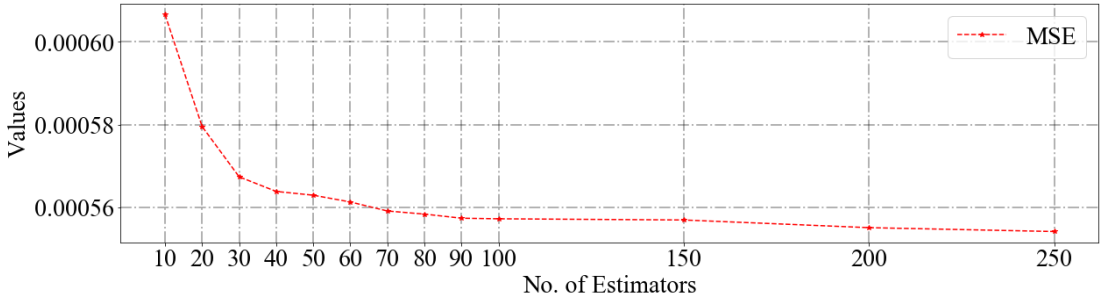

Figure 11: BAR with different number of estimators versus their corresponding MSE values

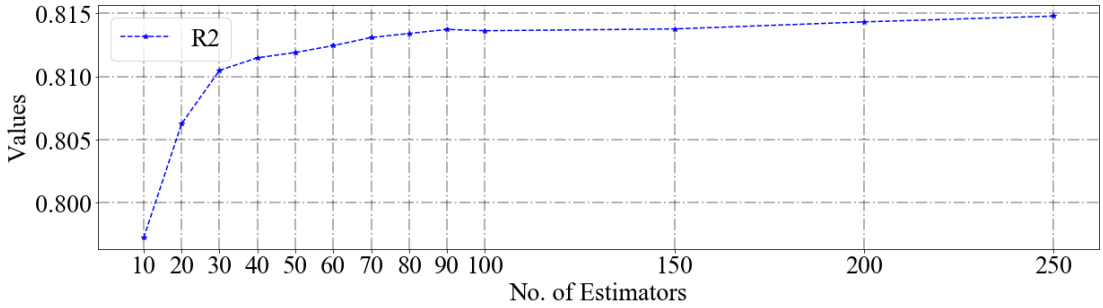

Figure 12: BAR with different number of estimators versus their corresponding R^2 values

We can infer from Fig. 11 and Fig. 12 that the values of MSE and R^2, becomes stagnant after 70 estimators; therefore 70 can be fixed as the number of estimators. To further optimise the result, the parameter **base_estimator** is tested with the best performing algorithm in Table 2 : Mulitple Linear Regression (MLR), Decision Tree Regression (max_depth = 6) (DTR(6)), Polynomial Regression (degree = 4) (PR(4)) and Support Vector Regression (kernel = linear) (SVR(L)).

Base Learner	MSE	RMSE	MAE	MDAE	EVS	R^2
MLR	0.000646	0.0254	0.0156	0.0101	0.784	0.784
DTR(6)	0.000576	0.0240	0.0138	0.0078	0.808	0.808
PR(4)	**0.000517**	**0.0227**	**0.0134**	**0.0076**	**0.827**	**0.827**
SVR(L)	0.001447	0.0380	0.0302	0.0267	0.697	0.515

Table 8: Performance of different Base Learners in BAR

Table 8 shows that BAR with base learner as Polynomial Regression (Degree = 4) performs better than the other base learners.

6.6.4 AdaBoost Regression

Fig. 13 and Fig. 14 shows the impact of number of estimators on MSE and R^2 values for ABR.

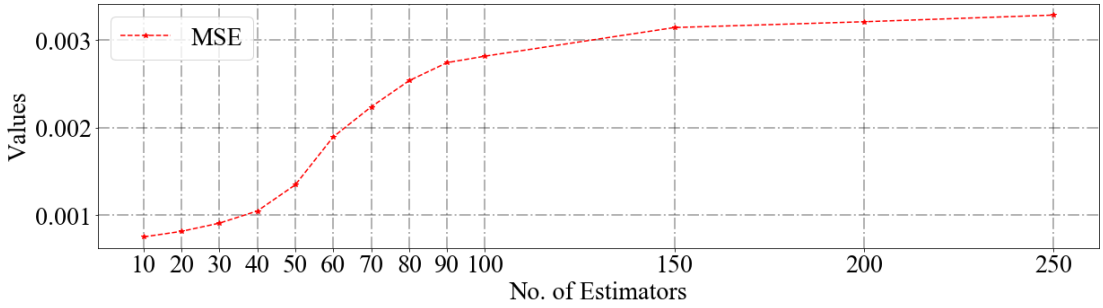

Figure 13: ABR with different number of estimators versus their corresponding MSE values

