MACHINE LEARNING AND DEEP LEARNING APPROACH FOR SOIL MOISTURE ESTIMATION USING REMOTE SENSING DATA

Ankit Kumar, Sanjay ram, Mohan Agarwala, Himanshu Raj, Aman Rastogi

[a]*Indian Institute of Information Technology-Allahabad, Jhalwa, Prayagraj, 211001, Uttarpradesh, India*

Abstract

In the paper, we have applied several machine learning model for prediction of soil moisture collected via remote sensing methods. We have highlighted the importance of prediction of soil moisture data, the problem statement, the previous work done in this topic. We have given our methodology and compared the outcomes from different models.

Keywords: Soil moisture, Active- passing microwave remote sensing, Random Forest, LSTM.

1. Introduction

The measurement of soil moisture, defined as the amount of water present in the soil per unit mass of dry soil, encompasses various forms of water held in the soil. These include capillary, gravitational, and hygroscopic water, and are influenced by factors such as temperature, vegetation, soil type and land use, topography, and rainfall. Soil moisture is a critical factor that affects plant growth, nutrient uptake, and soil physical and chemical properties, and is also used to identify agricultural droughts, early stage water deficit conditions, and to assist with crop planning. Moreover, it plays a crucial role in informing erosion and assessing potential risks of landslides, sinkholes, and other geological hazards, supporting weather forecasting and flood control efforts. Soil moisture is typically measured

Figure 1: Soil Moisture (14)

in different layers of the soil, including the top layer or A horizon, which is influenced by rainfall, temperature, and vegetation cover, and can fluctuate rapidly. The subsurface soil layer or B horizon, on the other hand, typically has a lower soil moisture content and varies depending on soil type, structure, and depth. Finally, the deep groundwater layer or C horizon, which is the deepest layer of soil moisture, can be an important source of water for deep-rooted plants and can influence the water-holding capacity of the soil.

1.1. Layers Of Soil

Soil moisture is a critical component of the hydrological cycle and an essential factor for agricultural productivity, water management, and ecosystem health. It is the amount of water present in soil, and it varies depending on different factors. The top layer of soil, also known as the A horizon, is the most active and influenced by factors such as rainfall, temperature, and vegetation cover. The soil moisture in this layer can fluctuate rapidly, and it is critical for plant growth and the availability of water to the soil.

On the other hand, the subsurface soil layer, known as the B horizon, typically has a lower soil moisture content than the topsoil. This layer is more affected by soil type, structure, and depth, and the soil moisture varies accordingly. The subsurface soil layer is crucial for water storage, especially during dry periods, and it helps in regulating the water flow to the topsoil.

The deep groundwater layer, also called the C horizon, is the deepest layer of soil moisture. It can be an essential source of water for deep-rooted plants and plays a crucial role in groundwater recharge. The amount of soil moisture in this layer can affect the water-holding capacity of the soil, which is critical for maintaining soil health and fertility. Overall, understanding the soil moisture content in different layers of the soil profile is crucial for effective water management, agricultural productivity, and ecosystem health.

1.2. Remote Sensing Methods for Soil Moisture Detection:

Remote sensing technology has revolutionized the way we monitor and measure soil moisture content. Unlike traditional methods, which rely on invasive soil sampling and physical measurements, remote sensing techniques provide a non-invasive, cost-effective, and efficient way of monitoring soil moisture content over large areas. The ability to collect data

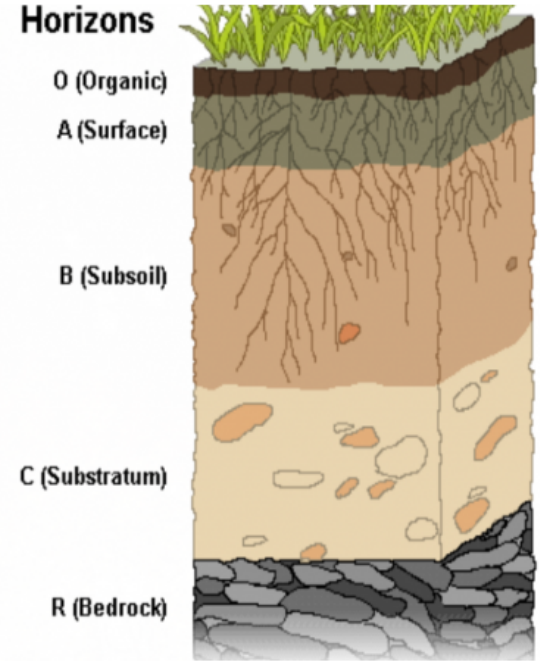


Figure 2: Layers of Soil
(1)

from different sensors and platforms, combined with sophisticated data processing techniques, allows for accurate and timely soil moisture monitoring. Additionally, remote sensing technology can provide valuable insights into soil moisture changes over time, helping researchers and practitioners make informed decisions related to water management, agriculture, and environmental conservation. Given the importance of soil moisture for various applications, remote sensing is becoming an increasingly popular tool for detecting and monitoring soil moisture content.

1.2.1. Optical remote sensing:

One of the methods for soil moisture detection is optical remote sensing. This technique uses sensors to measure visible and near-infrared light reflected by the Earth's surface, which can be used to estimate vegetation cover and, indirectly, soil moisture content. Although optical remote sensing can provide high spatial resolution measurements of soil moisture, its accuracy can be affected by atmospheric conditions, cloud cover, and the presence of shadows.

1.2.2. Land surface temperature:

Another indirect method for estimating soil moisture using remote sensing data is land surface temperature. This technique estimates soil moisture based on the relationship between land surface temperature and soil moisture. Land surface temperature can provide useful information on soil moisture changes over time, but it is sensitive to atmospheric conditions and other factors.

1.2.3. Vegetation indices:

Vegetation indices are also indirect techniques that use remote sensing data to estimate soil moisture based on the relationship between vegetation growth, land surface temperature, and soil moisture. While vegetation indices can provide useful information on soil moisture changes over time, they are sensitive to vegetation cover and may not accurately capture changes in bare soil moisture

1.2.4. Active microwave remote sensing:

Active microwave remote sensing is a technique that uses microwave radar sensors to emit a signal towards the ground and measure the reflected signal, which is sensitive to the soil moisture content. This method can provide high-resolution measurements of soil moisture, but its accuracy can be affected by vegetation cover and other surface features.

1.2.5. Passive microwave remote sensing:

Passive microwave remote sensing is another technique that uses microwave sensors to measure the thermal radiation emitted by the Earth's surface, including soil moisture. This method can provide frequent and accurate measurements of soil moisture, but its spatial resolution is relatively low.

1.2.6. Active-passive microwave remote sensing:

Active-passive microwave remote sensing is a powerful technique for measuring soil moisture content using both active and passive microwave sensors. Active sensors, such as the synthetic aperture radar (SAR), emit microwave pulses towards the ground and measure the time delay and strength of the returned signal, which is influenced by the dielectric properties of the soil. Passive sensors, such as the Advanced Microwave Scanning Radiometer (AMSR) on board the Aqua satellite, detect naturally emitted microwave radiation from the Earth's surface, which is also sensitive to soil moisture content. By combining the information obtained from active and passive sensors, it is possible to obtain accurate estimates of soil moisture content with high spatial resolution in the order of tens of meters, and with a high temporal resolution.

