21BCE056: Dhruvi Dholakia
21BCE193:Hardik Patel

NIRMA UNIVERSITY
INSTITUTE OF TECHNOLOGY
NAAC ACCREDITED 'A+' GRADE

DM INNOVATIVE

PREDICTIVE MODELING OF COTTON CROP PRODUCTION IN MAHARASHTRA USING MULTI-SOURCE EARTH OBSERVATION DATA MINING AND MACHINE LEARNING TECHNIQUES

CONTENT

Introduction

- This project focuses on leveraging multi-source Earth observation data and machine learning techniques to develop a predictive model for cotton crop production in Maharashtra,India. The study integrates various environmental and climatic factors, including meteorological parameters and satellite-derived variables, to assess their impact on cotton yield.The dataset encompasses a range of elements such as Area of Interest (AOI) name, year of observation, rainfall, surface temperature, leaf area index (LAI), fraction of photosynthetically active radiation (FPAT), wind speed, transpiration, air temperature, soil moisture, humidity, enhanced vegetation index (EVI), vapor pressure deficit (VPD), vapor pressure, minimum and maximum temperature, runoff, land surface temperature (LSTN, LSTD), surface net solar radiation, soil pH, and normalized difference water index (NDWI).
- A Python-based approach is employed to preprocess the data and develop predictive models using machine learning algorithms. Specifically, algorithms including Ridge Linear Regression, k-Nearest Neighbors, Support Vector Regression, Gradient Boosting Decision Tree, Random Forest, and Light Gradient Boosting Machine are implemented to predict cotton crop production based on the input features.
- The performance of each model is evaluated using metrics such as accuracy, error, and validation scores. Through rigorous evaluation and comparison of the predictive models, insights are gained into the most effective algorithms for predicting cotton yield in Maharashtra under varying environmental conditions. The outcomes of this project hold significant implications for agricultural planning, resource allocation, and decision-making in cotton cultivation, ultimately contributing to enhanced crop management practices and sustainable agricultural development in the region.

Objective

- Gather crucial agricultural data by using Geographic Information Systems (GIS).
- Accurately gather environmental data for particular places and crops by using geospatial analysis and satellite imagery.
- Develop very effective yield prediction models.
- To predict agricultural output, apply deep learning and machine learning approaches.
- Boost the accuracy of the model using the acquired agricultural data sets.
- For agricultural machine learning models, use explainable AI techniques.
- Recognise the inner workings of complex models used with data related to agriculture.
- Describe how models use environmental factors to project yields.

Data Collection

Used remote sensing and geographic information systems (GIS) to gather data.

Concentrated on areas with a high likelihood of cultivation and the cotton crop.

Indian state of Maharashtra's cotton-growing districts were chosen using yield statistics from the past.

Areas that produce little cotton were filtered.

Pixel-level data was extracted using Google Earth Engine.

Algorithm 1 Satellite Data Processing Algorithm

```
1:  Define feature list *features* $\leftarrow$ ['Yield', 'humidity', 'soilwater', 'surfT', 'ws',
    'transpiration', 'surface_net_solar_radiation', 'ph', 'airtemp', 'LAI', 'FPAT',
    'vpd', 'tmmn', 'tmmx', 'ro', 'vap', 'LSTN', 'LSTD', 'EVI', 'NDWI']
2:  Select Satellite Data Source
3:          Identify satellite with necessary *features*
4:  Extract Data
5:          Retrieve data for *features*
6:  Filter Data by Year
7:          Include only records from years 2000 to 2020
8:  Calculate Yearly Means
9:  for each year in 2000 to 2020 do
10:       Calculate mean of each feature
11: end for
12: Export Data
13:         Export yearly means to a CSV file:
```

Data Extraction

Datasets

- GLDAS-2.1: Global Land Data Assimilation System
- CHIRPS Daily: Climate Hazards Group InfraRed Precipitation with Station Data (Version 2.0 Final
- ERA5-Land Hourly - ECMWF Cli- mate Reanalysis
- MOD15A2H V6.1 MODIS combined Leaf Area Index (LAI) and Fraction of Photo- synthetically Active Radiation (FPAR)

Parameters

- Soil Moisture, Wind Speed, Humidity, Transpiration
- Precipitation
- surface net solar radiation, Temperature
- Leaf Area Index (LAI), Fraction of Photosyn- thetically Active Radia- tion (FPAR)