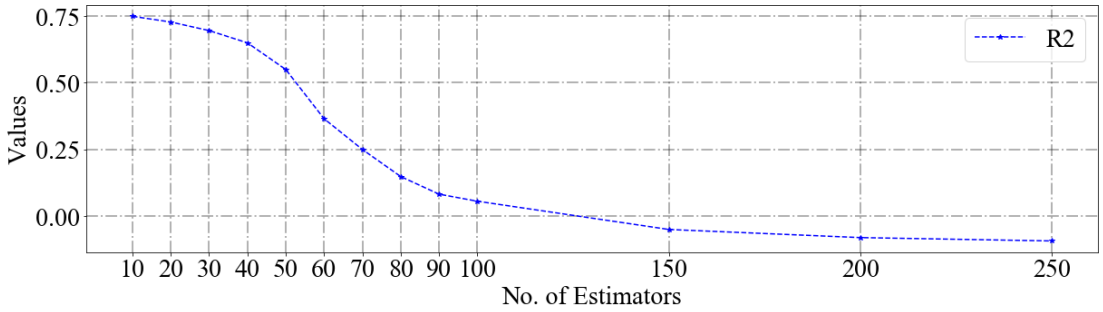

Figure 14: ABR with different number of estimators versus their corresponding R^2 values

It can be noted from Fig. 13 and Fig. 14 that as the number of estimators increases the MSE tends to increase and R^2 tends to decrease, indicating that the ensemble model has not served its purpose. Hence, it has not been explored further.

6.6.5 Gradient Boosting Regression

The MSE and R^2 values for the GBR with respective number of estimators is shown in Fig. 15 and Fig. 16.

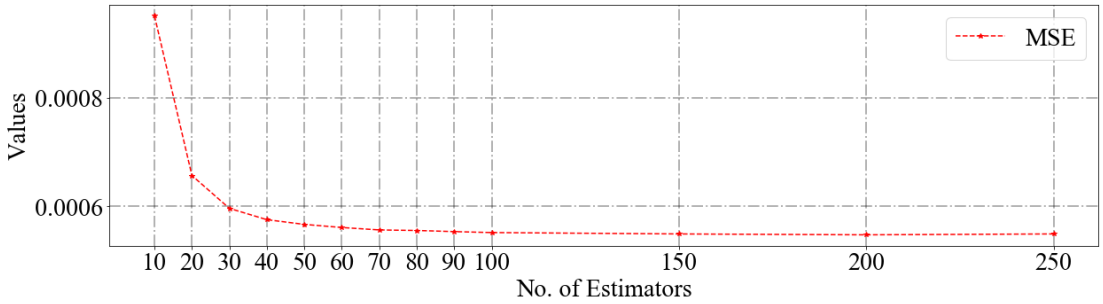

Figure 15: GBR with different number of estimators versus their corresponding MSE values

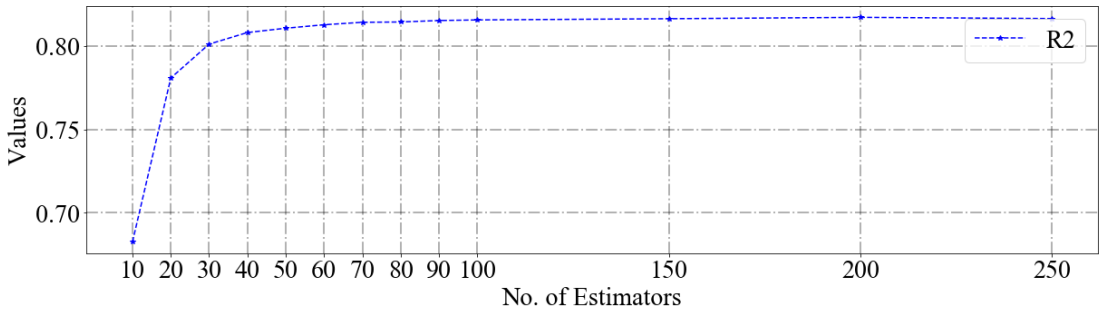

Figure 16: GBR with different number of estimators versus their corresponding R^2 values

From the graphs Fig. 15 and Fig. 16, 50 was chosen to be the optimal number of estimators for GBR. Now fixing the number of estimators as 50, the maximum depth is changed from 2 to 7 for which the performance of it is shown in Table 9.

