Real-Time Soil Moisture Prediction for Precision Farming

1. Introduction

- Efficient water management is a cornerstone of modern precision farming, enabling farmers to optimize resource utilization and enhance crop yields. Traditional methods of soil moisture assessment often involve extensive sensor deployment, which can be cost-prohibitive and logistically challenging, particularly for large agricultural fields. In response to these challenges, our project leverages machine learning to predict soil moisture levels in unsensed areas using data collected from a limited number of sensors.
- The innovative approach adopted in this project significantly reduces the dependency on widespread sensor placement, making precision farming more affordable and scalable. By integrating machine learning models, we aim to provide accurate and real-time predictions, empowering farmers to make informed irrigation decisions. This report summarizes our observations, findings, and contributions to advancing soil moisture prediction technology.

2.Key Objectives:

1. Predict soil moisture in unsensed areas using limited sensors.
2. Enhance irrigation planning via accurate, real-time estimates.

3. Project Setup

To implement this project, specific steps were undertaken to ensure robust data collection and preparation:

1. **Marked Coordinates:**
    - Central and peripheral points were identified for sensor deployment.
    - Data was collected from the following marked locations:
        - Center: (11,8)
        - Point 1: (13.5,10.5)
        - Point 2: (16,13)

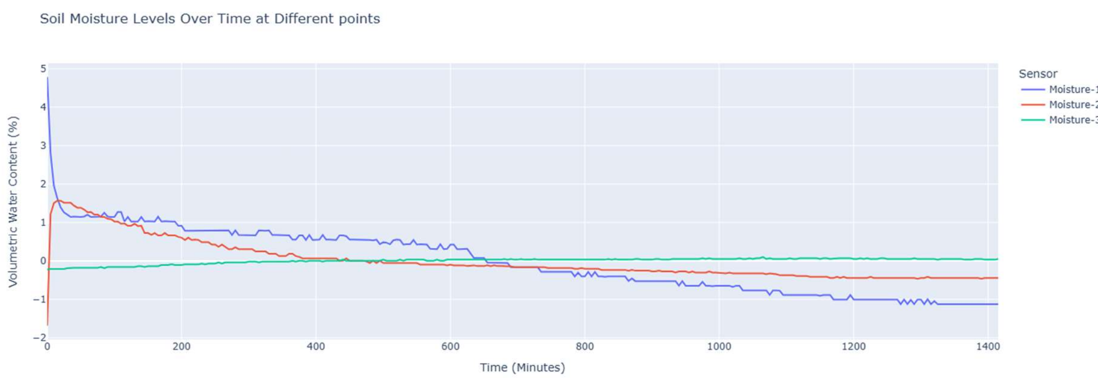
2. **Data Collection:**
    - Moisture data was recorded at each marked point individually over several days using a single sensor.
3. **Data Preprocessing:**
    - The collected datasets were split into manageable subsets.
    - Missing values were addressed using techniques such as Mean Imputation and Forward Fill (ffill).
4. **Calibration of Timestamps:**
    - Time alignment was performed across all datasets to create a unified timeline.
    - The datasets were then merged into a single table based on the common time column.

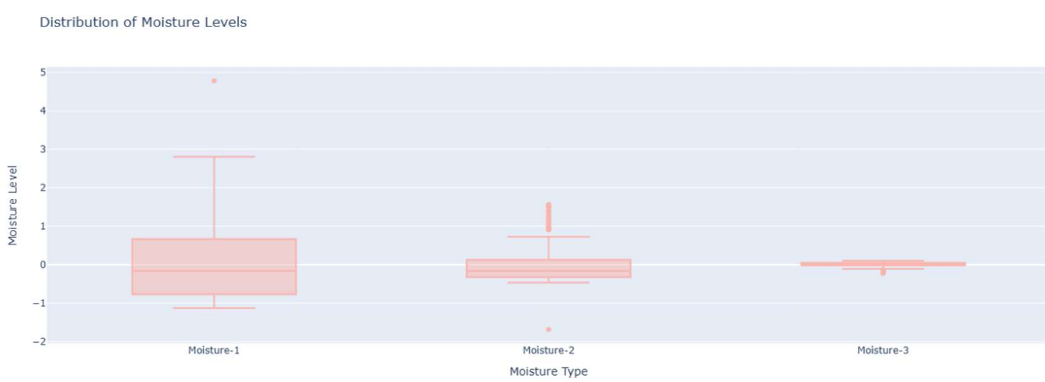
This structured setup ensured high-quality data inputs for subsequent machine learning model training and evaluation.

