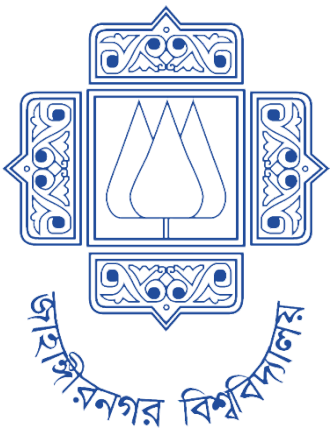
Institute of Information Technology (IIT)

Jahangirnagar University



ICT 4102– Artificial Intelligence Lab

LAB REPORT-04

SUBMITTED BY

Name: MD Zuleyenine Ibne Noman

Roll No: 2002

Exam Roll:192319

Submission Date: 10-08-2023

SUBMITTED TO

Md. Mahmudur Rahman

Lecturer, IIT

In :

In [8]:

```
import numpy as np
import pandas as pd
```

In [9]:

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

Out[9]:

	0.00632 18.00 2.310 0 0.5380 6.5750 65.20 4.0900 1 296.0 15.30 396.90 4.98 24.00
0	0.02731 0.00 7.070 0 0.4690 6.4210 78....
1	0.02729 0.00 7.070 0 0.4690 7.1850 61....
2	0.03237 0.00 2.180 0 0.4580 6.9980 45....
3	0.06905 0.00 2.180 0 0.4580 7.1470 54....
4	0.02985 0.00 2.180 0 0.4580 6.4300 58....

In [11]:

```
coumn_names=['CRIM','ZN','INDUS','CHAS','NOX','RM','AGE','DIS','RAD','TAX','PTRIO','B','
```

In [14]:

```
data=pd.read_csv('housing.csv',header=None,delimiter=r"\s+",names=coumn_names)
```

In [15]:

```
data.head()
```

Out[15]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRIO	B	LS
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296.0	15.3	396.90	4
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242.0	17.8	396.90	9
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242.0	17.8	392.83	4
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222.0	18.7	394.63	2
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222.0	18.7	396.90	5

In [16]:

```
data.shape
```

Out[16]:

```
(506, 14)
```

In [17]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 506 entries, 0 to 505
Data columns (total 14 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   CRIM    506 non-null    float64
 1   ZN      506 non-null    float64
 2   INDUS   506 non-null    float64
 3   CHAS    506 non-null    int64  
 4   NOX     506 non-null    float64
 5   RM      506 non-null    float64
 6   AGE     506 non-null    float64
 7   DIS     506 non-null    float64
 8   RAD     506 non-null    int64  
 9   TAX     506 non-null    float64
 10  PTRIO   506 non-null    float64
 11  B       506 non-null    float64
 12  LSTAT   506 non-null    float64
 13  MEDV    506 non-null    float64
dtypes: float64(12), int64(2)
memory usage: 55.5 KB
```

In [18]:

```
data.describe()
```

Out[18]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE
count	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000
mean	3.613524	11.363636	11.136779	0.069170	0.554695	6.284634	68.574901
std	8.601545	23.322453	6.860353	0.253994	0.115878	0.702617	28.148861
min	0.006320	0.000000	0.460000	0.000000	0.385000	3.561000	2.900000
25%	0.082045	0.000000	5.190000	0.000000	0.449000	5.885500	45.025000
50%	0.256510	0.000000	9.690000	0.000000	0.538000	6.208500	77.500000
75%	3.677083	12.500000	18.100000	0.000000	0.624000	6.623500	94.075000
max	88.976200	100.000000	27.740000	1.000000	0.871000	8.780000	100.000000

In [19]:

```
data.isnull().sum()
```

Out[19]:

```
CRIM     0
ZN       0
INDUS    0
CHAS     0
NOX      0
RM       0
AGE      0
DIS      0
RAD      0
TAX      0
PTRIO    0
B        0
LSTAT    0
MEDV     0
dtype: int64
```

In [20]:

```
corr=data.corr()
print(corr)
```

```
           CRIM        ZN     INDUS      CHAS       NOX        RM       AG
E  \
CRIM   1.000000 -0.200469  0.406583 -0.055892  0.420972 -0.219247  0.35273
4
ZN    -0.200469  1.000000 -0.533828 -0.042697 -0.516604  0.311991 -0.56953
7
INDUS  0.406583 -0.533828  1.000000  0.062938  0.763651 -0.391676  0.64477
9
CHAS  -0.055892 -0.042697  0.062938  1.000000  0.091203  0.091251  0.08651
8
NOX    0.420972 -0.516604  0.763651  0.091203  1.000000 -0.302188  0.73147
0
RM    -0.219247  0.311991 -0.391676  0.091251 -0.302188  1.000000 -0.24026
5
AGE    0.352734 -0.569537  0.644779  0.086518  0.731470 -0.240265  1.00000
0
DIS   -0.379670  0.664408 -0.708027 -0.099176 -0.769230  0.205246 -0.74788
1
RAD    0.625505 -0.311948  0.595129 -0.007368  0.611441 -0.209847  0.45602
2
TAX    0.582764 -0.314563  0.720760 -0.035587  0.668023 -0.292048  0.50645
6
PTRIO  0.289946 -0.391679  0.383248 -0.121515  0.188933 -0.355501  0.26151
5
B     -0.385064  0.175520 -0.356977  0.048788 -0.380051  0.128069 -0.27353
4
LSTAT  0.455621 -0.412995  0.603800 -0.053929  0.590879 -0.613808  0.60233
9
MEDV  -0.388305  0.360445 -0.483725  0.175260 -0.427321  0.695360 -0.37695
5
```

In [ ]:

```
            DIS       RAD       TAX     PTRIO         B     LSTAT      MED
V
CRIM  -0.379670  0.625505  0.582764  0.289946 -0.385064  0.455621 -0.38830
5
ZN     0.664408 -0.311948 -0.314563 -0.391679  0.175520 -0.412995  0.36044
5
INDUS -0.708027  0.595129  0.720760  0.383248 -0.356977  0.603800 -0.48372
5
CHAS  -0.099176 -0.007368 -0.035587 -0.121515  0.048788 -0.053929  0.17526
0
NOX   -0.769230  0.611441  0.668023  0.188933 -0.380051  0.590879 -0.42732
1
RM     0.205246 -0.209847 -0.292048 -0.355501  0.128069 -0.613808  0.69536
0
AGE   -0.747881  0.456022  0.506456  0.261515 -0.273534  0.602339 -0.37695
5
DIS    1.000000 -0.494588 -0.534432 -0.232471  0.291512 -0.496996  0.24992
9
RAD   -0.494588  1.000000  0.910228  0.464741 -0.444413  0.488676 -0.38162
6
TAX   -0.534432  0.910228  1.000000  0.460853 -0.441808  0.543993 -0.46853
6
PTRIO -0.232471  0.464741  0.460853  1.000000 -0.177383  0.374044 -0.50778
7
B      0.291512 -0.444413 -0.441808 -0.177383  1.000000 -0.366087  0.33346
1
LSTAT -0.496996  0.488676  0.543993  0.374044 -0.366087  1.000000 -0.73766
3
```

```
MEDV    0.249929 -0.381626 -0.468536 -0.507787  0.333461 -0.737663  1.00000
0
```

In [21]:

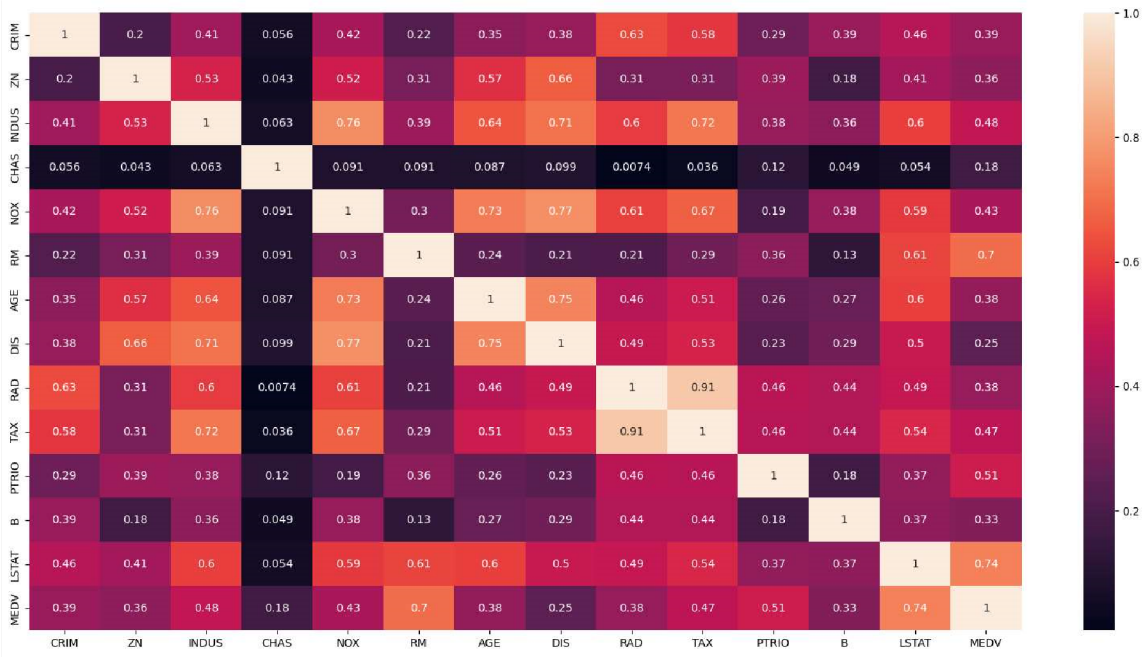
```
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [23]:

```
plt.figure(figsize=(20,10))
sns.heatmap(data.corr().abs(),annot=True)
```