Max Depth	MSE	RMSE	MAE	MDAE	EVS	R^2
2	0.00060	0.0246	0.0145	0.0080	0.796	0.796
3	0.00056	0.0237	0.0139	0.0077	0.810	0.810
4	0.00054	0.0233	0.0136	0.0076	0.818	0.817
5	**0.00053**	**0.0231**	0.0134	0.0075	**0.819**	**0.819**
6	0.00054	0.0233	0.0133	**0.0074**	0.817	0.817
7	0.00055	0.0234	**0.0132**	**0.0074**	0.815	0.815

Table 9: Performance of different Maximum Depths in GBR

It can be seen from Table 9 that out of the six performance measures the GBR with maximum depth five has performed well based on four performance measures. It can also

be noted that MDAE values for GBR with maximum depth six and maximum depth seven are the same indicating that the values tend to be stagnant after six. Also, the difference between the MAE and MDAE values for maximum depth five and other GBR models is minuscule. Hence it can be concluded that GBR with maximum depth five performs better than the others.

6.6.6 Extreme Gradient Boosting

Fig. 17 and Fig. 18 shows the impact of number of estimators on MSE and R^2 values for XGBR.

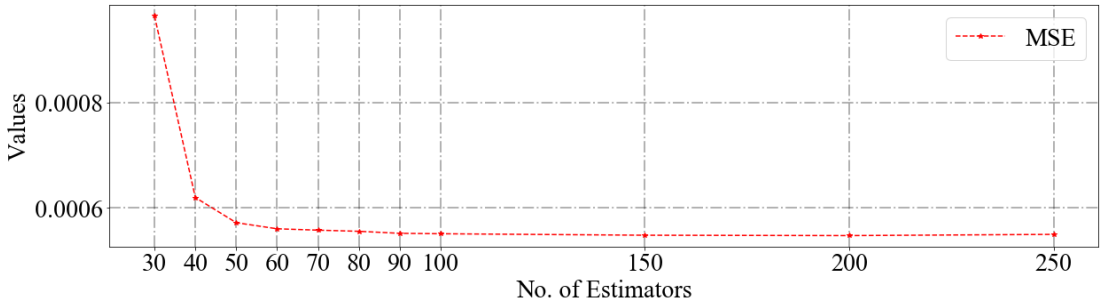

Figure 17: XGBR with different number of estimators versus their corresponding MSE values

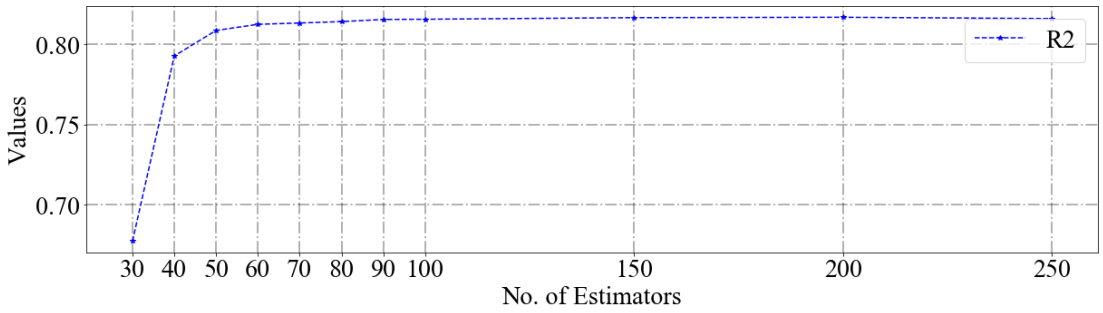

Figure 18: XGBR with different number of estimators versus their corresponding R^2 values

It is evident from Fig. 17 and Fig. 18 that the MSE and R^2 values becomes inert as the number of estimators goes beyond 50. Now fixing the number of estimators as 50, the maximum depth is changed from two to seven for which the performance of it is shown in Table 10.

Table 10 shows that XGBR with maximum depth as five performs better than the other XGBR models.

Max Depth	MSE	RMSE	MAE	MDAE	EVS	R^2
2	0.000615	0.0247	0.01528	0.00896	0.796	0.794
3	0.000574	0.0239	0.01465	0.00867	0.810	0.808
4	0.000550	0.0234	0.01430	0.00852	0.818	0.816
5	**0.000545**	**0.0233**	**0.01410**	**0.00838**	**0.820**	**0.818**
6	0.000549	0.0234	0.01400	0.00828	0.819	0.817
7	0.000554	0.0235	0.01395	0.00823	0.817	0.815

Table 10: Performance of different Maximum Depths in XGBR

6.7 Time Complexity Analysis of Ensemble Regression Models

Since the ensemble regression models in Section 6.6 performed similarly, the time complexity of the same is calculated and analysed to find the best model among these as shown in Table 11.