2. Literature review

Soil moisture is a crucial parameter in agricultural and environmental studies as it plays a significant role in plant growth and water balance. Accurate estimation of soil moisture is essential for various applications such as precision agriculture, water resource management, and flood prediction. Traditional methods for estimating soil moisture, such as gravimetric and TDR (Time Domain Reflectometry) techniques, are labor-intensive, time-consuming, and have limited spatial coverage. Therefore, remote sensing techniques have become popular for soil moisture estimation. In recent years, machine learning models have gained popularity in soil moisture prediction due to their ability to handle complex and nonlinear relationships between the input variables and the target variable.

Table 1: Comparison of Remote Sensing Methods

Methods	Properties	Advantages	Limitations
Optical	soil reflection	fine spatial resolution	broad coverage limited surface penetration cloud contamination many other noise sources
Thermal-Infrared	surface temperature	fine spatial resolution	broad coverage physical well understood limited surface penetration cloud contamination perturbed by meteorological conditions and vegetation
Active-microwave	brightness temperature, soil temperature	low atmospheric noise	moderate surface penetration physical well understood low spatial resolution perturbed by surface roughness and vegetation
Passive-microwave	backscatter coefficient, dielectric properties	low atmospheric noise	moderate surface penetration high spatial resolution limited swath width perturbed by surface roughness and vegetation

Support Vector Machine (SVM) regression is a machine learning model used for soil moisture prediction. SVM is a type of machine learning algorithm that uses a kernel function to map the input data into a higher-dimensional space, where it is easier to separate the data into different classes. (2), proposed a SVM model for soil moisture prediction. The authors used data from the Little River Experimental Watershed in Georgia, USA, and compared the performance of SVM with multiple linear regression and artificial neural networks. SVM outperformed the other two models in terms of accuracy. Another SVM-based model was proposed by (4), who used a deep learning approach to predict soil moisture. The authors used data from five weather stations in China and trained a deep neural network to predict soil moisture up to seven days in advance. The results showed that the model was highly accurate in predicting soil moisture.(6) proposed a machine learning-based approach for monitoring soil moisture content using unmanned aerial vehicles (UAVs) and hyperspectral imagery. The authors used a support vector regression model to predict soil moisture levels from hyperspectral data collected by UAVs. The results showed that the model was highly accurate in predicting soil moisture.(5) used SVM regression for predicting soil moisture in the Loess Plateau region of China. They found that SVM regression outperformed traditional regression models and was effective in handling the nonlinear relationship between soil moisture and other environmental variables.

One of the commonly used machine learning models for soil moisture prediction is the Random Forest Regression (RFR) model. RFR builds an ensemble of decision trees by randomly sampling the training data and input variables. Each decision tree is trained on a subset of the training data, and the final prediction is obtained by averaging the predictions of all decision trees. (15)) used RFR for predicting soil moisture in a study area in China. They found that RFR outperformed traditional regression models and showed promising results for soil moisture prediction.

(17) proposed a multilayer neural network model for soil moisture prediction. The authors used data from four sites in the Heihe River Basin in China and compared the performance of their model with multiple linear regression and SVM. The multilayer neural network outperformed the other two models in terms of accuracy. Song et al. (2016) proposed a deep learning-based cellular automata model for spatio-temporal distribution of soil moisture. The authors used data from four sites in China and trained a deep neural network to predict soil moisture levels at different spatial and temporal scales. The results showed that the model was highly accurate in predicting soil moisture.Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) that can handle long-term dependencies in sequential data. LSTM has been used for soil moisture prediction due to its ability to capture temporal dependencies in soil moisture data. For instance, (16) used LSTM for predicting soil moisture in the Yellow River Basin of China. They found that LSTM outperformed other machine learning models and showed promising results for soil moisture prediction.

(3) proposed a machine learning model to estimate surface soil moisture from remote sensing data. The authors used data from the Soil Moisture Active Passive (SMAP) satellite mission and trained a random forest model to predict soil moisture levels. The results showed that the model was highly accurate in estimating soil moisture. (7) proposed a framework for estimating all-weather fine resolution soil moisture from the integration of physics-based and machine learning-based algorithms. The authors used a combination of a land surface model and a random forest model to predict soil moisture levels. The results showed that the model was highly accurate in estimating soil moisture. (10)proposed a drought evaluation and forecast model based on soil moisture simulation. The authors used a combination of a hydrological model and a remote sensing model to predict soil moisture levels. The results showed that the model was highly accurate in predicting soil moisture.

3. PROBLEM STATEMENT

Develop a reliable and accurate method for estimating soil moisture content using Machine learning and Deep learning for Active-Passive Microwave remote sensing data.

Since accurate estimation of soil moisture is required for precision agriculture, water resource management and flood prediction. Traditional methods for measuring soil moisture are time consuming and expensive,and may not provide adequate spatial coverage.

Machine learning and deep learning methods have shown great potential in estimating soil moisture with the help of remote sensing data, which could provide a cost effective and time efficient solution.

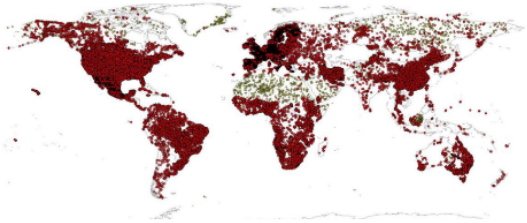


Figure 3:

4. Methodology

4.1. Dataset

4.1.1. SMAP Satelite

The SMAP project (11)(8)(12)(9)(13) of NASA uses a combination of active and passive sensors to collect data on soil moisture levels. The active sensor is a synthetic aperture radar (SAR), which transmits microwave pulses and measures the strength of the return signal. The passive sensor is a radiometer, which measures the natural microwave emissions from the Earth's surface.

The observatory is equipped with a 6-meter reflector antenna that rotates once every three seconds, allowing it to scan a swath of the Earth's surface 1000 km wide. The radar and radiometer work together to produce highly accurate soil moisture maps with a spatial resolution of about 10 km. To collect the data, the SMAP observatory orbits the Earth at an altitude of approximately 685 km, taking measurements over a 3-year period. The observatory covers the entire globe every 2-3 days, providing frequent and comprehensive soil moisture measurements.

The data collected by the SMAP observatory is transmitted to ground-based stations, where it is processed and analyzed using specialized algorithms to estimate the amount of soil moisture in the top 5 cm of the ground. The data is then validated using ground-based measurements and other data sources, such as climate models.

