

Experiment–4 (b)

Aim:

Apply regression algorithm and handle missing value and categorical data.

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.pipeline import Pipeline
8 from sklearn.compose import ColumnTransformer
9 from sklearn.impute import SimpleImputer
10 from sklearn.preprocessing import StandardScaler, OneHotEncoder
11 from sklearn.linear_model import LinearRegression
12 from sklearn.metrics import r2_score, mean_absolute_error
13
14 # Load Iris
15 iris = load_iris(as_frame=True)
16 df = iris.frame
17
18 # Add species name as categorical column
19 df['species'] = pd.Categorical.from_codes(iris.target, iris.target_names)
20
21 # Intentionally create missing values (for learning)
22 np.random.seed(42)
23 df.loc[np.random.choice(df.index, 20), 'sepal length (cm)'] = np.nan
24 df.loc[np.random.choice(df.index, 15), 'petal length (cm)'] = np.nan
25 print("Missing values created:")
26 print(df.isnull().sum())
27
28 # Target: predict petal_width
29 X = df.drop(['petal width (cm)', 'target'], axis=1)
30 y = df['petal width (cm)']
31
32 # Split
33 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
34     ↪ random_state=42)
35
36 # Preprocessing
37 numeric_features = ['sepal length (cm)', 'sepal width (cm)', 'petal length
38     ↪ (cm)']
39 categorical_features = ['species']
40 numeric_pipeline = Pipeline([
41     ('imputer', SimpleImputer(strategy='mean')),
```