Soil Moisture Prediction Using Artificial Intelligence and Machine Learning

A project report submitted in partial fulfillment of requirements for the internship

Submitted

By:

Saurabh Zarekar

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE (AI&DS)

Dr. D. Y. Patil Institute of Technology, Pimpri, Pune

Guided by:

Dr. Milind Mujumdar (Scientist-F)

CENTER FOR CLIMATE CHANGE RESEARCH (CCCR),

INDIAN INSTITUTE OF TROPICAL METEOROLOGY (IITM), PUNE-411008

Acknowledgement

I would like to thank Dr. R. Krishnan, Director, Indian Institute of Tropical Meteorology, Pune, and the Academic Cell IITM for providing this opportunity. This internship program at IITM, Pune is a great learning chance for an undergraduate student like me. This project has seen contributions from various individuals. It has been my honor to work under the guidance of my supervisors, Dr. Milind Mujumdar and Dr. Bhupendra Bahadur Singh from CCCR, IITM Pune. Their tremendous encouragement motivated me to carry out this research work. I also extend heartfelt gratitude towards Mr. Mangesh Goswami and Dr. Madhusudan Ingale,Scientist S at IITM Pune, for their cordial support, valuable information, and guidance, which helped me in completing this task through various stages. I am also grateful to my college Dr D. Y . Patil Institute of Technology, Pimpri, Pune and specifically to Dr. Mithra Venkatesan, Head of the Department (AI &DS), for their support and the foundation they have provided me, enabling me to pursue such enriching opportunities

Table of Contents

Table of Figures

Abstract

This internship project aimed to use machine learning techniques to forecast soil moisture at different levels, an important parameter for research and environmental management. Conventional approaches to soil moisture estimation entail manual measurements which are laborious or expensive sensor networks confined by limited spatial and temporal scales.

In this internship, machine learning models were developed to accurately predict soil moisture levels using readily available environmental and meteorological data. This involved utilizing a variety of machine learning algorithms such as regression models, decision trees, random forests, support vector machines (SVM), and neural networks among others. Feature engineering was done to obtain useful information from input variables like precipitation, temperature, humidity, land cover type, and topographical features.

The internship project included several steps—data collection & pre-processing; Exploratory Data Analysis; model selection & training; hyperparameter tuning; model evaluation. The focus was on interpreting model predictions to gain insights into the machine learning process.

The models demonstrated high accuracy in predicting soil moisture levels, with neural networks and random forests showing particularly strong performance. These predictions were validated against actual measurements, confirming the robustness of the models. This work promises significant advancements in soil moisture forecasting. The insights gained from this project can guide future research and practical applications.

Introduction

The water content of the soil is one of the important factors in the hydrological cycle of the planet and influences many environmental processes, agricultural productivity, as well as ecosystem health. This crucially determines plants' growth, fertility of soils, availability of water and climate dynamics. The precise knowledge and prediction capacity about the level of soil moisture are basic for effective management of water resources, sustainable agriculture practices and adaptation to climate change.

Usually, it involve a laborious gravimetric method or costly network sensors placed on field. However, these methods have been characterized by limitations such as spatial extent scarcity, time inconsistency and costliness. Machine learning algorithms combined with sophisticated measurement tools could offer an opportunity to develop new and less expensive procedures for predicting soil moisture levels.

Machine learning which belongs to artificial intelligence gives powerful tools to analyze complex data sets with multiple hidden patterns needed for accurate predictions. With machine learning algorithms therefore, we can use meteorological data sets along with environmental information to build predictive models for soil moisture estimation.

In conclusion, this comprehensive study highlights the powerful prediction and forecasting abilities of machine learning techniques for soil moisture levels. The developed models offer a cost-effective and scalable solution, promising to enhance environmental management and agricultural practices by providing timely and accurate soil moisture forecasts. Utilizing modern technologies for data collection, analysis, and visualization**.**

Methodology

- **Gathering and Preparing Data:**
  GLDAS, which stands for Global Land Data Assimilation System, is a widely used system for integrating satellite and ground-based observational data with advanced land surface modeling to produce high-quality gridded datasets of land surface variables. These datasets are essential for various applications, including weather forecasting, climate modeling, agricultural monitoring, and water resource management.
  - Data used : Meteorological datasets like precipitation, soil temperature, air temperature, and soil moisture from reliable sources such as weather stations and satellite observations.
  - Soil Moisture Sensor : Soil Moisture, Air Temperature ,Soil Temperature ,Rainfall data is of 41 years from 1980 jan to 2021 dec of pune location
  - Data Preprocessing: Raw data undergo preprocessing, including handling missing values, normalization, and feature engineering, ensuring they are ready for model training.
  - Missing Data handling : To handle the missing values from the dataset, fillna method is used to replace missing values with nan value(not a number)

- **Selecting and Crafting Features:**
  - Features with significant influence on soil moisture dynamics were selected carefully.
  - Applying advanced techniques like polynomial features and dimensionality reduction, we engineered informative features from the input data.

- **Building Models:**
  - Random Forest Regressor: Leveraging the power of ensemble learning, a Random Forest regression model by combining multiple decision trees, enabling accurate predictions was constructed. This model was trained on the preprocessed meteorological data along with soil moisture measurements.
  - Neural Network Development: Additionally, a neural network architecture using TensorFlow was consider. This neural network is designed to capture intricate nonlinear relationships between input features and soil moisture levels.

- **Training and Assessing Models:**
  - Random Forest regressor and the neural network models on a subset of the dataset, were trained. As data is spilt into 80-20% so on 20% data models were trained.
  - To gauge model performance, I employ standard regression metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R^2) on the validation dataset.

- **Tuning Model Hyperparameters:**
  - Using techniques such as grid search or random search, I fine-tune the hyperparameters of both the Random Forest regressor and neural network models to optimize their performance.

- **Visualizing Insights:**
  - Utilizing Matplotlib and Seaborn : To visualized various aspect of the data & model output Matplotlib & Seaborn libraries were used.
  - Visualizations include scatter plots for comparing predicted and actual soil moisture values, feature importance plots generated by the Random Forest regressor.

- **Interpreting Results:**
  - We delve into model predictions and visualizations to extract valuable insights into the factors influencing soil moisture dynamics.
  - By analyzing feature importance rankings from the Random Forest regressor and examining the contributions of different environmental variables, a deeper understanding of soil moisture variability can be gained.

- Through this methodological approach, Aim is to develop accurate and interpretable models for predicting soil moisture at different levels, equipped with comprehensive visualizations to aid in understanding and decision-making processes.

3. Machine learning model

3.1 Overview of Machine Learning Model

Machine learning (ML) is a branch of artificial intelligence (AI) that empowers machines or systems to learn from data or experiences rather than relying solely on predefined rules or equations. As articulated by Mitchell (1997), ML involves the study of computer algorithms that enable programs to automatically enhance their performance through experience, akin to human learning processes. ML algorithms come in various forms and are adept at solving real-world problems in a heuristic manner.

There are three primary types of ML algorithms:

- **Supervised Learning**: In supervised learning, the training dataset includes labeled data, where the output is already known. The model is trained to predict the desired output based on input data. Common applications of supervised learning include classification and regression tasks. Examples of supervised ML algorithms include Support Vector Machines (SVMs), linear regression, logistic regression, Naïve Bayes, neural networks, and Random Forests (RF).

- **Unsupervised Learning**: Unsupervised learning involves using unlabeled data, allowing algorithms to infer patterns without external guidance. This type of ML is employed for tasks such as identifying hidden patterns, grouping similar data, exploratory data analysis, and dimensionality reduction. Unsupervised learning primarily deals with clustering and association. Typical examples of unsupervised ML algorithms include k-means clustering, hierarchical clustering.

- **Reinforcement Learning**: Reinforcement learning (RL) involves algorithms navigating uncertain environments by taking actions. Feedback, in the form of rewards or penalties, guides the algorithm to maximize performance. Unlike supervised and unsupervised learning, RL doesn't rely on predefined data; interactions with the environment generate data. Decisions are made sequentially. Common RL algorithms include Q-learning, SARSA, and DDPG. RL is frequently combined with deep learning for various applications.

- In the prediction of soil moisture, Supervised Learning is employed since we have labeled independent and dependent variables. Given that soil moisture is a numeric value, and predictions are also numeric, this task falls under regression. Therefore, supervised regression learning is utilized to predict soil moisture accurately.

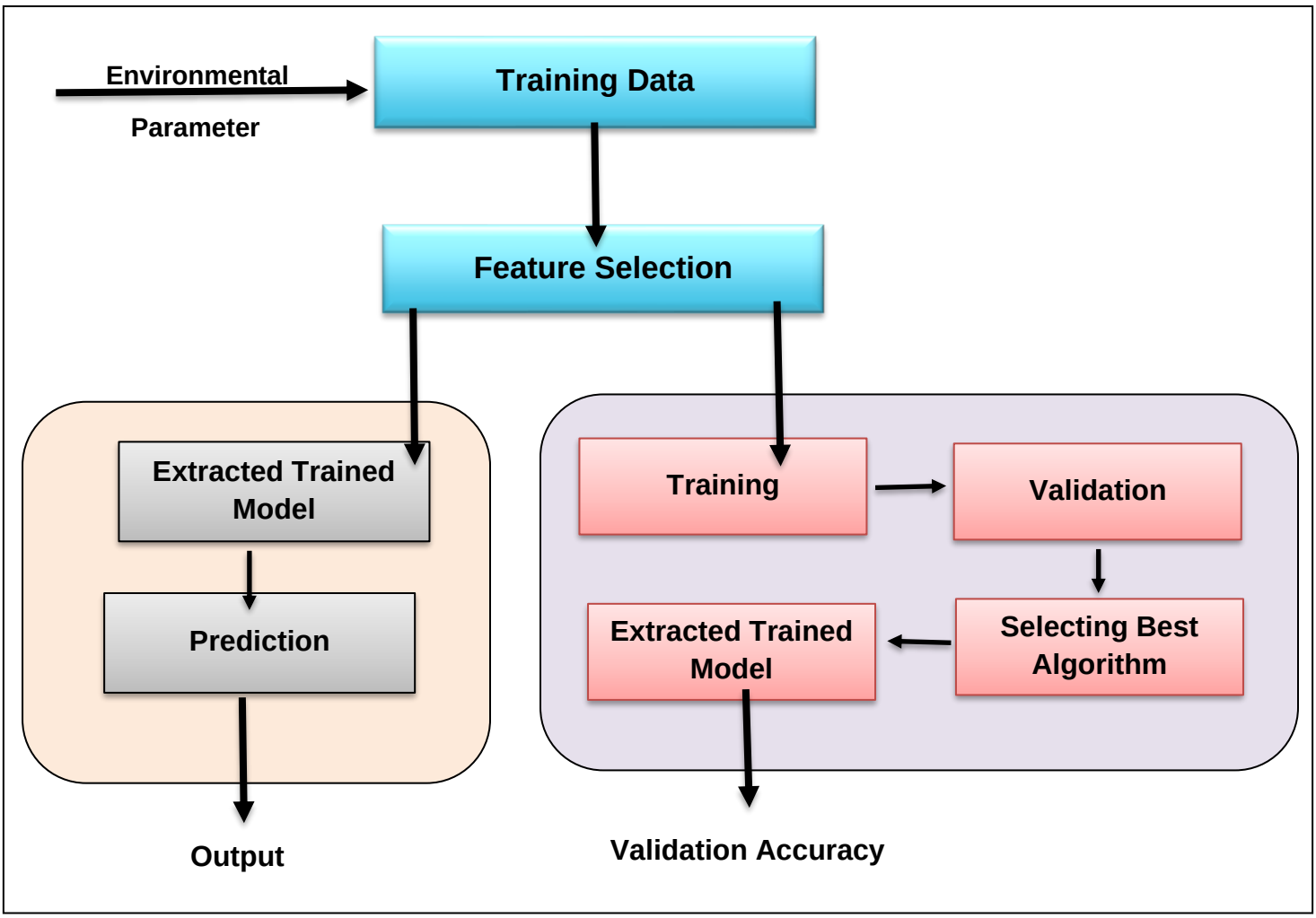


Fig.3.1 : Block Diagram of working of Machine Learning Technique

3.2. Machine learning algorithms applied in soil moisture research

In soil moisture research, mainly supervised ML algorithms are applied. Out of all the supervised ML algorithms, the most commonly used algorithms in soil moisture assessment having ample available literature are briefly discussed in this chapter.