4.1.2. Google Earth Engine

GEE is a cloud computing platform that assesses, stores, and analyzes data from a variety of satellites, including Sentinel, Landsat, and MODIS. The collection includes climate, atmosphere, surface temperature, land cover, terrain, cropland, and other geophysical data that are openly and freely available. The web-based Interactive Development Environment (IDE) and internet-based Application Programming Interface (API) are available in Python and JavaScript. It helps the researchers to reduce the burden of storing a large number of big data files locally. It saves the data pre-processing and formatting time with the advantage of accessing earth observation data. Earth engine explorer lets users manage and visualize data from several satellites, while earth engine time-lapse lets them see the earth's evolution over 40 years. GEE can process large geospatial datasets with global coverage.

4.1.3. Power Access Climate Data

Power Access Climate Data is a sophisticated web platform that is built on state-of-the-art technologies and tools, designed to provide users with comprehensive access to a wide range of climate data and analysis tools. The platform is built on a scalable and robust data processing and analysis framework, which integrates data from multiple sources, including ground-based sensors, satellite imagery, and climate models, among others.

One of the key features of the platform is its extensive range of climate models. These models are available in various resolutions, outputs, and time scales, and can be used to simulate and predict climate patterns and trends. Users can access these models in different formats, such as NetCDF, CSV, and JSON, making it easy for researchers and practitioners to perform their own analysis and modeling.

In addition to climate models, the platform also provides users with access to historical climate data spanning several decades. The data is available in various formats and can be downloaded by users for further analysis. Real-time weather data is also available on the platform, including temperature, humidity, pressure, wind speed and direction, and precipitation, from thousands of weather stations around the world.

The platform's visualization tools are designed to help users explore and analyze the data more easily. The interactive maps allow users to visualize the data spatially, while time-series charts and scatter plots enable users to compare and contrast different variables. Users can also overlay multiple variables on the same chart or map to perform more in-depth analysis.

The platform's analysis and forecasting tools use advanced statistical and machine learning algorithms to identify patterns and trends in the data, and forecast future climate scenarios. These tools are particularly useful for researchers, policymakers, and businesses that need to make informed decisions based on climate data.

4.1.4. Dataset Preparation

We generated a dataset on soil moisture levels using a combination of data sources and processing tools. We collected data on soil moisture levels from the SMAP project of NASA. Using Google Earth Engine, we processed the SMAP data sets

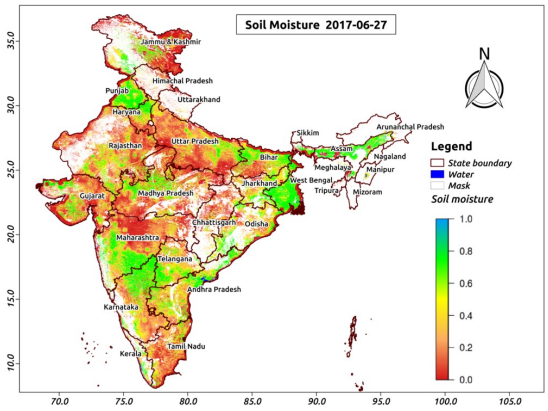


Figure 4:

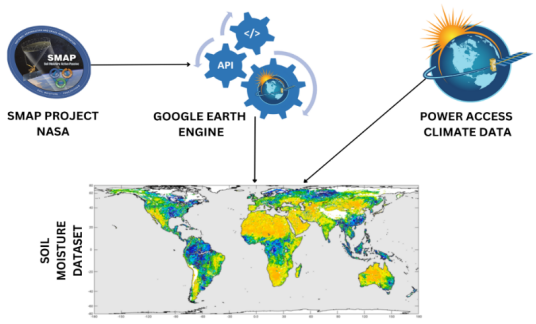


Figure 5: Dataset Preparation

and generated a dataset on soil moisture levels. To enhance the dataset, we also included additional parameters from Power Access Climate Data, such as temperature and precipitation data, to provide a more complete picture of the conditions affecting soil moisture levels. The resulting dataset provides a comprehensive picture of soil moisture levels on a global scale.

This dataset includes parameters like : surface and subsurface soil moisture (mm), soil moisture profile , surface and subsurface soil moisture anomalies (-) collected from the SMAP satelite. Surface soil moisture levels from:

- 20-25-mm are best for germinating and emerging a new crop, but can halt fieldwork and could damage newly-seeded crops that remain in the wet environment for an extended period of time.
- 15-20-mm of water are normally best for vigorous field activity.
- 10-mm or less will not support seed germination or early growth potentials for a recently emerged crop.

Subsurface soil moisture values of:

- > 100-mm indicates an abundance or at least favorable amount of moisture in the subsoil.
- > 100-mm indicates the sub-surface soil moisture storage is short but can still support a well-established crop.
- < 25-mm has very little sub-surface soil moisture and the crop could be severely stressed and reduce yields, especially if it occurs when the top-layer has little or no significant soil moisture and the crop is at a critical stage of growth.

And the parameters from Power Access climate data were chosen which directly impact the soil moisture content i.e., Wind direction(degree), Wind Speed(m/s), Surface Pressure(Kpa), Due point(c), Temperature At 2meters height(cel),Earth temp(cel), precipitation (mm per day).

For diversification of our project, we included five cities from different geographical locations in India. The cities are: Purulia, Ludhiana, Dharampuri, Jodhpur, Kandhawal.

4.2. Machine Learning Models

4.2.1. Linear Regression

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_p x_p + \epsilon \quad (1)$$

where:

y is the dependent variable (soil moisture content) $x_1, x_2, ..., x_p$ are the independent variables (wind speed, wind direction, pressure, temperature, etc.) β_0 is the y-intercept (the value of y when all independent variables are equal to zero) $\beta_1, \beta_2, ..., \beta_p$ are the coefficients (the values that represent the change in y for a one-unit change in each independent variable) ϵ is the error term (the difference between the predicted value and the actual value of the dependent variable) To train the linear regression model, we need to estimate the values of the coefficients β that minimize the sum of the squared errors between the predicted values and the actual values of the dependent variable. This can be done using the least squares method:

$$\beta = (X_{train}^T X_{train})^{-1} X_{train}^T y_{train} \quad (2)$$

where X_{train}^T is the transpose of the matrix of input features X_{train}. To make predictions on new data, we can multiply the matrix of input features X_{test} by the vector of coefficients β:

$$\hat{y} = X_{test}\beta \quad (3)$$

where $\hat{y}$ is the vector of predicted values.

4.2.2. Support Vector Regression

To train an SVM model for regression, we first need to choose a kernel function that will map the input features into a higher-dimensional space. A popular choice for SVM regression is the radial basis function (RBF) kernel:

$$K(x_i, x_j) = \exp(-\gamma ||x_i - x_j||^2) \quad (4)$$

where x_i and x_j are the input feature vectors for two data points, and γ is a hyperparameter that controls the width of the kernel function.