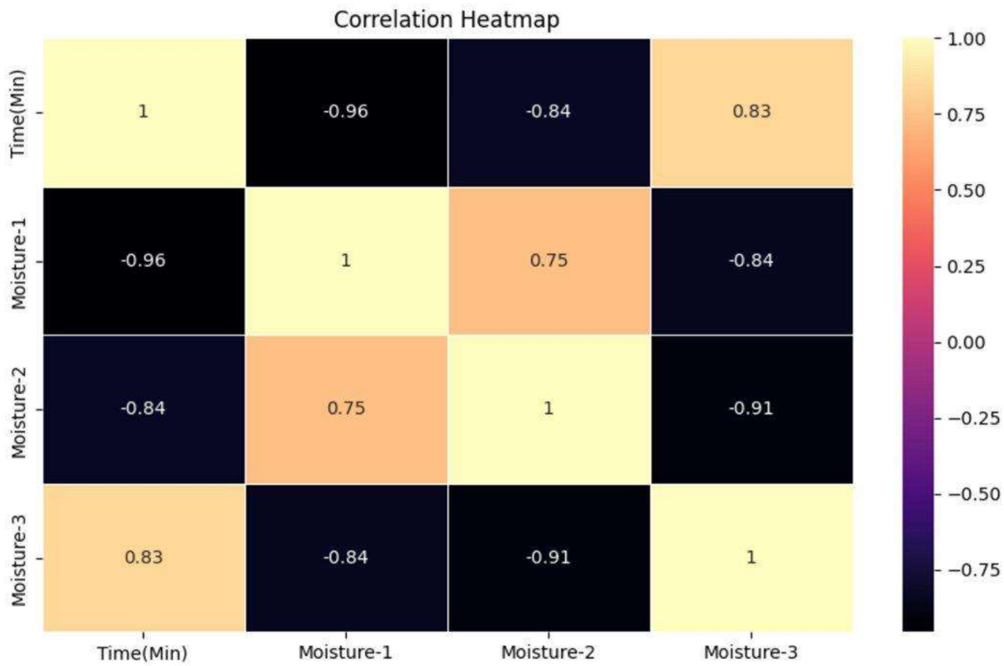
4.Dataset After data handling:

	Time(Min)	Moisture-1	Moisture-2	Moisture-3
0	0	22.84	8.36	5.33
1	5	20.86	11.26	5.35
2	10	20.01	11.55	5.35
3	15	19.69	11.61	5.35
4	20	19.45	11.61	5.35

Visualising the data using plotly:







5.Model Selection and Evaluation

To identify the most suitable machine learning model for soil moisture prediction, we evaluated several algorithms using cross-validation techniques. Key models tested include Linear Regression, Decision Tree, Random Forest, and XGBoost. The evaluation metrics comprised Mean Squared Error (MSE) and R-Squared values, focusing on minimizing prediction error and maximizing model accuracy.

