# DROUGHT PREDICTION BASED ON SPI WITH VARYING TIMESCALES USING MACHINE LEARNING AND DEEP LEARNING MODELS

**A Project Report**

***Submitted by***


**Mr.JAYESH RAJ**

**REG NO : TKM20MEAI09**

**SEMESTER : IV**


***In partial fulfillment for the award of the degree of***

**MASTER OF TECHNOLOGY**
**IN**
**Mechanical Engineering (Artificial Intelligence)**

**Under the guidance of**

**Dr. Adarsh S**


**Thangal Kunju Musaliar College of Engineering**
**Kerala**


**JULY 2022**

# DECLARATION

I undersigned hereby declare that the project report "DROUGHT PREDICTION BASED ON SPI WITH VARYING TIMESCALES USING MACHINE LEARNING AND DEEP LEARNING MODELS", submitted for partial fulfillment of the requirements for the award of degree of Master of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of Dr. Adarsh S. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the university and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other university.

Place: Kollam
Date: JAYESH RAJ

## Thangal Kunju Musaliar College of Engineering
## Centre for Artificial Intelligence

# C E R T I F I C A T E

This is to certify that, this report titled ***DROUGHT PREDICTION BASED ON SPI WITH VARYING TIMESCALES USING MACHINE LEARNING AND DEEP LEARNING MODELS*** is a bonafide record of the **Project** completed by **JAYESH RAJ (TKM20MEAI09)**, under our guidance and supervision, in partial fulfillment of the requirements for the award of the degree, **M.Tech in Mechanical Engineering (Artificial Intelligence)** in **APJ Abdul Kalam Technological University** .

| Internal Supervisor | Project coordinator | Head of the Department |
|---|---|---|
| Dr. Adarsh S<br>Professor<br>Dept of Civil Engg<br>TKMCE | Prof. Sumod Sundar<br>Assistant Professor<br>Centre for Artificial Intelligence<br>TKMCE | Dr. Imthias Ahamed T.P.<br>Professor & HOD<br>Centre for Artificial Intelligence<br>TKMCE |

Internal Examiner

External Examiner

# ACKNOWLEDGEMENT

A successful project is a fruitful culmination of efforts by many people, some directly involved and some others indirectly, by providing support and encouragement. Firstly I would like to thank the almighty for giving me the wisdom and grace for making my project a successful one. I thank him for steering me to the shore of fulfillment under his protective wings

I express my sincere gratitude to **Dr. T A Shahul Hameed**, Principal of TKMCE, and **Dr. Imthias Ahamed T.P.**, Professor and Head of the Department, Centre for Artificial Intelligence, TKMCE, for their constant support and encouragement throughout the project work.

With a profound sense of gratitude, I would like to express my heartfelt thanks to my guide **Dr. Adarsh S**,Professor, Department of Civil Engineering, TKMCE, for his expert guidance, cooperation and immense encouragement. I would like to express my heartfelt thanks to our project coordinator **Prof.Sumod Sundar** Assistant Professor, Centre for Artificial Intelligence, TKMCE,for his constant support and encouragement throughout the project work. I also extend my thanks to the entire faculty and staff of the Centre for Artificial Intelligence, TKMCE, who has encouraged me throughout this work.

**JAYESH RAJ**

## Abstract

Drought modeling is an important issue because it is required for curbing or mitigating its effects, alerting the people to its consequences and water resources planning. The monitoring and forecasting of droughts plays a key role in the assessment of ecosystem health and mitigating the impact of extreme weather events on human society. This study investigates the capability of a deep learning method, Long Short-Term Memory (LSTM), in forecasting drought calculated from monthly rainfall data of Palakkad, Kasaragod and Punalur in Kerala. Due to the complexity of the drought phenomena and the requirements for its assessment, several indices have been developed and used in assessing drought events. Among these indices, SPI (Standardized Precipitation Index) is recommended by the WMO (World Meteorological Organization). Root Mean Square Error (RMSE), Mean Absolute Error (MAE) , Mean Square Error (MSE), Coefficient of Determination ($R^2$) , Radar Plot,Box Plot and Violin Plot of SPI-1, SPI-3, SPI-6 and SPI-12 for different models like Random Forest, Support Vector Regression (SVR) and LSTM are compared with each other in order to find the best model. The overall results showed that the LSTM method performed superior to the Random Forest and SVR in forecasting drought based on SPI-1, SPI-3, SPI-6 and SPI-12. From the study it is proven that SPI-12 shows better performance in LSTM time series prediction.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 General

Precipitation is a field that is random in character and as such makes drought prediction a complex task. Drought materializes from a deficiency of rainfall over a period of time. Drought can be classified as a meteorological drought, agricultural drought, hydrological drought and socio-economical drought. Drought over short timescales (months) characterizes meteorological drought, whereas long-term scales (years) showcase hydrological drought. Drought causes immense damage to the environment, economy and society. Agricultural droughts can lead to agricultural losses which in turn result in higher suicide rates among farmers, higher cost of food production, lower hydrological energy output and depleted water supply and tourism. They are also responsible for excessive heat waves, limitation of water supplies for consumption, high stress due to failed harvests, etc. Research-based knowledge, monitoring, prediction, management and mitigation have proven invaluable in reducing the extent and impact of drought on our economy and society. Therefore, there is a vital requirement to give an accurate prediction of drought occurrence. An early prediction can help in an early warning for drought management.

Droughts begin with a deficiency in precipitation, which then slowly propagates to a reduction in soil moisture conditions, causing agricultural drought and declines in streamflow, leading to hydrological drought and finally impacting society's social and economic aspects. Generally drought can be classified into four categories are (a) meteorological drought due to precipitation shortage (degree of dryness) over a certain period for a specific region, (b) hydrological drought due to the presence of below-average surface and subsurface flow for a longer time duration that accelerates inadequate water supply, (c) agricultural drought due to low soil water availability to support agricultural growth, and (d) socio-economic drought, which defines the imbalances in supply and demand of droughtdependent socio-economic commodities. The assessment and monitoring of drought using drought indices are more appropriate than the direct use of hydro-meteorological indicators.

More specifically, indicators are hydro-meteorological variables used to define drought situation such as rainfall and temperature. On the other hand, drought indices are obtained by numerically using hydro-meteorological inputs and the drought indicators. It should be noted that a drought variable should be able to quantify the drought for different time scales for which a long time series is essential. The most commonly used time scale for drought analysis is a year, followed by a month. Although the yearly time scale is long, it can also be used to abstract information on the regional behavior of droughts.

The ultimate objective of drought prediction is to prepare a mitigation plan in advance, rather than resolve intellectual curiosity about nature. Drought forecasting plays an important role in mitigating the negative effects of drought [1]; hence, various approaches for predicting droughts are stochastic methods, combined statistical and dynamical models, categorical prediction, machine learning approaches, deep learning and hybrid models. Agricultural droughts determined by soil moisture must be predicted several months ahead for proper and rapid resource allocation [1], because this allocation can mitigate the effects of upcoming droughts by supplying timely water and guaranteeing suitable crop growth and availability of food resources.

Meteorological drought is a consequence of complex individual interactions making it challenging to predict. Many droughts up to a month in advance and in rare cases it may be possible to predict drought conditions more than a year in advance. The main objective of modeling is to improve drought information systems by incorporating the latest advances in monitoring and prediction and advanced information delivery platforms. Also, a good model should be able to predict the occurrence, severity and duration of the drought accurately.

Through different machine learning (ML) models such as Support Vector Regression (SVR) and Random Forest(RF) have proved to make good predictions, efforts are made to find better models. Developments in deep learning and its ability to find a non-linear relationship between the parameters by mimicking the human brain sparks interest in creating more accurate models. This project investigates the capability of a deep learning model, long short-term memory (LSTM) in outperforming previous ML models that have proved to be efficient in time series forecasting

## 1.2 Objectives

The objectives of this project are as follows:

- To develop LSTM model for the prediction of Short, Medium and Long-term droughts in Palakkad,Kasaragod and Punalur in Kerala.
- To compare the performance of the deep learning model (LSTM) with Machine Learning models such as Random Forest, Support Vector Regression for drought prediction.

## 1.3 Organization of work

All the procedures and steps adopted to complete the project are explained in this report. The report is organized into six chapters. Chapter 1 gives an introduction to the topic of the project, its objectives. Chapter 2 deals with the literature relevant to this study. Chapter 3 provides the proposed methodology for the work and Chapter 4 provides a basic idea about the study region. Chapter 5 gives the results obtained from the project. Chapter 6 provides the conclusion of the report.

# Chapter 2

# Literature review

## 2.1 General

This chapter presents the Literature Review of different types of drought, different types of drought indices and prediction of drought index using Machine Learning and Deep Learning.

## 2.2 Drought indices and prediction models

There are many works done focused on this drought prediction. All the researchers and research papers focused on or follow for defining drought is that Drought simply means it is the deficiency of precipitation. Precipitation is random in characteristics. So, the prediction of drought is a complex task. But it is very essential to have a drought monitoring system to predict drought. For this task deep learning and machine learning techniques can predict more efficiently by drought indices.

**Mckee et al. (1993)**[2]Introduced the work" The relationship of drought frequency and duration to time scales.A new definition of drought has been proposed which explicitly specifies time scales and utilizes a standardized precipitation index. Drought frequency decreases inversely and duration increases linearly with time scale. Frequency and duration of random climate and actual climate are very similar.The new definition allows a consistent set of information to be calculated including drought beginning, ending, intensity, and magnitude. It also produces monitoring information of index values,probability, percent of average, and precipitation deficit during drought.
**Zargar A et al.(2011)**[3] There are hundreds of drought indices have so far been proposed, some of which are operationally used to characterize drought using gridded maps at regional and national levels. These indices correspond to different types of drought, including meteorological, agricultural,and hydrological drought. By quantifying severity levels and declaring drought's start and end, drought indices currently aid in a variety of operations including drought early warning and monitoring and contingency planning. Given their variety and ongoing development, it is crucial to provide a comprehensive overview of all available drought indices and their difference and examines the trend in their development. This paper reviews 74 operational and proposed drought indices and describes research directions.
**Agana and Homaifar(2017)**[4] studied a drought prediction model with a deep belief network with two Restricted Boltzmann machines (RBM). The performance are evaluated

by RMSE and MAE values. The deep belief network with two RBM are compared with classical MLP and SVR. Where the Deep belief network shows fewer error values than MLP and SVR.

**Poornima and Pushpalatha (2019)**[5]Proposed a work "Drought prediction based on Standardized Precipitation Index and Standardized Precipitation Evapotranspiration Index with varying time scales using LSTM recurrent neural network", where they developed 12 models of LSTM were, the first input of LSTM is given SPI and SPEI values and second input as average humidity and average temperature with respect to 1,3,6 timescales and the performance of LSTMs are compared with classical ARIMA models. The result shows that LSTM gives a better prediction and less error than the classical ARIMA model.

**Karimi et al. (2019)**)[6] analyzed the meteorological drought at seven stations of Karkheh Basin in Iran using monthly precipitation data and SPI index for 1, 3, 6, 9, and 12 monthly timescales. Forecasting of SPI3 time series was performed using ARIMA models. Based on the results, the ARIMA model was considered a useful tool for forecasting drought. The use of the other forecasting such as wavelet transforms, support vector machine (SVM), and ANN were suggested to forecast drought for future studies.

**Kaur and Sood (2020)**)[7]Introduced a work based on "Deep learning-based drought assessment and prediction framework" a new framework that predicts drought has a data collection layer where the input data are collected and stored. A fog layer helps to reduce the dimensions and then passed to the loud layer where the severity of the drought is evaluated. ANN optimized with a Genetic algorithm. Then they are passed to DNN which performs a comparison at last SVM predicts the drought.

**Bouaziz et al. (2021)**[8] studied the effectiveness of the Extreme Learning Machine (ELM) in Standardized Precipitation Index (SPI) in various timescales to classify and track drought events based on Climate Hazards Group InfraRed Precipitation with Station (CHIRPS) rainfall data for the period between 1981 and 2019 over different time scales (1, 3, 6, 9, 12, 15, 18, and 24 months). Forecasting of meteorological droughts was based on using SPI as input for ELM algorithms. This study confirms the utility of using SPI with ELM to forecast meteorological drought and detect temporal patterns of drought with a high coefficient of 6 determination (R2). The computed SPI values from this study help to exhibit high reliability between wet and dry periods over the different time scales, and the amplitude of fluctuations decreases with increasing time scale.

**Pham et al. (2021)**[9] Studied Coupling singular spectrum analysis with a least square support vector machine to improve the accuracy of drought forecasting. In this study they propose a model to forecast drought SPI based on singular spectrum analysis (SSA) and least square support vector machine (LSSVM) with two case evaluation forecasting per- performances are evaluated of the LSSVM-based model with and without coupling SSA and for both models, different inputs like antecedent SPIs and antecedent accumulated monthly rainfall are used and they are pre-processed by SSA. Two cases are considered for the test, the first case is about LSSVM using antecedent SPI as input (LSSVM1). Another model in which LSSVM coupled with SSA is considered an antecedent SPI as inputs were developed (LSSVM2) and in another case, SSA-LSSVM-based models using antecedent accumulated monthly rainfall are considered as inputs. Then compared to SSA-LSSVM2 then predicted for SPI1 and SPI3 timescales. The selected sub-division was the Tseng-Wen reservoir catchment in southern Taiwan and the result shows that SSA-LSSVM2 is better than LSSVM1 which shows that preprocessed by SSA can increase the accuracy of the other case, SSA-

LSSVM2 and SSA-LSSVM3. Where SSA-LSSVM3 show better performance.

## 2.3 Summary

In a place like Kerala, there are diverse climatic conditions, which make it very difficult to predict drought.Time series Artificial intelligence(AI) models can be used for effective drought prediction in these areas.Multivariate approaches are rarely used for drought predictions.So models considering multiple inputs would be a great advancement for time series drought prediction.

# Chapter 3

# Methodology

## 3.1 Overview of Standardized precipitation index

Many indices are used to predict drought depending upon the situation, among them is Standardized precipitation index[SPI][9].SPI is based on the percentage values, therefore sometimes it becomes complex. The reason for developing the index is to get the index to avoid complex calculations and these index values can be used in regions all over the world. The deficiency of rainfall affects many water-related phenomena like soil moisture, humidity, temperature, groundwater etc.., and this makes them develop the SPI index's values are simple in the calculation, fairly easy to use, and flexible and the only requirement is that need is the time series precipitation values. These indexes are very effective to detect drought and flood in an equal range of measurements.SPI values were implemented from precipitation values depending upon different timescales, through those values we can obtain the severity of drought.SPI calculations for a region are developed from long-term precipitation values. These long-term precipitation values fit in a probability distribution and transformed into a normal distribution. The mean value of the series is always zero. Here, the positive values show the wet conditions and negative values show the dry conditions. The values near zero give the normal condition. As shown in Table 3.1 [2]

### 3.1.1 Computation of SPI

The index is the number of standard deviations got from the observed precipitation deviating from the long-term mean, for the normal distribution. The distribution of precipitation is not normally distributed initially. So, precipitation must be transformed using gamma function [10] or normal distribution or Pearson distribution. The SPI values show the deviation of precipitation. The different timescales help in taking decisions on various water-associated issues values can be calculated from 1 to 72 months timescales. But, commonly used best timescales were 1-24months. for 50-60 years we can use 24month or more. Considering the 42 years (1979-2021) time series SPI-1, SPI-3, SPI-6, and SPI-12 are used for this work.

