

## LAB-1

To Do - 1

1.

import pandas as pd

file\_path = '/content/housing.csv'

df = pd.read\_csv(file\_path)

print("Sample data:")

print(df.head())

print("\n")

O/p

	longitude	latitude	housing-median-age
0	-122.23	37.88	41.0

total-rooms

total-bedsrooms

population

880.0

129.0

322.0

households

median-income

median-house-value

126.0

8.3252

452600.0

Ocean-proximity

NEAR BAY

2.

print(df.info)

O/p

0	-122.23	41.0	880.0	129.0
---	---------	------	-------	-------

37.88

322.0

126.0

8.3252

452600.0

NEAR BAY.



3.

print(df.describe())

o/p	count	longitude	latitude
		20640.00	20640.00
	mead	-119.5	
	std	2.00	
	min	-124	
	25%	-121	
	50%	-118	
	75%	-118	
	max	-114.31	

4.

print(df['ocean proximity'].value\_counts())

o/p ocean proximity

L1H OCEAN 9136

INLAND 65351

Near Ocean 2656

near bay 2290

Island 5

Name: count, dtype: int64.

5.

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

longitude 0

latitude 0

housing median age. 0

total\_rooms 0

total\_bathrooms

207

population 0

households 0

median\_income 0

median\_house\_value 0

ocean\_proximity 0

dtype: int64







gender - Ordinal Encoder

workclass, education, marital status, occupation, relationship  
race, native country - OneHot Encoder

oe = OrdinalEncoder(categories=[["Female", "Male"]])

df[gc] = oe.fit\_transform(df[gc])

mcc = ['workclass', 'education', 'native country']

ohc = OneHotEncoder(sparse\_output=False, drop='first')

ed = ohc.fit\_transform(df[mcc])

O/P)

gender

1.0

1.0

0.0

native-country vs

1.0

1.0

1.0

### 3. Min Max Scaling

→ Scales data to a specific range

→ Highly sensitive to outliers

→ Preserves original distribution

Standardization.

→ Doesn't bound data to a specific range.

→ Less sensitive to outliers

→ Transforms to normal distribution.

Used for

Bounded data and algorithms

Sensitive feature range

is better for algorithms

that assume normally distributed data when dealing with outliers.