Models	Time taken for Training (sec)	Time taken for Testing (sec)
RFR	12.501	0.0343
ETR	2.840	0.053
GBR	2.143	0.006
XGBR	0.956	0.007
BAR	120.656	0.911

Table 11: Time Complexity Analysis of Ensemble Regression Models

The seconds in Table 11 indicate the average time taken per iteration by the models to train and test the data. It can be observed that XGBR takes the most minimal to train and test the data and BAR takes the most maximal to perform the same. Through which it can be concluded that XGBR performs best as a plain ensemble regression model.

6.8 Performance Analysis of Hybrid Ensemble Regression Models

A Hybrid Ensemble Model is a combination of two or more different ensemble models, combined to improve the performance of the same. Our initial intention was to find the best ensemble regression model, but the performance measures of all the optimised models were almost similar, which led us to build a hybrid ensemble regression model. For this purpose, we have taken optimised the ensemble regression models from Section 6.6 and tried to combine them with other ensemble techniques such as Simple Averaging, Weighted Averaging and Stacking. For comparing the performance of the above mentioned three models, the predicted results of optimised models on the training set and the testing set are saved in separate comma separated files, which also contains the actual rainfall value for the respective predicted values.

6.8.1 Simple Averaging

It is the process of averaging the selected models' predicted results and evaluating the averaged result with the actual dependent value. For this purpose, we have performed simple averaging with different combinations of ensemble regression models to find the best combination that predicts rainfall the best. Table 12 shows the performance of the different combinations of ensemble regression models using simple averaging.

It can be observed from Table 12 that BAR and GBR is the best combination of ensemble regression model using simple averaging for the chosen dataset. It can also be noted that in all the highlighted cells, BAR is present as one of the ensemble regression models.

6.8.2 Weighted Averaging

Simple averaging is a particular case of weighted averaging where all the weights are equal to one divided by the number of models. The advantage of using weighted averaging is that the user has the choice of entering more weight to a model that performs better so that it contributes more to the averaging process. In order to perform the weighted averaging more systematically, the weights for the models were chosen based on the R^2 values of the respective ensemble regression models. The formula for the weight is given in (9).

$$w_{\mathrm{i}} = \frac{{R^2}_{\mathrm{i}}}{\sum_{i=1}^{N} {R^2}_{\mathrm{i}}} \tag{9}$$

Where ${R^2}_{\mathrm{i}}$ is the R^2 value of i^{th} model and N is the number of models considered for weighted average. As the models mentioned in Section 6.6 performed almost equally good, the result of the weighted average version of the Hybrid Ensemble Regression Model does not differ much from the result of the simple average version.

6.8.3 Stacking

Stacking uses the prediction made on the training set by multiple models as input to a regression algorithm to predict the actual dependent attribute. The prediction various combinations of the ensemble regression models are taken as input for this purpose and MLR is used to predict the actual rainfall as shown Table 13.

Based on the values in Table 13 it can be inferred that ETR and XGBR perform the best compared to the other combinations. It can also be observed that the combination with minimal value in MSE, RMSE, MAE and MDAE is also the same combination with