SPI is very less complex and fairly easy to calculate. The only requirement to calculate SPI values is the precipitation values. The SPI is applicable in all climate regions, and SPI values for very different climates can be compared. The ability of SPI to be computed for short periods of record that contain missing data is also valuable for those regions that may

be data-poor or lacking long term, cohesive datasets. The program used to calculate SPI is easy to use and readily available.
The SPI values can be calculated as

$$SPI = \ (t - \frac{c_0 + c_1 t + c_2 t^2}{1 + d_1 t + d_2 t^2 + d_3 t^2}); t = \sqrt{\log \frac{1}{(H(x))^2}}; 0 < H(x) \leq 0.5 \tag{3.1}$$

$$SPI = \ t - \frac{c_0 + c_1 t + c_2 t^2}{1 + d_1 t + d_2 t^2 + d_3 t^2}; t = \sqrt{\log \frac{1}{(1 - (H(x)))^2}}; 0.5 < H(x) < 1 \tag{3.2}$$

where "x" is the monthly rainfall, c0 = 2.515517, c1 = 0.802853, c2 = 0.010328, d1 = 1.432788, d2 = 0.189269, d3 = 0.001308, and "H(x)" is the average likelihood of the data series being translated into an incomplete gamma distribution function. The distribution of the gamma function is expressed as follows:

$$g(x) = \frac{1}{\beta^\alpha I(\alpha)} x^{\alpha - 1} e^{-x\beta^{-1}}; x > 0 \tag{3.3}$$

### 3.1.2 Types of SPI

SPI can be calculated from several time periods(timescales)

- 1-month SPI: SPI-1 values mean, for one month shows the 30-days period representing the normal precipitation percentage. Here, the SPI is an accurate representation of precipitation distribution because the distribution is normalized. So, a 1-month period means, it compares all the months in the year. Therefore, they show a short-term effect and these are applicable to the crop loss problems related to precipitation.

- 3-months SPI: They are used for computing for shorter accumulation periods and can be used as an indicator for immediate impacts for reduced soil moisture, snowpack and flow in smaller creeks

- 6-months SPI: SPI6 is dealing with the same 6 months period.For example, January is the start all the January to June are taken into consideration every year. This type of SPI shows a seasonal to medium-term trend in rainfall.

- 12-month SPI: SPI-12 is very much similar to SPI-3 and SPI-6 .here considering 12 consecutive months in the data. Which takes all month's data in the year to calculate the SPI-12 value. they are related to streamflow, reservoir levels and groundwater level at longer time scales.

Table 3.1: Drought classification by SPI value

| | |
|---|---|
| 2.0+ | Extremely wet |
| 1.5 to 1.99 | Very wet |
| 1.0 to 1.49 | Moderately wet |
| − .99 to .99 | Near normal |
| − 1.0 to − 1.49 | Moderately dry |
| − 1.5 to − 1.99 | Severely dry |
| − 2 and less | Extremely dry |

## 3.2 Long Short-Term Memory (LSTM)

Among the various types of recurrent neural networks[12] (RNNs), LSTM is recognized as an advanced form of it, which can cover the flaws of general RNN structure through long-term dependency learning.

LSTM is the advanced version of Recurrent neural network[12](RNN).Hochreiter and Schmidhuber proposed LSTM for the first time in 1997, although it has been improved and generalized progressively by numerous scholars.RNN suffer from short-term memory due to a vanishing gradient problem. To avoid this problem LSTM is developed, which can work with longer data sequences' and can solve the vanishing gradient problem of RNNs.LSTM is a more advanced version of RNNs that can preserve important information from the earlier part of the sequence and carry it forward.LSTM recurrent unit is more complex than that of RNN, they improve learning but required more computational resources. The key elements in the LSTM are cell, input gates, output gates and forget gates.

- Cells: it is also known as the memory state, which is like an information highway It is a chain structure, that contains four neural networks and many memory blocks called cells. Information is retained by cells and memory manipulation is done by the gates.
- Forget gates: forget gates are gates which can manipulate memory and retain by cells In LSTM the information that is not useful in the cell state is removed by the forget gate. the unwanted or no longer used are ignored.
- Input gates: additional information for the cell state is given by Input gates. they decide what information is given to the cell. The information is regulated using the sigmoid function and filters.
- Output gates:extract the useful information from cell and output is given.

The central role of an LSTM model is held by 'cell state' that maintains its state over time.Which is a conveyor belt through which information just flows, unchanged.Information can be added to or removed from the cell state in LSTM and is regulated by gates. These gates optionally let the information flow in and out of the cell. It contains a point wise

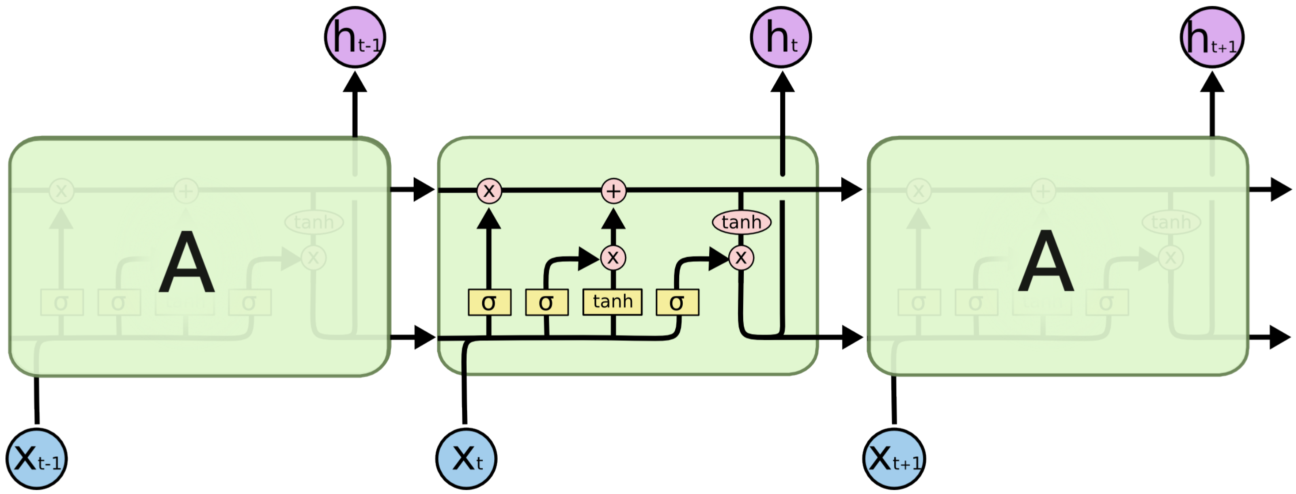


Figure 3.1: LSTM cell

multiplication operation and a sigmoid neural net layer that assist the mechanism.The sigmoid layer gives out numbers between zero and one, where zero means ‘nothing should be let through,’ and one means ‘everything should be let through.’ The LSTM architecture is shown in Figure 3.1 LSTM is a deep learning algorithm which was implemented using the TensorFlow library.

- Units: They are Positive integer, dimensionality of the output space.
- Activation: Activation function to use.The Default activation function is tanh. If you pass None no activation will applied
- recurrent activation:Activation function to use for the recurrent step. Default: sigmoid (sigmoid). If you pass None, no activation is applied
- use bias: Boolean (default True), whether the layer uses a bias vector.
- kernel initializer: Initializer for the kernel weights matrix, used for the linear transformation of the inputs. Default: glorot uniform.
- recurrent initializer:Initializer for the recurrent kernel weights matrix, used for the linear transformation of the recurrent state. Default: orthogonal.
- bias initializer:Initializer for the bias vector. Default: zeros.
- unit forget bias Boolean (default True). If True, add 1 to the bias of the forget gate at initialization. Setting it to true will also force bias initializer=”zeros”.
- kernel regularizer Regularizer function applied to the kernel weights matrix. Default: None.
- recurrent regularizer Regularizer function applied to the recurrent kernel weights matrix. Default: None.

- bias regularizer Regularizer function applied to the bias vector. Default: None.
- activity regularizer Regularizer function applied to the output of the layer (its "activation"). Default: None.
- kernel constraint Constraint function applied to the kernel weights matrix. Default: None.
- recurrent constraint Constraint function applied to the recurrent kernel weights matrix. Default: None.
- bias constraint Constraint function applied to the bias vector. Default: None.
- Dropout Float between 0 and 1. Fraction of the units to drop for the linear transformation of the inputs. Default: 0.
- recurrent dropout Float between 0 and 1. Fraction of the units to drop for the linear transformation of the recurrent state. Default: 0.
- return sequences Boolean. Whether to return the last output. in the output sequence, or the full sequence. Default: False.
- return state Boolean. Whether to return the last state in addition to the output. Default: False.
- go backwards Boolean (default False). If True, process the input sequence backwards and return the reversed sequence.
- Stateful Boolean (default False). If True, the last state for each sample at index i in a batch will be used as initial state for the sample of index i in the following batch.
- time major The shape format of the inputs and outputs tensors. If True, the inputs and outputs will be in shape [timesteps, batch, feature], whereas in the False case, it will be [batch, timesteps, feature]. Using time major = True is a bit more efficient because it avoids transposes at the beginning and end of the RNN calculation. However, most TensorFlow data is batch-major, so by default this function accepts input and emits output in batch-major form.
- Unroll Boolean (default False). If True, the network will be unrolled, else a symbolic loop will be used. Unrolling can speed-up a RNN, although it tends to be more memory-intensive. Unrolling is only suitable for short sequences.
- Inputs A 3D tensor with shape [batch, timesteps, feature].
- Mask Binary tensor of shape [batch, timesteps] indicating whether a given timestep should be masked (optional, defaults to None). An individual True entry indicates that the corresponding timestep should be utilized, while a False entry indicates that the corresponding timestep should be ignored.
- Training Python boolean indicating whether the layer should behave in training mode or in inference mode. This argument is passed to the cell when calling it. This 20 is only relevant if dropout or recurrent dropout is used (optional, defaults to None).

- initial state List of initial state tensors to be passed to the first call of the cell (optional, defaults to None which causes creation of zero-filled initial state tensors)

## 3.3 Support Vector Regression(SVR)

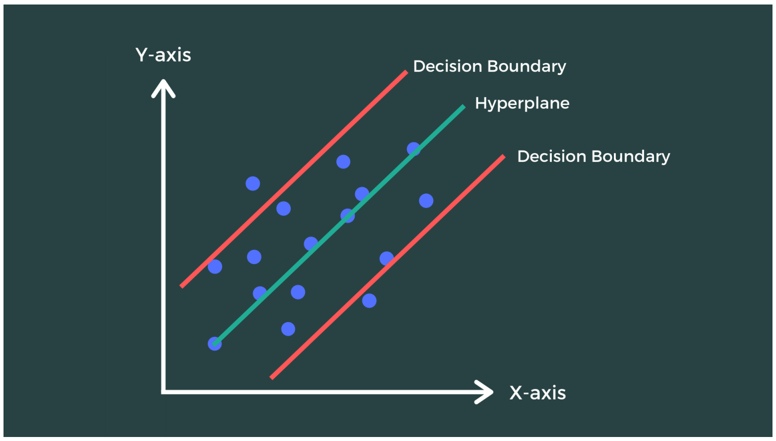


Figure 3.2: SVR architecture

Support Vector Regression is a popular machine learning model.Which was initially proposed by Drucker et al. in 1996. SVR is associated with learning algorithms that analyse data for classification and regression analysis. SVR can be said that it is a combination of SVM with regression. They can be used in classification problems or assigning classes. when the given data are not linearly separable. In other words, SVR can be said that SVMs that solve the regression problems are called SVR. Commonly used three kernels in SVR are Linear kernel, Polynomial kernel, and Radial basis function. The basic sample model for Support Vector Regression is shown in Figure 3.2

kernels used for SVR

- Linear kernel: they calculate the dot products between two given observations. Compared to others Linear kernel shows less performance
- Polynomial kernel: They allow curved lines in the input space and they show moderate performance
- Radial basis function: they create a complex region within the feature space. Always shows better results as compared to other kernels.

The parameters in the model are C and epsilon.

- C(complexity parameter):They are float type, the default value is 1.0 It is a Regularization parameter. The strength of the regularization is inversely proportional to C. Must be strictly positive.
- epsilon : It is float type the default value is 0.1 Epsilon in the epsilon-SVR model. It specifies the epsilon-tube within which no penalty is associated in the training loss function with points predicted within a distance epsilon from the actual value.

## 3.4 Random Forest (RF)

The random forest (RF) method is an ensemble learning technique proposed by Leo Breiman and Adele Cutler in 1996. It has been effectively used in dealing with a number of prediction problems. It is a machine-learning algorithm that combines a giant set of selection trees to improve the prediction overall performance of the classification and regression trees (CART) method. Each decision tree of RF is grown by means of the use of a randomly selected bootstrap sample from the original data set, and the final result of RF is the average end result of all the trees. Compared to the regression methods, the number of parameters needed to be defined in the RF is very few. There are only two essential parameters, consisting of the number of variables used in every tree-building technique and the range of trees built in the forest. The quantity of trees constructed in the forest has a great impact on the end result of RF. The inadequate quantity of trees would end result in poor forecasting performance, while the excessive number of trees might also lead to problematic predictors.

The parameters represent different properties such as

- n estimators : int, default=100 The number of trees in the forest
- criterion: "squared error", "absolute error", "poisson" .The function to measure the quality of a split. Supported criteria are "squared error" for the mean squared error, which is equal to variance reduction as feature selection criterion, "absolute error" for the mean absolute error, and "poisson" which uses reduction in Poisson deviance to find splits. Training using "absolute error" is significantly slower than when using "squared error".
- max depth : int, default=None The maximum depth of the tree. If None, then nodes are expanded until all leaves are pure or until all leaves contain less than min samples split samples.
- min samples split int or float, default=2 .The minimum number of samples required to split an internal node
- min samples leaf : int or float, default=1 .The minimum number of samples required to be at a leaf node. A split point at any depth will only be considered if it leaves at least min samples leaf training samples in each of the left and right branches. This may have the effect of smoothing the model, especially in regression.
- min weight fraction leaf : float, default=0.0 .The minimum weighted fraction of the sum total of weights (of all the input samples) required to be at a leaf node. Samples have equal weight when sample weight is not provided.
- max features : "sqrt", "log2", None, int or float, default=1.0 .The number of features to consider when looking for the best split
- max leaf nodes : int, default=None .Grow trees with max leaf nodes in best-first fashion. Best nodes are defined as relative reduction in impurity. If None then unlimited number of leaf nodes.
- min impurity decrease float, default=0.0 .A node will be split if this split induces a decrease of the impurity greater than or equal to this value. 12

- bootstrap : bool, default=True Whether bootstrap samples are used when building trees. If False, the whole dataset is used to build each tree.
- oob score : bool, default=False .Whether to use out-of-bag samples to estimate the generalization score. Only available if bootstrap=True.
- n jobs : int, default=None.The number of jobs to run in parallel.
- random state : int, RandomState instance or None, default=None .Controls both the randomness of the bootstrapping of the samples used when building trees (if bootstrap=True) and the sampling of the features to consider when looking for the best split at each node (if max features less than n features).
- verbose : int, default=0 .Controls the verbosity when fitting and predicting.
- warm start : bool, default=False .When set to True, reuse the solution of the previous call to fit and add more estimators to the ensemble, otherwise, just fit a whole new forest.
- ccp alpha : non-negative float, default=0.0 .Complexity parameter used for Minimal Cost-Complexity Pruning. The subtree with the largest cost complexity that is smaller than ccp alpha will be chosen. By default, no pruning is performed.
- max samples : int or float, default=None .If bootstrap is True, the number of samples to draw from X to train each base estimator.