Out[23]:

```
<Axes: >
```



In [ ]:

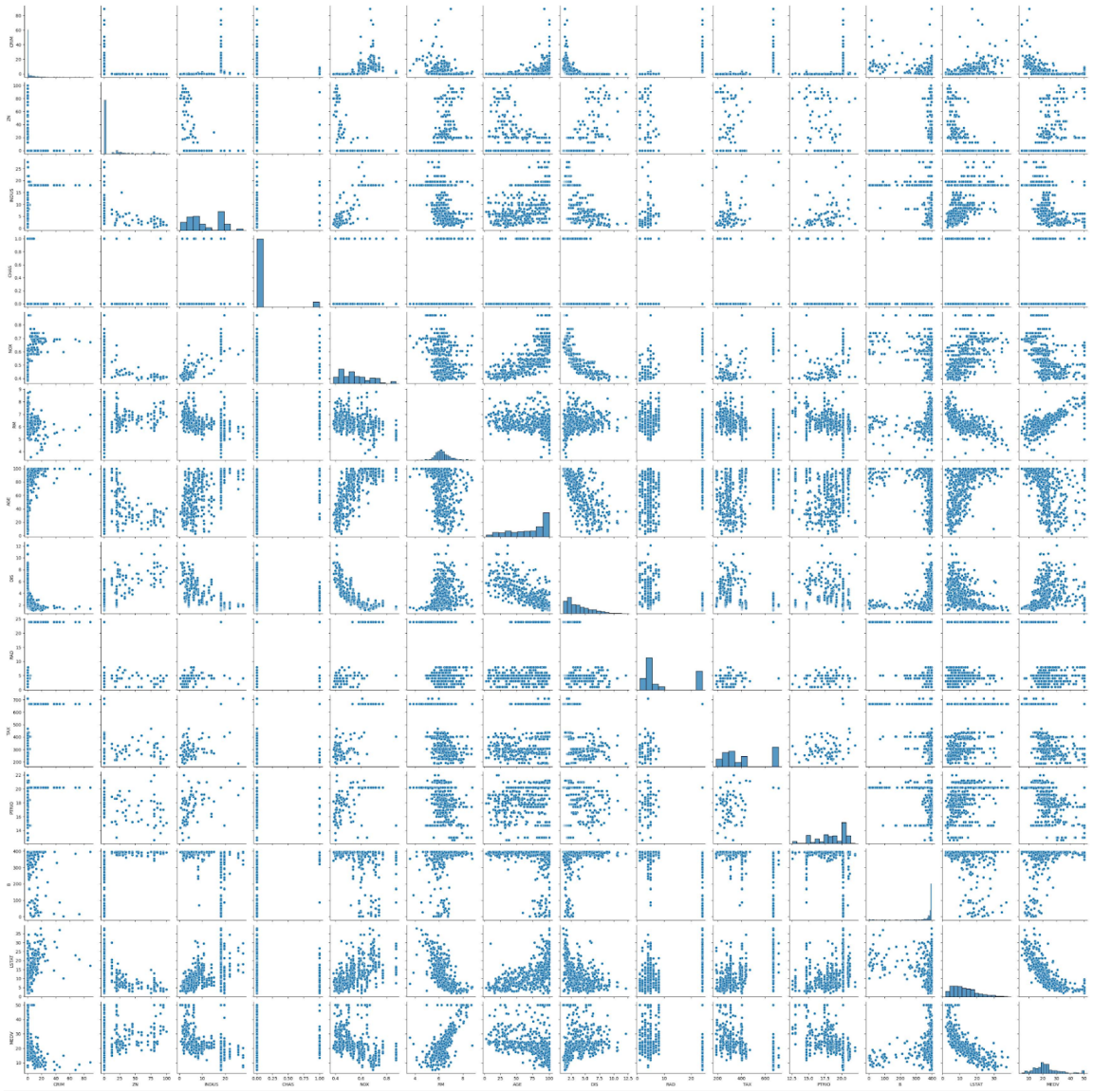
```
```

24

```
sns.pairplot(data)
```

Out[24]:

```
<seaborn.axisgrid.PairGrid at 0x229333dd660>
```



In [26]:

```
X=data.drop('MEDV',axis=1)
```

In [27]:

```
Y=data['MEDV']
```

In [ ]:

28

```
X
```

Out[28]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRIO	B	L
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296.0	15.3	396.90	
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242.0	17.8	396.90	
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242.0	17.8	392.83	
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222.0	18.7	394.63	
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222.0	18.7	396.90	
...	...	...	...	...	...	...	...	...	...	...	...	...	
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273.0	21.0	391.99	
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273.0	21.0	396.90	
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273.0	21.0	396.90	
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273.0	21.0	393.45	
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273.0	21.0	396.90	

506 rows × 13 columns

In [29]:

```
Y
```

Out[29]:

```
0      24.0
1      21.6
2      34.7
3      33.4
4      36.2
       ...
501    22.4
502    20.6
503    23.9
504    22.0
505    11.9
Name: MEDV, Length: 506, dtype: float64
```

In [31]:

```
from sklearn.model_selection import train_test_split
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.3,random_state=42)
```

In [32]:

```
from sklearn.linear_model import LinearRegression
```

In [ ]:

```
```

33

```
model =LinearRegression()
```

In [34]:

```
model.fit(X_train,Y_train)
```

Out[34]:

```
LinearRegression()
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [35]:

```
Y_predict=model.predict(X_test)
```

In [37]:

```
print(model.intercept_)
```

```
31.63108403569189
```

In [39]:

```
print(model.coef_)
```

```
[-1.33470103e-01  3.58089136e-02  4.95226452e-02  3.11983512e+00
 -1.54170609e+01  4.05719923e+00 -1.08208352e-02 -1.38599824e+00