Combinations	MSE	RMSE	MAE	MDAE	EVS	R^2
RFR	0.000544	0.02332	0.01329	0.00744	0.8182	0.8182
ETR	0.000556	0.02358	0.01326	0.00742	0.8143	0.8143
BAR	0.000518	0.02275	0.01336	0.00761	0.8270	0.8270
GBR	0.000539	0.02322	0.01338	0.00752	0.8199	0.8199
XGBR	0.000544	0.02331	0.01410	0.00839	0.8202	0.8184
RFR, ETR	0.000545	0.02335	0.01323	0.00740	0.8178	0.8178
RFR, BAR	0.000515	0.02270	0.01310	0.00729	0.8278	0.8278
RFR, GBR	0.000536	0.02316	0.01328	0.00746	0.8208	0.8208
RFR, XGBR	0.000537	0.02318	0.01359	0.00782	0.8209	0.8205
ETR, BAR	0.000519	0.02278	0.01308	**0.00728**	0.8265	0.8265
ETR, GBR	0.000538	0.02320	0.01323	0.00743	0.8201	0.8201
ETR, XGBR	0.000539	0.02322	0.01355	0.00779	0.8202	0.8197
BAR, GBR	**0.000513**	**0.02265**	0.01313	0.00735	**0.8285**	**0.8285**
BAR, XGBR	0.000514	0.02268	0.01339	0.00775	0.8285	0.8280
GBR, XGBR	0.000538	0.02320	0.01368	0.00789	0.8205	0.8201
RFR, ETR, BAR	0.000522	0.02285	0.01308	0.00728	0.8254	0.8254
RFR, ETR, GBR	0.000537	0.02319	0.01322	0.00742	0.8202	0.8202
RFR, ETR, XGBR	0.000538	0.02320	0.01342	0.00766	0.8203	0.8201
RFR, BAR, GBR	0.000517	0.02275	0.01310	0.00731	0.8270	0.8270
RFR, BAR, XGBR	0.000518	0.02276	0.01327	0.00754	0.8270	0.8268
RFR, GBR, XGBR	0.000535	0.02314	0.01349	0.00770	0.8212	0.8210
ETR, BAR, GBR	0.000518	0.02278	**0.01308**	0.00730	0.8266	0.8266
ETR, BAR, XGBR	0.000519	0.02279	0.01324	0.00753	0.8266	0.8264
ETR, GBR, XGBR	0.000536	0.02315	0.01345	0.00768	0.8210	0.8208
BAR, GBR, XGBR	0.000518	0.02276	0.01332	0.00760	0.8270	0.8268
RFR, ETR, BAR, GBR	0.000522	0.02285	0.01309	0.00731	0.8254	0.8254
RFR, ETR, BAR, XGBR	0.000522	0.02286	0.01321	0.00747	0.8254	0.8253
RFR, ETR, GBR, XGBR	0.000535	0.02315	0.01338	0.00760	0.8210	0.8209
RFR, BAR, GBR, XGBR	0.000520	0.02282	0.01326	0.00751	0.8261	0.8259
ETR, BAR, GBR, XGBR	0.000521	0.02283	0.01323	0.00750	0.8259	0.8258
RFR, ETR, BAR, GBR, XGBR	0.000523	0.02288	0.01322	0.00746	0.8251	0.8250

Table 12: Performance of the different combinations of ensemble regression models using simple averaging

maximal value in EVS and R^2. To enhance the performance of the stacking, the regression algorithms with better performance from Table 2 is shown in Table 14.

Combinations	MSE	RMSE	MAE	MDAE	EVS	R^2
RFR	0.000323	0.01797	0.01082	0.00624	0.8956	0.8955
ETR	0.000337	0.01836	0.01092	0.00630	0.8909	0.8908
GBR	0.000312	0.01766	0.01070	0.00620	0.8990	0.8990
XGBR	0.000297	0.01724	0.01072	0.00631	0.9039	0.9038
BAR	0.000336	0.01833	0.01199	0.00757	0.8912	0.8912
RFR, ETR	0.000321	0.01791	0.01078	0.00619	0.8963	0.8962
RFR, GBR	0.000310	0.01759	0.01064	**0.00613**	0.89986	0.8998
RFR, XGBR	0.000302	0.01738	0.01065	0.00615	0.9023	0.9022
RFR, BAR	0.000303	0.01742	0.01122	0.00689	0.9018	0.9018
ETR, GBR	0.000309	0.01758	0.01060	0.00617	0.9000	0.9000
ETR, XGBR	**0.000296**	**0.01722**	**0.01059**	0.00619	**0.9041**	**0.9040**
ETR, BAR	0.000310	0.01762	0.01138	0.00705	0.8995	0.8995
GBR, XGBR	0.000301	0.01735	0.01067	0.00626	0.9026	0.9026
GBR, BAR	0.000302	0.01737	0.01120	0.00687	0.9023	0.9023
XGBR, BAR	0.000297	0.01725	0.01120	0.00692	0.9037	0.9037
RFR, ETR, GBR	0.000309	0.01757	0.01062	0.00613	0.9001	0.9001
RFR, ETR, XGBR	0.000301	0.01734	0.01062	0.00614	0.9027	0.9026
RFR, ETR, BAR	0.000303	0.01742	0.01122	0.00689	0.9018	0.9018
RFR, GBR, XGBR	0.000304	0.01745	0.01063	0.00613	0.9015	0.9014
RFR, GBR, BAR	0.000300	0.01731	0.01114	0.00683	0.9030	0.9030
RFR, XGBR, BAR	0.000297	0.01724	0.01114	0.00684	0.9037	0.9037
ETR, GBR, XGBR	0.000301	0.01734	**0.01059**	0.00618	0.9027	0.9026
ETR, GBR, BAR	0.000301	0.01734	0.01117	0.00685	0.9026	0.9026
ETR, XGBR, BAR	0.000297	0.01724	0.01116	0.00686	0.9038	0.9038
GBR, XGBR, BAR	0.000300	0.01732	0.01119	0.00687	0.9029	0.9028
RFR, ETR, GBR, XGBR	0.000304	0.01742	0.01060	0.00613	0.9018	0.9017
RFR, ETR, GBR, BAR	0.000300	0.01731	0.01114	0.00684	0.9030	0.9029
RFR, ETR, XGBR, BAR	0.000297	0.01725	0.01114	0.00685	0.9037	0.9037
RFR, GBR, XGBR, BAR	0.000300	0.01731	0.01114	0.00683	0.9030	0.9030
ETR, GBR, XGBR, BAR	0.000300	0.01731	0.01116	0.00685	0.9030	0.9030
RFR, ETR, GBR, XGBR, BAR	0.000300	0.01731	0.01114	0.00685	0.9030	0.9030

Table 13: Performance of Stacking using MLR and different combinations of Ensemble Regression Models

The values in Table 14 indicate that using Polynomial Regression with degree as four with ETR and BAR as input models performs best compared to the other combinations

Combinations	MLR	PR(4)	DTR(6)	SVR(L)
RFR	0.0003229	0.0003128	0.0003249	0.0020901
ETR	0.0003372	0.0003211	0.0003612	0.0023848
GBR	0.0003120	0.0002993	0.0003050	0.0022134
XGBR	0.0002971	0.0002912	0.0003012	0.0021389
BAR	0.00033615	0.0003319	0.0003372	0.0018789
RFR, ETR	0.0003207	0.0003239	0.0003298	0.0021026
RFR, GBR	0.0003096	0.0002990	0.0003097	0.0019985
RFR, XGBR	0.0003020	0.0002961	0.0003156	0.0020423
RFR, BAR	0.0003034	0.0002800	0.0003157	0.0016469
ETR, GBR	0.0003090	0.0003026	0.0003153	0.0022194
ETR, XGBR	**0.0002966**	0.0003002	0.0003155	0.0022417
ETR, BAR	0.0003104	**0.0002740**	0.0003073	0.0018460
GBR, XGBR	0.0003009	0.0002897	**0.0003005**	0.0022121
GBR, BAR	0.0003017	0.0002797	0.0003097	0.0017552
XGBR, BAR	0.0002975	0.0002985	0.0003082	0.0017626
RFR, ETR, GBR	0.0003087	0.0003091	0.0003086	0.0019680
RFR, ETR, XGBR	0.0003008	0.0003250	0.0003286	0.0019868
RFR, ETR, BAR	0.0003034	0.0002878	0.0003045	0.0016915
RFR, GBR, XGBR	0.0003044	0.0002910	0.0003150	0.0019651
RFR, GBR, BAR	0.0002997	0.0002933	0.0003116	0.0017331
RFR, XGBR, BAR	0.0002974	0.0002933	0.0003087	0.0017056
ETR, GBR, XGBR	0.0003008	0.0002950	0.0003146	0.0021658
ETR, GBR, BAR	0.0003008	0.0002741	0.0003028	0.0017978
ETR, XGBR, BAR	0.0002971	0.0002831	0.0003041	0.0017220
GBR, XGBR, BAR	0.0003001	0.0003180	0.0003065	0.0017623
RFR, ETR, GBR, XGBR	0.0003035	0.0003047	0.0003275	0.0019471
RFR, ETR, GBR, BAR	0.0002998	0.0003328	0.0003031	0.0016921
RFR, ETR, XGBR, BAR	0.0002974	0.0002908	0.0003050	**0.0016264**
RFR, GBR, XGBR, BAR	0.0002996	0.0003424	0.0003052	0.0017228
ETR, GBR, XGBR, BAR	0.0002997	0.0003530	0.0003028	0.0017967
RFR, ETR, GBR, XGBR, BAR	0.0002997	0.0003687	0.0003030	0.0016920

Table 14: Performance of Stacking with different Regression Algorithms

and other regression algorithms as it did in Table 2.

On drawing a comparison between Hybrid Ensemble Regression Model using Simple Averaging versus Stacking, it can be concluded that Stacking performs twice as good as

the results produced by simple averaging and weighted averaging.

6.9 LSTM based Neural Network Analysis

A Neural network consists of complexly connected neurons which helps in learning to become more profound and learn essential details of the dataset. For this purpose, we are testing two different types of neural networks, one the Standard Neural Network and the other is Long Short Term Memory Neural Network. Both the above mentioned neural networks are trained in the same fashion, i.e. using Repeated K Fold Cross-Validation with ten splits and ten repeats, and the number of epochs is 10, and the batch size is 500 (For computation purposes).

According to a study in Artificial Intelligence [26], the major factors affecting a neural network are Network Complexity, Problem Complexity and Learning Complexity. To start with optimisation we need to find the right number of neurons in order to reduce over-fitting. Table 15 shows the performance comparison of LSTM with different set of neurons.