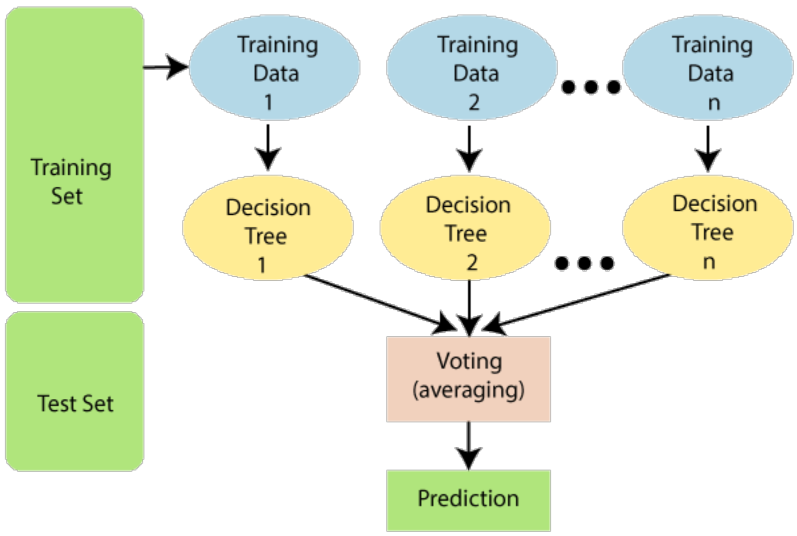


Figure 3.3: Random forest architecture

RF is a supervised learning model that uses the ensemble learning technique for regression. It consists of an aggregate of more than one model tree algorithm to make a greater accurate prediction.It commonly performs excellently on many problems, together with features with non-linear relationships. Disadvantages: there is no interpretability, overfitting can also easily occur, and we need to pick out the wide variety of trees to consist of in the model. The general model of the Random Forest is shown in Figure 3.3 .

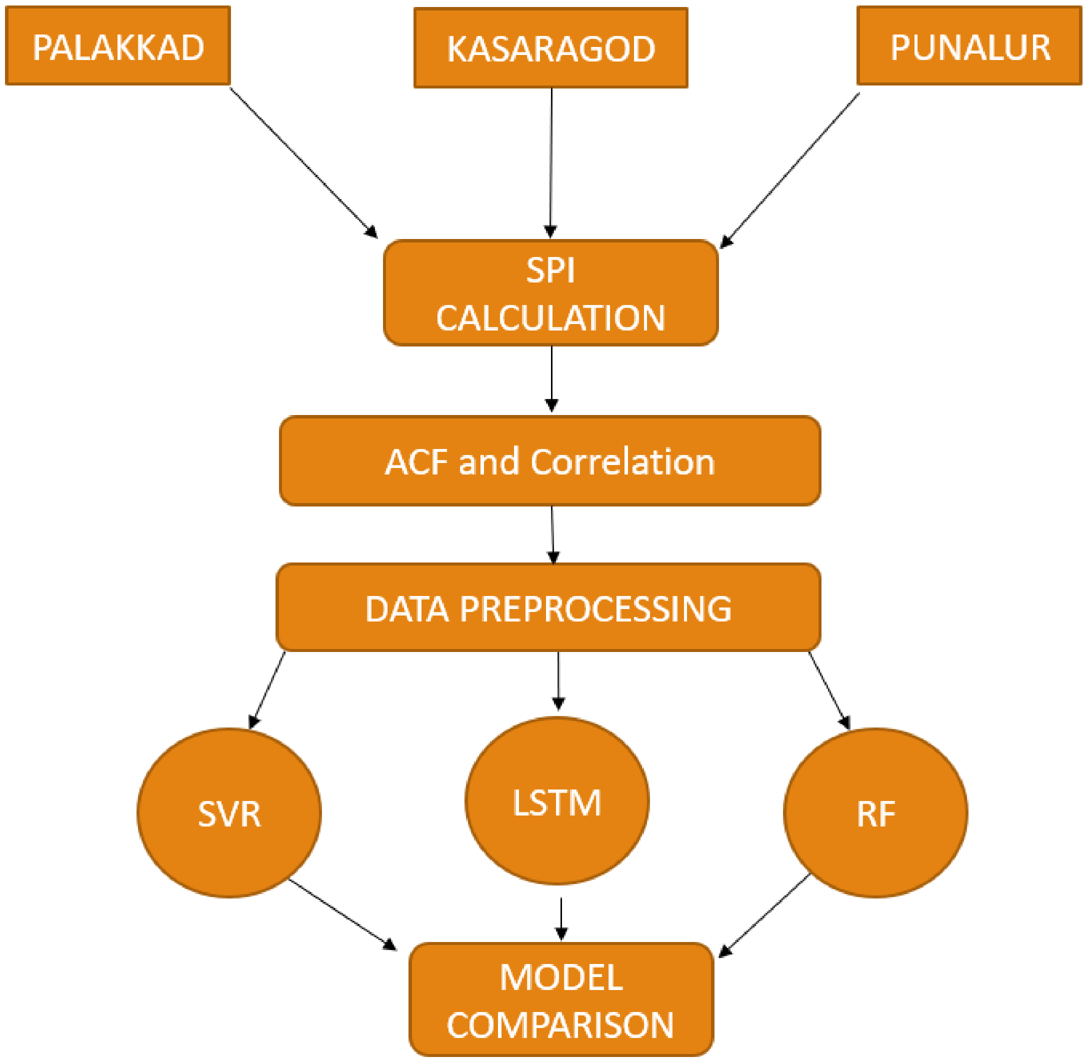


Figure 3.4: Work flow

### 3.4.1 Work flow

The flow of work is shown in Figure 3.4. Initially collected three monthly datasets (Kasaragod, Palakkad and Punalur) separately. Then its SPI-1, SPI-3, SPI-6 and SPI-12 values are calculated from precipitations with respect the corresponding dates of precipitations. Then required lags are set to SPI values with respect to auto-correlation function and correlation co-efficient to other features. Then, pre-processed data are given as input to models (LSTM, SVR and RF) separately. Finally, the predicted outputs performance are compared.

# Chapter 4

# Study region and data collection


Figure 4.1: Selected Locations

## 4.1 Dataset description

The Meteorological data of the Palakkad,Kasaragod and Punalur locations are obtained from the NASA POWER official website(https://power.larc.nasa.gov/data-access-viewer/). Consists of various climatic parameters corresponding to the selected longitudes and latitudes of locations. The meteorological data are collected monthly. Selected features are precipitation, maximum temperature, minimum temperature, relative humidity, soil moisture, root-soil moisture, and surface soil wetness. From the long-term precipitation data, SPI-1, SPI-3, SPI-6 and SPI- 12 values are calculated using R packages. Then arranged the SPI values with respect to corresponding months. The map location is shown in Figure 4.1

- Palakkad:Latitude: 10°4627 N and Longitude: 76°3922" E
- Kasaragod:Latitude of 12°30'36.81" N and Longitude 74°59'6.6"E
- Punalur:Latitude of 9°0'35.16" N and Longitude 76°55'47.05" E

# Chapter 5

# Results and discussions

## 5.1 LSTM Model development

The LSTM RNN model proposed in this work is about a multivariate case where maximum temperature, relative humidity, soil moisture, root-soil moisture, and surface soil wetness are inputs given to the LSTM to predict the corresponding SPI values corresponding to the timescale. This work uses one layer of LSTM which consists of 1000 cells. The back-propagation through time. A dropout layer is included between the two hidden layers for regularization. It will randomly exclude 50% of the activations of the previous layer from propagating to prevent overfitting. The root mean square (RMS) loss is reduced using the Adam optimizer which can handle sparse gradients on the noisy dataset and little memory is enough, therefore using Adam gives more memory efficiency.LSTM model with three datasets is checked with four cases of timescales(SPI-1, SPI-3, SPI-6, SPI-12)

## 5.2 SVM Model development

Support Vector Regression proposed in this work is used radial basis function as the kernel, which is the best and most efficient kernel than the linear kernel and the polynomial kernel is very much popular because of its similarity to the K-nearest neighbourhood algorithm. RBF can overcome the space complexity problem as RBF kernel SVR just needs to store the support vectors during training and not the entire dataset. The proposed SVR model in this work uses a Batch size of 100 and then used a c-value (complexity parameter) of 1.0

## 5.3 RF Model development

Random forest is a type of supervised learning method.RF can be said a random decision forest, It is an ensemble learning approach for both classification and regression.The parameters used in Random forest model are n estimators = 200 and Random state = 1. They create a multitude of decision trees at training time.RF works based on the idea of ensemble learning. They mix multiple classifiers to solve a complicated problem and to enhance the overall performance of the model. When the quantity of trees in the forest leads to greater accuracy and prevents the trouble of overfitting.

## 5.4 Data preprocessing

To convert the time series into a supervised learning problem, the optimal lagged inputs were decided according to the correlation analysis. For this analysis the Partial Auto-Correlation Function (PACF) and Auto-Correlation Function (ACF) plots are used. The ACF plots of the different time series of SPI-1, SPI-3, SPI-6 and SPI-12 are to be considered. And also, correlation features to outputs are analyzed optimal lags are obtained.
From the meteorological parameters (precipitation, maximum temperature, minimum temperature, relative humidity, soil moisture, root-soil moisture, and surface. soil wetness.) Precipitation is neglected because precipitation is already considered to calculate the SPI values and by performing the correlation test it has found that maximum temperature is highly correlated to SPI values than minimum temperature. Therefore, minimum temperature is also ignored. hence obtained the input features to the models are maximum temperature, relative humidity, soil moisture, root-soil moisture, and surface soil wetness.

### 5.4.1 Preprocessing for Datasets

The correlation plots of Palakkad for SPI-1,SPI-3,SPI-6 and SPI-12 are shown in Figure 5.1, 5.2, 5.3 and 5.4 respectively.Similarly for Kasaragod and Punalur the correlation plots are shown in Figure 5.9, 5.10, 5.11 and 5.12.There optimal lagged inputs and outputs are shown in Table 5.2, 5.1 and 5.3 respectively

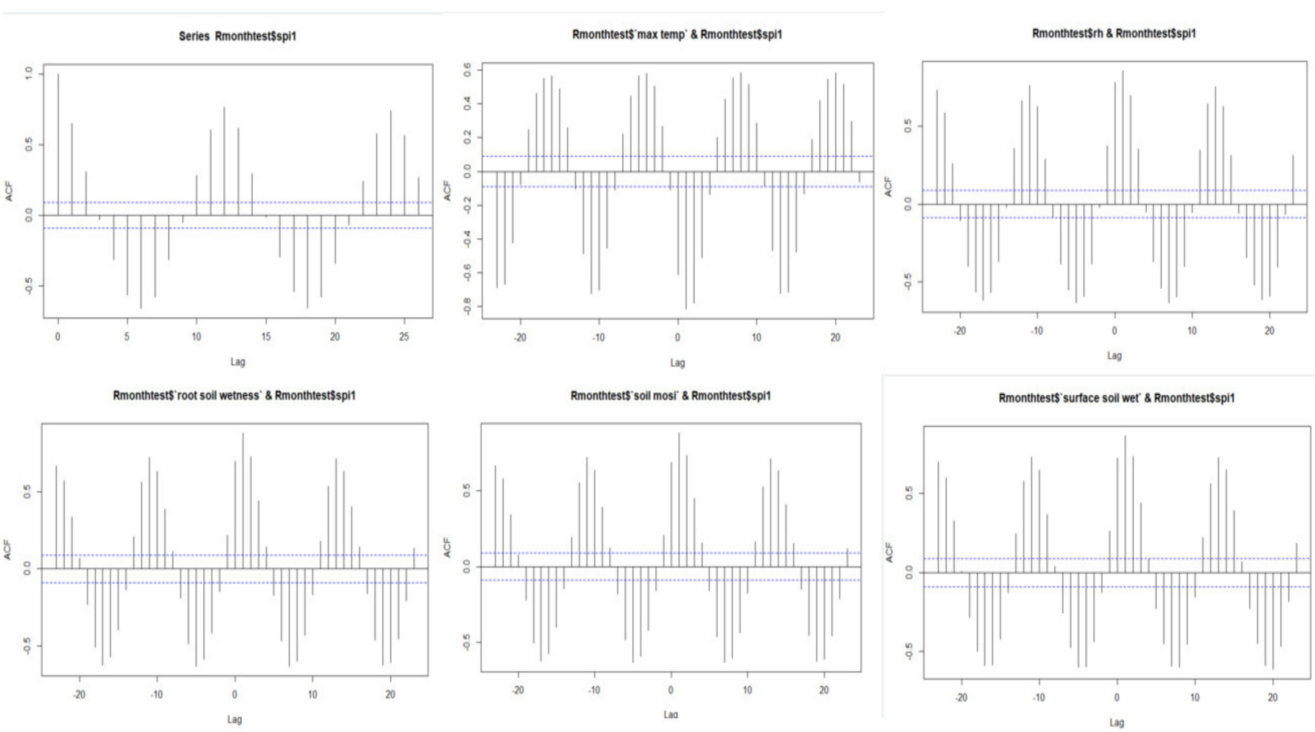


Figure 5.1: Correlation Plots for SPI-1(Palakkad)

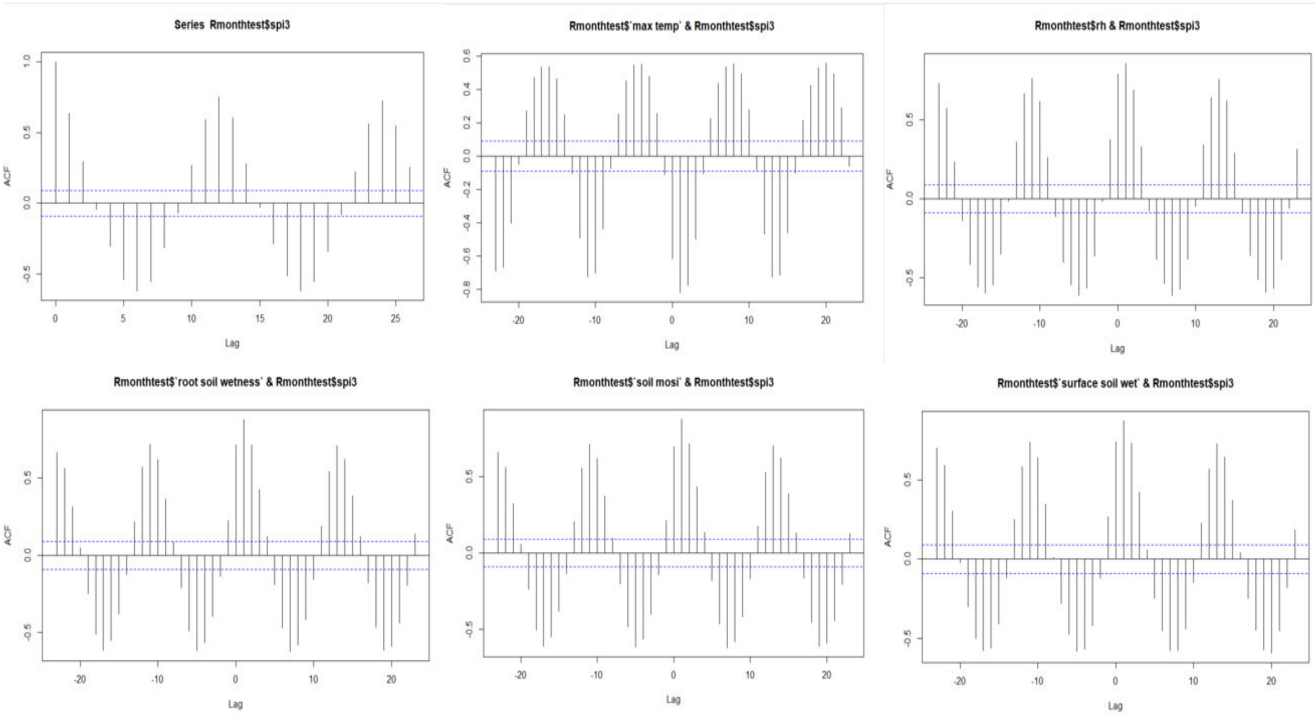


Figure 5.2: Correlation Plots for SPI-3(Palakkad)

