

MODEL LAB

1) House price prediction

Aim: To Analyze House data And predict House prices Using Regression Model

Algorithm:

- Read the House dataset
- Clean And preprocess the data
- Split data into Training And Testing Sets
- Train a Regression Model
- predict House prices

Program:

```
import pandas as pd  
from sklearn.linear_model import  
LinearRegression  
data = pd.read_csv("house-data.csv")  
data.fillna(data.mode().iloc[0], inplace=True)  
x = data.drop(["price"], axis=1)  
y = data["price"]  
model = LinearRegression()  
model.fit(x, y)  
print(model.predict(x))
```

Output:

Area = 1800sqft

Bedrooms = 3

Predicted Price = ₹ 5,400,000.

Result: House prices were successfully predicted Using Regression..

2) Candidate Elimination Algorithm

Aim: To Find all Hypotheses Consistent with Given Training Examples Using Candidate Elimination.

Algorithm:

- Initialize Specific and General Hypotheses
- Read Training Examples
- Generalize for Positive Examples
- Specialize for Negative Examples
- Display Final Hypotheses

Program:

```
import csv  
data = list(csv.reader(open("Training-data.csv")))  
Hypotheses = data[1]  
S = Hypotheses[0][:-1]  
for h in Hypotheses:  
    if h != Hypotheses:  
        if h[-1] == "Yes":  
            for i in range(len(h)):  
                if h[i] != S[i]:  
                    S[i] = "?"  
print ("Specific Hypothesis", S)
```

Output:

Final Specific Hypothesis(S):

[?, Large, Light? Thick]

Result: The Consistent Hypothesis Was Obtained Using

Candidate Elimination.

3) Linear Regression

Aim:- To Implement Linear Regression And Evaluate its prediction Capability

Algorithm:

- Load dataset
- Separate input And Output Variables
- Train Output Value
- predict Output Value
- Display predictions

Program:

```
import pandas as pd  
from SkLearn.Linear_model import  
LinearRegression.  
data = pd.read_csv("data.csv")  
x = data['x'])  
y = data('Y')  
Model = LinearRegression()  
Model.fit(x, y)  
print(Model.predict(x))
```

Output:

Regression Equation: $y = 2x$

For $x=5$,

Predicted $y=10$.

Result: Linear Regression Correctly Modeled The Relationship between x and y.

4) Em Algorithm

Aim: To cluster data using the Expectation/Maximization Algorithm.

Algorithm:

- Initialize cluster parameters
- perform Expectation Step
- perform Maximization Step
- Repeat Until Convergence.
- Output Clusters

Program:

```
import numpy as np
from sklearn.mixture import GaussianMixture
z = np.array([1, 2, 3, 9, 10])
model = GaussianMixture(n_components=2)
model.fit(z)
print(model.predict(z))
```

Output:

Data point	Cluster
1	0
2	0
3	0
8	1
9	1
10	1

Result: The Em Algorithm Successfully Clustered The Data into Two Groups.