

Experiment–6

Aim:

Program to implement simple and multiple linear regression.

Code:

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 from sklearn.datasets import load_iris
6 from sklearn.model_selection import train_test_split
7 from sklearn.linear_model import LinearRegression
8 from sklearn.metrics import r2_score, mean_absolute_error,
   ↪ mean_squared_error
9
10 # Load Iris
11 iris = load_iris(as_frame=True)
12 df = iris.frame
13 df.columns = ['sepal_length', 'sepal_width', 'petal_length',
   ↪ 'petal_width', 'species']
14
15 print("Dataset Shape:", df.shape)
16 print(df.head())
17
18 # Target: petal_width
19 y = df['petal_width']
20
21 # 1. SIMPLE LINEAR REGRESSION (Best single feature: petal_length)
22 X_simple = df[['petal_length']]
23
24 X_train_s, X_test_s, y_train_s, y_test_s = train_test_split(X_simple, y,
   ↪ test_size=0.2, random_state=42)
25
26 model_simple = LinearRegression()
27 model_simple.fit(X_train_s, y_train_s)
28 y_pred_s = model_simple.predict(X_test_s)
29
30 print("\n" + "="*60)
31 print("1. SIMPLE LINEAR REGRESSION (petal_length → petal_width)")
32 print("="*60)
33 print(f"Slope: {model_simple.coef_[0]:.4f}")
34 print(f"Intercept: {model_simple.intercept_:.4f}")
35 print(f"R2 Score: {r2_score(y_test_s, y_pred_s):.4f}")
36 print(f"RMSE: {np.sqrt(mean_squared_error(y_test_s, y_pred_s)):.4f}")
37
38 # Plot Simple Regression
```