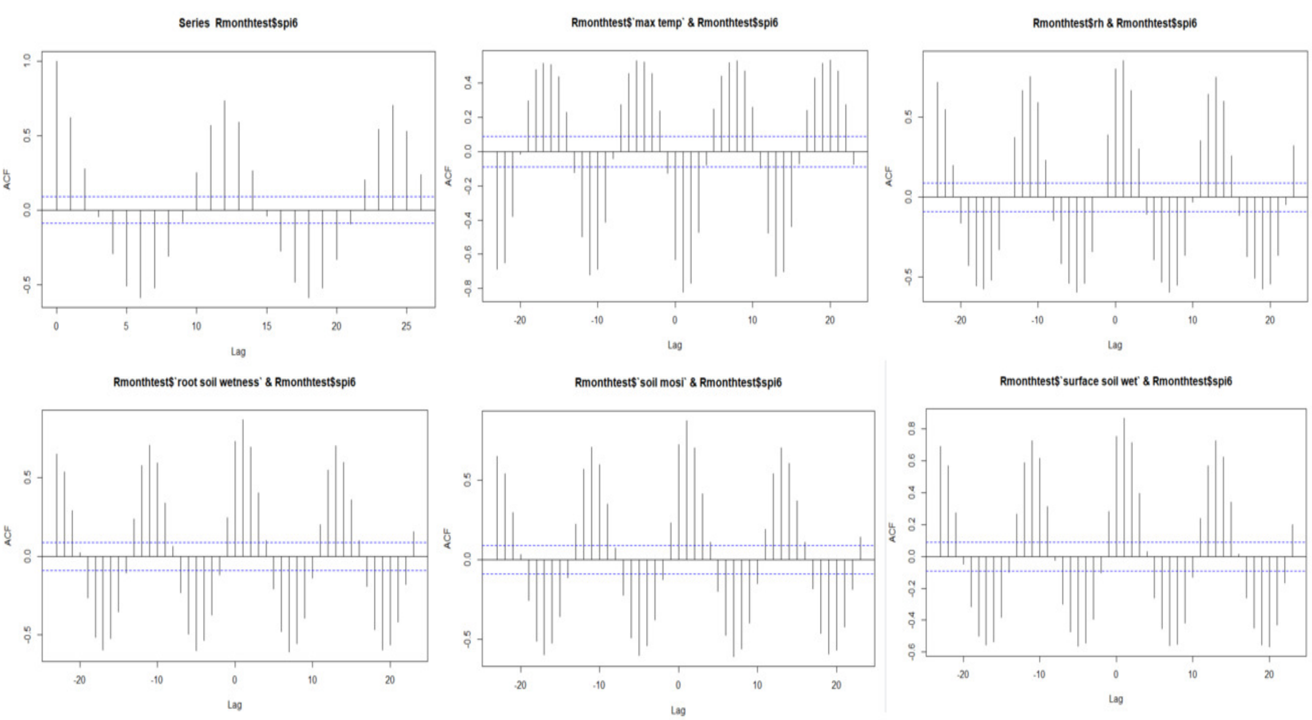


Figure 5.3: Correlation Plots for SPI-6(Palakkad)

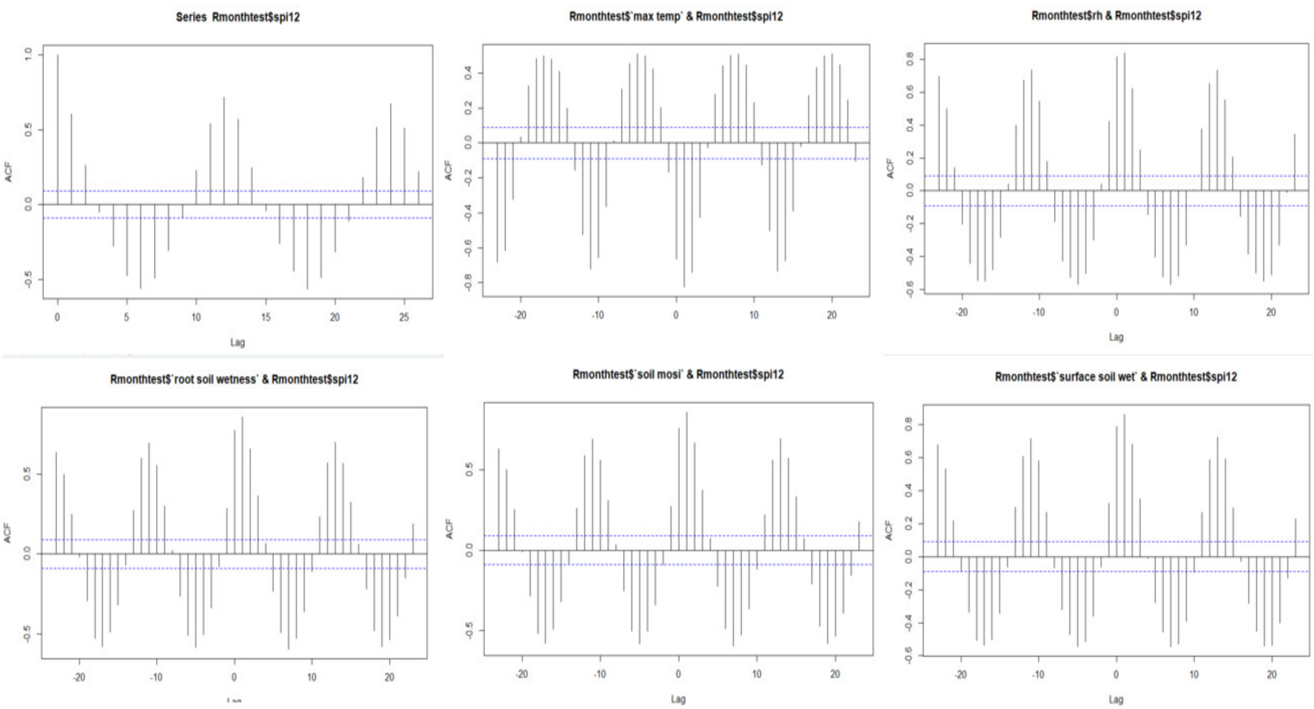


Figure 5.4: Correlation Plots for SPI-12(Palakkad)

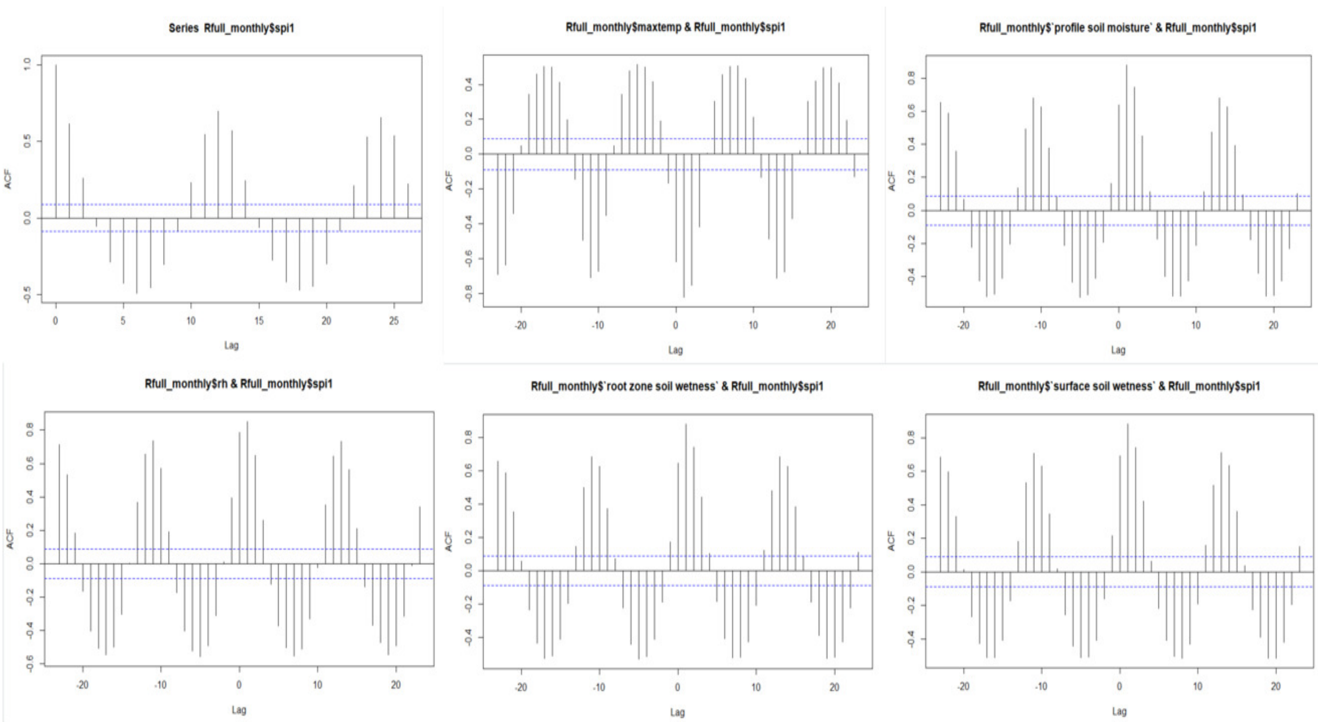


Figure 5.5: Correlation Plots for SPI-1(Punalur)

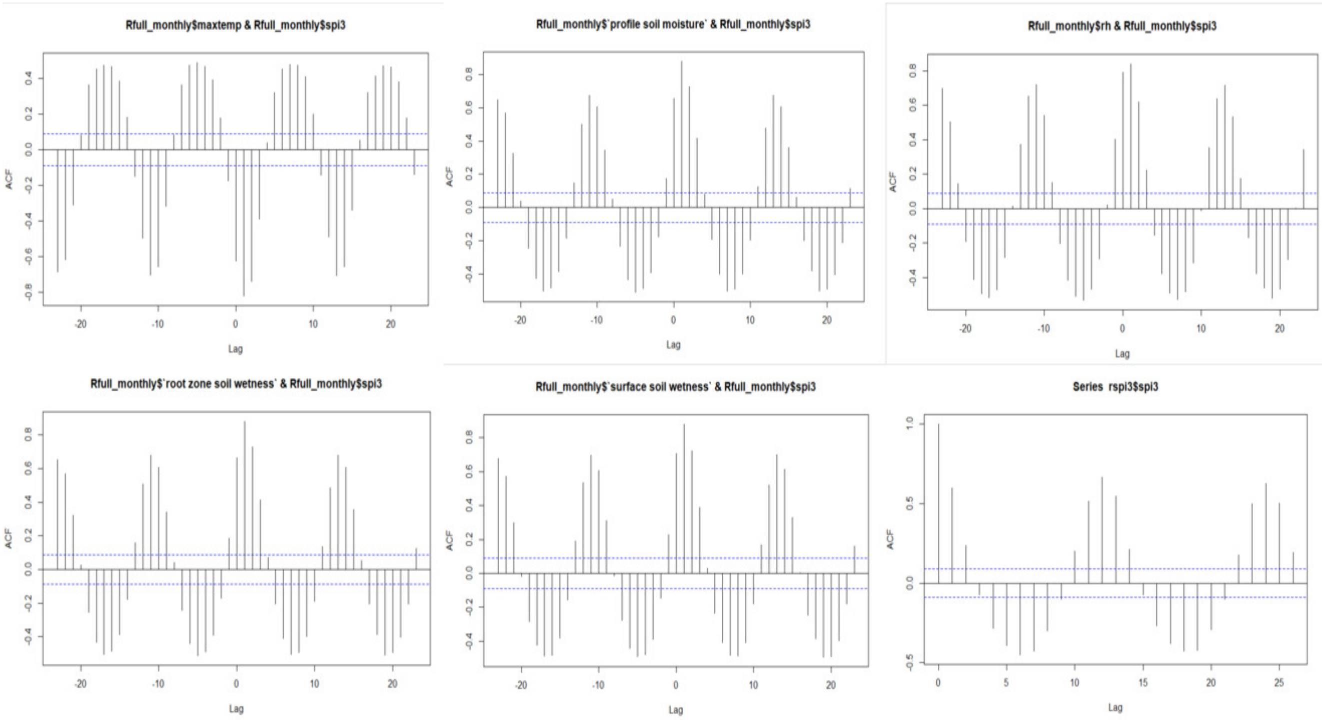


Figure 5.6: Correlation Plots for SPI-3(Punalur)

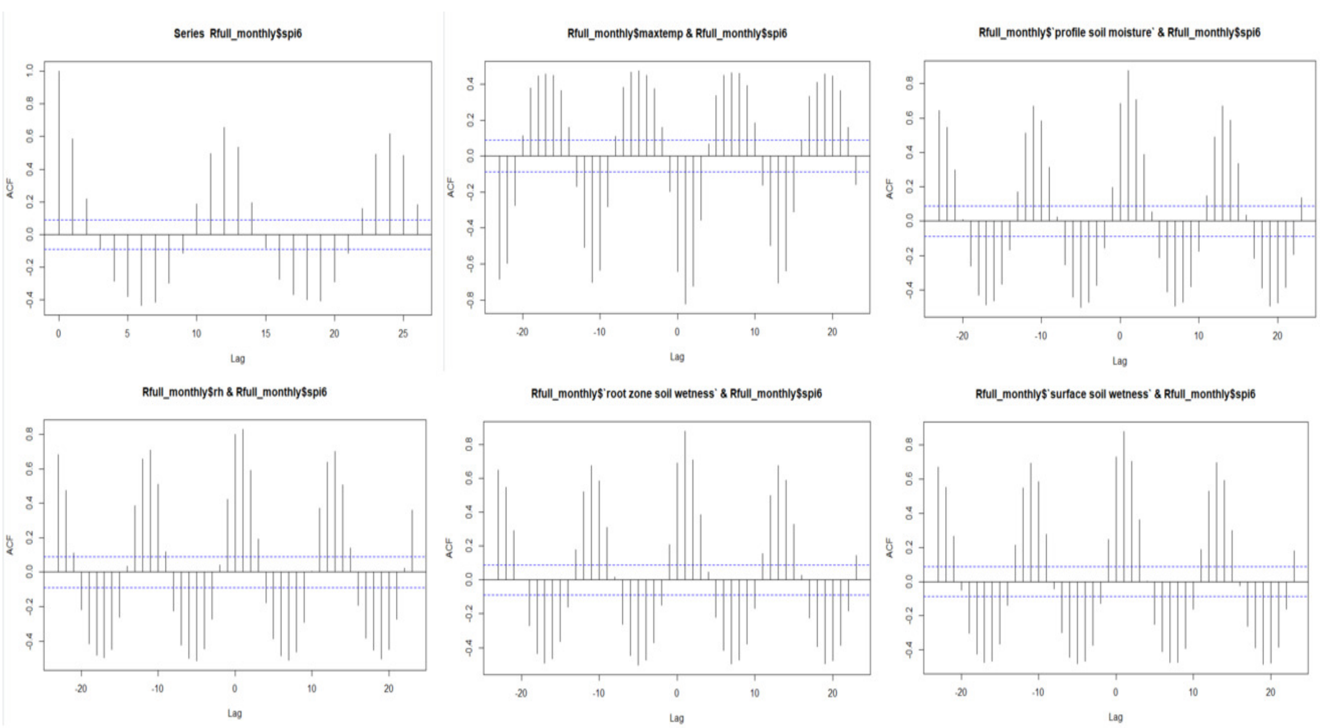


Figure 5.7: Correlation Plots for SPI-6(Punalur)