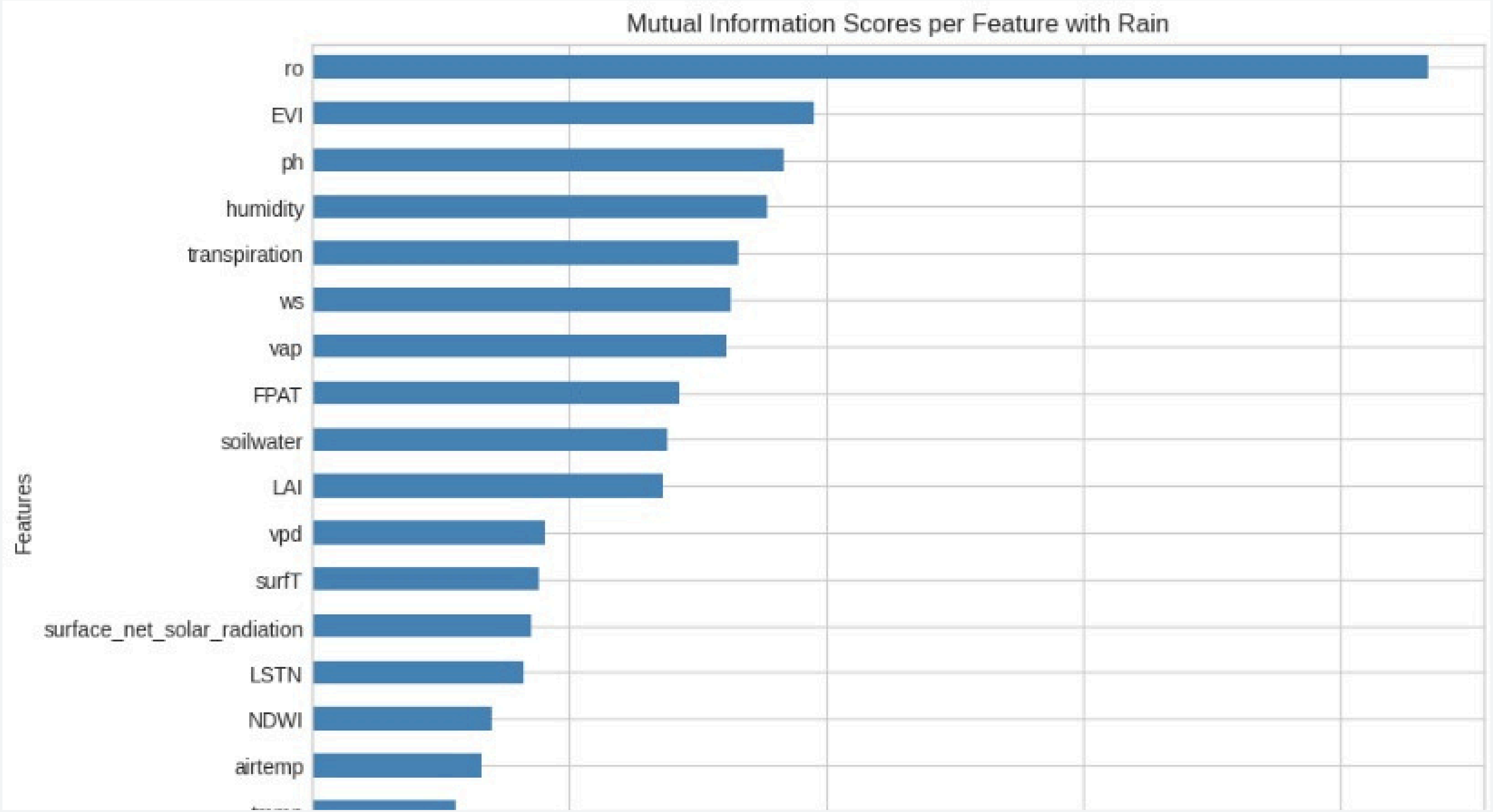
Data Extraction

Datasets

- TerraClimate is a dataset of monthly climate and climatic water balance for global terres- trial surfaces.
- MOD11A2 V6.1 product pro- vides an average 8-day land surface temperature (LST)
- MOD13A2 V6.1 product pro- vides two Vegetation Indices (VI): the Enhanced Vegetation Index (EVI)
- MODIS Terra Daily NDWI

Parameters

- Vapor pressure deficit, Vapor pressure,Maximum temperature,Minimum temperature, Runoff
- Day land surface tem- perature, Night land surface temperature
- the Enhanced Vegetation Index (EVI)
- NDWI



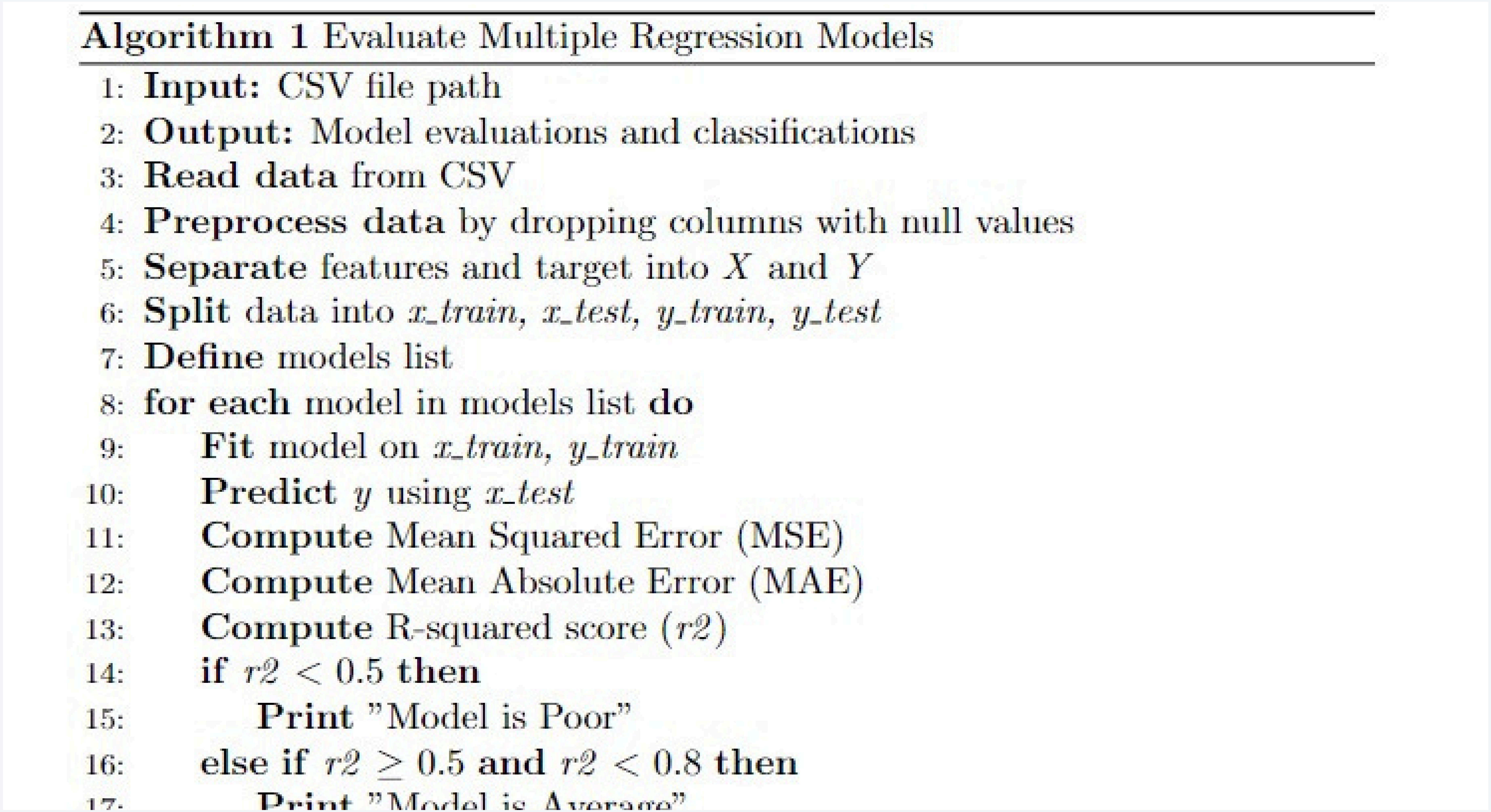
Mutual Information Scores per Feature with Rain
Features
ro
EVI
ph
humidity
transpiration
ws
vap
FPAT
soilwater
LAI
vpd
surfT
surface_net_solar_radiation
LSTN
NDWI
airtemp

Final Dataset

- Data at the district level for yield forecasts and predictive modelling.
- When training machine learning models, incorporates historical yield as the goal variable.

Feature Selection

- Calculate how dependent the input and output variables are on one another.
- Determine which attributes offer the most information about the output.
- Comprises both linear and non-linear connections.
- Non-parametric technique without any data presumptions



Algorithm 1 Evaluate Multiple Regression Models

```
1:  Input: CSV file path
2:  Output: Model evaluations and classifications
3:  Read data from CSV
4:  Preprocess data by dropping columns with null values
5:  Separate features and target into $X$ and $Y$
6:  Split data into $x\_train$, $x\_test$, $y\_train$, $y\_test$
7:  Define models list
8:  for each model in models list do
9:      Fit model on $x\_train$, $y\_train$
10:     Predict $y$ using $x\_test$
11:     Compute Mean Squared Error (MSE)
12:     Compute Mean Absolute Error (MAE)
13:     Compute R-squared score ($r2$)
14:     if $r2 < 0.5$ then
15:         Print "Model is Poor"
16:     else if $r2 \geq 0.5$ and $r2 < 0.8$ then
17:         Print "Model is Average"
```

ML Algorithms

- RidgeLinear Regression
- k-Nearest Neighbors
- Support Vector Regression
- Gradient Boosting Decision Tree
- Random Forest
- Light Gradient Boosting Machine

Results

```
Model: Ridge Linear Regression
R-Accuracy: 0.6960174896378097
Model is Average
-------------------
Model: k-Nearest Neighbors
R-Accuracy: 0.6780996390207099
Model is Average
-------------------
Model: Support Vector Regression
R-Accuracy: 0.05356118116183095
Model is Poor
-------------------
Model: Gradient Boosting Decision Tree
R-Accuracy: 0.8134435172226813
Model is Good
-------------------
Model: Random Forest
R-Accuracy: 0.8307899125836846
Model is Good
-------------------
Model: Light Gradient Boosting Machine
R-Accuracy: 0.8322682481111563
Model is Good
------------------
Model: DNN
R-Accuracy: 0.8322682481111563-
```