Once we have chosen the kernel function, we can train the SVM model to find the hyperplane that maximizes the margin

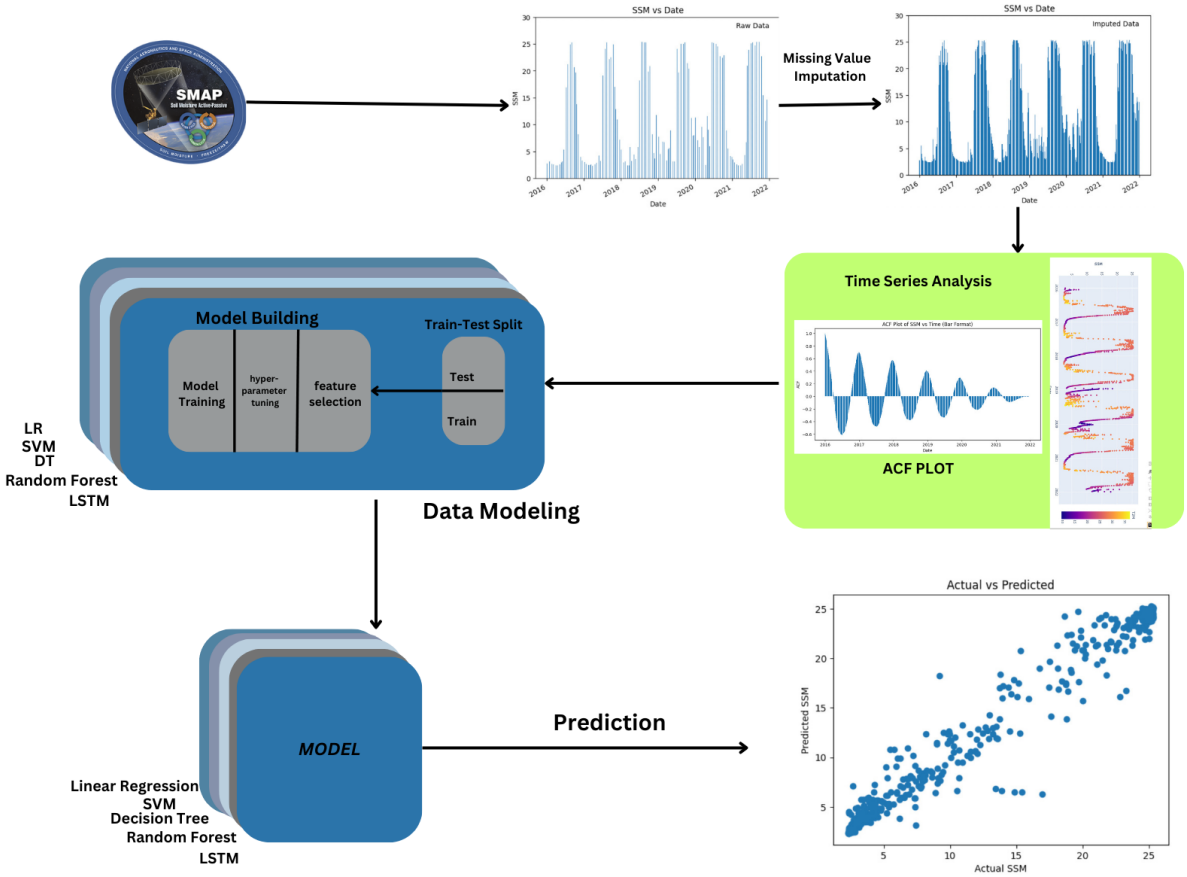


Figure 6: Proposed Methodology

between the support vectors and the decision boundary. The decision function for an SVM regression model can be expressed as:

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)K(x, x_i) + b \tag{5}$$

where x is the input feature vector for a new data point, n is the number of training examples, α_i and α_i^* are the Lagrange multipliers for the ith training example and its corresponding slack variable, $K(x, x_i)$ is the kernel function evaluated at x and x_i, and b is a bias term.

To train the SVM model, we need to solve the optimization problem:

$$\min_{\alpha,\alpha \cdot b} \frac{1}{2}(\alpha - \alpha')^T K(\alpha - \alpha') + C\sum_{i=1}^{n}(\xi_i + \xi_i') \tag{6}$$

subject to the constraints:

$\sum_{i=1}^{n}(\alpha_i - \alpha_i') = 0\ 0 \le \alpha_i, \alpha_i^{\le} C\ y_i - f(x_i) \le \epsilon + \xi_i\ f(x_i) - y_i \le \epsilon + \xi_i^*$

where C is a hyperparameter that controls the trade-off between maximizing the margin and minimizing the training error, ξ_i and ξ_i^* are slack variables that allow for points to be on the wrong side of the decision boundary, and ϵ is a parameter that controls the width of the margin.

To make predictions on new data, we can simply evaluate the decision function $f(x)$ for each data point. The predicted values can be obtained as:

$$\hat{y} = f(X_{test}) \tag{7}$$

where X_{test} is the matrix of input features for the testing data.

4.2.3. Decision Tree

Given a training set $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$, where $\mathbf{x}_i$ is a vector of D features and y_i is the corresponding target variable, the decision tree algorithm recursively partitions the feature space into a set of rectangles R_j such that the target variable is nearly constant in each rectangle. The algorithm works as follows:

Define a recursive function T that takes as input a dataset D and a set of candidate splitting functions $\mathcal{F}$. The function T re-

turns a decision tree that minimizes a certain splitting criterion, such as the Gini impurity or information gain.

At each step of the recursion, T selects the best splitting function f^* from $\mathcal{F}$ by minimizing the impurity or maximizing the information gain of the resulting partitions.T creates a new node in the tree and partitions the data into subsets $D_1, \ldots, D_K$ according to the values of $f^*(\mathbf{x})$. The subsets are then recursively passed to T to create child nodes.

The recursion terminates when either a maximum depth is reached, the impurity or information gain falls below a certain threshold, or the number of samples in a node falls below a certain threshold.To make a prediction for a new input $\mathbf{x}$, the decision tree traverses the tree from the root node to a leaf node, following the path determined by the splitting functions.The prediction is the average of the target variable of the training examples that fall into the leaf node.

4.2.4. Random Forest

Given a training set $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$, where $\mathbf{x}_i$ is a vector of D features and y_i is the corresponding target variable, the random forest regression algorithm creates an ensemble of decision trees to make predictions. The algorithm works as follows:

Define a hyperparameter B, the number of trees in the forest.

For each tree $b = 1, \ldots, B$:

a. Draw a bootstrap sample of size N from the training set, with replacement.

b. Randomly select a subset of m features, where $m << D$.

c. Train a decision tree on the bootstrap sample using the selected features. The tree is constructed by recursively partitioning the feature space into a set of rectangles, as in the decision tree algorithm.

To make a prediction for a new input $\mathbf{x}$, the random forest regression algorithm averages the predictions of all trees in the forest.

The random forest regression algorithm is an extension of the decision tree algorithm that reduces the tendency of decision trees to overfit the training data.

4.2.5. LSTM

Let $X \in \mathbb{R}^{N \times T \times D}$ be the input dataset with N samples, T time steps, and D features. The features are wind speed, wind direction, pressure, temperature, and time. Let $Y \in \mathbb{R}^{N \times 1}$ be the output variable, which is the soil moisture content.