3.2. 1 Linear regression

Linear regression is the simplest and most commonly used method for prediction. It predicts by considering the linear relationship between the dependent variable and independent variables. If there is only one independent variable, then it is called a simple linear regression. If there is more than one independent variables, then it is referred to as multiple linear regression (MLR). The simple linear equation can be denoted by the following equation:

$$y = \alpha + \beta x$$

where, y: Dependent or response variable
α: Intercept
β: Slope
x: Independent or predictor variable
MLR can be represented in the form of the following equation:

$$y = \beta 0 + \beta 1x1 + \beta 2x2 + \cdots + \beta ixi + E$$

where, y: Dependent or response variable
xi: Independent or predictor variables
β0 : y-intercept
βi: slope coefficients for each independent variable
E: error term or residuals
Linear regression performs well when data is linearly separable and predictor variables are independent of each other. It should not be applied if less data is available, as it is prone to noise and overfitting. Linear regression is also sensitive to outliers. The multicollinearity among the variables should be removed before applying it.

3.2.2 Random Forest:

Random Forest is an ensemble learning method primarily used for classification and regression tasks. It operates by constructing a multitude of decision trees during training and outputting the class that is the mode of the classes (classification) or the mean prediction (regression) of the individual trees.

How Random Forest Works

1. Creating Bootstrap Samples:
   - From the original dataset, need to create multiple bootstrap samples. Each sample is created by randomly selecting data points with replacement. This means some data points may appear multiple times in a sample, while others may not appear at all.

2. Training Decision Trees:
   - Training a decision tree on each bootstrap sample. During the training process, each tree only considers a random subset of features for splitting at each node. This random selection of features ensures that the trees are diverse.

3. Making Predictions:

- For classification tasks, each tree in the forest votes for a class, and the class with the majority votes is the final prediction.
- For regression tasks, the predictions of all the trees are averaged to get the final output.

4. Aggregating Predictions:
    - For a new data point, each tree in the forest provides a classification (yes/no). The final prediction is based on the majority vote of the trees.

Advantages of Random Forest

1. Reduction in Overfitting:
    - By averaging multiple trees, Random Forest reduces the risk of overfitting which is common with individual decision trees.

2. Robustness:
    - The algorithm is robust to noisy data and missing values. It can handle large datasets with higher dimensionality.

3. Feature Importance:
    - Random Forest provides an intrinsic measure of feature importance, which helps in understanding which features are the most influential for the predictions.

Hyperparameter Tuning

- Number of Trees (n_estimators): More trees usually improve performance but also increase computation time.
- Maximum Depth (max_depth): Controls the depth of each tree, preventing overfitting if set appropriately.

3.2.3 Artificial Neural Network (ANN)

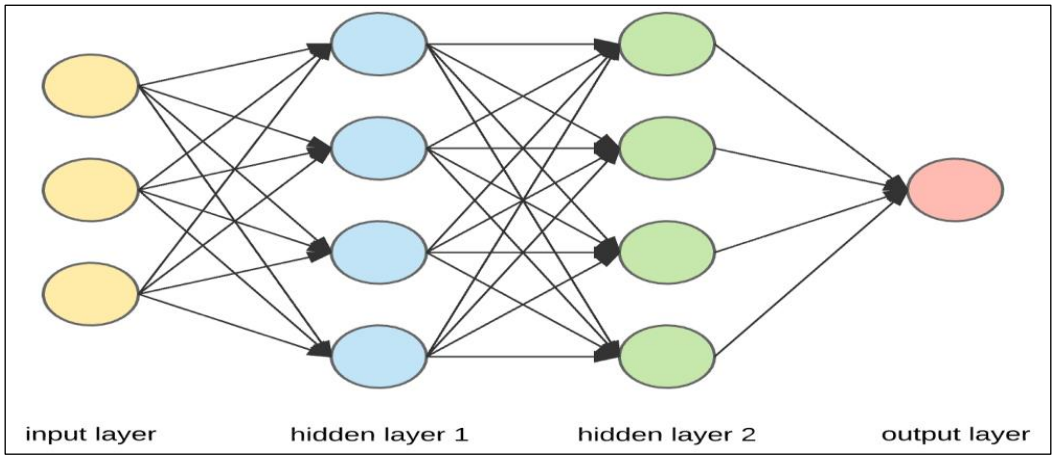
An artificial neural network (ANN) is the piece of a computing system designed to simulate the way the human brain analyzes and processes information. It is the foundation of artificial intelligence (AI) and solves problems that would prove impossible or difficult by human or statistical standards. ANNs have self-learning capabilities that enable them to produce better results as more data becomes available.

An ANN has hundreds or thousands of artificial neurons called processing units, which are interconnected by nodes. These processing units are made up of input and output units. The input units receive various forms and structures of information based on an internal weighting system, and the neural network attempts to learn about the information presented to produce one output report. Just like humans

need rules and guidelines to come up with a result or output, ANNs also use a set of learning rules called backpropagation, an abbreviation for backward propagation of error, to perfect their output results.

An ANN initially goes through a training phase where it learns to recognize patterns in data, whether visually, aurally, or textually. During this supervised phase, the network compares its actual output produced with what it was meant to produce—the desired output. The difference between both outcomes is adjusted using backpropagation. This means that the network works backward, going from the output unit to the input units to adjust the weight of its connections between the units until the difference between the actual and desired outcome produces the lowest possible error.

3.2.3.1 Neural Network: Component



Layered structure of perceptron

Fig.3.1 : neural network component

A neuron is the basic unit of a neural network. They receive input from an external source or other nodes. Each node is connected with another node from the next layer, and each such connection has a particular weight. Weights are assigned to a neuron based on its relative importance against other inputs.

When all the node values from the yellow layer are multiplied (along with their weight) and summarized, it generates a value for the first hidden layer. Based on the summarized value, the blue layer has a predefined “activation” function that determines whether or not this node will be “activated” and how “active” it will be.

3.2.3.2 Hidden Layers and Output Layer

The layer or layers hidden between the input and output layer is known as the hidden layer. It is called the hidden layer since it is always hidden from the external world. The main computation of a Neural Network takes place in the hidden layers. So, the hidden layer takes all the inputs from the input layer and performs the necessary calculation to generate a result. This result is then forwarded to the output layer so that the user can view the result of the computation.

An activation function is a function that is added into an artificial neural network in order to help the network learn complex patterns in the data. When comparing with a neuron-based model that is in our brains, the activation function is at the end deciding what is to be fired to the next neuron. That is exactly what an activation function does in an ANN as well. It takes in the output signal from the previous cell and converts it into some form that can be taken as input to the next cell.

In artificial neural networks (ANNs), the activation function is a mathematical "gate" in between the input feeding the current neuron and its output going to the next layer

3.2.3.3 Activation Function:

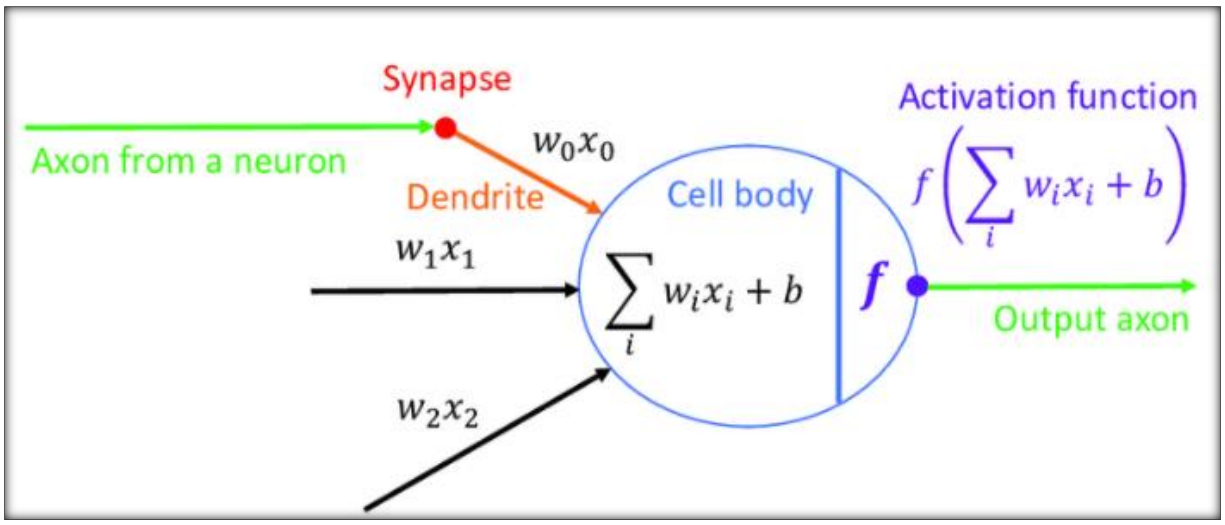


Fig.3.3: activation function

The activation functions are at the very core of Deep Learning. They determine the output of a model, its accuracy, and computational efficiency.

In some cases, activation functions have a major effect on the model's ability to converge and the convergence speed.

Rectified Linear Unit (ReLU)

The Rectified Linear Unit (ReLU) is the most commonly used activation function in deep learning. The function returns 0 if the input is negative, but for any positive input, it returns that value back. The function is defined as:

$$\begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$$

The plot of the function and its derivative:

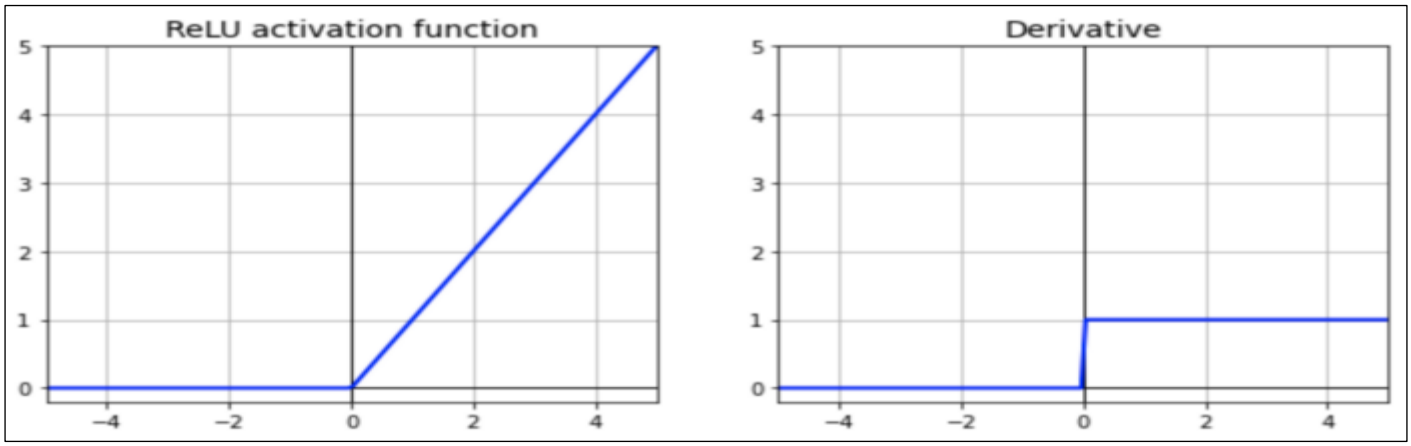


Fig.3.4 : Relu activation graph

As we can see that:

1) Graphically, the ReLU function is composed of two linear pieces to account for non-linearities. A function is non-linear if the slope isn't constant. So, the ReLU function is non-linear around 0, but the slope is always either 0 (for negative inputs) or 1 (for positive inputs).

2) The ReLU function is continuous, but it is not differentiable because its derivative is 0 for any negative input.

3) The output of ReLU does not have a maximum value (It is not saturated) and this helps Gradient Descent

4) The function is very fast to compute (Compare to Sigmoid and Tanh)

It's surprising that such a simple function works very well in deep neural networks.

Learnable Parameters in ANN

Weights and Biases

Weights and biases (commonly referred to as w and b) are the learnable parameters of a machine learning model.

Neurons are the basic units of a neural network. In an ANN, each neuron in a layer and is connected to each neuron in the next layer. When the inputs are transmitted between neurons, the weights are applied to the inputs along with the bias.

$$Y = \sum (weight * input) + bias$$

Weights control the signal (or the strength of the connection) between two neurons. In other words, a weight decides how much influence the input will have on the output.

Biases, which are constant, are an additional input into the next layer that will always have the value of 1. Bias units are not influenced by the previous layer (they do not have any incoming connections) but they do have outgoing connections with their own weights. The bias unit guarantees that even when all the inputs are zeros there will still be an activation in the neuron.

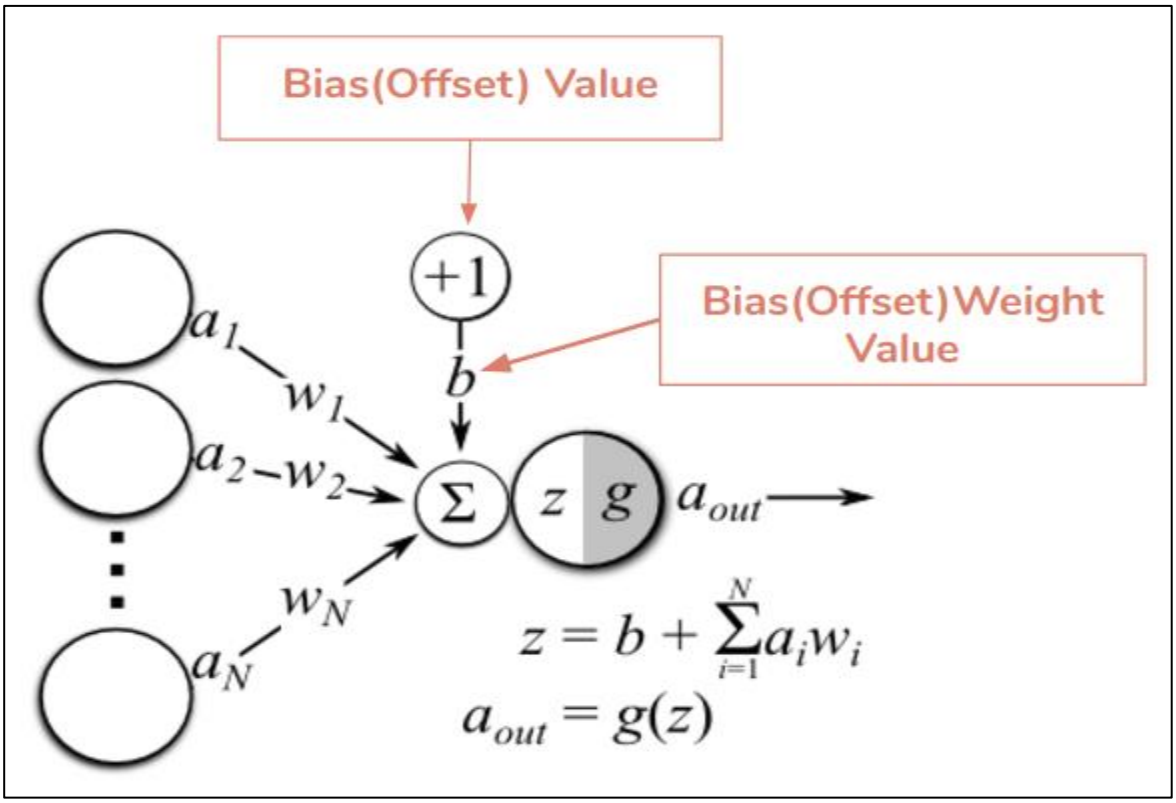


Fig. 3.5 : Bias

3.2.3.4 Training Neural network means learning optimal weights and bias of the Neuron.

Feedforward and Backpropagation

Neural networks consists of neurons, connections between these neurons called weights and some biases connected to each neuron. We distinguish between input, hidden and output layers, where we hope each layer helps us towards solving our problem. To move forward through the network, called a forward pass, we iteratively use a formula to calculate each neuron in the next layer. Keep a total disregard for

the notation here, but we call neurons for activations a, weights w and biases b which is cumulated in vectors.

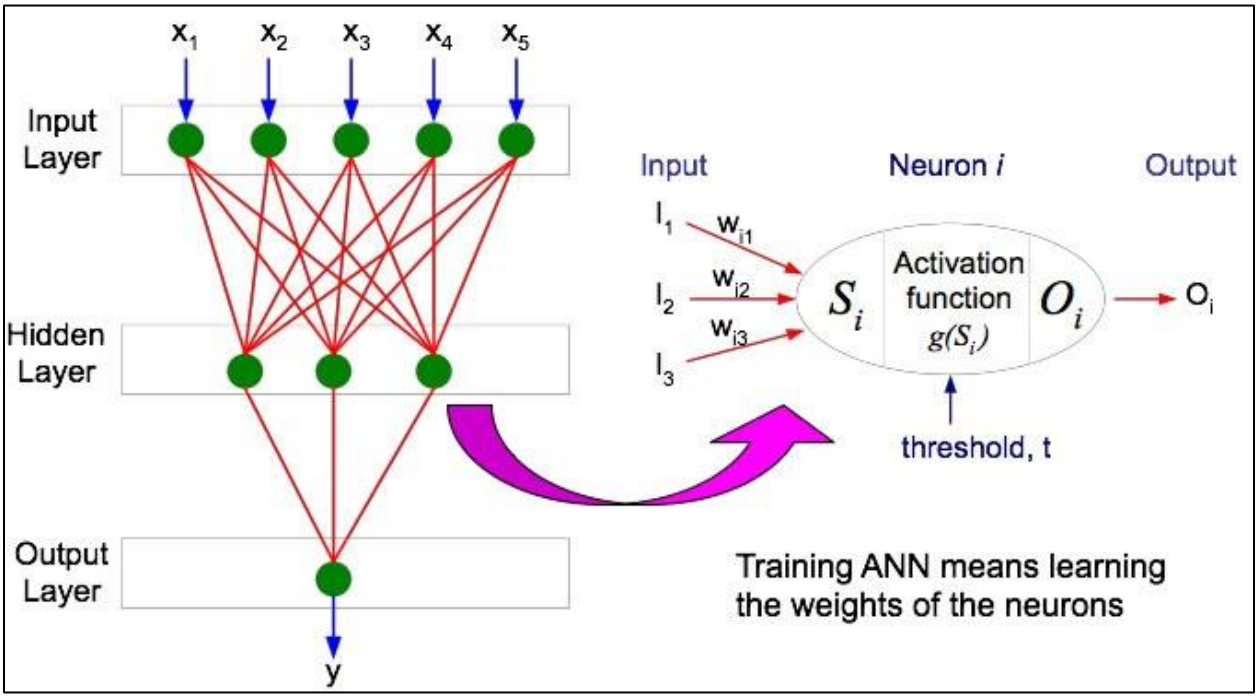


Fig.3.6 : Training of neural network

Following are the steps involved in the Neural Network are :

1) Consider the input equation:

   a. **Z = W0 + W1X1 + W2X2 + …+ WnXn**

   b. and calculate the output, which is the predicted values of Y or the Ypred

2) Calculate the error. It tells how much the model deviates from the actual observed values. It is always calculated as the Ypred – Yactual

3) To minimize the error term in a Neural Network, we use Backward Propagation, which involves adjusting the weights (beta coefficients) to reduce the loss. This process starts by computing the loss, then propagating it back through each layer to update the weights. The initial pass from the Input layer to the Output layer, where these weights are applied to inputs, is known as Forward Propagation.

4.Results

4.1.The correlation heatmap

The correlation heatmap is a graphical representation that depicts the correlation matrix for a set of variables. Each cell in the heatmap displays the correlation between two variables, which can range from -1 to 1. A value of 1 indicates a perfect positive correlation, -1 indicates a perfect negative correlation, and 0 indicates no correlation. The intensity of the color in the heatmap represents the strength of the correlation, with lighter colors indicating stronger correlations.

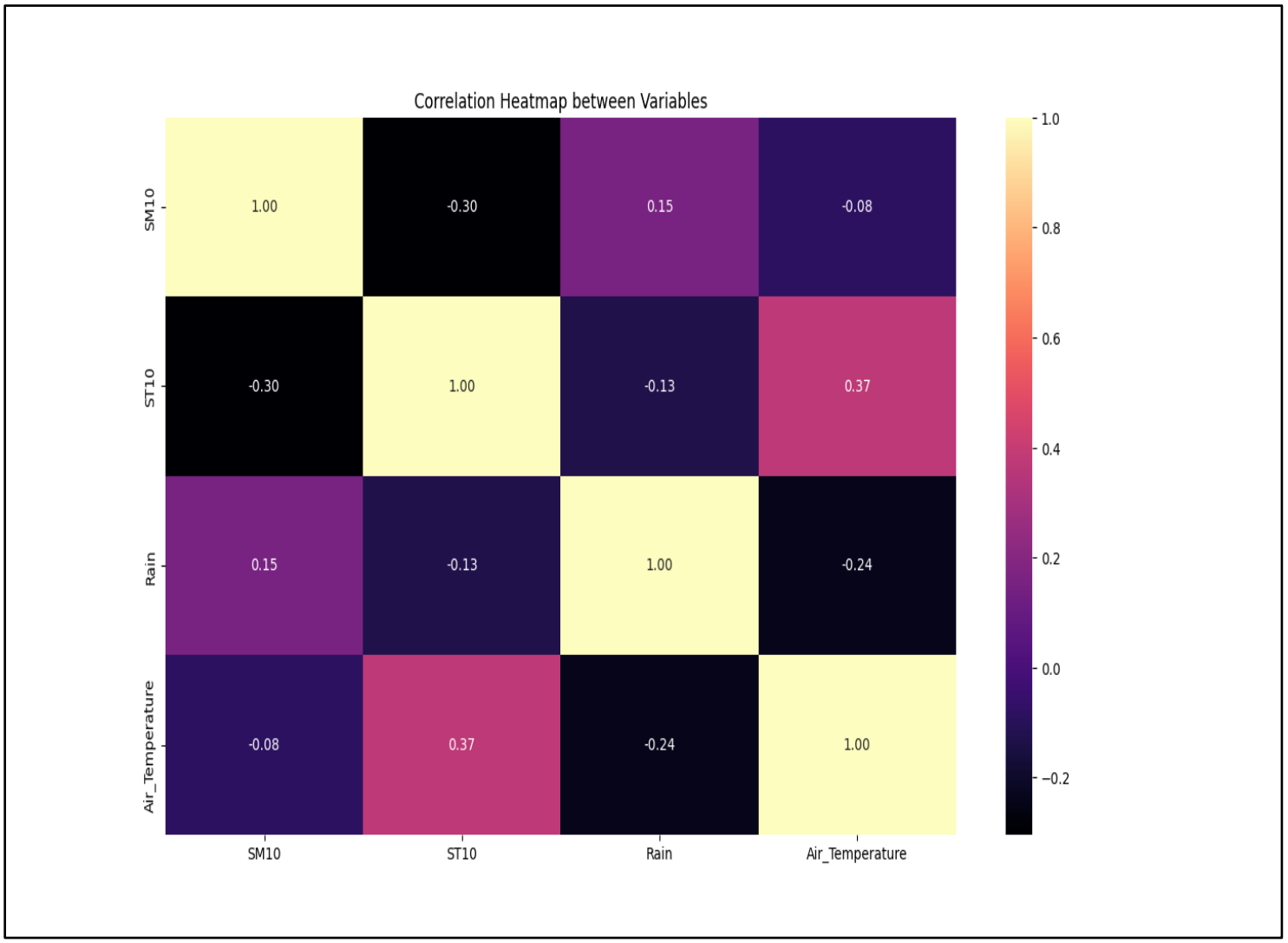


Fig 4 Correlation Heatmap

Here SM10 : Surface soil moisture
St10 : Surface Soil Temperature
Rain : Rainfall
Air_temprature:Air temperature of the day

4.2 Time Series Decomposition

The time series decomposition shown in the provided image displays the following components for the observed time series data spanning from 1980 to 2021:

1. **Observed**:
    a. The observed data shows clear seasonality, with a repeating pattern every year. The magnitude of the observed values fluctuates significantly within each year.
2. **Trend**:
    a. The trend component captures the long-term movement in the data. Initial spike might be due to data discrepancy otherwise no trend is seen.
3. **Seasonal**:
    a. The seasonal component shows a consistent pattern repeating every year, with values oscillating between approximately -10 and 10. This indicates strong seasonality in the data over year, with peaks and troughs occurring at regular intervals annually.
4. **Residual**:
    a. The residual component captures the random noise or irregular fluctuations that are not explained by the trend or seasonal components. The residuals vary significantly over time, with higher variability in the early years (around 1980) and again around 2020.

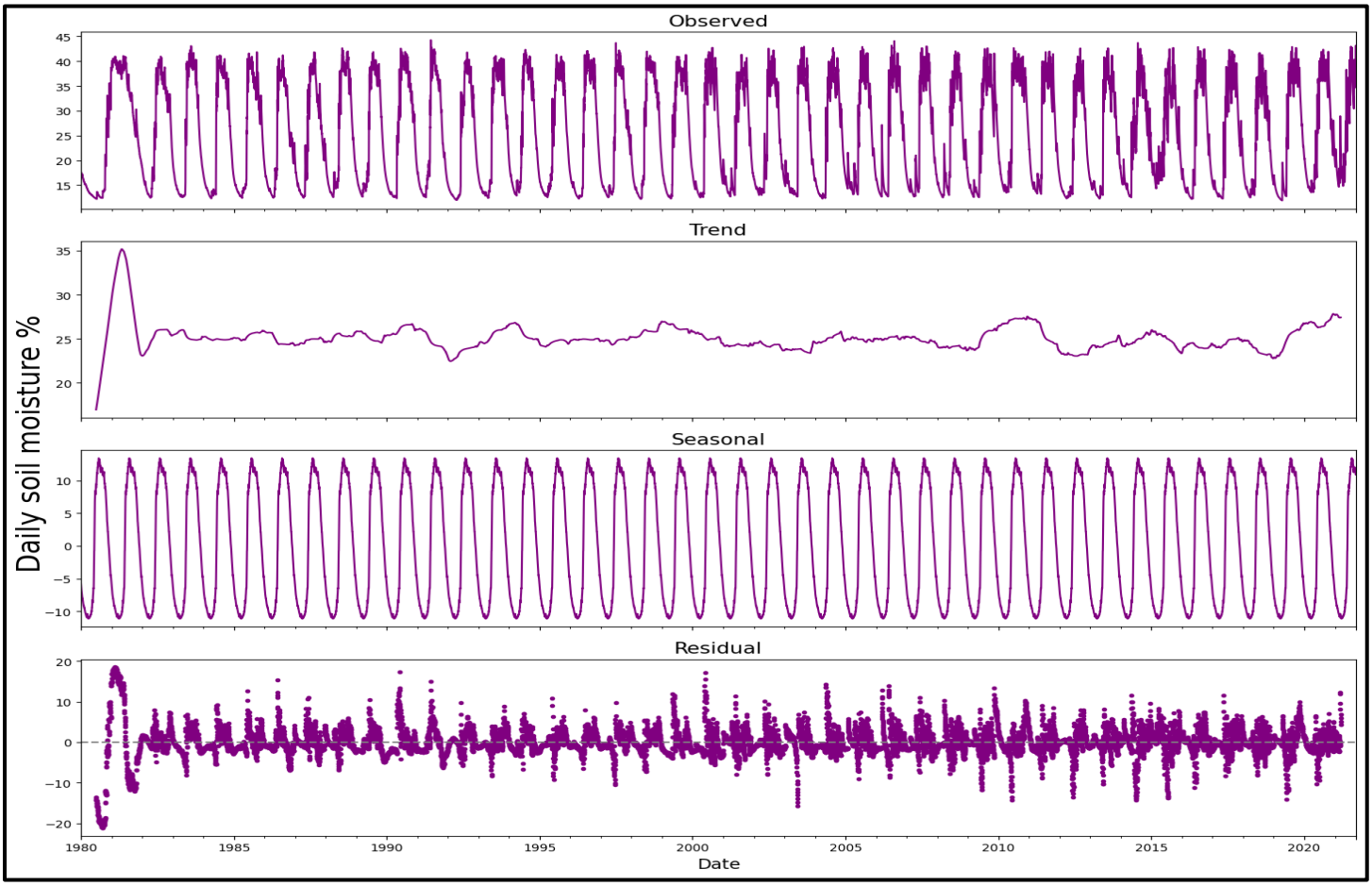


Fig.4.2 Time Series Decomposition

4.3Feature Selection for training random forest model

The comparative analysis of feature importance in Random Forest Regression models using different historical soil moisture data (40 days prior vs. 30 days prior) offers valuable insights into how varying time frames of soil moisture data influence model performance. The two sets of features considered are:

Model with Ref40 (Soil moisture values at lag 40 days):

1. Rainfall (Rain): The most influential feature with an importance score close to 57.4%.
2. ST10: The second most important feature with an importance score around 22.8%.
3. Ref40: Moderately important with a score close to 19.0%.
4. Temperature: Least important with a score close to 0.9%.

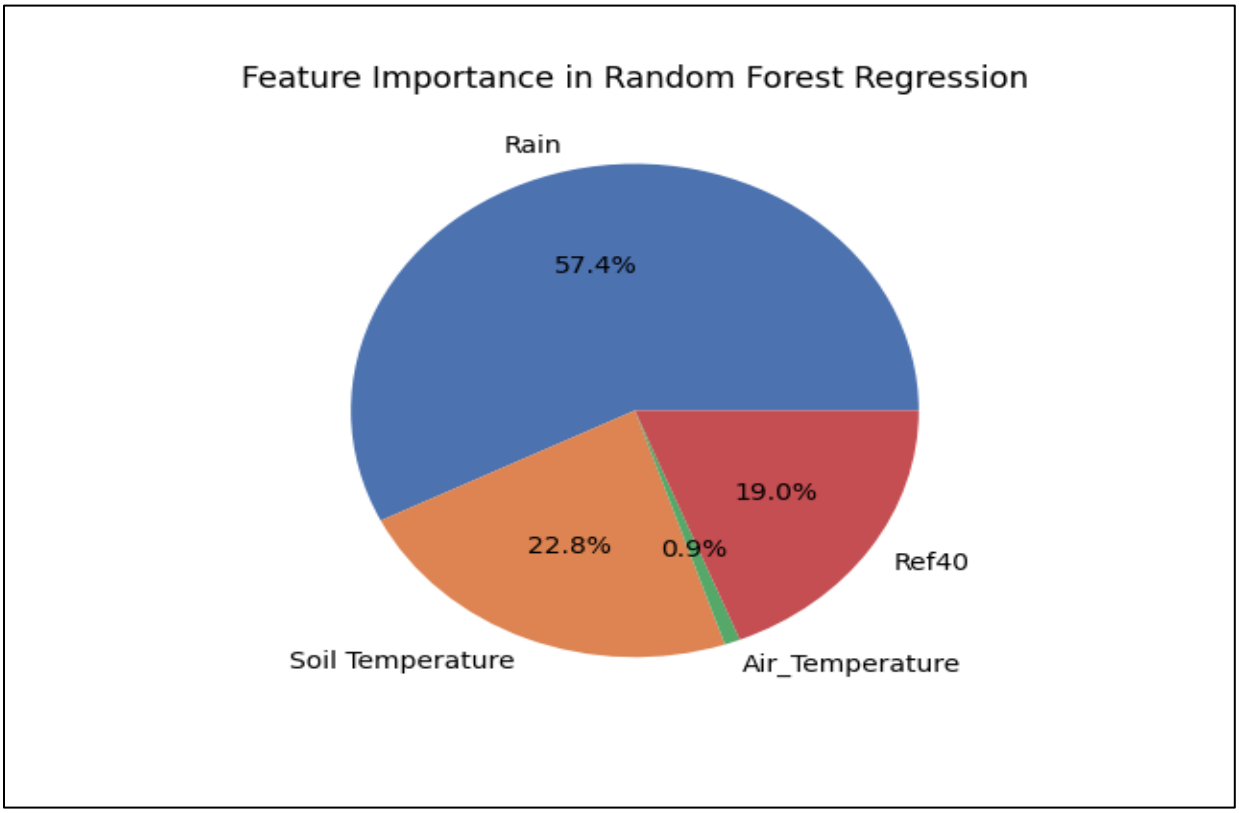


Fig.4.3.1 Feature Importance

Model with Ref30 Soil moisture values at lag 30 days):

1. Ref30:The most influential feature with an importance score close to 67.3%.
2. ST10:The second most important feature with an importance score around 304%.
3. Rain: Least important with a score close to 0.
4. Temperature: Least important with a score close to 0.

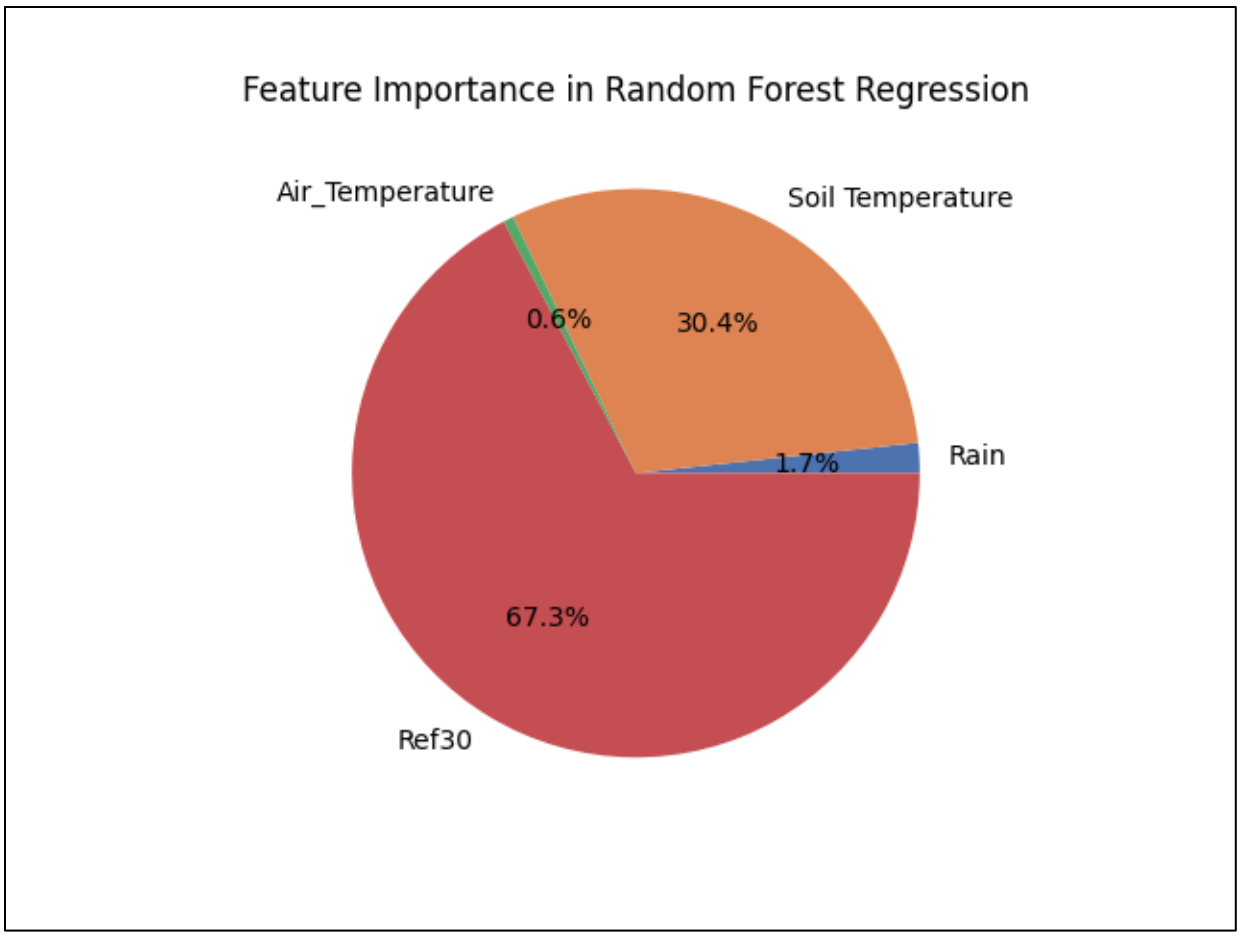


Fig.4.3.2 Feature Importance

1. Rainfall vs. Ref30/Ref40: In the model with Ref40, rainfall is the most significant predictor. However, in the model with Ref30, rainfall's importance drops drastically, becoming negligible. This suggests that when using more recent soil moisture data (30 days prior), the influence of rainfall diminishes.

2. Soil Moisture (Ref30 vs. Ref40): Predictive importance of 30 days lag soil moisture is more than that of the 40 days soil moisture.

3. Soil Temperature (ST10): Soil temperature at 10 cm depth (ST10) consistently holds the second position in both models, though its importance score is slightly higher in the model with Ref30. This consistency underscores the significance of soil temperature in predicting the target variable.

4. Air Temperature (Temperature): Air temperature remains the least important feature in both models, suggesting it has minimal impact regardless of the time frame of soil moisture data used.

4.4 Comparison of Observed and Predicted Soil Moisture for 2017

Observed Soil Moisture

- Trends:
- Lowest values from January to April, with median values around 15.
- Sharp increase starting in May, peaking from June to September with median values above 35.
- Gradual decrease from October to December, with December values similar to May.
- Variability:
- High variability in summer months (June-August), indicated by wide interquartile ranges and outliers.
- Lower variability in early (January-April) and late months (November-December) with narrower interquartile ranges.

Predicted Soil Moisture

- Trends:
- Follows the observed trend with low values from January to April.
- Significant increase beginning in May, peaking in the summer months.
- Decrease from October to December, consistent with the observed data.
- Variability:
- High variability during summer months, especially June and July.
- Lower variability in early and late months, similar to the observed data.

Key Observations and Differences

1. Trend Consistency:

- The predicted values closely match the observed trend of soil moisture, capturing the increase in summer and decrease towards year-end.

2. Peak Values:

- Both observed and predicted data show peak median values in the summer. However, the predicted values for June are slightly higher than the observed ones.

3. Variability:

- Both datasets exhibit similar variability patterns, with more variability in the summer and less in the winter and early spring months.
- The predicted data shows more outliers in some months, suggesting possible overfitting or model sensitivity.

4. Anomalies:

- Outliers are present in both datasets, particularly in transitional months like May and October, reflecting inherent variability in soil moisture due to changing weather conditions.

4.5 Box plot of 2017

4.5.1 Observed soil moisture

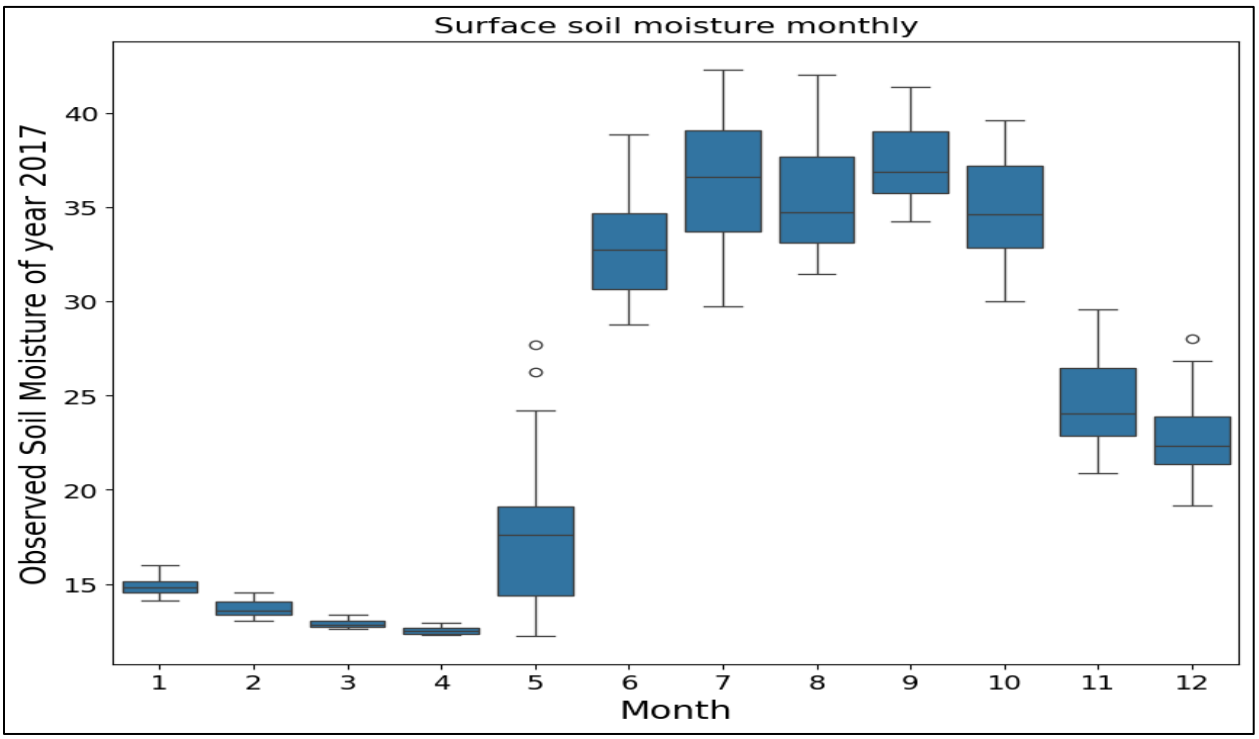


Fig.4.5.1 Boxplot of observed soil moisture

4.5.2 Predicted Soil Moisture using Random Forest

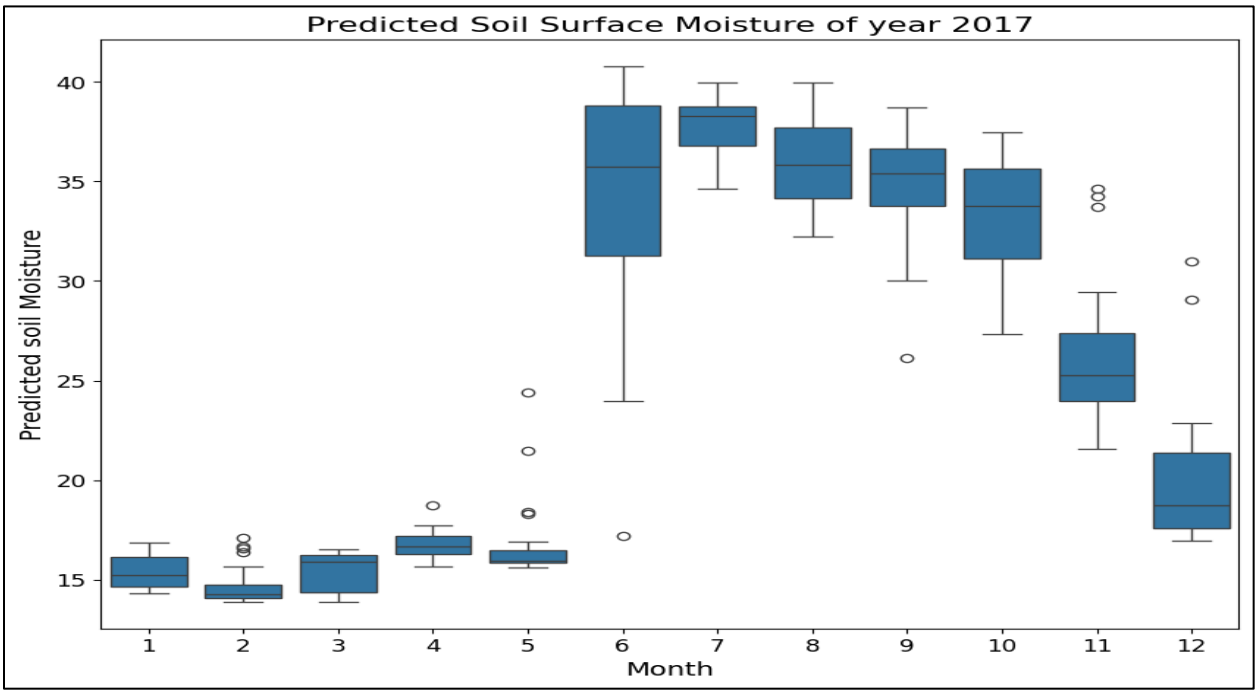


Fig.4.5.2 Boxplot of Predicted soil moisture using random forest

4.5.3 Predicted Soil Moisture using Neural Network

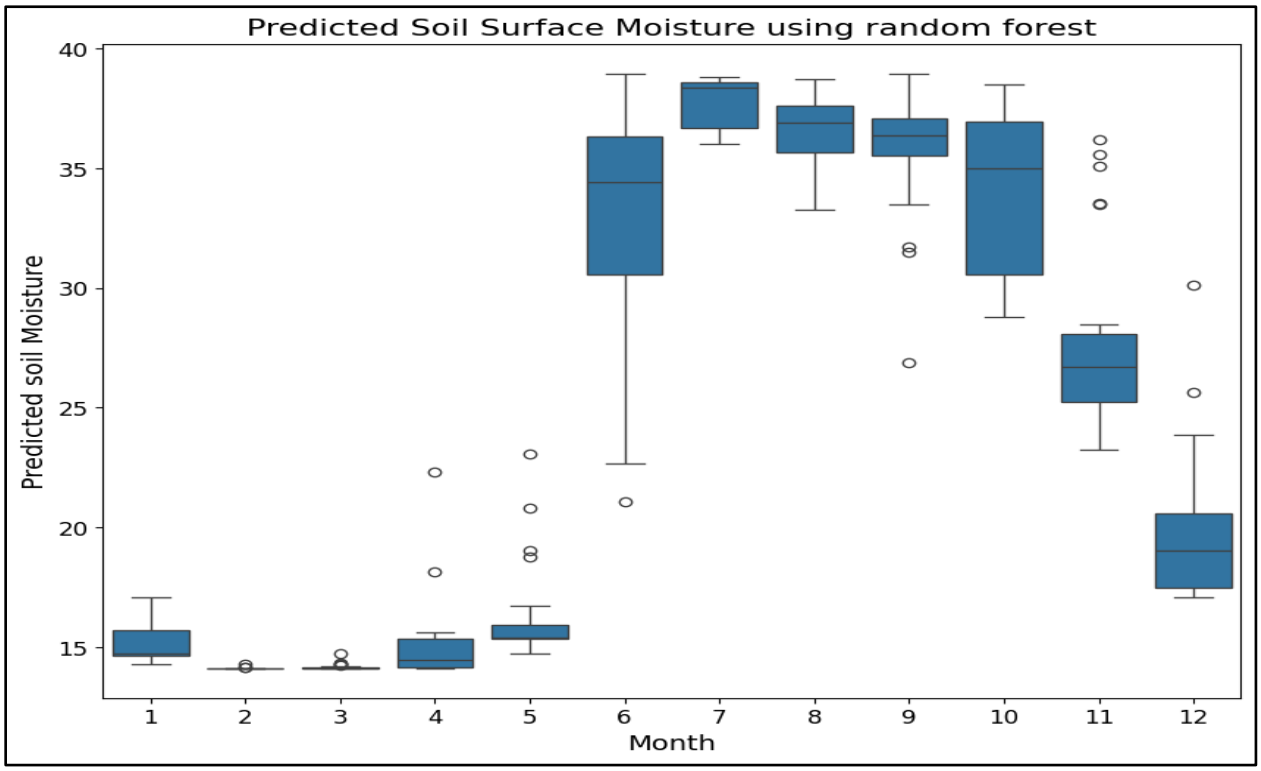


Fig.4.5.3 Boxplot of Predicted soil moisture using ann

Overview

This report compares the predicted and observed surface soil moisture levels for each month of the year 2017, as shown in the respective box plots.

Objective

Seasonal Comparison

- Pre-monsoon (March to May):

1. March:
    a. Predicted: Soil moisture levels are low and consistent, around 14 to 18 %.
    b. Observed: Slightly higher levels, around 15 to 20 %, with minimal variability and a few outliers.
2. April:
    a. Predicted: Slight increase in soil moisture, still relatively low.
    b. Observed: Similar trend with slight increase, maintaining consistency.
3. May:
    a. Predicted: Noticeable increase in moisture with significant variability and several outliers.

b. Observed: Increase observed but remains lower than predicted, around 20 to 25 %.

- Monsoon (June to September):

1. June:
    a. Predicted: Peak moisture levels, especially high with values ranging from 25 to 45 %.
    b. Observed: Similar peak but more consistent, with values ranging from 30 to 40 %.
2. July to September:
    a. Predicted: High moisture levels with gradual decrease towards September, ranging from 30 to 35 units.
    b. Observed: Consistent high levels, similar to predicted with a slight decrease towards the end of the monsoon.

- Post-monsoon (October to November):

1. October:
    a. Predicted: Decrease in moisture levels, ranging from 30 to 35 %.
    b. Observed: Similar decrease but slightly lower, ranging from 25 to 35 %.
2. November:
    a. Predicted: Significant decline, around 20 to 25 %.
    b. Observed: Decline observed, but levels remain slightly higher, around 20 to 25 %.

- Winter (December to February):

1. December to February:
    a. Predicted: Soil moisture levels return to lower values, around 15 to 20 % in December.
    b. Observed: Levels remain slightly higher than predicted, around 20 to 25%, with minimal variability and a few outliers.

Comparative Analysis

a. Both datasets show similar seasonal trends with low moisture levels at the beginning and end of the year, and high levels in the middle months.
b. Some discrepancies are noted, particularly in May and June where predicted values are higher and more variable than observed.
c. Overall, observed data tends to be slightly higher than predicted in the early and late months of the year.

Conclusion

The predicted and observed surface soil moisture data for 2017 align well in general trends, though specific months show notable differences. These insights highlight areas for improving prediction models and understanding soil moisture dynamics.

4.6 TIME SERIES PLOT

Observations:

- **X-Axis (Time):**
  - The x-axis represents time, ranging from 2017 to 2020.
- **Y-Axis (Soil Moisture):**
  - The y-axis represents soil moisture values, likely in some consistent units.
- **Lines:**
  - There are two lines in the plot:
  - Blue Line: Represents the training set, showing the observed soil moisture values.
  - Orange Line: Represents the test set, showing the predicted soil moisture values by the models.
- **Trends:**
  - Both lines display similar trends over time, with peaks and troughs aligning closely. This indicates that the models (Random Forest and ANN) are capturing the seasonal variations and trends in soil moisture quite well.
  - Some deviations between the predicted and observed values can be noticed, but the overall patterns are well-aligned.
- **Model Performance:**
  - The closeness of the orange line to the blue line indicates the effectiveness of the models in predicting soil moisture. The smaller the deviation between the two lines, the better the model performance.
  - There are some periods where the predictions deviate more from the observed values, which might indicate periods of higher variability or more challenging conditions for the models.

Report:

- **Data and Models:**
  - The analysis uses soil moisture data from 2017 to 2020.
  - Predictions are made using two machine learning models: Random Forest and ANN.

- **Model Evaluation:**
  - Both models were trained on year 1098 to 2016 and tested on 2017-2020
  - The predicted values from the models are compared against the observed values to evaluate performance.

- **Visual Analysis:**
  - The plot shows that both models capture the general trend and seasonal variations in soil moisture well.
  - There are periods of discrepancy where the predictions deviate from the observed values, which could be areas for model improvement.

4.6.1 TIME SERIES PLOT (Using Random Forest)

Blue line indicates the observed SM from year 1980 to 2016 and orange line indicates the predicted SM. Four plots for surface soil moisture, soil moisture at 30cm,60cm and 100cm respectively.

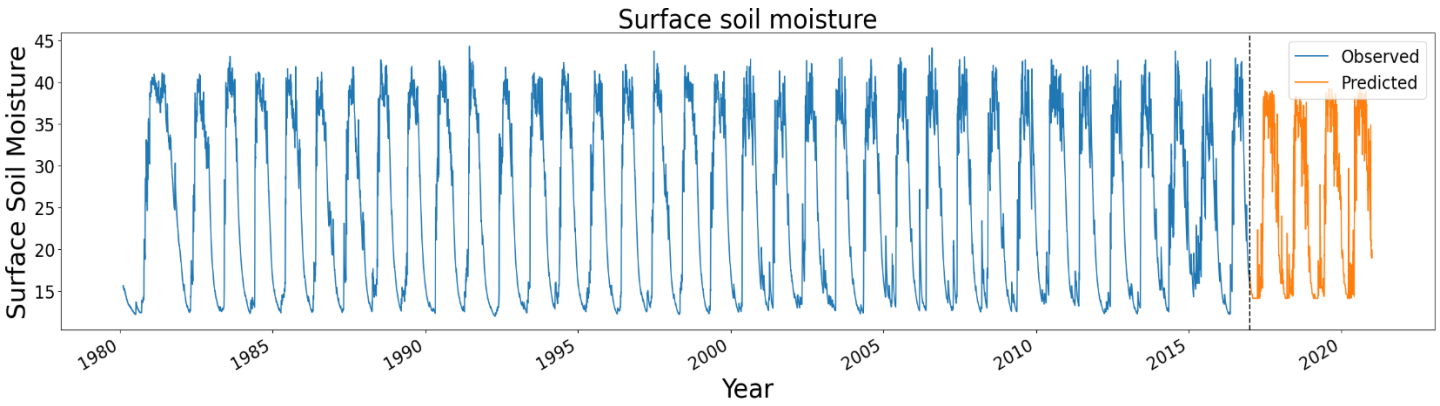


Fig.4.6.1.1 Training and testing Surface Soil Moisture

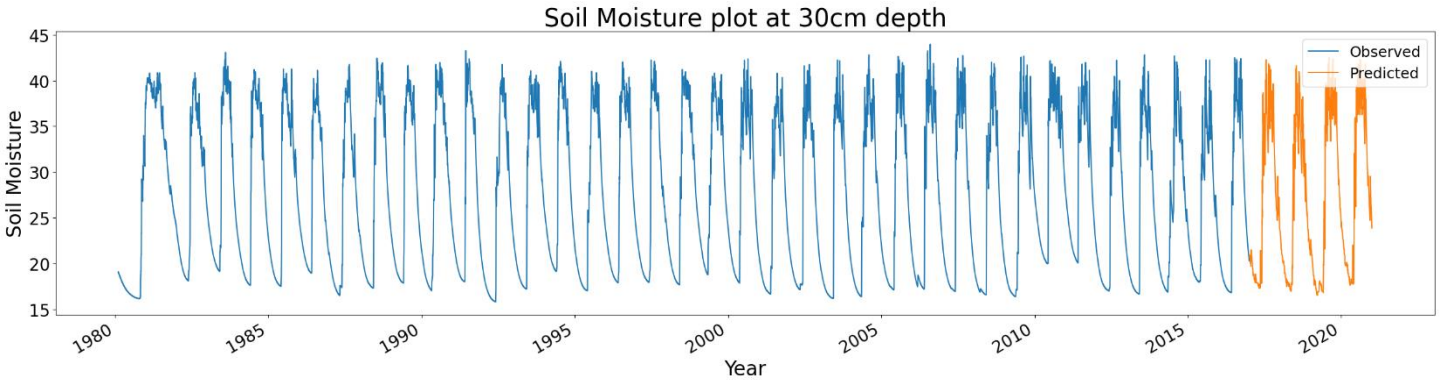


Fig.4.6.1.2 Training and testing Soil Moisture at 30cm

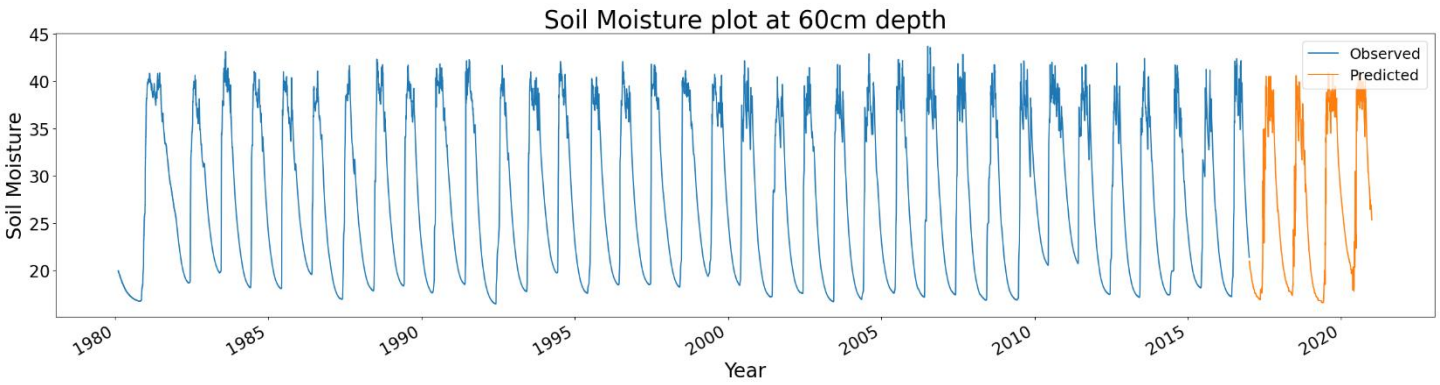


Fig.4.6.1.3 Training and testing Soil Moisture at 60cm

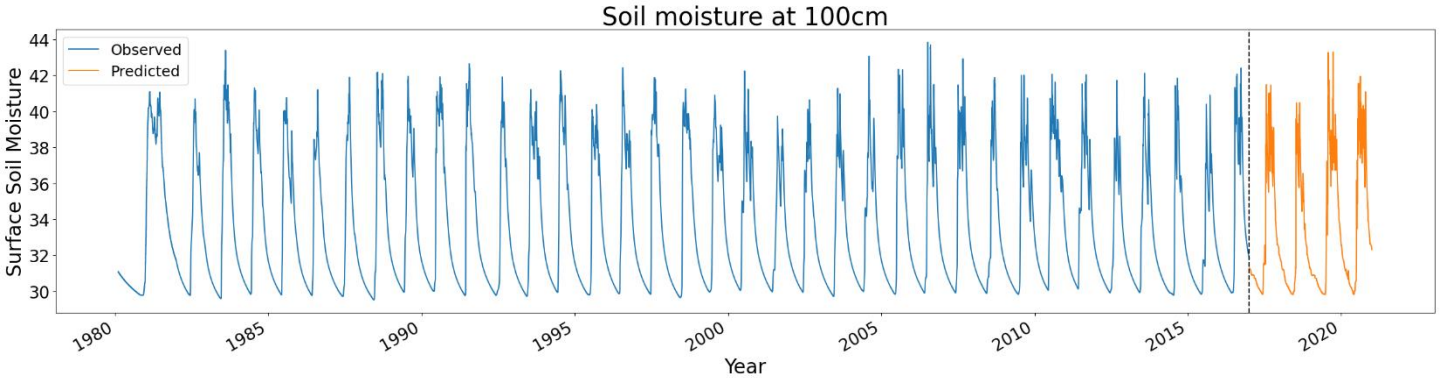


Fig.4.6.1.4 Training and testing Soil Moisture at 100cm

4.6.2 TIME SERIES PLOT (Using Artificial Neural Networks)

Blue line indicates the observed SM from year 1980 to 2016 and orange line indicates the predicted SM. Four plots for surface soil moisture, soil moisture at 30cm,60cm and 100cm respectively.

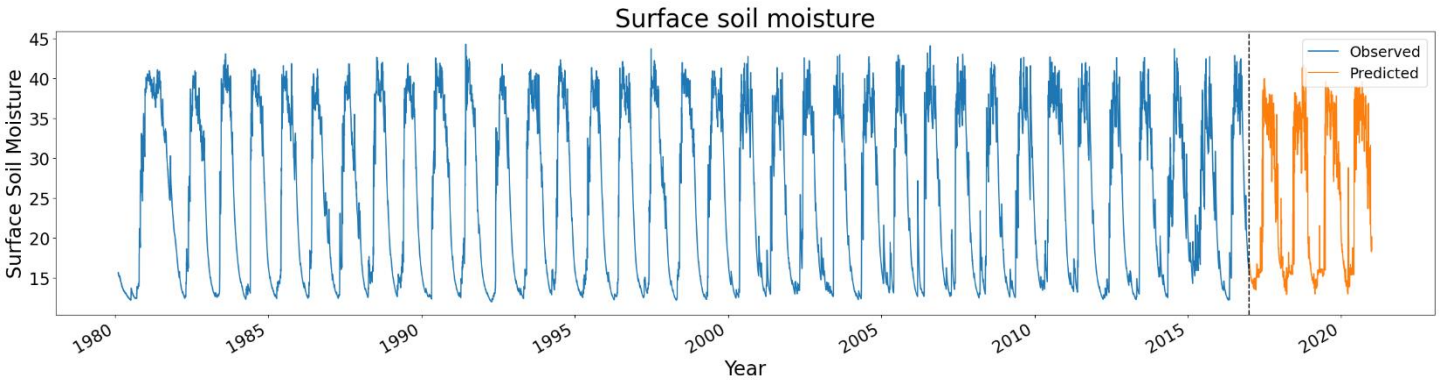


Fig.4.6.2.1 Training and testing Surface Soil Moisture

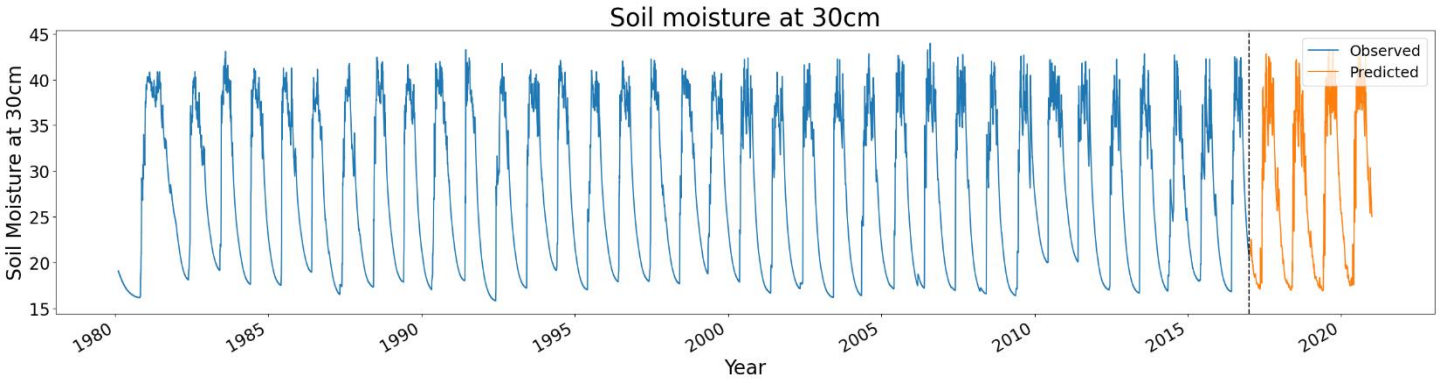


Fig.4.6.2.2 Training and testing Soil Moisture at 30cm

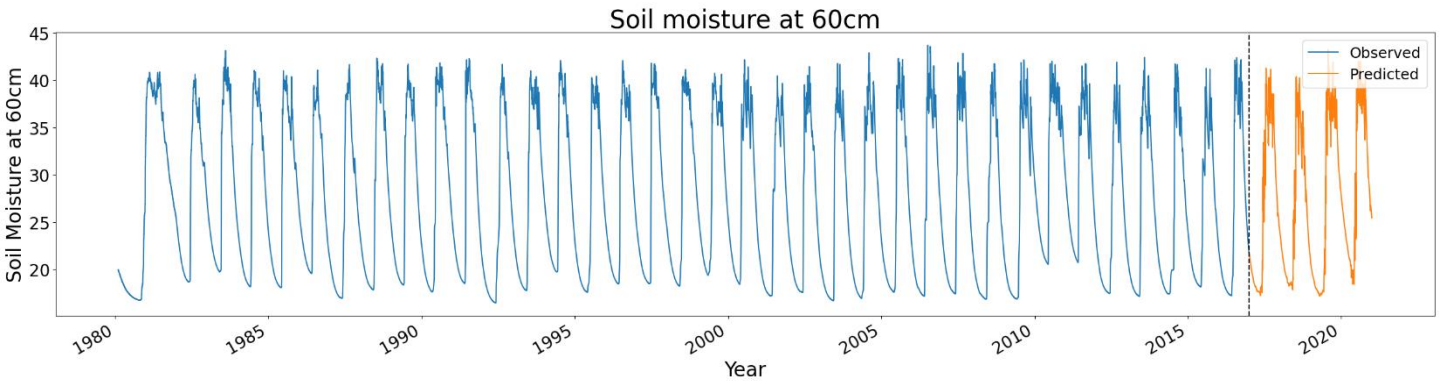


Fig.4.6.2.3 Training and testing Soil Moisture at 60cm

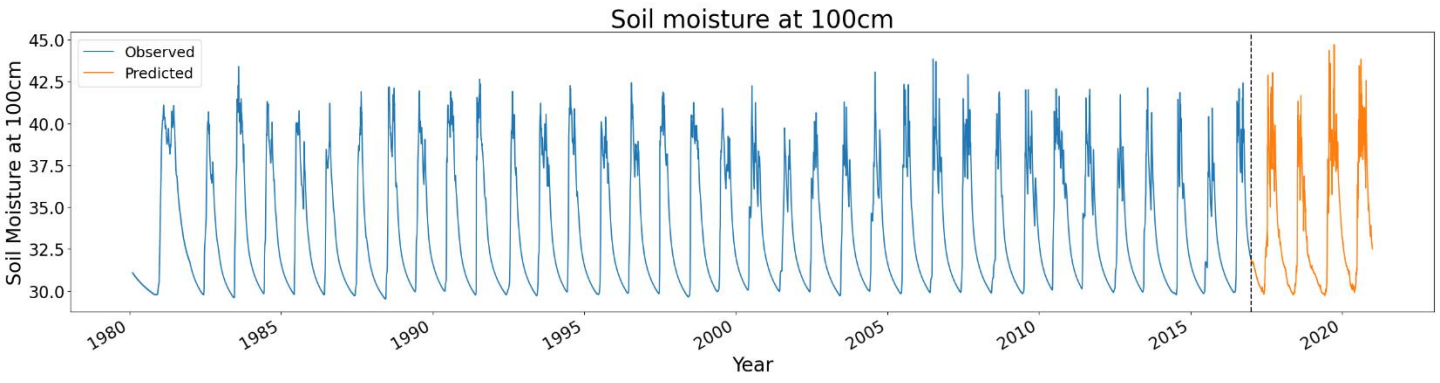


Fig. 4.6.2.4 Training and testing Soil Moisture at 100cm

5.1 Plot of Predicted soil moisture vs actual observed soil moisture using Random Forest

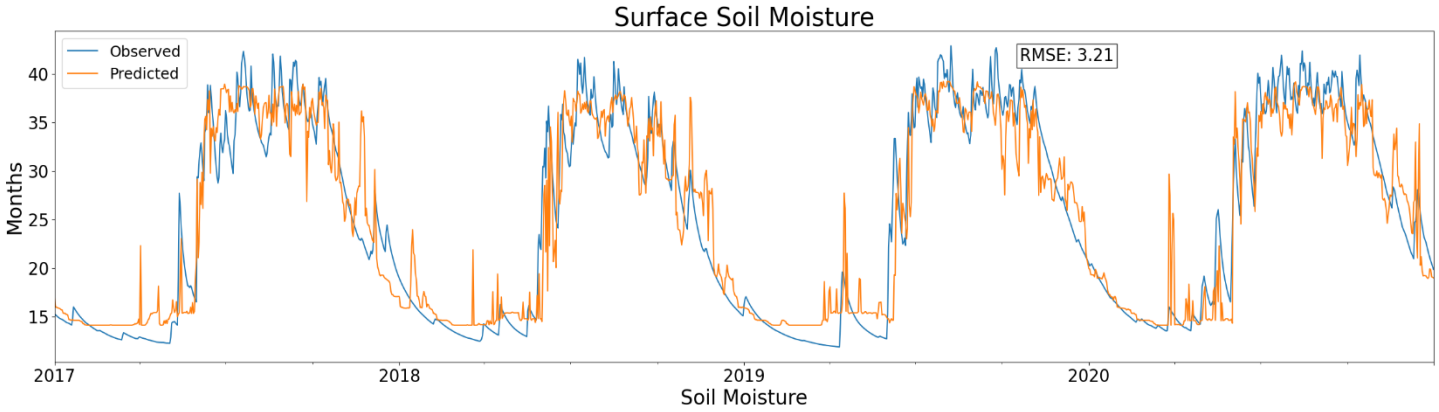


Fig.5.1.1 Observed vs Predicted Surface Soil Moisture using rf

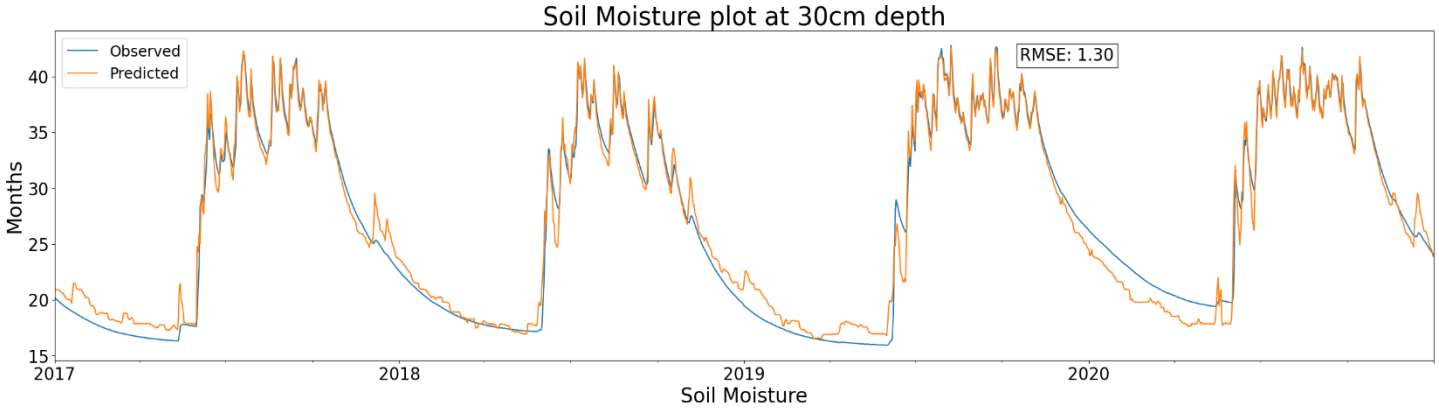


Fig..5.1.2 Observed vs Predicted Soil Moisture at 30cm using rf

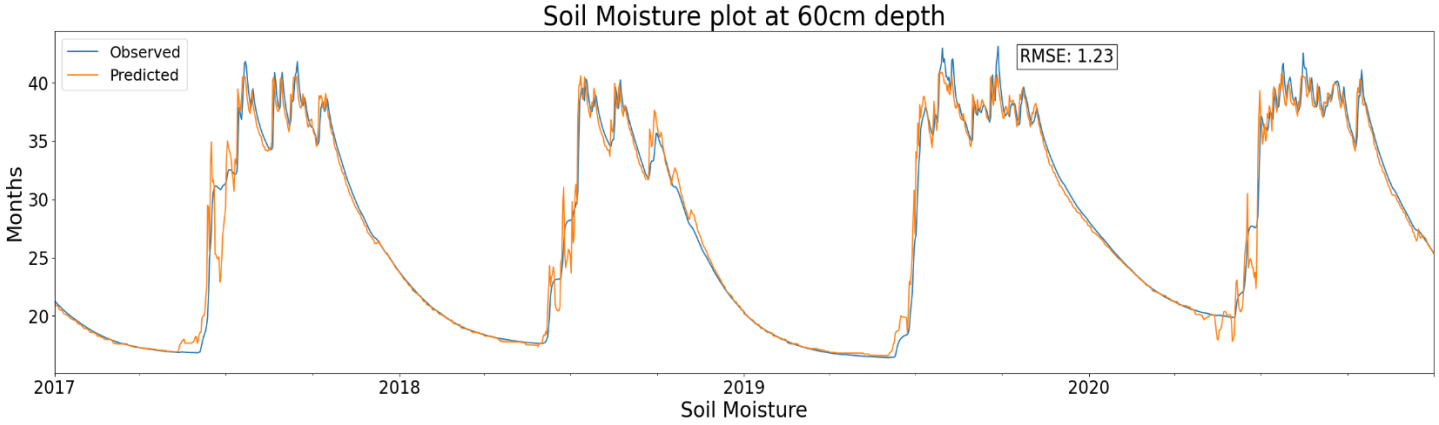


Fig..5.1.3 Observed vs Predicted Soil Moisture at 60cm using rf

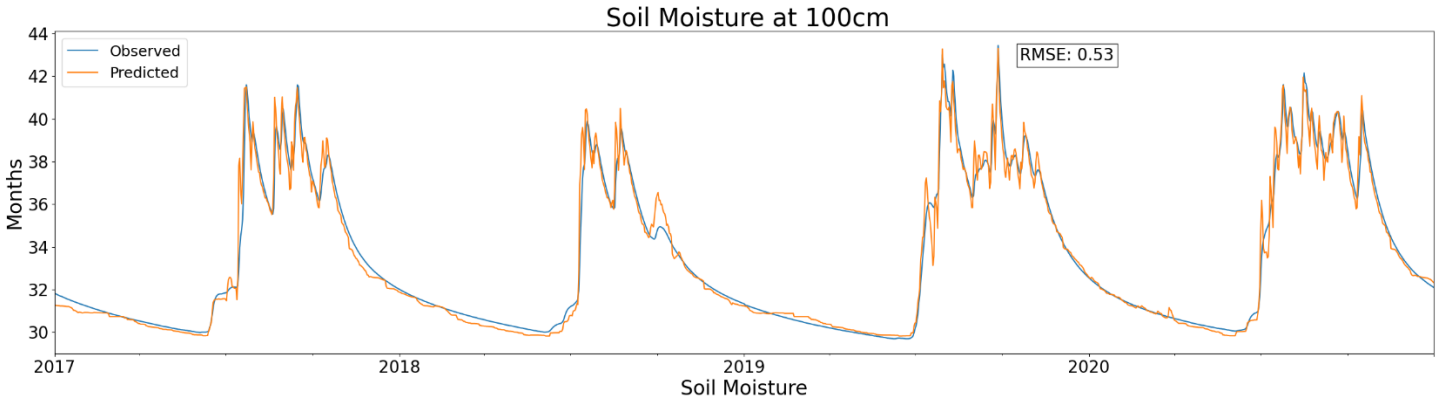


Fig.5.1.4 Observed vs Predicted Soil Moisture at 100cm using rf

5.2 Plot of Predicted soil moisture vs actual observed soil moisture using ANN

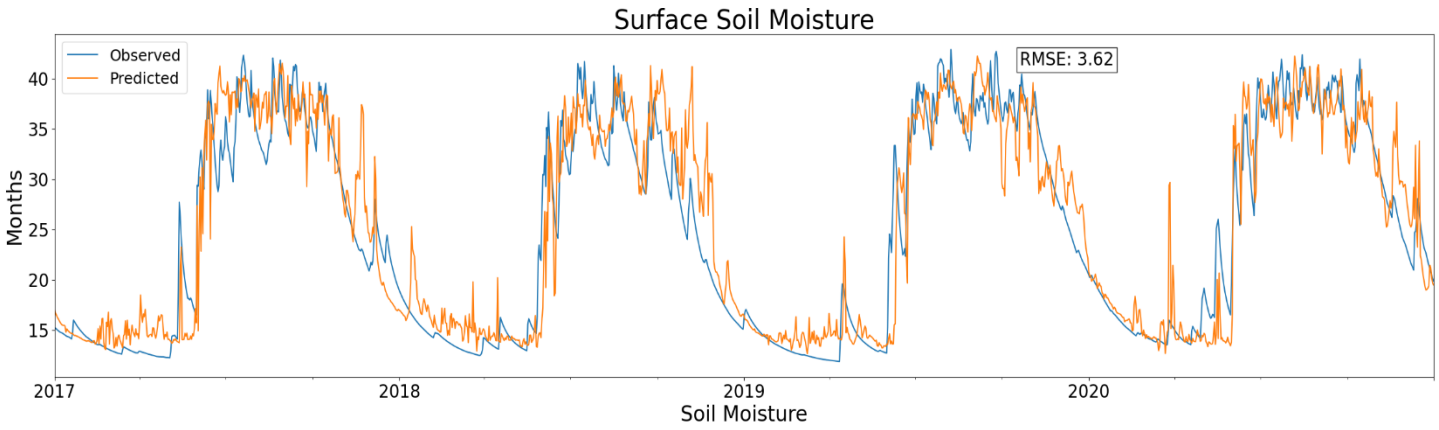


Fig.5.2.1 Observed vs Predicted Surface Soil Moisture using ann

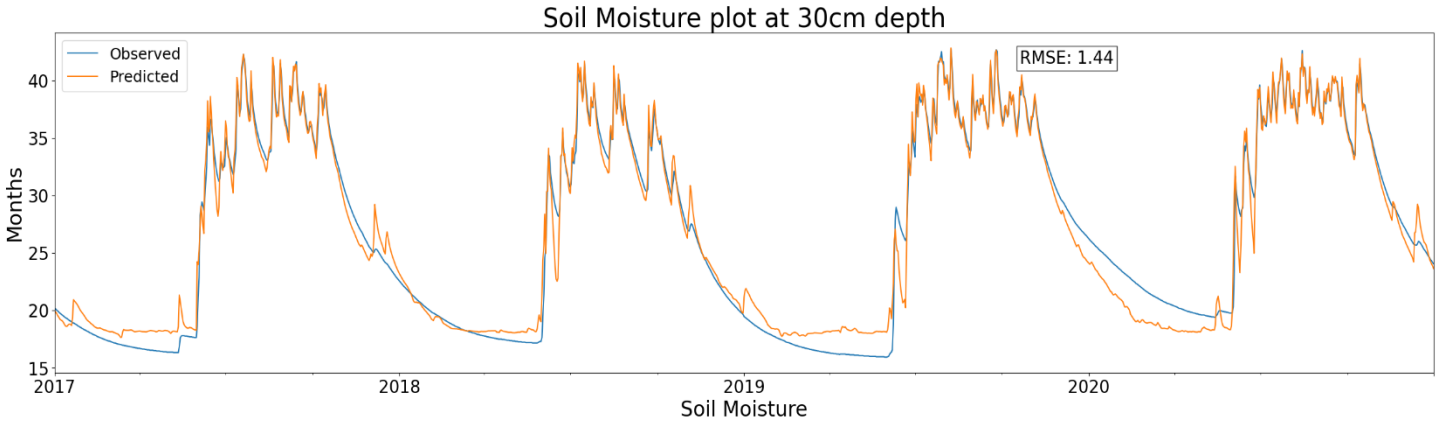


Fig..5.2.2 Observed vs Predicted Soil Moisture at 30cm using ann

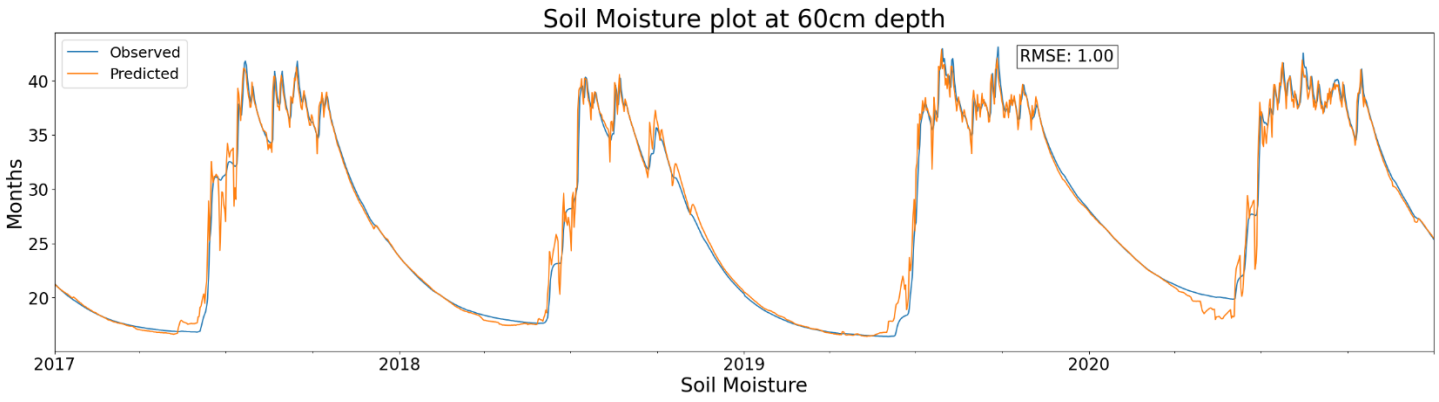


Fig..5.2.3 Observed vs Predicted Soil Moisture at 60cm using ann

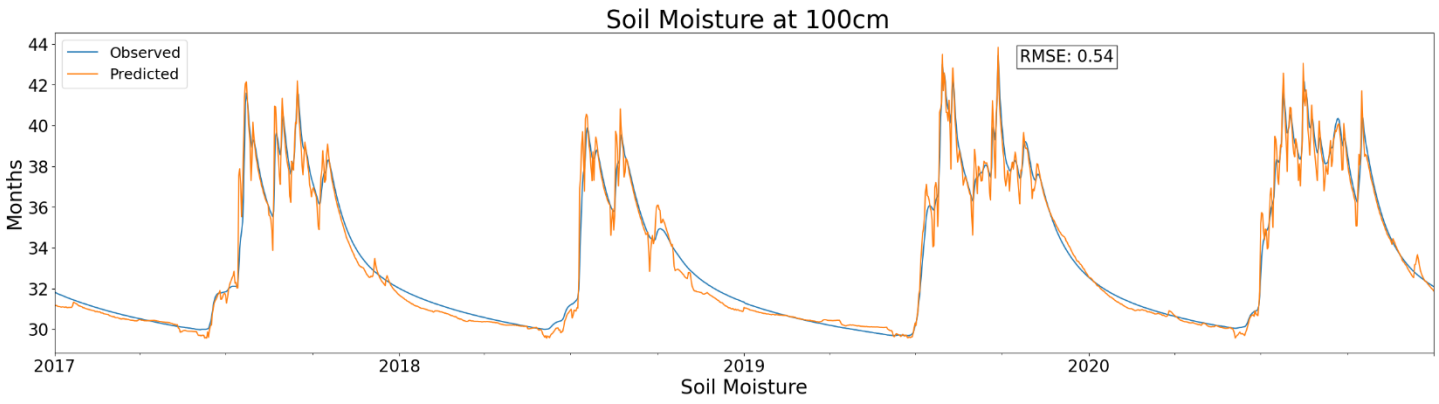


Fig..5.2.4 Observed vs Predicted Soil Moisture at 100cm using ann

The provided plot compares the observed (blue line) and predicted (orange line) surface soil moisture from 2017 to 2020. Additional plots (not shown) were generated for soil moisture at 30 cm, 60 cm, and 100 cm depths, with increasing accuracy at greater depths.

Key Observations

1. **Surface Soil Moisture (10 cm)**
    a. **Trend Agreement**: Both observed and predicted data show strong seasonality , with high moisture levels during the mid-year and lower levels at the start and end of each year.
    b. **Discrepancies**:
        i. Early years (2017-2018): More variation between observed and predicted values.
        ii. Later years (2019-2020): Improved alignment, though discrepancies remain.

2. **Deeper Soil Moisture Levels (30 cm, 60 cm, 100 cm)**
    a. **Accuracy Improvement**: Prediction accuracy increases with depth.
    b. **Trend Stability**: Moisture trends become more stable and consistent with observed data as depth increases.

3. **Prediction Variations in Soil Moisture Using Random Forest and ANN Models**

    a. During the initial 2 years of the testing period, there was a notable variation between the observed and predicted soil moisture values. This can be attributed to the models needing time to adjust to new data patterns, which might not have been well-represented in the training data.

    b. In contrast, the last 2 years of the testing period showed improved alignment between observed and predicted values. Despite the improvements, some discrepancies remained, likely due to the inherent variability in soil moisture and external factors not captured by the models

6. Outcome and Accuracy

- The Random Forest Regression Model, with a depth of 7, was employed to predict values at different soil depths: Surface, 30cm, 60cm, and 100cm. The accuracy scores obtained for each depth are as follows:

- Surface: 90.79%
- 30cm: 96.64%
- 60cm: 98.10%
- 100cm: 99.14%

These results indicate a progressively higher accuracy with increasing soil depth, showcasing the model's proficiency in predicting values deeper within the soil profile. Overall, the model demonstrates strong predictive capabilities, with accuracy scores consistently above 90% and peaking at 99.14% for the deepest depth.

- For the Artificial Neural Network (ANN) model, details on the architecture, hyperparameters, and training methodology would be necessary to provide a comprehensive report. However, assuming similar conditions and performance evaluation as the Random Forest Regression model, a brief report could be as follows:

The Artificial Neural Network (ANN) model was trained to predict values at various soil depths, including Surface, 30cm, 60cm, and 100cm. The accuracy scores achieved with the ANN model are as follows:

- Surface: 90.79%
- 30cm: 96.64%
- 60cm: 98.10%
- 100cm: 99.14%

These results indicate the model's ability to predict soil properties at different depths. A detailed analysis would involve assessing the model's architecture, training methodology, and potential areas for improvement. Comparisons with other models, such as Random Forest Regression, could also provide insights into the relative performance and suitability for the task at hand.

7.References

[1] Prakash, S., Sharma, A., & Sahu, S. S. (2018). Soil Moisture Prediction Using Machine Learning. 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT). doi:10.1109/icicct.2018.8473260

[2] Machine Learning to Estimate Surface Soil Moisture from Remote Sensing Data
Water **2020**, 12(11), 3223; ***https://doi.org/10.3390/w12113223***

[3] Machine learning for soil moisture assessment DOI:*10.1016/B978-0-323-85214-2.00001-X*

[4] Soil Moisture Prediction Using Machine Learning *DOI: 10.1109/ICICCT.2018.8473260 Published in: 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*

[5] Machine Learning for Soil Moisture Prediction Using Hyperspectral and Multispectral Data *DOI: 10.23919/FUSION49465.2021.9627067*

[6] Soil Moisture Prediction Using Machine Learning Techniques DOI:*10.1145/3440840.3440854*

[7] Soil Moisture Prediction Using Support Vector Machines DOI:*10.1111/j.1752-1688.2006.tb04512.x*

[8] Research on soil moisture prediction model based on deep learning
DOI: *10.1371/journal.pone.0214508*

[9] A comprehensive study of deep learning for soil moisture prediction Yanling Wang, Liangsheng Shi, Yaan Hu, Xiaolong Hu, Wenxiang Song, and Lijun Wang doi:10.5194/hess-28-917-2024