REPORT: LOGISTIC REGRESSION MODEL FOR PREDICTING RAIN IN AUSTRALIA

1. Objective

The objective of this project is to build and evaluate a logistic regression model that predicts whether it will rain tomorrow based on historical weather data from Australia.

2. Dataset

The dataset used for this project is the "Weather Dataset" available on Kaggle. It contains daily weather observations from various locations across Australia.

3. Data Preprocessing

The data preprocessing steps include:

- **Loading and Cleaning the Data**: The dataset is downloaded using the opendatasets package, and rows with missing values in the target columns RainToday and RainTomorrow are removed.
- **Splitting the Data**: The dataset is divided into training, validation, and test sets based on the year of observation. Data from years before 2015 is used for training, data from 2015 is used for validation, and data from years after 2015 is used for testing.
- **Feature Engineering**:
    - Identifying numeric and categorical columns.
    - Imputing missing values in numeric columns using the mean strategy.
    - Scaling numeric features using MinMaxScaler.
    - One-hot encoding categorical features using OneHotEncoder.

4. Model Training

A logistic regression model is trained on the preprocessed training data. The following steps are performed:

- **Feature Selection**: Selecting numeric and one-hot encoded categorical features for training.
- **Model Initialization and Training**: Initializing and fitting a logistic regression model using the liblinear solver.

5. Model Evaluation

The model's performance is evaluated using accuracy score and confusion matrix on the validation and test sets. The following steps are performed:

- **Accuracy Score**: Calculating the proportion of correct predictions.
- **Confusion Matrix**: Visualizing the confusion matrix to analyze the model's performance in terms of true positives, false positives, true negatives, and false negatives.

6. Prediction Function

A helper function predict_input is created to predict the likelihood of rain for a new input. This function:

- Imputes and scales numeric features.

- One-hot encodes categorical features.
- Makes predictions using the trained logistic regression model.

Explanation of Key Sections

- **Data Loading and Cleaning**: The dataset is downloaded and cleaned by removing rows with missing target values.
- **Data Splitting**: The dataset is split into training, validation, and test sets based on the year.
- **Feature Engineering**: Missing numeric values are imputed, numeric features are scaled, and categorical features are one-hot encoded.
- **Model Training**: A logistic regression model is trained on the preprocessed training data.
- **Model Evaluation**: The model is evaluated on validation and test sets using accuracy score and confusion matrix. Visualization of the confusion matrix helps in understanding the performance.
- **Prediction Function**: A function predict_input is created to make predictions on new data inputs, showcasing how to preprocess new inputs and make predictions using the trained model.

Results and Insights

- **Accuracy**: The training, validation, and test accuracy scores are printed, providing an overview of the model's performance on different data splits.
- **Confusion Matrix**: The confusion matrix for validation and test sets is plotted, helping in understanding the distribution of true positives, false positives, true negatives, and false negatives.
- **New Input Prediction**: The predict_input function demonstrates how to handle new data inputs and make predictions using the trained model. The example input returns a prediction and probability of rain.This approach ensures a robust preprocessing pipeline and evaluation framework, making it easier to understand the model's performance and deploy it for real-world predictions.