We first preprocess the dataset by scaling the data to have zero mean and unit variance and splitting it into training, validation, and test sets.

The LSTM model is defined as follows. Let $h_t \in \mathbb{R}^{N \times H}$ be the hidden state of the LSTM at time t, where H is the number of hidden units. The hidden state is computed using the LSTM function:

$h_t = LSTM(xt, ht - 1)$

where $LSTM$ is the LSTM function.

We then apply a linear transformation to the hidden state to obtain the output:

$y_t = Wh_t + b$

where $W \in \mathbb{R}^{1 \times H}$ is the weight matrix, and $b \in \mathbb{R}^1$ is the bias vector.

To train the LSTM model, we use backpropagation through time (BPTT) to compute the gradients of the loss with respect to the weights and biases. The gradients are then used to update the weights and biases using stochastic gradient descent (SGD) or a variant thereof.

During training, we also use dropout regularization to prevent overfitting. Once the LSTM model is trained, we can use it to make predictions on new input data.

4.3. Evaluation Metrics

In machine learning, loss functions play a critical role in training a model to make accurate predictions. A loss function, also known as a cost function, measures the error between the predicted and actual values of the target variable. The goal of training a machine learning model is to minimize this error by adjusting the model's parameters through an optimization process.

4.3.1. MSE

The Mean Squared Error (MSE) is a commonly used loss function in regression problems. It measures the average of the squared differences between the predicted and actual values. The mathematical implementation of MSE is as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 \tag{8}$$

where n is the number of observations, Y_i is the actual value of the ith observation, and $\hat{Y}_i$ is the predicted value of the ith observation. The goal of training a regression model is to minimize the MSE, which means the predicted values should be as close as possible to the actual values. In other words, the lower the MSE, the better the model performs.

4.3.2. RMSE

The Root Mean Squared Error (RMSE) is a commonly used evaluation metric for regression problems. It is similar to MSE, but takes the square root of the average of the squared differences between the predicted and actual values, which provides a more interpretable metric in the same units as the target variable. The mathematical implementation of RMSE is as follows:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2} \tag{9}$$

where n is the number of observations, Y_i is the actual value of the ith observation, and $\hat{Y}_i$ is the predicted value of the ith observation. The goal of training a regression model is to minimize the RMSE, which means the predicted values should be as close as possible to the actual values. In other words, the lower the RMSE, the better the model performs.

4.3.3. MAE

The Mean Absolute Error (MAE) is another commonly used loss function for regression problems. It measures the average of the absolute differences between the predicted and actual values. The mathematical implementation of MAE is as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y}_i| \tag{10}$$

where n is the number of observations, Y_i is the actual value of the ith observation, and $\hat{Y}_i$ is the predicted value of the ith observation. The goal of training a regression model is to minimize the MAE, which means the predicted values should be as close as possible to the actual values. In other words, the lower the MAE, the better the model performs. Unlike MSE, MAE is not sensitive to outliers as it takes absolute differences.

4.3.4. MAPE

The Mean Absolute Percentage Error (MAPE) is a commonly used evaluation metric for regression problems. It measures the percentage difference between the predicted and actual values. The mathematical implementation of MAPE is as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Y_i - \hat{Y}_i}{Y_i}\right| \times 100 \tag{11}$$

where n is the number of observations, Y_i is the actual value of the ith observation, and $\hat{Y}_i$ is the predicted value of the ith observation. The goal of training a regression model is to minimize the MAPE, which means the predicted values should be as close as possible to the actual values. In other words, the lower the MAPE, the better the model performs. MAPE is a useful metric as it provides a relative measure of the prediction error, making it easier to compare across different target variables with different scales. However, MAPE can be problematic when the actual value is zero, resulting in an undefined or infinite value.

5. Results

We split the dataset into training and testing dataset and implemented all the above discussed machine learning models; and evaluated their performance on the basis of the loss functions discussed. Two tables were drawn that shows the loss function values for all the regression models for all the different datasets.

For better visualisation, the box-plots for all the loss functions for all the models were drawn for ludhiana.

5.1. Ludhiana:

Linear Regression (LR) has the lowest MAE and MAPE, indicating that it is performing the best among all models in terms of accuracy and precision. Support Vector Machine (SVM) has the highest MSE and RMSE, indicating that it is performing the worst among all models in terms of overall error and deviation from the actual values. Decision Tree (DT) and Random Forest

Table 2: Loss Function For SSM Prediction

City	Models	MSE	RMSE	MAE	MAPE
Ludhiana	LR	0.59	0.77	0.51	7.52%
	SVM	0.26	0.51	0.25	4.24%
	DT	1.02	1.01	0.75	7.17%
	RF	0.69	0.83	0.46	5.57%
	LSTM	0.06	0.24	0.16	2.80%
Kandhamal	LR	0.53	0.73	0.53	6.69%
	SVM	0.27	0.52	0.25	2.98%
	DT	0.94	0.97	0.48	5.05%
	RF	0.47	0.68	0.36	3.75%
	LSTM	0.08	0.28	0.18	2.00%
Jodhpur	LR	0.20	0.45	0.29	6.57%
	SVM	0.20	0.44	0.18	3.93%
	DT	0.56	0.75	0.31	5.00%
	RF	0.26	0.51	0.23	3.88%
	LSTM	0.03	0.17	0.11	2.56%
Purulia	LR	0.37	0.61	0.44	6.27%
	SVM	0.18	0.42	0.22	3.06%
	DT	0.78	0.89	0.49	5.17%
	RF	0.46	0.68	0.37	3.85%
	LSTM	0.06	0.24	0.17	2.28%
Dharmapuri	LR	0.76	0.87	0.59	9.31%
	SVM	0.26	0.51	0.24	3.80%
	DT	1.78	1.33	0.80	8.43%
	RF	0.96	0.98	0.57	5.95%

Table 3: Loss Function For SUSM

City	Models	MSE	RMSE	MAE	MAPE
Ludhiana	LR	0.80	0.89	0.62	2.20%
	SVM	2.50	1.58	0.63	2.51%
	DT	2.85	1.69	0.98	2.92%
	RF	1.64	1.28	0.75	2.26%
	LSTM	0.18	0.42	0.29	1.08%
Kandhamal	LR	1.46	1.21	0.87	2.36%
	SVM	10.51	3.24	1.12	2.87%
	DT	4.04	2.01	1.36	2.66%
	RF	2.12	1.46	0.97	1.98%
	LSTM	0.59	0.77	0.61	1.68%
Jodhpur	LR	0.47	0.69	0.47	3.32%
	SVM	2.35	1.53	0.47	2.58%
	DT	1.06	1.03	0.49	2.64%
	RF	0.50	0.71	0.35	1.95%
	LSTM	0.14	0.38	0.26	1.86%
Purulia	LR	0.97	0.98	0.72	2.41%
	SVM	2.12	1.46	0.67	1.97%
	DT	2.02	1.42	0.87	2.41%
	RF	1.09	1.05	0.63	1.79%
	LSTM	0.27	0.52	0.40	1.24%
Dharmapuri	LR	1.46	1.21	0.87	2.36 %
	SVM	10.51	3.24	1.12	2.87 %
	DT	4.63	2.15	1.43	2.15%
	RF	2.12	1.46	0.97	1.98 %
	LSTM	0.51	0.71	0.57	1.43 %