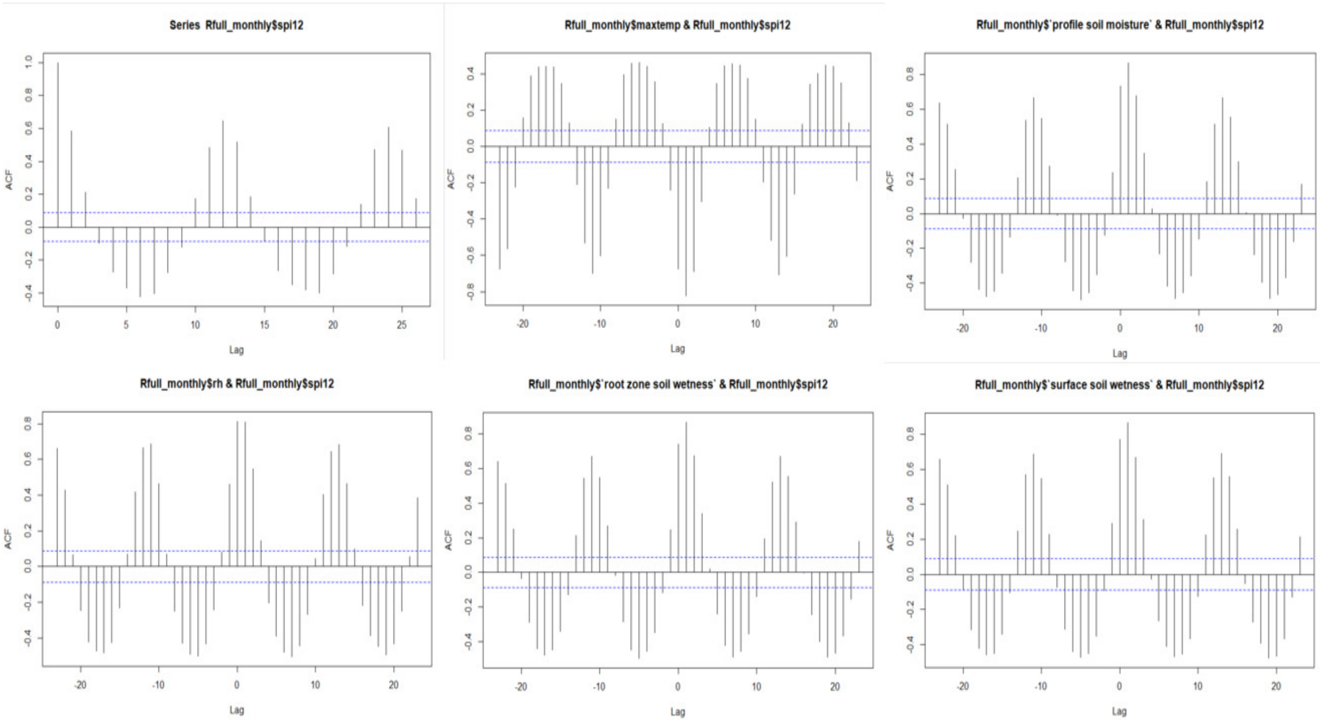


Figure 5.8: Correlation Plots for SPI-12(Punalur)

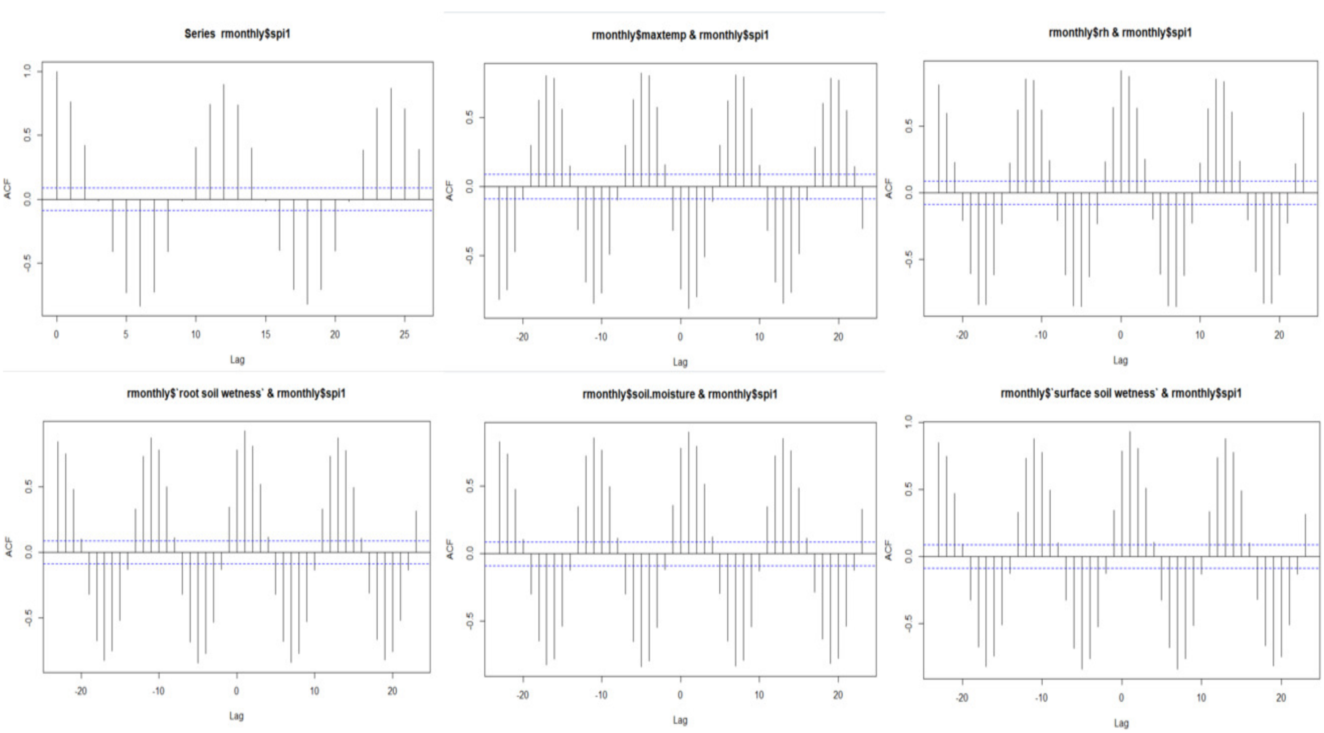


Figure 5.9: Correlation Plots for SPI-1(Kasaragod)

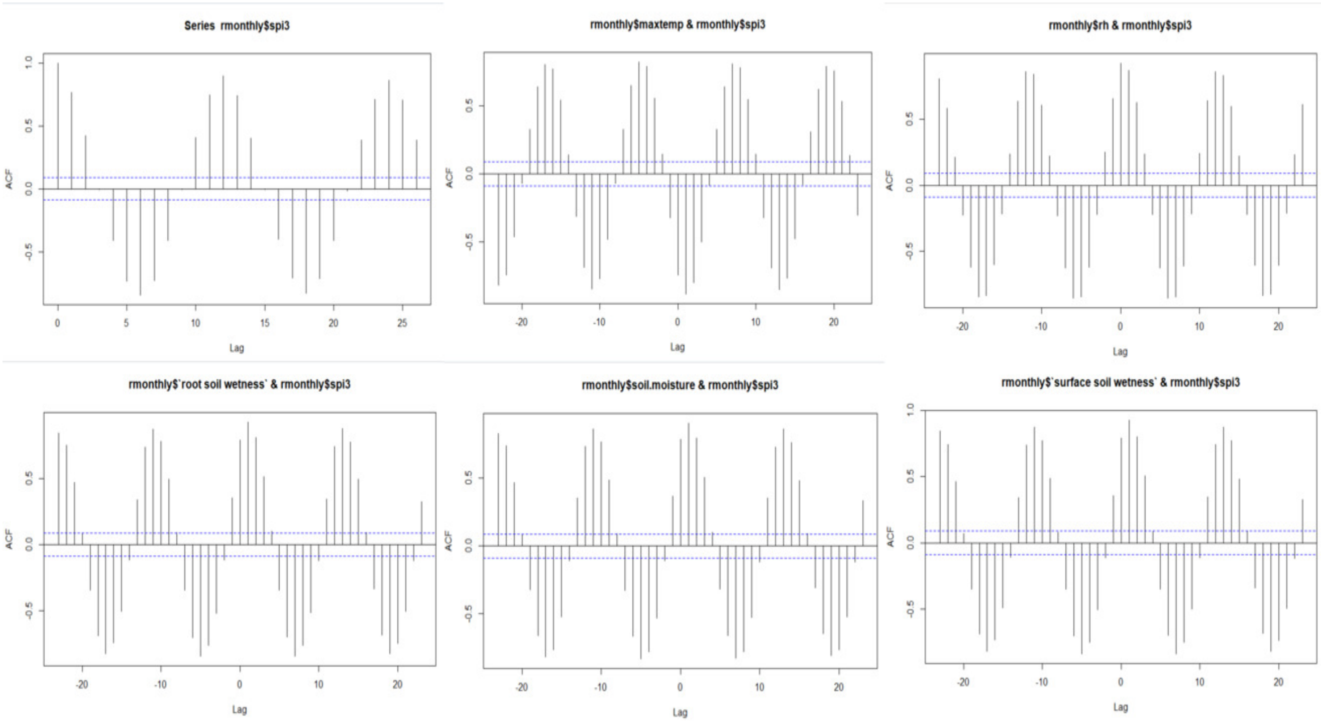


Figure 5.10: Correlation Plots for SPI-3(Kasaragod)

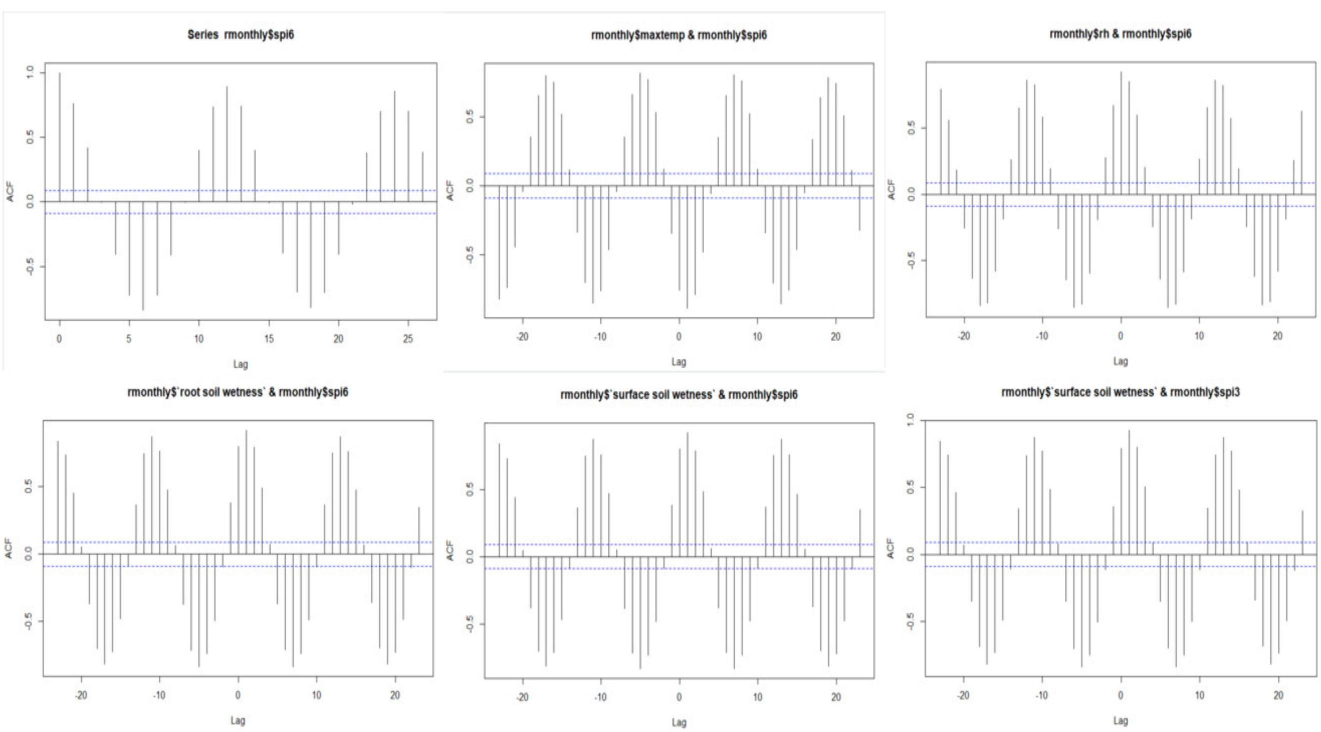


Figure 5.11: Correlation Plots for SPI-6(Kasaragod)

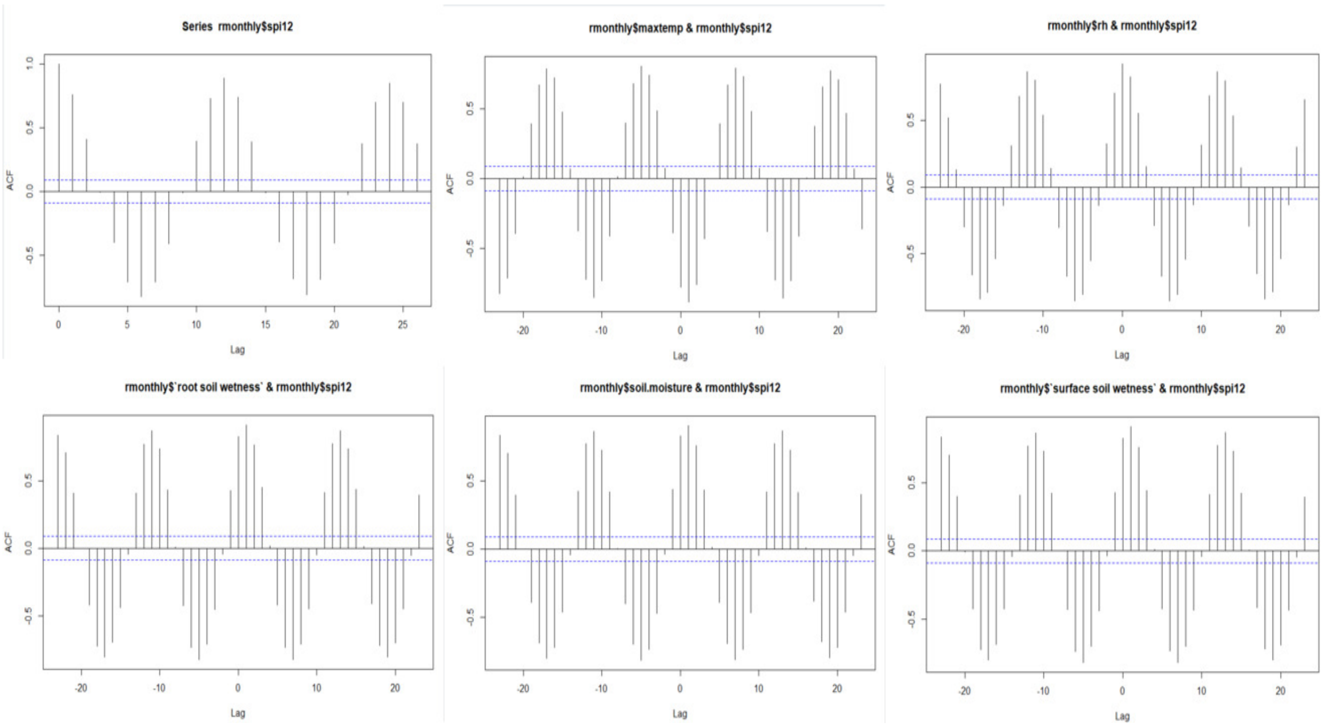


Figure 5.12: Correlation Plots for SPI-12(Kasaragod)