1. **Linear Regression:**
    - MSE Scores: [0.0513, 0.0471, 0.0186, 0.0227, 0.1509]
    - Mean MSE: 0.0581 ± 0.0482
    - R-Squared Scores: [0.7298, 0.7190, 0.7758, 0.6982, 0.2846]
    - Mean R-Squared: 0.6415 ± 0.1802
2. **Decision Tree:**
    - MSE Scores: [0.0010, 0.0019, 0.0005, 0.0004, 0.0943]
    - Mean MSE: 0.0196 ± 0.0373
    - R-Squared Scores: [0.9849, 0.9831, 0.9689, 0.9680, 0.6503]
    - Mean R-Squared: 0.9110 ± 0.1305
3. **Random Forest:**
    - MSE Scores: [0.0016, 0.0053, 0.0004, 0.0002, 0.1015]
    - Mean MSE: 0.0218 ± 0.0399
    - R-Squared Scores: [0.9866, 0.9742, 0.9795, 0.9780, 0.6284]

- Mean R-Squared: 0.9093 ± 0.1405

4. **XGBoost:**
   - MSE Scores: [0.0014, 0.0924, 0.0005, 0.0003, 0.0942]
   - Mean MSE: 0.0378 ± 0.0453
   - R-Squared Scores: [0.9857, 0.6717, 0.9723, 0.9727, 0.6537]
   - Mean R-Squared: 0.8512 ± 0.1541

Model Selection: Based on the evaluation metrics, the Decision Tree and Random Forest models demonstrated superior performance with the lowest MSE and highest R-Squared values. Random Forest, with its ensemble learning approach, offers robustness and generalization capabilities, making it the preferred choice for this project.

Voting Regressor Implementation

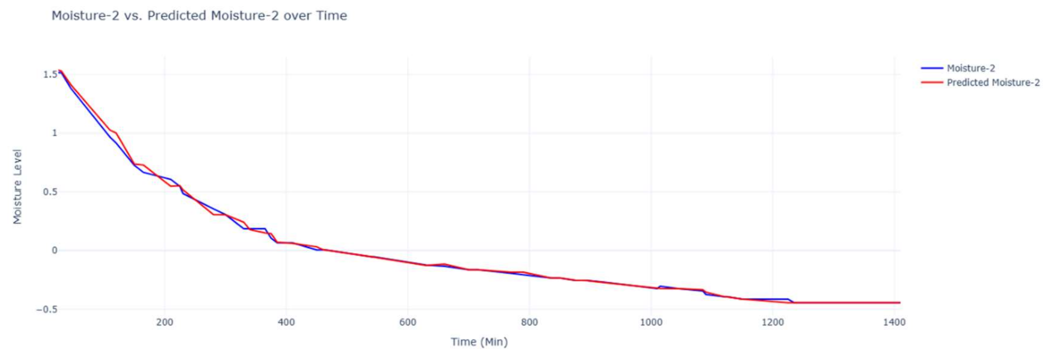
To further enhance prediction accuracy, we implemented a Voting Regressor by combining three models: Decision Tree, Random Forest, and XGBoost. This ensemble technique aggregates predictions from multiple models to improve overall reliability and performance.

The Voting Regressor was trained using the following setup:

```
model = VotingRegressor(
    estim Loading...
        ('DecisionTree', DecisionTreeRegressor(random_state=42)),
        ('RandomForest', RandomForestRegressor(random_state=42)),
        ('XGBoost', XGBRegressor(objective='reg:squarederror', random_state=42))
    ]

        )

mod1=model.fit(X_train, train_df['Moisture-2'])

mod2=model.fit(X_train, train_df['Moisture-3'])
```

The Voting Regressor demonstrated consistent results across multiple moisture columns. Predicted vs. actual moisture levels were visualized to validate model accuracy, showing strong alignment and minimal deviations.

Visualizations: (Actual vs Predicted values)



These plots highlight the model's ability to generalize well to unseen data, confirming its suitability for precision farming applications.

6.Model-2 Training

Using data from these marked points (time, moisture, distance), we trained a machine learning model, specifically a Random Forest Regressor. The goal was to predict moisture levels at points not directly measured, using only a small number of sensors (2 or 3). The dataset was modified to include relevant features for this purpose.