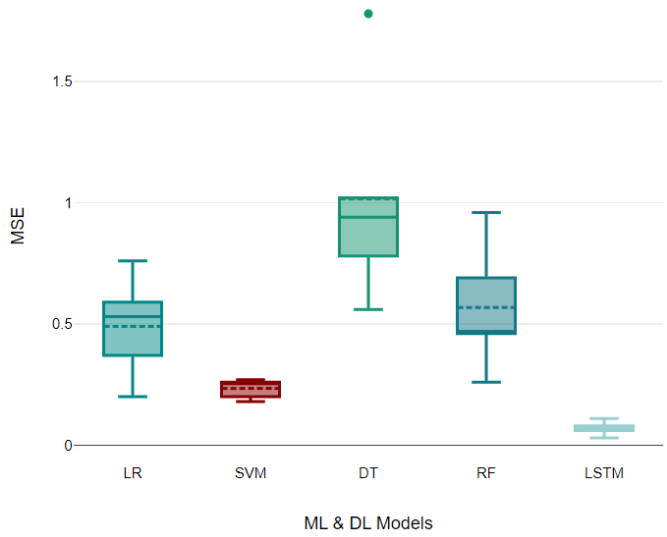


Figure 7: MSE for SSM

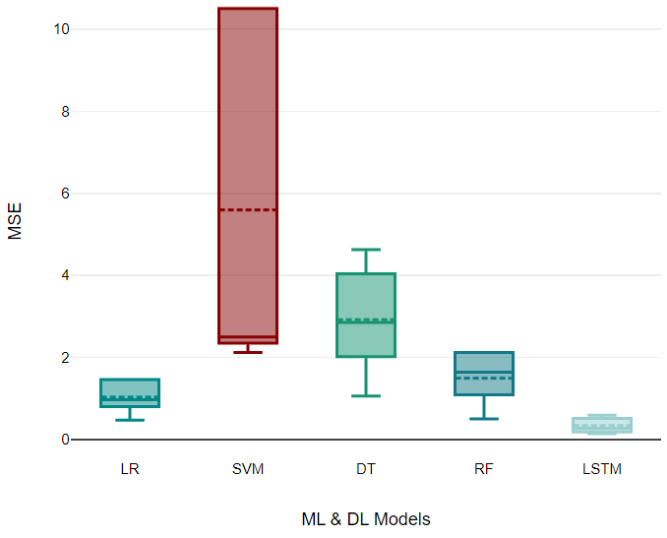


Figure 8: MSE for SUSM

(RF) have comparable performance in terms of all the loss functions, but RF has a slightly better performance in terms of MSE and MAPE. Long Short-Term Memory (LSTM) has the lowest MSE, RMSE, and MAPE, indicating that it is performing the best among all models in terms of minimizing the overall error and deviation from the actual values.

5.2. Kandhamal:

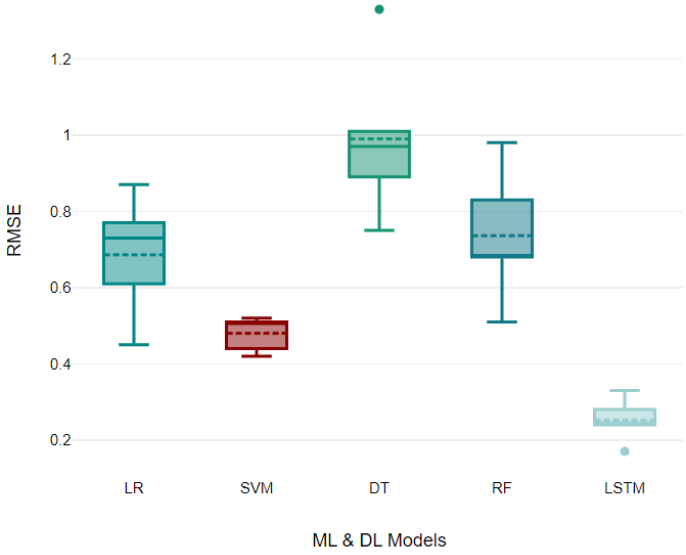
LR has the lowest MAE and MAPE, indicating that it is performing the best among all models in terms of accuracy and precision. SVM has the highest MSE and RMSE, indicating that it is performing the worst among all models in terms of overall error and deviation from the actual values. DT has the highest MAE and MAPE, indicating that it is performing the worst among all models in terms of accuracy and precision. RF has the lowest MSE, RMSE, MAE, and MAPE, indicating that it is performing the best among all models in terms of minimizing the overall error and deviation from the actual values. LSTM has a good performance, with the lowest MSE, RMSE, and MAPE among all models except RF, indicating that it is performing well in terms of minimizing the overall error and deviation from the actual values.

5.3. Jodhpur:

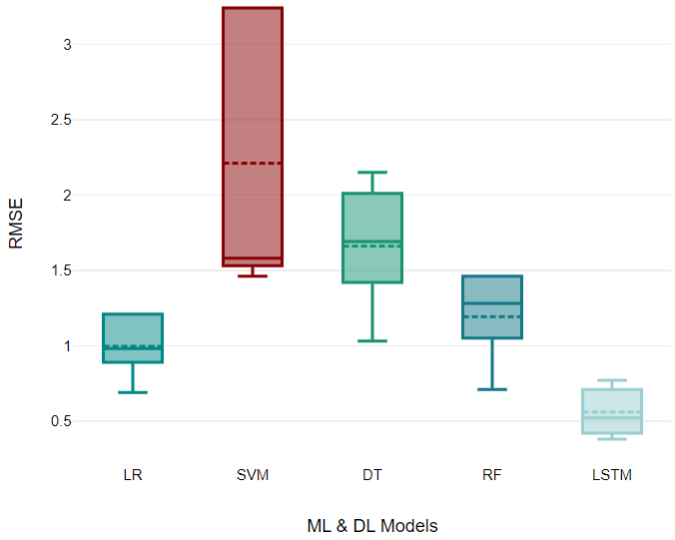
LR has the lowest MAE and MAPE, indicating that it is performing the best among all models in terms of accuracy and precision. SVM has the highest MSE and RMSE, indicating that it is performing the worst among all models in terms of overall error and deviation from the actual values. DT and RF have comparable performance in terms of all the loss functions, but RF has a slightly better performance in terms of MSE and MAPE. LSTM has the lowest MSE, RMSE, and MAPE, indicating that it is performing the best among all models in terms of minimizing the overall error and deviation from the actual values.

