Data Collection and Preprocessing Phase

Date	18 July 2024
Team ID	SWTID1721319573
Project Title	Blueberry Yield Prediction
Maximum Marks	6 Marks

Data Exploration and Preprocessing Template

Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	Description
Data Overview	The dataset includes columns like clonesize, honeybee, bumbles, andrena, osmia, and various weather-related features, with dimensions of X rows and Y columns. Basic statistics such as mean, median, and standard deviation will be calculated.
Univariate Analysis	Examine each variable individually to understand their distributions and central tendencies. Calculate statistics like mean, median, mode, and standard deviation for each feature.
Bivariate Analysis	Explore relationships between pairs of variables using correlation coefficients and scatter plots. For example, assess how `clonesize` relates to `yield`.
Multivariate Analysis	Analyze patterns involving multiple variables. Use techniques like heatmaps and pair plots to understand interactions and dependencies among features.
Outliers and Anomalies	Identify outliers using statistical methods (e.g., IQR) and visualization (e.g., box plots). Apply transformations or filtering techniques to address these anomalies.
Data Preprocessing Code Screenshots	

Loading Data	In [3]: ``` import pandas as pd from sklearn.model_selection import train_test_split from sklearn.ensemble import RandomForestRegressor from sklearn.metrics import mean_squared_error import joblib import matplotlib.pyplot as plt import seaborn as sns import pickle from sklearn.linear_model import LinearRegression ``` In [4]: ``` # Load dataset df = pd.read_csv("C:\\Users\\angel\\OneDrive\\Desktop\\WildBlueberryPollinationSimulationData.csv") ``` In [5]: ``` # Display the first few rows of the dataset print(df.head()) ``` Row# clonesize honeybee bumbles andrena osmia MaxOfUpperTRange \ 0 0 37.5 0.75 0.25 0.25 0.25 86.0 1 1 37.5 0.75 0.25 0.25 0.25 86.0 2 2 37.5 0.75 0.25 0.25 0.25 94.6 3 3 37.5 0.75 0.25 0.25 0.25 94.6 4 4 37.5 0.75 0.25 0.25 0.25 86.0 MinOfUpperTRange AverageOfUpperTRange MaxOfLowerTRange MinOfLowerTRange \ 0 52.0 71.9 62.0 30.0
Handling Missing Data	In [9]: ``` # Data Cleaning and preparation # Identify missing values missing_values = df.isnull().sum() # Handle missing values df = df.dropna() # Verify no missing values remain print("Missing values after handling:\n", df.isnull().sum()) ``` Missing values after handling: Row# 0 clonesize 0 honeybee 0 bumbles 0 andrena 0 osmia 0 MaxOfUpperTRange 0 MinOfUpperTRange 0 AverageOfUpperTRange 0 MaxOfLowerTRange 0 MinOfLowerTRange 0 AverageOfLowerTRange 0 RainingDays 0 AverageRainingDays 0 fruitset 0 fruitmass 0
Data Transformation	In [10]: ``` # Data Transformation from sklearn.preprocessing import StandardScaler # Select columns to be scaled columns_to_scale = ['clonesize', 'honeybee', 'bumbles'] # Initialize scaler scaler = StandardScaler() # Scale the selected columns df[columns_to_scale] = scaler.fit_transform(df[columns_to_scale]) # Display transformed data print(df.head()) ```
Feature Engineering	In [11]: ``` # Feature Engineering # Create new feature 'average_temp' as the average of 'MaxOfUpperTRange' and 'MinOfUpperTRange' df['average_temp'] = (df['MaxOfUpperTRange'] + df['MinOfUpperTRange']) / 2 # Display data with new feature print(df[['MaxOfUpperTRange', 'MinOfUpperTRange', 'average_temp']].head()) ``` MaxOfUpperTRange MinOfUpperTRange average_temp 0 86.0 52.0 69.0 1 86.0 52.0 69.0 2 94.6 57.2 75.9 3 94.6 57.2 75.9 4 86.0 52.0 69.0
Save Processed Data	In [12]: ``` # Save Processed Data # Save the cleaned and transformed dataset to a new CSV file df.to_csv('C:\\Users\\angel\\OneDrive\\Desktop\\WildBlueberryPollinationSimulationData.csv', index=False) # Verify the saved data saved_data = pd.read_csv('C:\\Users\\angel\\OneDrive\\Desktop\\WildBlueberryPollinationSimulationData.csv') print(saved_data.head()) ```