Table 5.1: Input and Output parameters used for forecasting SPI (Kasaragod)

| OUTPUTS | INPUTS |
|---|---|
| SPI-$1_{(t)}$ | $spi1_{(t-1)}$, $spi1_{(t-2)}$, $rh_{(t-7)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |
| SPI-$3_{(t)}$ | $spi3_{(t-1)}$, $spi3_{(t-2)}$, $rh_{(t-7)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |
| SPI-$6_{(t)}$ | $spi6_{(t-1)}$, $spi6_{(t-2)}$, $rh_{(t-6)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |
| SPI-$12_{(t)}$ | $spi12_{(t-1)}$, $spi12_{(t-2)}$, $rh_{(t-6)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |

Table 5.2: Input and Output parameters used for forecasting SPI (Palakkad)

| OUTPUTS | INPUTS |
|---|---|
| SPI-1 $_{(t)}$ | $spi1_{(t-1)}$, $spi1_{(t-2)}$, $rh_{(t-1)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |
| SPI-3 $_{(t)}$ | $spi3_{(t-1)}$, $spi3_{(t-2)}$, $rh_{(t-1)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |
| SPI-6 $_{(t)}$ | $spi6_{(t-1)}$, $spi6_{(t-2)}$, $rh_{(t-1)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |
| SPI-12 $_{(t)}$ | $spi12_{(t-1)}$, $spi12_{(t-2)}$, $rh_{(t-1)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |

Table 5.3: Input and Output parameters used for forecasting SPI (Punalur)

| OUTPUTS | INPUTS |
|---|---|
| SPI-1 $_{(t)}$ | $spi1_{(t-1)}$, $spi1_{(t-2)}$, $rh_{(t-1)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |
| SPI-3 $_{(t)}$ | $spi3_{(t-1)}$, $spi3_{(t-2)}$, $rh_{(t-1)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |
| SPI-6 $_{(t)}$ | $spi6_{(t-1)}$, $spi6_{(t-2)}$, $rh_{(t-1)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |
| SPI-12 $_{(t)}$ | $spi12_{(t-1)}$, $spi12_{(t-2)}$, $rh_{(t-1)}$, $temp_{(t-1)}$, $sm_{(t-1)}$, $sw_{(t-1)}$, $ssw_{(t-1)}$ |

## 5.5 Discussions

Based on the observation and predicted values for training and testing for different SPI time series MAE, MSE, RMSE and $R^2$ values were computed. Graphical methods such as Radar plots, Box plots and Violin plots were also used to analyses the performance.In Figures 5.13, 5.14 and 5.15 various graphs are shown for different model prediction for SPI-1, SPI-3, SPI-6, and SPI-12 timescales for Palakkad,Kasaragod and Punalur respectively for the last 100 months.

**Predictions for locations**

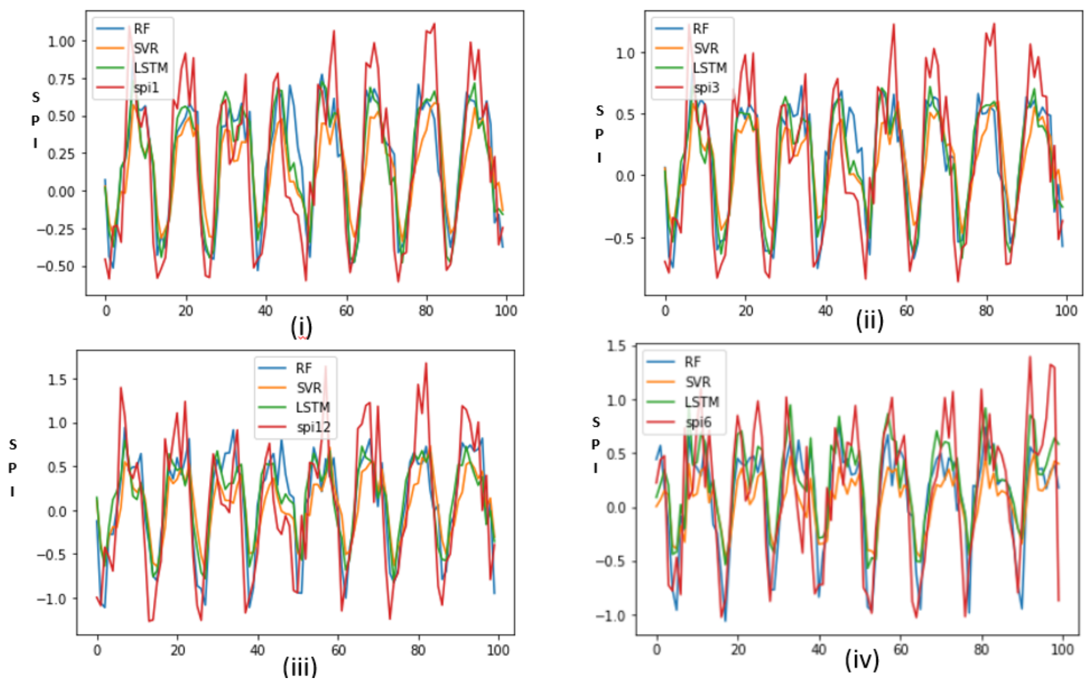


Figure 5.13: Different model prediction for SPI-1(Palakkad), (ii)Different model prediction for SPI-3(Palakkad), (iii)Different model prediction for SPI-6(Palakkad), (iv)Different model prediction for SPI-12(Palakkad)

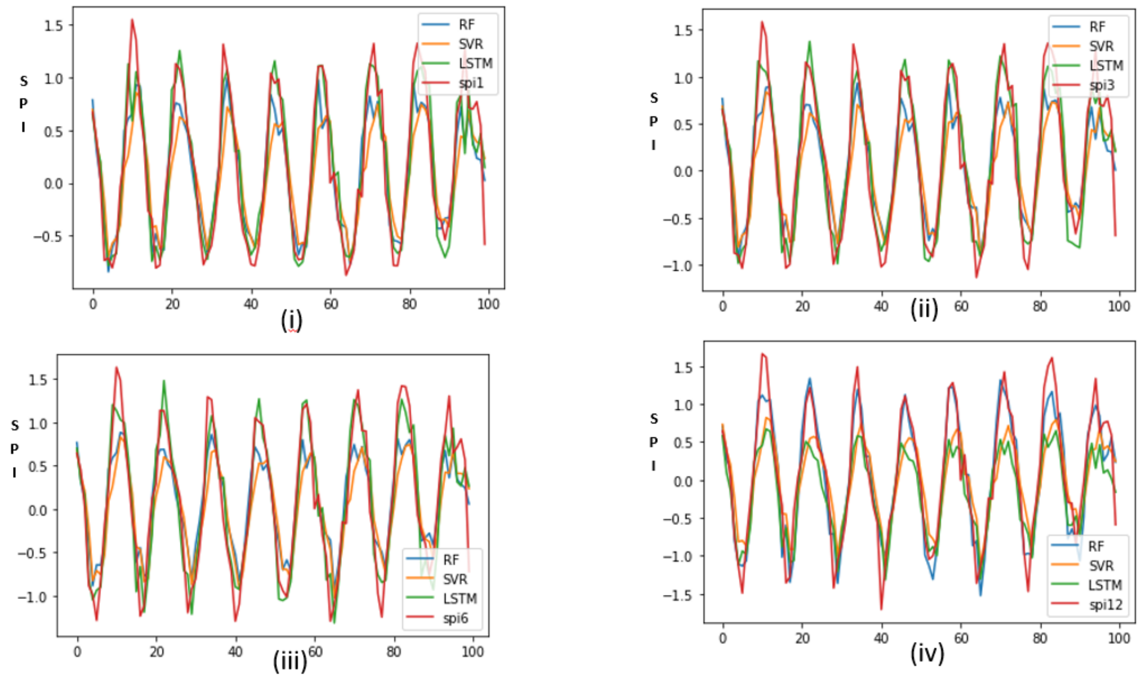


Figure 5.14: Different model prediction for SPI-1(Kasaragod), (ii)Different model prediction for SPI-3(Kasaragod), (iii)Different model prediction for SPI-6(Kasaragod), (iv)Different model prediction for SPI-12(Kasaragod)

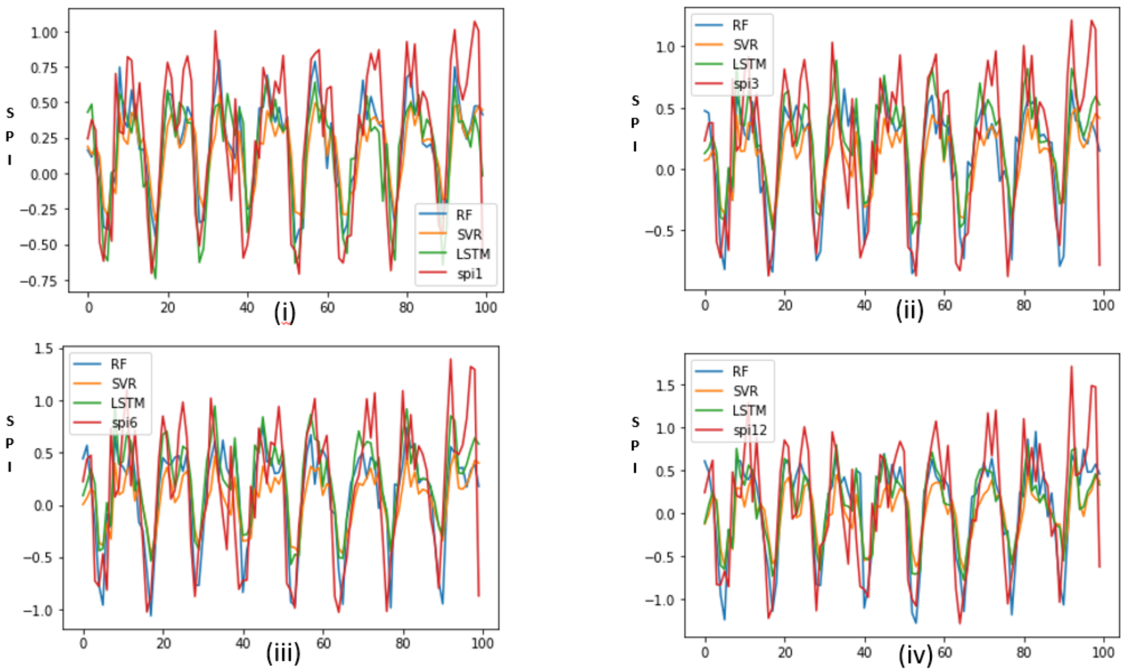


Figure 5.15: Different model prediction for SPI-1(Punalur), (ii)Different model prediction for SPI-3(Punalur), (iii)Different model prediction for SPI-6(Punalur), (iv)Different model prediction for SPI-12(Punalur)

Table 5.4: Performance evaluators of SPI-1 Prediction for Training and Testing (Palakkad)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.092 | 0.304 | 0.136 | 0.244 | 0.282 | 0.123 |
| MSE | 0.014 | 0.139 | 0.029 | 0.095 | 0.113 | 0.022 |
| RMSE | 0.121 | 0.374 | 0.172 | 0.308 | 0.337 | 0.147 |
| $R^2$ | 0.954 | 0.509 | 0.605 | 0.639 | 0.652 | 0.716 |

Table 5.5: Performance Evaluators of SPI-3 Prediction for Training and Testing (Palakkad)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.116 | 0.372 | 0.133 | 0.346 | 0.344 | 0.119 |
| MSE | 0.022 | 0.207 | 0.028 | 0.173 | 0.172 | 0.021 |
| RMSE | 0.149 | 0.455 | 0.169 | 0.370 | 0.415 | 0.150 |
| $R^2$ | 0.954 | 0.479 | 0.591 | 0.623 | 0.594 | 0.672 |

Table 5.6: Performance Evaluators of SPI-6 Prediction for Training and Testing (Palakkad)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.132 | 0.431 | 0.127 | 0.353 | 0.401 | 0.118 |
| MSE | 0.029 | 0.284 | 0.026 | 0.186 | 0.236 | 0.020 |
| RMSE | 0.171 | 0.533 | 0.162 | 0.431 | 0.486 | 0.145 |
| $R^2$ | 0.955 | 0.461 | 0.571 | 0.598 | 0.556 | 0.632 |

Table 5.7: Performance Evaluators of SPI-12 Prediction for Training and Testing (Palakkad)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.159 | 0.504 | 0.121 | 0.401 | 0.459 | 0.112 |
| MSE | 0.043 | 0.403 | 0.024 | 0.250 | 0.330 | 0.019 |
| RMSE | 0.208 | 0.635 | 0.157 | 0.500 | 0.574 | 0.141 |
| $R^2$ | 0.950 | 0.419 | 0.538 | 0.575 | 0.493 | 0.523 |

The values of the Performance Evaluators of SPI-1, SPI-3,SPI-6 and SPI-1 Palakkad testing and training are shown in Tables 5.4,5.5, 5.6 and 5.7.LSTM model shows higher $R^2$ values and lower errors than SVR and RF in SPI-1,SPI-3 and SPI-6.Where for SPI-12 RF has higher $R^2$ value (0.575) than SVR and LSTM but, in training phase RF has 0.950 $R^2$ value and for LSTM the $R^2$ value for training is 0.538 and for testing 0.523.This indicates the over fitting condition of RF,hence the LSTM model show more efficiency than other two models for SPI-12.

Table 5.8: Performance Evaluators of SPI-1 Prediction for Training and Testing (Kasaragod)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.084 | 0.317 | 0.093 | 0.194 | 0.302 | 0.079 |
| MSE | 0.013 | 0.181 | 0.014 | 0.070 | 0.145 | 0.011 |
| RMSE | 0.114 | 0.426 | 0.118 | 0.268 | 0.381 | 0.097 |
| $R^2$ | 0.977 | 0.700 | 0.792 | 0.853 | 0.764 | 0.845 |

For performance evaluators of Kasaragod with respect to SPI-1,SPI-3,SPI-6 and SPI-12 timescales the training and testing performance measures are shown in Table 5.8, 5.9, 5.10 and 5.11 respectively.Here in all testing phase RF show high $R^2$ values than SVR and LSTM.But,in training phase RF has very high $R^2$ values than that of testing phase of RF.But for LSTM in all timescales shows a near $R^2$ values and less errors in both testing and training phase,hence it indicates the over-fitting condition of RF.Therefore, LSTM model can perform more efficiently than RF.