No. of Neurons	**MSE**	**RMSE**	**MAE**	**MDAE**	**EVS**	**R^2**
10	0.000609	0.02462	0.01419	0.00811	0.7974	0.7966
20	0.000603	0.02450	0.01417	0.00810	0.7995	0.7981
30	0.000652	0.02483	0.01462	0.00860	0.8035	0.7822
40	**0.000590**	**0.02425**	**0.01406**	**0.00808**	**0.8043**	**0.8028**
50	0.000611	0.02460	0.01431	0.00831	0.8010	0.7957

Table 15: Performance of LSTM for number of neurons in Layer 1

From Table 15 it can be inferred that the number of neurons required in layer one of LSTM is 40. The MSE value also indicates that it's performance is similar to the optimised version of the Polynomial Regression with degree four of the Generic-Regression Model and the optimized Ensemble Regression Models. However, on combining the Ensemble Regression Model using stacking, it was able to produce an MSE of 0.000266. In a similar way the LSTM neural network can also be optimised by increasing the number of layers in the network, using dropout regularisation to remove unwanted neurons to prevent overfitting randomly or by the activation function in the network.

7 CONCLUSION

In this project, we have developed a regression model that predicts rainfall with minimum error and captures sudden fluctuations in it. Based on the analysis, it was observed that the Generic-Regression Model using Polynomial Regression with degree 4 outperforms all the other models and predicts the rainfall in all the districts with comparatively low error rates. However, the Cluster-Based Model using Polynomial Regression captures variation in most of the districts and performs better than the Generic-Regression Model only by a fractional value. Hence, it can be concluded that Generic-Regression Model is the best model to predict rainfall for the state of Tamil Nadu, India.

Six ensemble models namely RFR, ETR, BAR, ADR, GBR and XGBR has been trained with entire states data as Generic-Regression model performed well. As all the models performed similarly, time complexity analysis was done to find the best model and XGBR happens to take minimum time to train and test. Finally, all the Ensemble models predictions are combined using Ensemble techniques like Simple Averaging, Weighted Averaging and Stacking. Stacking performs two times better compared to all Preliminary models, Simple Averaging, Weighted Averaging and Basic LSTM.

Also, on an analysis of variation of rainfall among the formed clusters, it was concluded that the districts in the eastern half and western half of the state have distinct patterns of rainfall across the months.

REFERENCES

[1] WTTC. Country Reports 2017 - India. Technical report, 2017.

[2] Jinghao Niu and Wei Zhang. Comparative analysis of statistical models in rainfall prediction. In *2015 IEEE International Conference on Information and Automation*, pages 2187–2190. IEEE, aug 2015.

[3] V.P Tharun, Ramya Prakash, and S. Renuga Devi. Prediction of Rainfall Using Data Mining Techniques. In *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pages 1507–1512. IEEE, apr 2018.

[4] Andrew Kusiak, Xiupeng Wei, Anoop Prakash Verma, and Evan Roz. Modeling and Prediction of Rainfall Using Radar Reflectivity Data: A Data-Mining Approach. *IEEE Transactions on Geoscience and Remote Sensing*, 51(4):2337–2342, apr 2013.

[5] Kesheng Lu and Lingzhi Wang. A Novel Nonlinear Combination Model Based on Support Vector Machine for Rainfall Prediction. In *2011 Fourth International Joint Conference on Computational Sciences and Optimization*, pages 1343–1346. IEEE, apr 2011.

[6] Sandeep Kumar Mohapatra, Anamika Upadhyay, and Channabasava Gola. Rainfall prediction based on 100 years of meteorological data. In *2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN)*, pages 162–166. IEEE, oct 2017.

[7] Sankhadeep Chatterjee, Bimal Datta, Soumya Sen, Nilanjan Dey, and Narayan C. Debnath. Rainfall prediction using hybrid neural network approach. In *2018 2nd International Conference on Recent Advances in Signal Processing, Telecommunications & Computing (SigTelCom)*, pages 67–72. IEEE, jan 2018.

[8] R. Venkata Ramana, B. Krishna, S. R. Kumar, and N. G. Pandey. Monthly Rainfall Prediction Using Wavelet Neural Network Analysis. *Water Resources Management*, 27(10):3697–3711, aug 2013.

[9] Mislan, Haviluddin, Sigit Hardwinarto, Sumaryono, and Marlon Aipassa. Rainfall Monthly Prediction Based on Artificial Neural Network: A Case Study in Tenggarong Station, East Kalimantan - Indonesia. *Procedia Computer Science*, 59:142–151, jan 2015.

