LAB - I

Filename = "housing.csv"

i import pandas as pd.

i. Load the csv file.

df = pd.read_csv('housing.csv')

ii. Display information

print(df.info())

iii. Display statistical information.

print(df['ocean_proximity'].value_counts())

iii. Display statistical information.

print(df.describe())

v Display which attribute (cols) has null values.

print(df.isnull().sum())

To Do 2.

1. Diabetes

print(df.isnull().sum())

op:- I P

class 0

dtype: int64

Adult issue

age 0

income 0

dtype: int64

There are no missing values.

2. Gender & Class (Diabetes)

ordinal encoder or one hot encoder

ordinal_encoder = OrdinalEncoder(categories = [['F', 'M'
handle_unknown = 'use_encoded_value',
unknown_value = -1)

df['Gender_Encoded'] = ordinal_encoder.fit_transform(df[['Gender']])

one_hot_encoder = OneHotEncoder()

encoded_data = one_hot_encoder.fit_transform(df[['Class']])

O/P :-

Gender_Encoded	Class_N	Class_P
0.0	1.0	0.0
1.0	1.0	0.0
0.0	1.0	0.0

gender - ordinal Encoder

workclass, education, marital status, occupation, relationship, race, nature country - One hot Encoder -

oe = OrdinalEncoder(categories = [['Female', 'Male'
df[[oc]] = oe.fit_transform(df[[oc]])

O/P :-

gender	native-country US - - -
1.0	1.0
1.0	1.0
0.0	1.0

3. Min Max Scaling
→ Scales data to a specific range
→ Highly sensitive to outliers
→ preserves original distribution
Used for Bounded data and algorithms sensitive feature range

Standardization.
→ Does not bound data to a specific range
→ less sensitive to outliers
→ transform to normal distribution.
is better for algorithms that assume normally distributed data or when dealing with outliers.