Table 5.9: Performance Evaluators of SPI-3 Prediction for Training and Testing (Kasaragod)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.096 | 0.357 | 0.088 | 0.213 | 0.333 | 0.073 |
| MSE | 0.016 | 0.219 | 0.012 | 0.072 | 0.174 | 0.008 |
| RMSE | 0.126 | 0.468 | 0.112 | 0.268 | 0.417 | 0.074 |
| $R^2$ | 0.977 | 0.711 | 0.789 | 0.876 | 0.769 | 0.836 |

Table 5.10: Performance Evaluators of SPI-6 Prediction for Training and Testing (Kasaragod)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.103 | 0.391 | 0.083 | 0.238 | 0.361 | 0.072 |
| MSE | 0.017 | 0.259 | 0.011 | 0.089 | 0.204 | 0.008 |
| RMSE | 0.133 | 0.509 | 0.164 | 0.298 | 0.452 | 0.097 |
| $R^2$ | 0.978 | 0.707 | 0.785 | 0.864 | 0.758 | 0.845 |

Table 5.11: Performance Evaluators of SPI-12 Prediction for Training and Testing (Kasaragod)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.111 | 0.413 | 0.079 | 0.249 | 0.381 | 0.071 |
| MSE | 0.019 | 0.293 | 0.009 | 0.101 | 0.234 | 0.008 |
| RMSE | 0.141 | 0.542 | 0.099 | 0.319 | 0.484 | 0.090 |
| $R^2$ | 0.978 | 0.709 | 0.783 | 0.860 | 0.741 | 0.789 |

Table 5.12: Performance Evaluators of SPI-1 Prediction for Training and Testing (Punalur)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.106 | 0.336 | 0.150 | 0.287 | 0.321 | 0.141 |
| MSE | 0.017 | 0.159 | 0.034 | 0.119 | 0.149 | 0.030 |
| RMSE | 0.132 | 0.399 | 0.184 | 0.345 | 0.386 | 0.170 |
| $R^2$ | 0.950 | 0.455 | 0.519 | 0.568 | 0.504 | 0.562 |

Table 5.13: Performance Evaluators of SPI-3 Prediction for Training and Testing (Punalur)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.125 | 0.387 | 0.145 | 0.334 | 0.367 | 0.135 |
| MSE | 0.023 | 0.215 | 0.032 | 0.167 | 0.200 | 0.028 |
| RMSE | 0.154 | 0.464 | 0.179 | 0.409 | 0.448 | 0.166 |
| $R^2$ | 0.950 | 0.448 | 0.504 | 0.498 | 0.464 | 0.525 |

Table 5.14: Performance Evaluators of SPI-6 Prediction for Training and Testing (Punalur)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.143 | 0.445 | 0.139 | 0.366 | 0.414 | 0.129 |
| MSE | 0.031 | 0.289 | 0.029 | 0.208 | 0.258 | 0.025 |
| RMSE | 0.177 | 0.538 | 0.172 | 0.451 | 0.508 | 0.159 |
| $R^2$ | 0.949 | 0.443 | 0.504 | 0.485 | 0.442 | 0.507 |

Table 5.15: Performance Evaluators of SPI-12 Prediction for Training and Testing (Punalur)

| | TRAINING | | | TESTING | | |
|---|---|---|---|---|---|---|
| Performance Measures | RF | SVR | LSTM | RF | SVR | LSTM |
| MAE | 0.976 | 0.493 | 0.120 | 0.388 | 0.467 | 0.113 |
| MSE | 0.030 | 0.378 | 0.022 | 0.231 | 0.314 | 0.018 |
| RMSE | 0.196 | 0.615 | 0.150 | 0.481 | 0.560 | 0.137 |
| $R^2$ | 0.954 | 0.443 | 0.521 | 0.545 | 0.447 | 0.502 |

For performance evaluators of Punalur with respect to SPI-1,SPI-3,SPI-6 and SPI-12 timescales the training and testing performance measures are shown in Table 5.12, 5.13, 5.14 and 5.15 respectively.Here LSTM models shows a better performance in $R^2$ values and errors than SVR and LSTM in SPI-3 and SPI-6.But in the case of SPI-1 SVR and for SPI-12 RF shows a better performance than LSTM.For training phase of SVR and RF show very distanced $R^2$ value to testing $R^2$ values of SVR and RF(it shows the over-fitting condition of SVR and RF in SPI-1 and SPI-12 respectively).In the same case of LSTM the $R^2$ values are very much similar and less errors are obatined in both testing and training.Therfore LSTM show better performance than other two models with less errors.

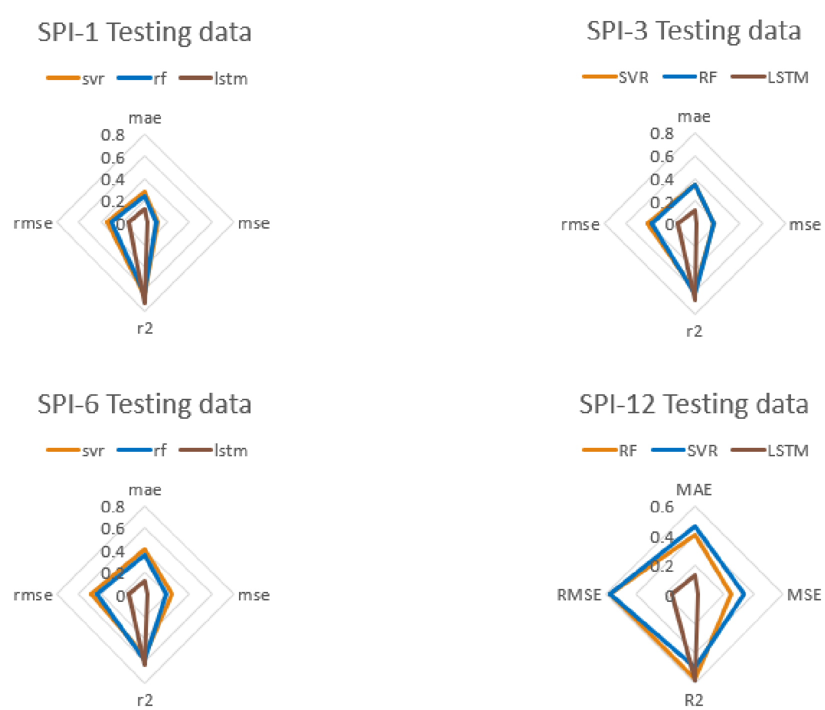


Figure 5.16: Radar Plots of SPI timescales Testing Data Performance Evaluators(Palakkad)


Figure 5.17: Radar Plots of SPI timescales Testing Data Performance Evaluators(Kasargod)


Figure 5.18: Radar Plots of SPI timescales Testing Data Performance Evaluators(Punalur)

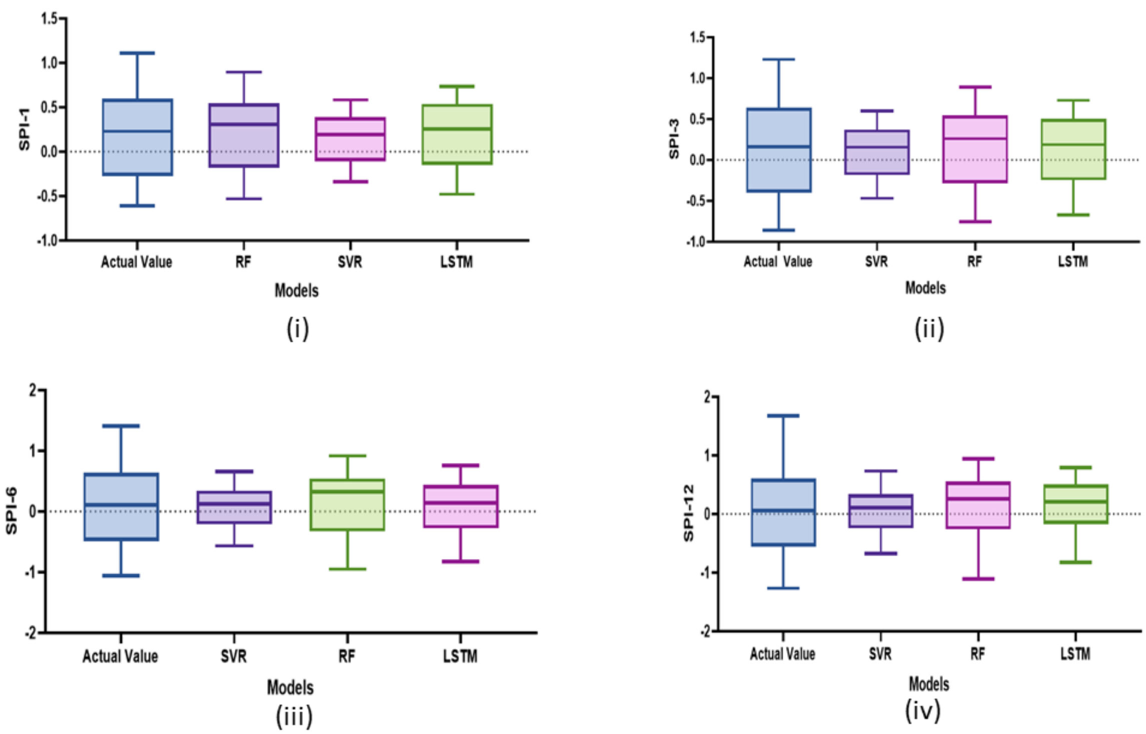


Figure 5.19: (i)Box plot of SPI-1 Testing Predictions (Palakkad),(ii)Box plot of SPI-3 Testing Predictions(Palakkad),(iii)Box plot of SPI-6 Testing Predictions(Palakkad),(iv)Box plot of SPI-12 Testing Predictions(Palakkad)

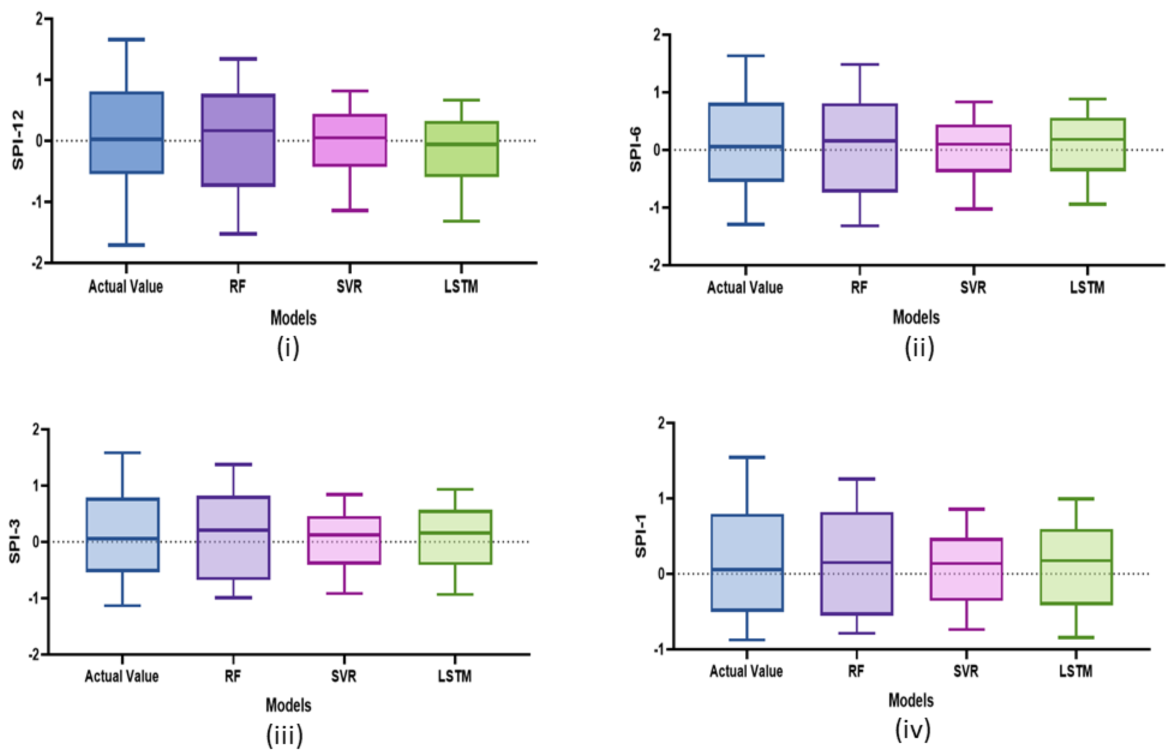


Figure 5.20: (i)Box plot of SPI-1 Testing Predictions (Kasaragod),(ii)Box plot of SPI-3 Testing Predictions(Kasaragod),(iii)Box plot of SPI-6 Predictions(Kasaragod),(iv)Box plot of SPI-12 Testing Predictions(Kasaragod)

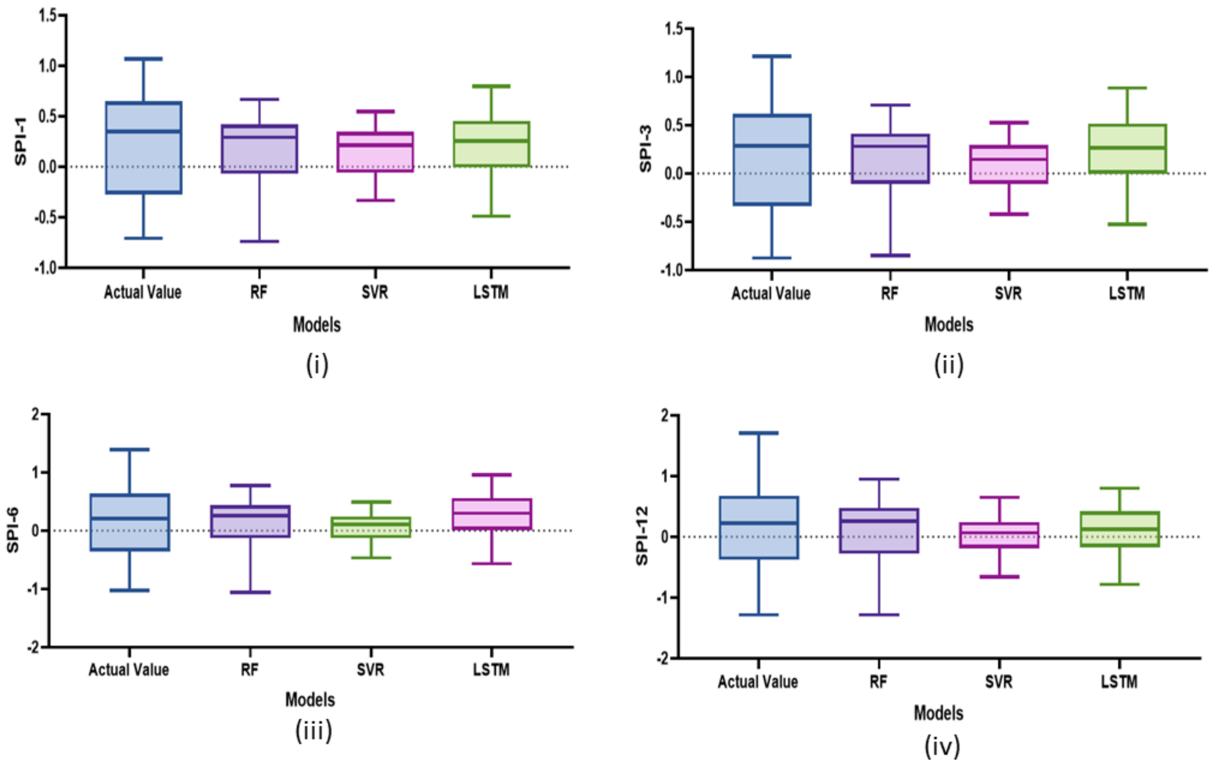


Figure 5.21: (i)Box plot of SPI-1 Testing Predictions (Punalur),(ii)Box plot of SPI-3 Testing Predictions(Punalur),(iii)Box plot of SPI-6 Testing Predictions(Punalur),(iv)Box plot of SPI-12 Testing Predictions(Punalur)

The Box plot for the different time series testing predictions for Palakkad, Kasaragod and Punalur are shown in Figure 5.19, 5.20 and 5.21 respectively. In Box Plot, the start of the box, i.e., the lower quartile represents 25% of the data set and the end of the box, i.e., the upper quartile represents 75% of the data. The bold black line in the box represents the median value of the data. The difference between the lower quartile and upper quartile is called the inter-quartile range. In the box plot, the whiskers are generally defined as 1.5 times the inter-quartile range. Anything outside the whiskers is considered as an outlier. The shorter the box plot, then the data will be more consistent. Skewness of the data can be identified by observing the shape of the box plot. If the box plot is symmetric, it means that the data follows a normal distribution. If the box plot is not symmetric it shows that the data is skewed.

The boxplot of Palakkad for SPI-1, SPI-3, SPI-6 and SPI-12 LSTM median value is closer to the median value of actual value than SVR and RF, which indicate the LSTM is more consistent. Considering the Kasaragod boxplots RF is symmetric to actual values it means RF follows a normal distribution but, in SPI-1, SPI-3 and SPI-12 LSTM median value is closer to actual values, therefore LSTM is very consistent other than SVR and RF. For the last location Punalur, SPI-1,SPI-3,SPI-6 and SPI-12 LSTM and RF have similar median closer to actual value median and for SVR ,it shows less consistence. Box plots for the different time series testing predictions for Kasaragod, Punalur and Palakkad are show in Figures 5.22, 5.23 and 5.24.

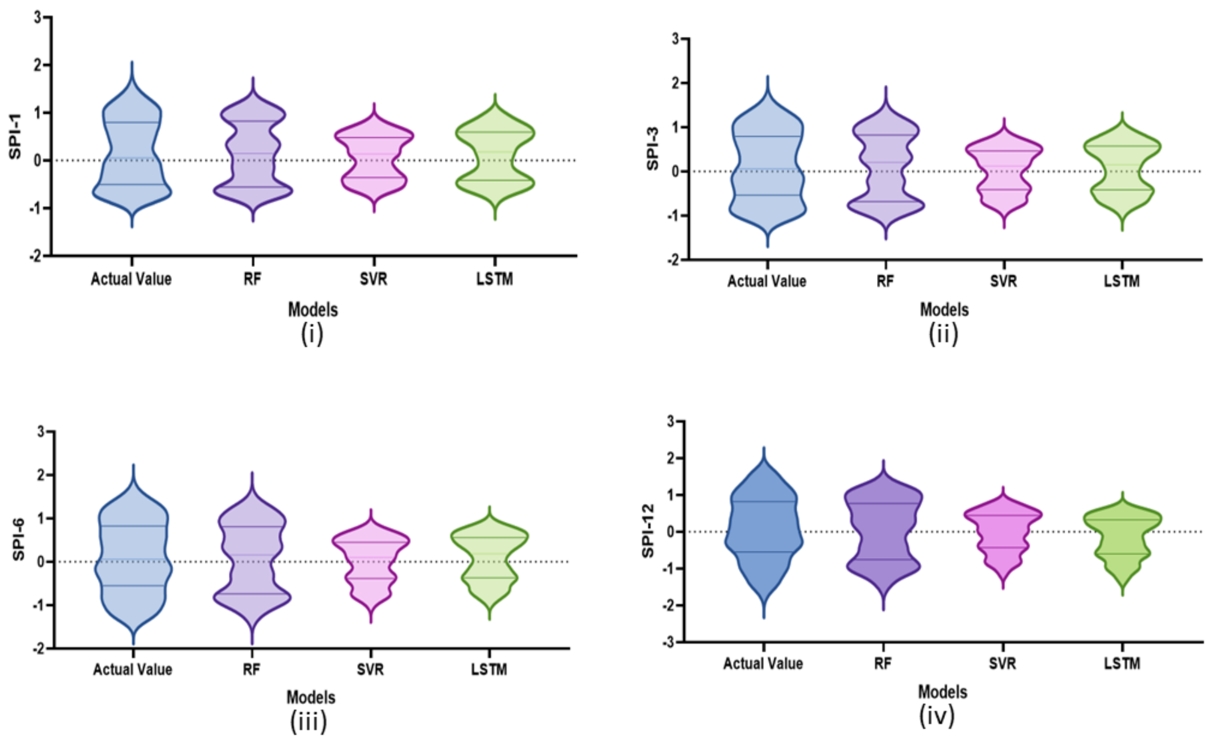


Figure 5.22: (i)Violin plot of SPI-1 Testing Predictions (Kasaragod),(ii)Violin plot of SPI-3 Testing Predictions(Kasaragod),(iii)Violin plot of SPI-6 Testing Predictions(Kasaragod),(iv)Violin plot of SPI-12 Testing Predictions(Kasaragod)

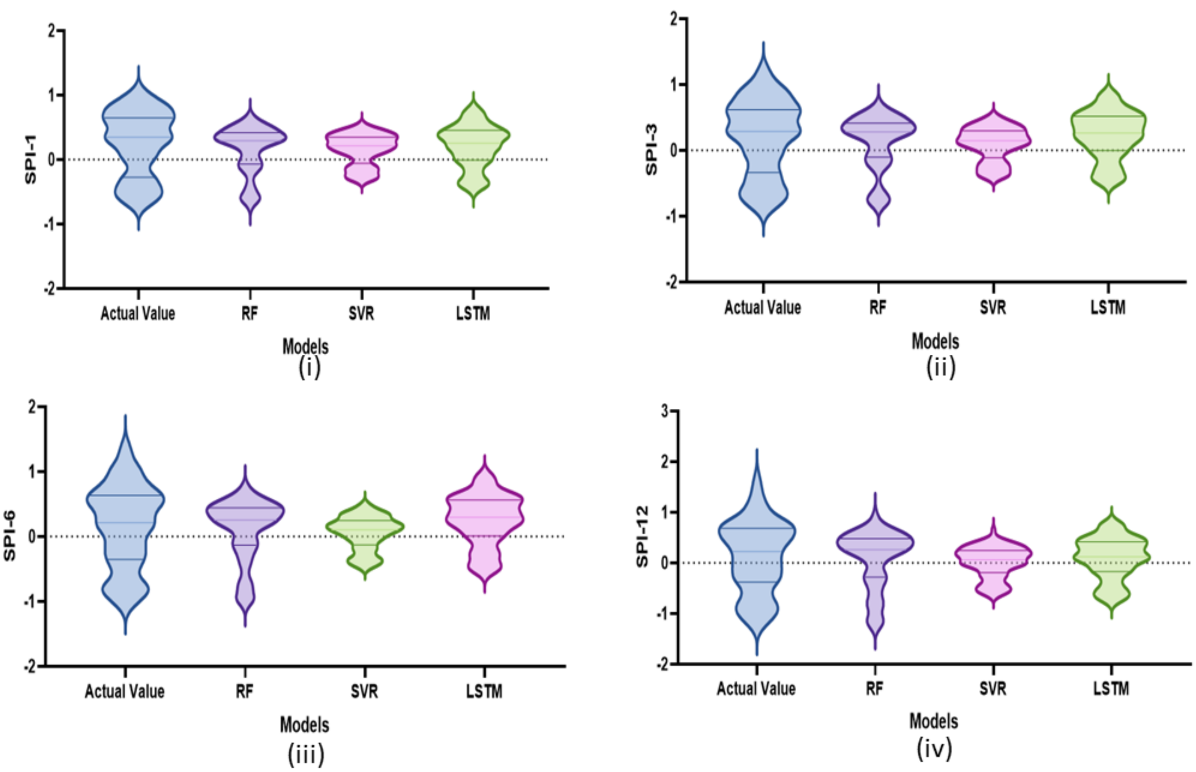


Figure 5.23: (i)Violin plot of SPI-1 Testing Predictions (Punalur),(ii)Violin plot of SPI-3 Testing Predictions(Punalur),(iii)Violin plot of SPI-6 Testing Predictions(Punalur),(iv)Violin plot of SPI-12 Testing Predictions(Punalur)

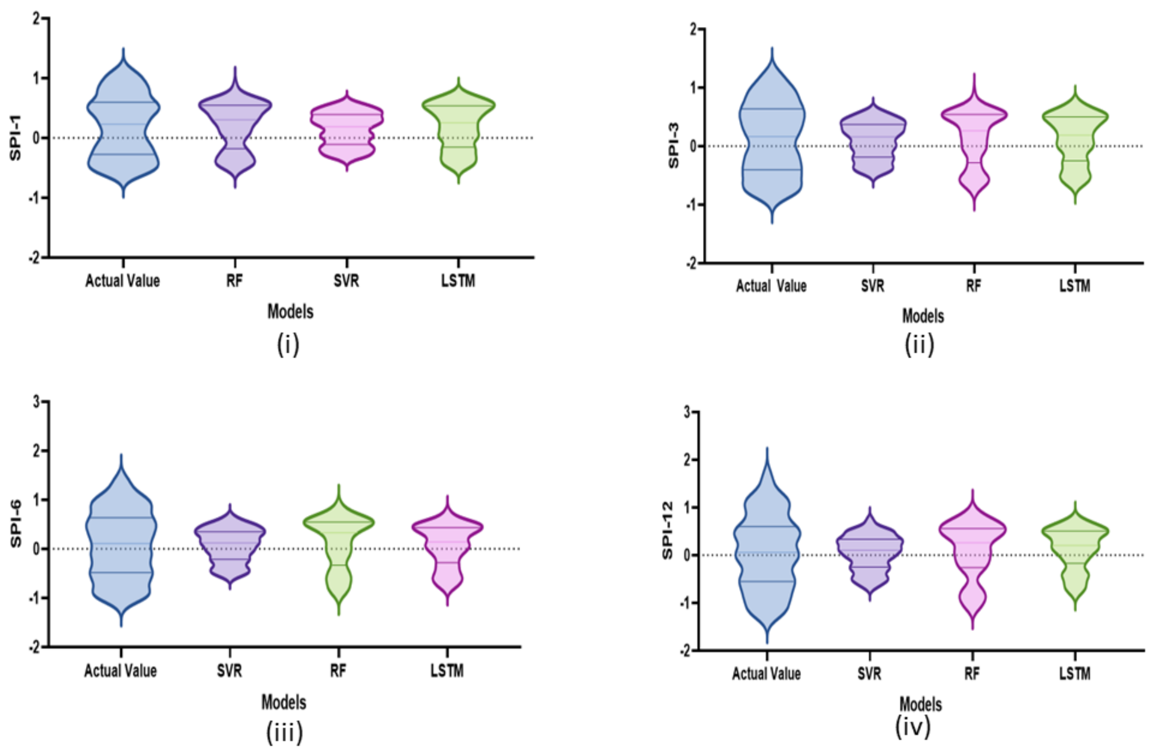


Figure 5.24: (i)Violin plot of SPI-1 Testing Predictions (Palakkad),(ii)Violin plot of SPI-3 Testing Predictions(Palakkad),(iii)Violin plot of SPI-6 Testing Predictions(Palakkad),(iv)Violin plot of SPI-12 Testing Predictions(Palakkad)

Violin plots for the different time series testing predictions for Kasaragod, Punalur and Palakkad are shown in Figure 5.22, 5.23 and 5.24. Violin plots are similar to box plots, except that they show the probability density of the data at different values usually smoothed by a kernel density estimator. The distribution of the data at the median in the violin plot can be used to determine the best model prediction. Thus, LSTM shows an overall performance in the three locations.

# Chapter 6

# Conclusion

In this study, different Machine Learning models such as Random Forest, SVR and Deep Learning model LSTM were applied for short, medium and long-term drought prediction of Palakkad, Kasaragod and Punalur in Kerala. The errors like Mean Absolute Error (MAE), Mean Square Error (MSE), Root Mean Square Error (RMSE) and Coefficient of Determination (R2) were computed. Different plots like Radar Plot, Box Plot and Violin Plot were used. The important conclusion drawn from the study are:

- Some Machine Learning algorithms performs best during training, but failed to perform well during testing. (In Kasaragod location dealing with SPI-1,SPI-3,SPI-6 and SPI-12 RF shows high R2 values in testing and failed to perform well during training and similarly, in Punalur location RF shows better performance in SPI-1 and RF shows better in SPI-12.)

- The graphical plots shows that the overall performance of LSTM ,the best among the three models where, Box Plot and Violin Plot shows the distributions of predictions and LSTM have almost same medium value as that of actual values which indicate that LSTM is more consistent than SVR and RF also when the timescales of SPI increase the prediction efficiency also increases for LSTM.

  The limitations include the unavailability of larger amount of historical data for LSTM to learn the Trends and fluctuations so as to generalize the problem, otherwise the model will show better performance.

# References


[1] Hao Z, Singh VP, Xia Y. Seasonal drought prediction: advances, challenges, and future prospects. Reviews of Geophysics. ;56(1)(2018):108-41.

[2] McKee, TB.,Doesken NJ, and Kleist J. "The relationship of drought frequency and duration to time scales." In Proceedings of the 8th Conference on Applied Climatology, vol. 17, no. 22 (1993): 179-183.

[3] Zargar A, Sadiq R, Naser B, Khan FI. A review of drought indices. Environmental Reviews. (2011):333-49.

[4] Agana NA and Homaifar A. "A deep learning based approach for long-term drought prediction."in SoutheastCon (2017):1-8.

[5] Poornima, S., and M. Pushpalatha. "Drought prediction based on SPI and SPEI with varying timescales using LSTM recurrent neural network." Soft Computing 23, no. 18 (2019): 8399-8412.

[6] Karimi, M,Melesse AM,Khosravi K,Mamuye M, and Zhang J. "Analysis and prediction of meteorological drought using SPI index and ARIMA model in the Karkheh River Basin, Iran." In Extreme Hydrology and Climate Variability,Elsevier,(2019):343-353.

[7] Kaur, A, and Sood SK. "Deep learning based drought assessment and prediction framework." Ecological Informatics 57 (2020): 101067.

[8] Bouaziz, M,Medhioub E, and Csaplovisc E. "A machine learning model for drought tracking and forecasting using remote precipitation data and a standardized precipitation index from arid regions." Journal of Arid Environments 189 (2021): 104478.

[9] Pham, QB,Yang T-C,Kuo C-M,Tseng H-W, and Yu P-S. "Coupling singular spectrum analysis with least square support vector machine to improve accuracy of SPI drought forecasting." Water Resources Management 35, no. 3 (2021): 847-868.

[10] Cheval and Sorin. "The standardized precipitation index–an overview." Romanian Journal of Meteorology 12, no. 1-2 (2015):17-64..

[11] Medsker LR, Jain LC. Recurrent neural networks. Design and Applications. (2001);5:64-7.