5.4. Purulia:

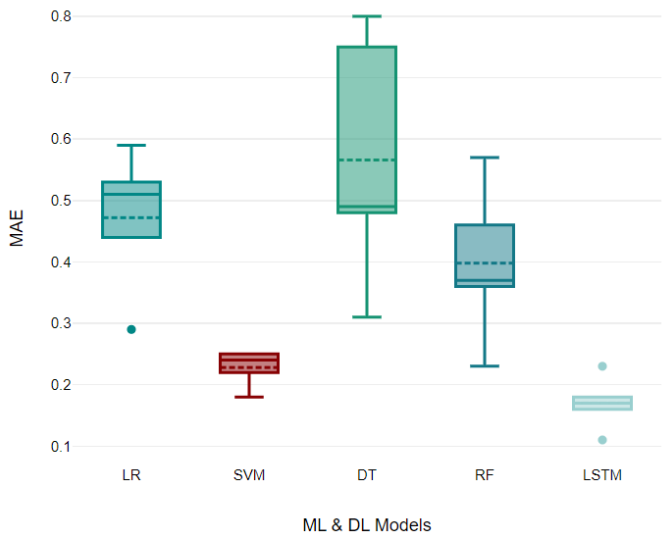
LR has the lowest MAE and MAPE, indicating that it is performing the best among all models in terms of accuracy and precision. SVM has the highest MSE and RMSE, indicating that it is not performing well in terms of accuracy and precision compared to other models. However, it has a relatively low MAE and MAPE, indicating that it is able to make more accurate predictions for smaller values. DT and RF have similar performances, with RF performing slightly better in terms of MSE and RMSE, while DT has a lower MAE and MAPE. This suggests that RF is better at making predictions for larger values, while DT is better for smaller values. LSTM has the lowest MSE, RMSE, MAE, and MAPE, indicating that it is performing the best among all models in terms of accuracy and precision. This is expected as LSTM is a type of deep learning model that can capture complex relationships between variables and is known to perform well in time-series data analysis.



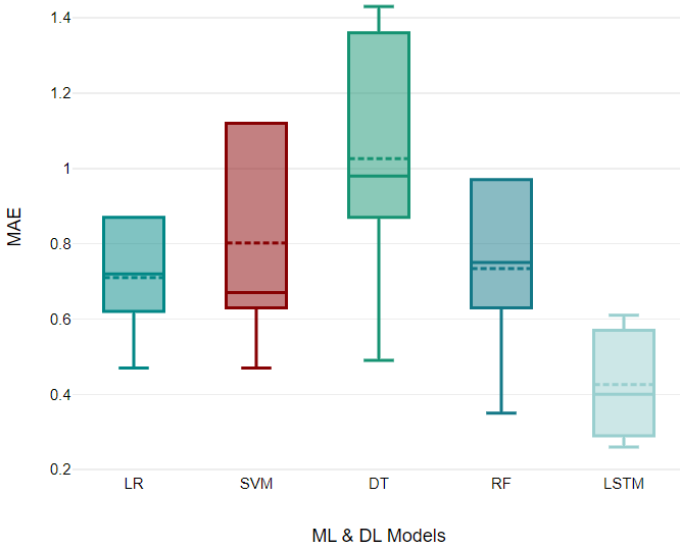
RMSE for SSM



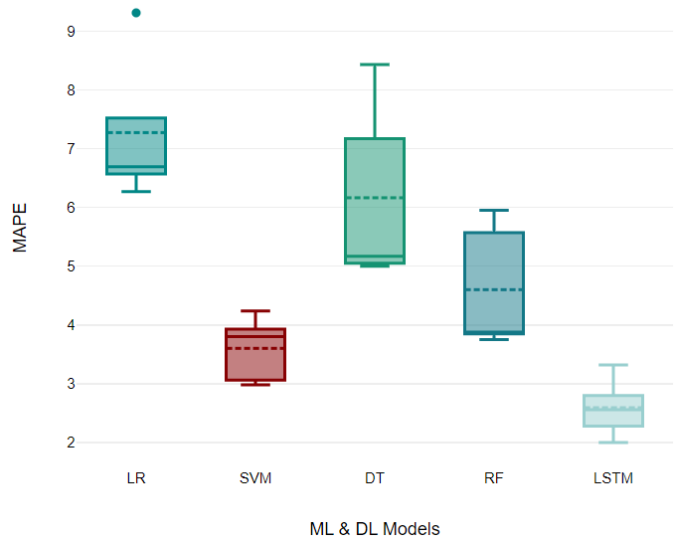
RMSE for SUSM



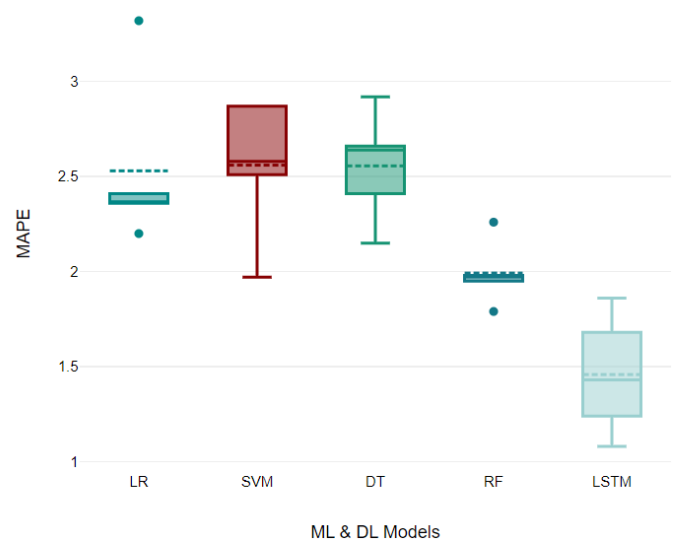
MAE for SSM



MAE for SUSM



MAPE for SSM



MAPE for SUSM

Figure 9: Overall caption for the six images

5.5. *Dharmapuri*

Moving on to the results for Dharmapuri, we see a similar trend as Purulia. LR again performs the best among all models in terms of accuracy and precision, with the lowest MAE and MAPE. SVM performs the worst with the highest MSE and RMSE, but has a relatively low MAE and MAPE. DT and RF have similar performances, with RF performing slightly better in terms of MSE and RMSE, while DT has a lower MAE and MAPE. This suggests that RF is better at making predictions for larger values, while DT is better for smaller values. LSTM again has the lowest MSE, RMSE, MAE, and MAPE, indicating that it is performing the best among all models in terms of accuracy and precision.

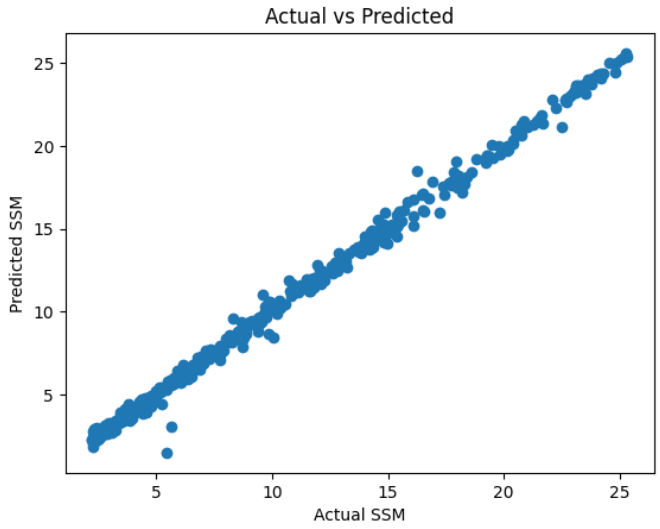


Figure 10: LSTM comparison for SSM

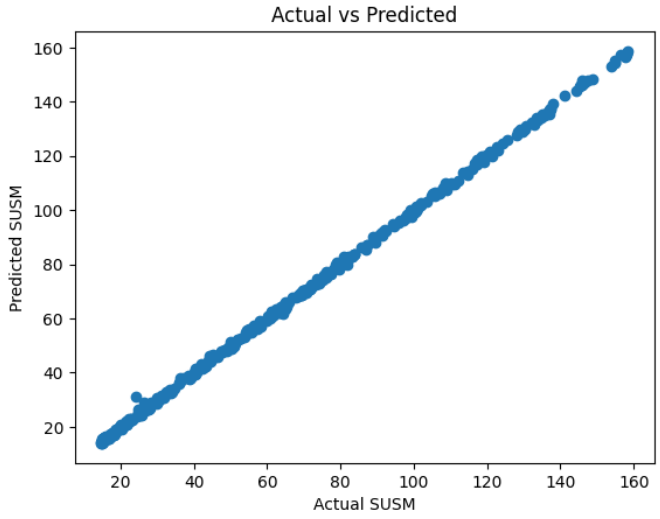


Figure 11: LSTM comparison for SUSM

Additionally, graphs were plotted for the actual vs predicted values by LSTM model for both SSM and SUSM. These are a common way of evaluating the performance of a machine learning model, here equation of the line is y = x, which represents the ideal scenario where the predicted values are equal to the actual values.

6. Summary and conclusions

In conclusion, the combination of active and passive microwave remote sensing has proven to be an effective method for detecting soil moisture. The integration of machine learning algorithms has improved the accuracy of soil moisture prediction. Among all the models, random forest has performed the best, demonstrating its potential for soil moisture prediction. These findings highlight the importance of remote sensing and machine learning techniques for soil moisture detection, which can aid in better management of water resources and improve agricultural productivity. Further research in this area could lead to the development of more accurate and efficient methods for soil moisture detection and prediction. We can use the SMAP dataset to predict soil moisture in areas where much research has not been done and boost agricultural output in remote areas and mountain terrains.

References

[1] Figure 1: Collection areas of iaedes albopictus/i strains from major rice growing areas of punjab, pakistan (wikimedia commons: https://commons.wikimedia.org/wiki/file:soil$_h$*orizons.svg*).

[2] U. Acharya, A. L. M. Daigh, and P. G. Oduor. Machine learning for predicting field soil moisture using soil, crop, and nearby weather station data in the red river valley of the north. *Soil Systems*, 5(4):57, sep 2021.

[3] H. Adab, R. Morbidelli, C. Saltalippi, M. Moradian, and G. A. F. Ghalhari. Machine learning to estimate surface soil moisture from remote sensing data. *Water*, 12(11):3223, 2020.

[4] Y. Cai, W. Zheng, X. Zhang, L. Zhangzhong, and X. Xue. Research on soil moisture prediction model based on deep learning. *PloS one*, 14(4):e0214508, 2019.

[5] Y. Feng, W. Hao, H. Li, N. Cui, D. Gong, and L. Gao. Machine learning models to quantify and map daily global solar radiation and photovoltaic power. *Renewable and Sustainable Energy Reviews*, 118:109393, 2020.

[6] X. Ge, J. Wang, J. Ding, X. Cao, Z. Zhang, J. Liu, and X. Li. Combining uav-based hyperspectral imagery and machine learning algorithms for soil moisture content monitoring. *PeerJ*, 7:e6926, 2019.

[7] P. Leng, Z. Yang, Q.-Y. Yan, G.-F. Shang, X. Zhang, X.-J. Han, and Z.-L. Li. A framework for estimating all-weather fine resolution soil moisture from the integration of physics-based and machine learning-based algorithms. *Computers and Electronics in Agriculture*, 206:107673, 2023.

[8] I. E. Mladenova, J. D. Bolten, W. Crow, N. Sazib, and C. Reynolds. Agricultural drought monitoring via the assimilation of smap soil moisture retrievals into a global soil water balance model. *Frontiers in Big Data*, 3, 2020.

[9] I. E. Mladenova, J. D. Bolten, W. T. Crow, N. Sazib, M. H. Cosh, C. J. Tucker, and C. Reynolds. Evaluating the operational application of smap for global agricultural drought monitoring. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(9):3387–3397, 2019.

[10] M. A. Rajib, V. Merwade, and Z. Yu. Multi-objective calibration of a hydrologic model using spatially distributed remotely sensed/in-situ soil moisture. *Journal of hydrology*, 536:192–207, 2016.

[11] N. Sazib, J. D. Bolten, and I. E. Mladenova. Leveraging nasa soil moisture active passive for assessing fire susceptibility and potential impacts over australia and california. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:779–787, 2022.

[12] N. Sazib, J. D. Bolten, and I. E. Mladenova. Leveraging nasa soil moisture active passive for assessing fire susceptibility and potential impacts over australia and california. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:779–787, 2022.

[13] N. Sazib, l. E. Mladenova, and J. D. Bolten. Assessing the impact of enso on agriculture over africa using earth observation data. *Frontiers in Sustainable Food Systems*, 4, 2020.

[14] S. F. Sweere, I. Valtchanov, M. Lieu, A. Vojtekova, E. Verdugo, M. Santos-Lleo, F. Pacaud, A. Briassouli, and D. Cámpora Pérez. Deep learning-based super-resolution and de-noising for xmm-newton images. *Monthly Notices of the Royal Astronomical Society*, 517(3):4054–4069, 2022.

[15] W. Wu, C. Zucca, A. S. Muhaimeed, W. M. Al-Shafie, A. M. Fadhil Al-Quraishi, V. Nangia, M. Zhu, and G. Liu. Soil salinity prediction and mapping by machine learning regression in c entral m esopotamia, i raq. *Land degradation & development*, 29(11):4005–4014, 2018.

[16] Q. Yuan, H. Shen, T. Li, Z. Li, S. Li, Y. Jiang, H. Xu, W. Tan, Q. Yang, J. Wang, et al. Deep learning in environmental remote sensing: Achievements and challenges. *Remote Sensing of Environment*, 241:111716, 2020.

[17] N. Zhu, X. Liu, Z. Liu, K. Hu, Y. Wang, J. Tan, M. Huang, Q. Zhu, X. Ji, Y. Jiang, et al. Deep learning for smart agriculture: Concepts, tools, applications, and opportunities. *International Journal of Agricultural and Biological Engineering*, 11(4):32–44, 2018.