  2.42727340e-01 -8.70223437e-03 -9.10685208e-01  1.17941159e-02
 -5.47113313e-01]
```

In [41]:

```
coef
```

```
---------------------------------------------------------------------------
-
NameError                                 Traceback (most recent call las
t)
Cell In[41], line 1
----> 1 coef

NameError: name 'coef' is not defined
```

In [ ]:

```

```

42

```
data.head(5)
```

Out[42]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRIO	B	LS
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296.0	15.3	396.90	4
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242.0	17.8	396.90	9
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242.0	17.8	392.83	4
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222.0	18.7	394.63	2
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222.0	18.7	396.90	5

In [43]:

```
model.predict([[0.02729,0.0,7.07,0,0.469,7.185,61.1,4.9671,2,242.0,17.8,392.83,4.03]])
```

```
C:\ProgramData\anaconda3\lib\site-packages\sklearn\base.py:420: UserWarnin
g: X does not have valid feature names, but LinearRegression was fitted wi
th feature names
  warnings.warn(
```

Out[43]:

```
array([30.94992991])
```

In [44]:

```
model.predict([[0.06905,0.0,2.18,0,0.458,7.147,54.2,6.0622,3,222.0,18.7,396.90,5.33]])
```

```
C:\ProgramData\anaconda3\lib\site-packages\sklearn\base.py:420: UserWarnin
g: X does not have valid feature names, but LinearRegression was fitted wi
th feature names
  warnings.warn(
```

Out[44]:

```
array([28.20837173])
```

In [45]:

```
import sklearn import metrics
```

```
  Cell In[45], line 1
    import sklearn import metrics
                   ^
SyntaxError: invalid syntax
```

In [ ]:

```
```

In [46]:

```
from sklearn import metrics
```

```
    48
```

```
print('MAE:',metrics.mean_absolute_error(Y_test,Y_predict))
```

```
MAE: 3.1627098714574355
```

In [49]:

```
print('MSE:',metrics.mean_squared_error(Y_test,Y_predict))
```

```
MSE: 21.5174442311775
```

In [51]:

```
from sklearn.metrics import confusion_matrix, f1_score, roc_curve, auc
```

In [ ]:

```
```

In [ ]:

```
```

In [ ]:

```
```