What Model-2 Predicts: Model-2 estimates the moisture levels at specific unsensed locations based on time, distance from the sensor, and initial moisture levels at the center. This predictive capability allows users to calculate moisture trends at customizable distances beyond the fixed distances (e.g., 2.5 and 3.0 units) in Model-1. By providing inputs such as center moisture and elapsed time, the model predicts moisture levels dynamically, enabling flexible and location-specific irrigation planning.

Modified Dataset:

Time(Min)	Moisture-1	Moisture_Point	Predicted_Moisture	Distance
0	4.797113	Moisture-2	-1.684754	3.535534
0	4.797113	Moisture-3	-0.228620	6.403124
5	2.817113	Moisture-3	-0.208620	6.403124
5	2.817113	Moisture-2	1.215246	3.535534
10	1.967113	Moisture-2	1.505246	3.535534

Implementation:

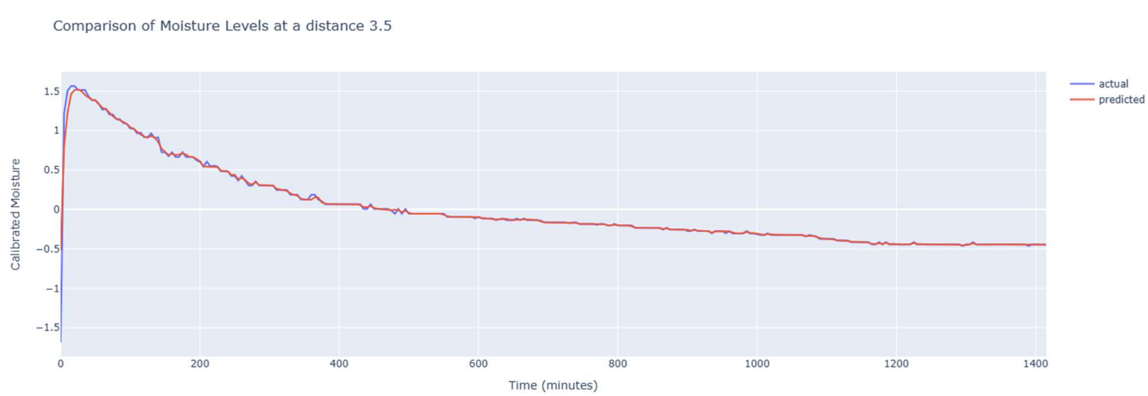
```
# Features (X) and Target (y)
X = df_melted_sorted[['Time(Min)', 'Moisture-1', 'Distance']]
y = df_melted_sorted['Predicted_Moisture']

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Train a model (e.g., Random Forest Regressor)
model = RandomForestRegressor(random_state=42)
model.fit(X_train, y_train)
```

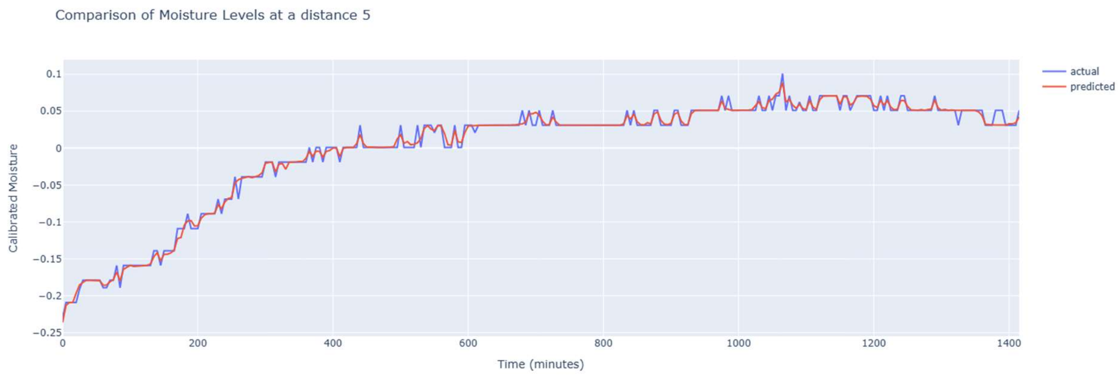
Evaluation Results:

- Mean Squared Error: 0.00033
- R-Squared: 0.9975

The model successfully predicted moisture levels with high accuracy, as demonstrated by the low MSE and high R-Squared values. The deployment of this model provides a reliable framework for estimating moisture levels at unsensed points in real-time precision farming scenarios.

Visualizations:



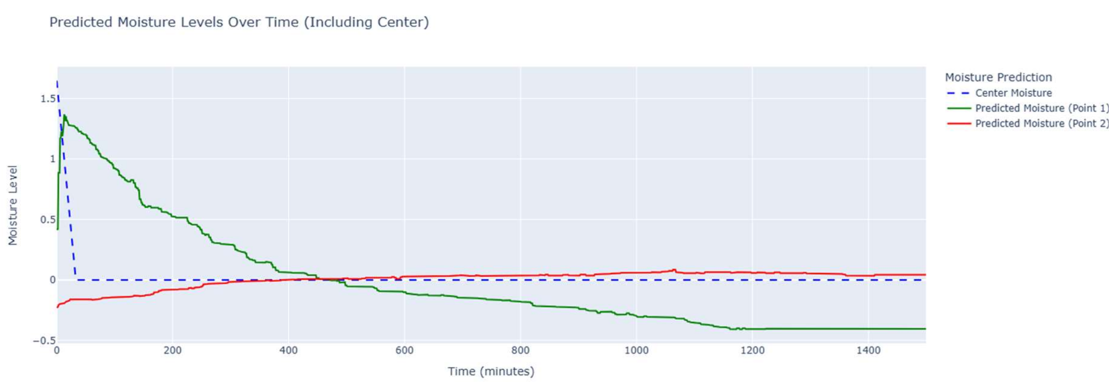


7.Simulated Data for Moisture Prediction

To gain further insights into soil moisture behavior, we simulated its variation over time at the center and two sensor points (Point 1 and Point 2). This simulation aimed to model the gradual absorption and evaporation of moisture. The distances to the sensor points were fixed at 3.5 and 6.0 units, respectively.

Explanation:

- Initial Moisture at Center: The starting moisture level at the center was set to 1.647113, decreasing linearly by 0.05 per minute over time.
- Prediction for Points 1 and 2: Using the trained model, moisture levels were predicted for Points 1 and 2 as time progressed.
- Visualization: The resulting plot illustrated moisture trends at the center (dashed line) and the two sensor points (solid lines in distinct colors).



The simulation and plot confirmed the model's capability to generalize and capture the dynamic nature of moisture levels over time, providing valuable predictions for effective irrigation planning.

8.Key Observations and Findings

- Machine learning models effectively reduce dependency on dense sensor grids by predicting moisture levels in unsensed areas.
- The Random Forest and Voting Regressor models demonstrated exceptional accuracy, with R-squared values exceeding 0.99 in several scenarios.
- Simulated data confirmed the model's robustness in dynamic and real-time moisture behavior.
- Moisture level predictions showed strong alignment with observed trends, enabling better irrigation planning.

9.Future Enhancements

- Model Expansion: Incorporate additional variables such as temperature, humidity, and soil type for more precise predictions.
- Automation: Deploy optimized sensor grids via drones or autonomous robots.
- Integration: Develop user-friendly dashboards for real-time monitoring and decision support for farmers.
- Scalability: Test and adapt the solution for large-scale industrial agriculture with diverse crops and environmental conditions.

10.Conclusion and Future Work

- This project demonstrated the feasibility and effectiveness of using machine learning for soil moisture prediction in precision farming. By leveraging limited sensor data, we achieved accurate predictions of moisture levels, enabling smarter irrigation management. The integration of ensemble models, such as Random Forest and Voting Regressor, ensured robustness and reliability in predictions.
- Future efforts will focus on expanding the model to include more environmental factors, automating deployment mechanisms, and enhancing scalability for diverse agricultural settings. This work lays the foundation for sustainable and efficient water management practices in precision agriculture.