[10] Aishwarya Himanshu Manek and Parikshit Kishor Singh. Comparative study of neural network architectures for rainfall prediction. In *2016 IEEE Technological*

Innovations in ICT for Agriculture and Rural Development (TIAR), pages 171–174. IEEE, jul 2016.

[11] Yajnaseni Dash, S.K. Mishra, and B.K. Panigrahi. Rainfall prediction of a maritime state (Kerala), India using SLFN and ELM techniques. In *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, pages 1714–1718. IEEE, jul 2017.

[12] Dennis van Heijst, Rob Potharst, and Michiel van Wezel. A support system for predicting eBay end prices. *Decision Support Systems*, 44(4):970–982, mar 2008.

[13] Alberto Torres-Barrán, Álvaro Alonso, and José R. Dorronsoro. Regression tree ensembles for wind energy and solar radiation prediction. *Neurocomputing*, 326-327:151–160, jan 2019.

[14] Seokho Kang and Pilsung Kang. Locally linear ensemble for regression. *Information Sciences*, 432:199–209, mar 2018.

[15] Yajnaseni Dash, Saroj Kanta Mishra, Sandeep Sahany, and Bijaya Ketan Panigrahi. Indian summer monsoon rainfall prediction: A comparison of iterative and non-iterative approaches. *Applied Soft Computing*, 70:1122–1134, sep 2018.

[16] Yu Xiang, Ling Gou, Lihua He, Shoulu Xia, and Wenyong Wang. A SVR-ANN combined model based on ensemble EMD for rainfall prediction. *Applied Soft Computing*, 73:874–883, dec 2018.

[17] Tomoaki Kashiwao, Koichi Nakayama, Shin Ando, Kenji Ikeda, Moonyong Lee, and Alireza Bahadori. A neural network-based local rainfall prediction system using meteorological data on the Internet: A case study using data from the Japan Meteorological Agency. *Applied Soft Computing*, 56:317–330, jul 2017.

[18] Yajnaseni Dash, Saroj K. Mishra, and Bijaya K. Panigrahi. Rainfall prediction for the Kerala state of India using artificial intelligence approaches. *Computers & Electrical Engineering*, 70:66–73, aug 2018.

[19] S. Renuga Devi, P. Arulmozhivarman, C. Venkatesh, and Pranay Agarwal. Performance comparison of artificial neural network models for daily rainfall prediction. *International Journal of Automation and Computing*, 13(5):417–427, oct 2016.

[20] Afan Galih Salman, Yaya Heryadi, Edi Abdurahman, and Wayan Suparta. Single Layer & Multi-layer Long Short-Term Memory (LSTM) Model with Intermediate Variables for Weather Forecasting. *Procedia Computer Science*, 135:89–98, jan 2018.

[21] Shiluo Xu and Ruiqing Niu. Displacement prediction of Baijiabao landslide based on empirical mode decomposition and long short-term memory neural network in Three Gorges area, China. *Computers & Geosciences*, 111:87–96, feb 2018.

[22] Duo Zhang, Geir Lindholm, and Harsha Ratnaweera. Use long short-term memory to enhance Internet of Things for combined sewer overflow monitoring. *Journal of Hydrology*, 556:409–418, jan 2018.

[23] Unjin Pak, Chungsong Kim, Unsok Ryu, Kyongjin Sok, and Sungnam Pak. A hybrid model based on convolutional neural networks and long short-term memory for ozone concentration prediction. *Air Quality, Atmosphere & Health*, 11(8):883–895, oct 2018.

[24] Qingxin Xiao, Weilu Li, Peng Chen, and Bing Wang. Prediction of Crop Pests and Diseases in Cotton by Long Short Term Memory Network. pages 11–16. Springer, Cham, aug 2018.

[25] Chieko Palanisami, Kuppannan and R. Ranganathan, C and Senthilnathan, S and UMETSU. Diversification of Agriculture in Coastal Districts of Tamil Nadu- a Spatio- Temporal Analysis. page 673, 2009.

[26] Factors Affecting the Performance of Artificial Neural Network Models. pages 51–85. Springer, Berlin, Heidelberg, 2008.

PUBLICATIONS

1. **Paper Title**: Forecast of Rainfall Quantity and its Variation using Environmental Features
   Authors: Preetham Ganesh, Harsha Vardhini Vasu and Dayanand Vinod
   Conference: 2019 2nd International Conference on Innovations in Power & Advanced Computing Technologies (i-PACT)
   Publisher: IEEE
   Date of Conference: 22nd and 23rd of March 2019
   Submission Date: 28th February 2019
   Status: Accepted and recommended for publication