MACHINE LEARNING

23CSE301

(Linear Regression)

NAME: LAKSHANA B

ROLL.NO: CH.SC.U4CSE23028

Exercises:

2. Conduct Linear regression on a model that predicts the body weight of a person. The dependent variable is thus the body weight, while the height, age and gender are chosen as independent variables. The following example data set is available:

Weight	Height	Age	Gender
79	1.80	35	Male
69	1.68	39	Male
73	1.82	25	Male
95	1.70	60	Male
82	1.87	27	Male
55	1.55	18	Female
69	1.50	89	Female
71	1.78	42	Female
64	1.67	16	Female
69	1.64	52	Female

AIM:

To perform **Linear Regression** to predict the body weight of a person (dependent variable) based on independent variables: height, age, and gender, using the given dataset.

PROCEDURE:

Import necessary libraries (e.g., pandas, sklearn)

- **Create a dataset** with columns: Weight, Height, Age, Gender.
- **Encode the 'Gender'** column into numeric values using LabelEncoder.
- **Define independent variables** (`Height`, `Age`, `Gender`) as features (X) and `Weight` as the target (y).
- **Split the data** into training and test sets (80/20 split).
- **Train the Linear Regression model** on training data.
- **Predict the weights** on test data using the trained model.
- **Evaluate the model** using R^2 Score and Mean Squared Error (MSE).
- **Display the predicted and actual weights**, and the performance metrics.

PROGRAM:

[1]:

```python
# Step 1: Import libraries
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score

# Step 2: Create dataset from the given table
data = {
    'Weight': [79, 69, 73, 95, 82, 55, 69, 71, 64, 69],
    'Height': [1.80, 1.68, 1.82, 1.70, 1.87, 1.55, 1.50, 1.78, 1.67, 1.64],
    'Age':    [35, 39, 25, 60, 27, 18, 89, 42, 16, 52],
    'Gender': ['Male', 'Male', 'Male', 'Male', 'Male', 'Female', 'Female', 'Female', 'Female', 'Female']
}

df = pd.DataFrame(data)

# Step 3: Encode Gender (Male=0, Female=1)
le = LabelEncoder()
df['Gender_encoded'] = le.fit_transform(df['Gender'])  # Male=1, Female=0 or vice versa

# Step 4: Define features (X) and target (y)
X = df[['Height', 'Age', 'Gender_encoded']]
y = df['Weight']

# Step 5: Train/Test split (optional since data is small, but good practice)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Step 6: Train the Linear Regression model
model = LinearRegression()
model.fit(X_train, y_train)

# Step 7: Predict and evaluate
y_pred = model.predict(X_test)

# Print results
print("Predicted Weights:", y_pred)
print("Actual Weights:", list(y_test))
print("R² Score:", r2_score(y_test, y_pred))
print("Mean Squared Error:", mean_squared_error(y_test, y_pred))
```

```python
# Optional: See model coefficients
print("\nModel Coefficients:")
print(f"Height: {model.coef_[0]:.2f}")
print(f"Age: {model.coef_[1]:.2f}")
print(f"Gender_encoded: {model.coef_[2]:.2f}")
print(f"Intercept: {model.intercept_:.2f}")
```

```
Predicted Weights: [56.85651025 79.10576115]
Actual Weights: [64, 69]
R² Score: -11.252468343750825
Mean Squared Error: 76.57792714844265

Model Coefficients:
Height: 32.84
Age: 0.32
Gender_encoded: 14.62
Intercept: -3.06
```

[ ]:

[ ]: