```
import pandas as pd
from sklearn.linear_model import LinearRegression
import matplotlib.pyplot as ply

data = pd.read_csv("/content/HousingData.csv")

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

```
CRIM       20
ZN         20
INDUS      20
CHAS       20
NOX         0
RM          0
AGE        20
DIS         0
RAD         0
TAX         0
PTRATIO     0
B           0
LSTAT      20
MEDV        0
dtype: int64
```

```
data.fillna({"CRIM":(data["CRIM"].mean())},inplace=True)

data.fillna({"ZN":(data["ZN"].mean())},inplace=True)
data.fillna({"INDUS":(data["INDUS"].mean())},inplace=True)
data.fillna({"LSTAT":(data["LSTAT"].mean())},inplace=True)
data.fillna({"AGE":(data["AGE"].mean())},inplace=True)




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

```
CRIM        0
ZN          0
INDUS       0
CHAS       20
NOX         0
RM          0
AGE         0
DIS         0
RAD         0
TAX         0
PTRATIO     0
B           0
LSTAT       0
```

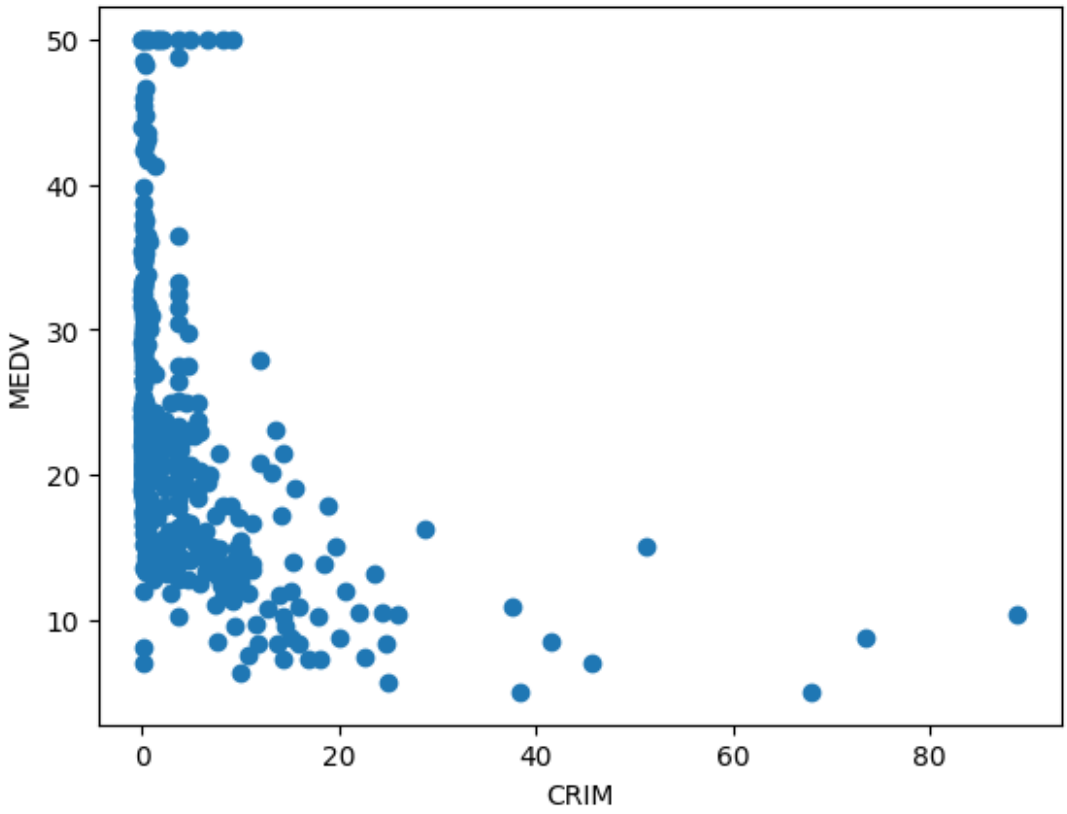
```
MEDV        0
dtype: int64
```

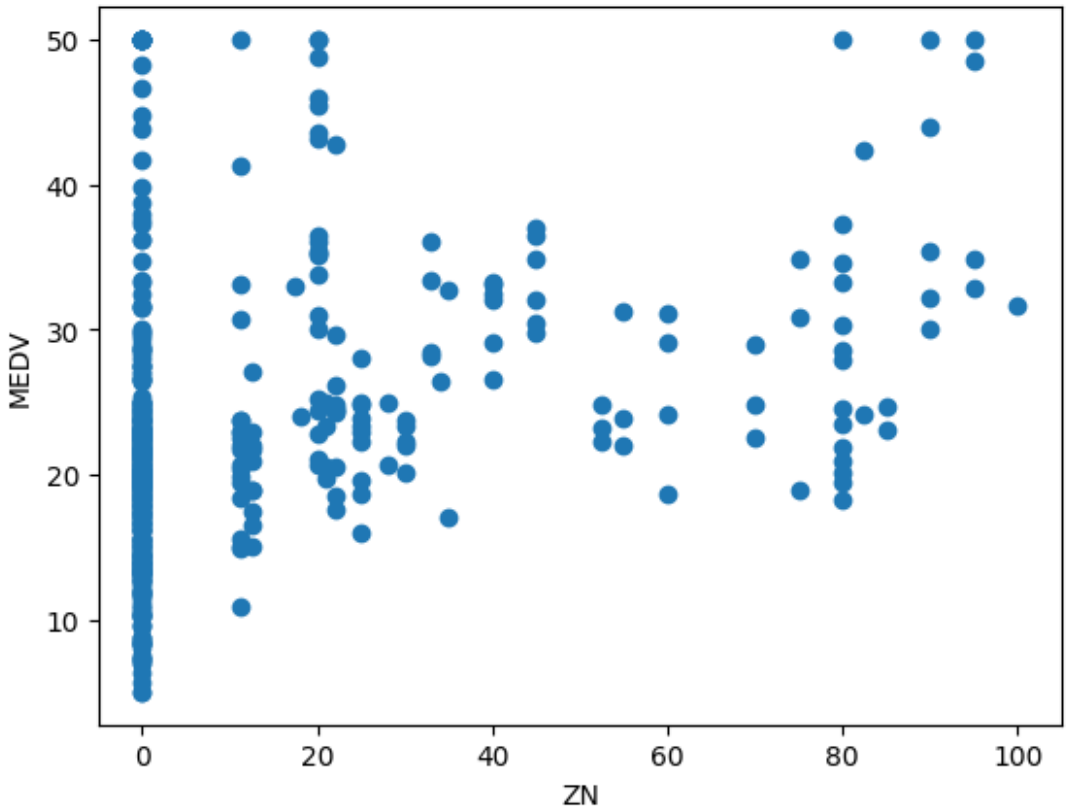
```
X = data.drop(["MEDV","CHAS"],axis='columns')
Y = data["MEDV"]

model = LinearRegression()
from sklearn.model_selection import train_test_split
X_train,X_test,Y_train,Y_test =
train_test_split(X,Y,test_size=0.2,random_state=20)
model.fit(X_train,Y_train)
from sklearn.metrics import mean_squared_error

model.predict(X_test)
print(mean_squared_error(Y_test,model.predict(X_test)))
model.score(X_test,Y_test)
```

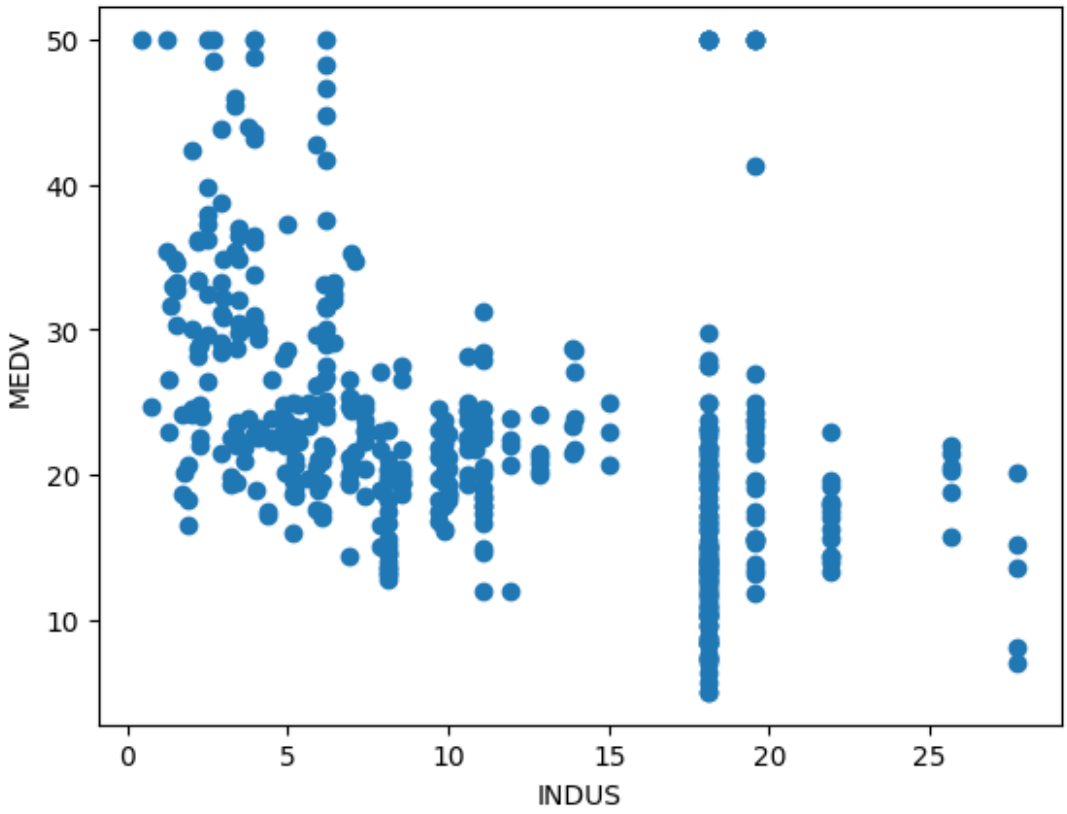
```
17.375922218514397
```

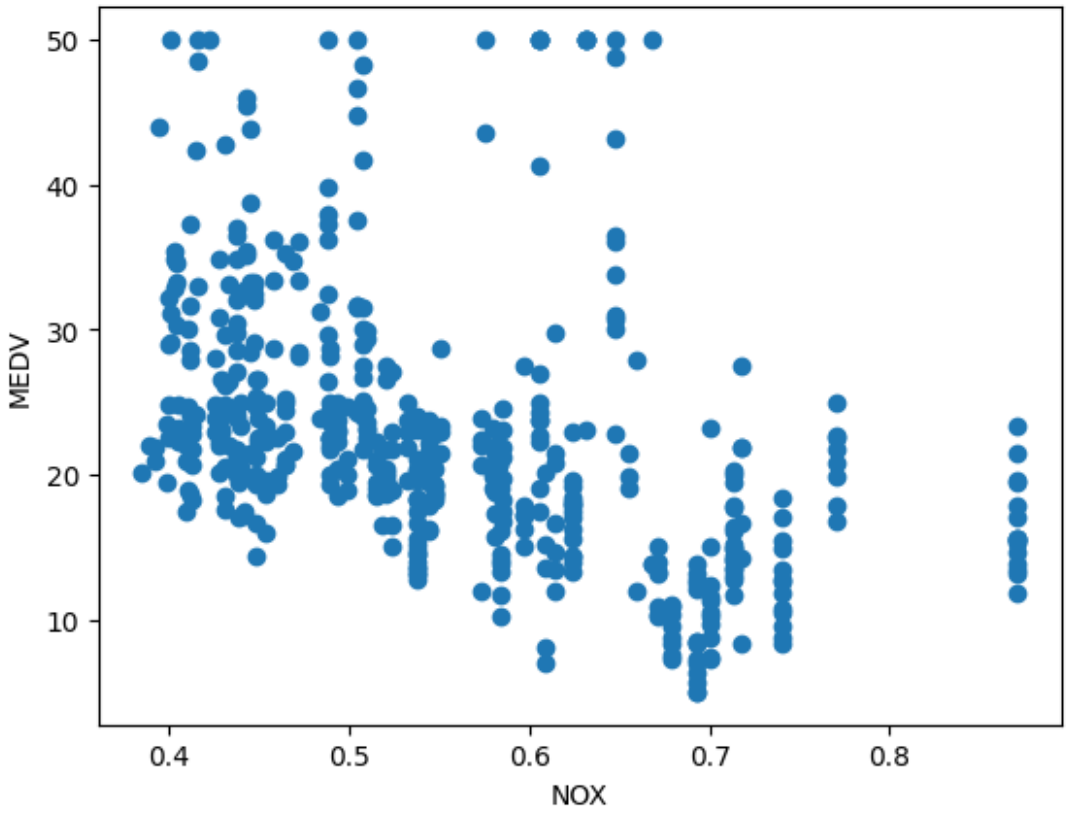
```
0.7302103217000235
```

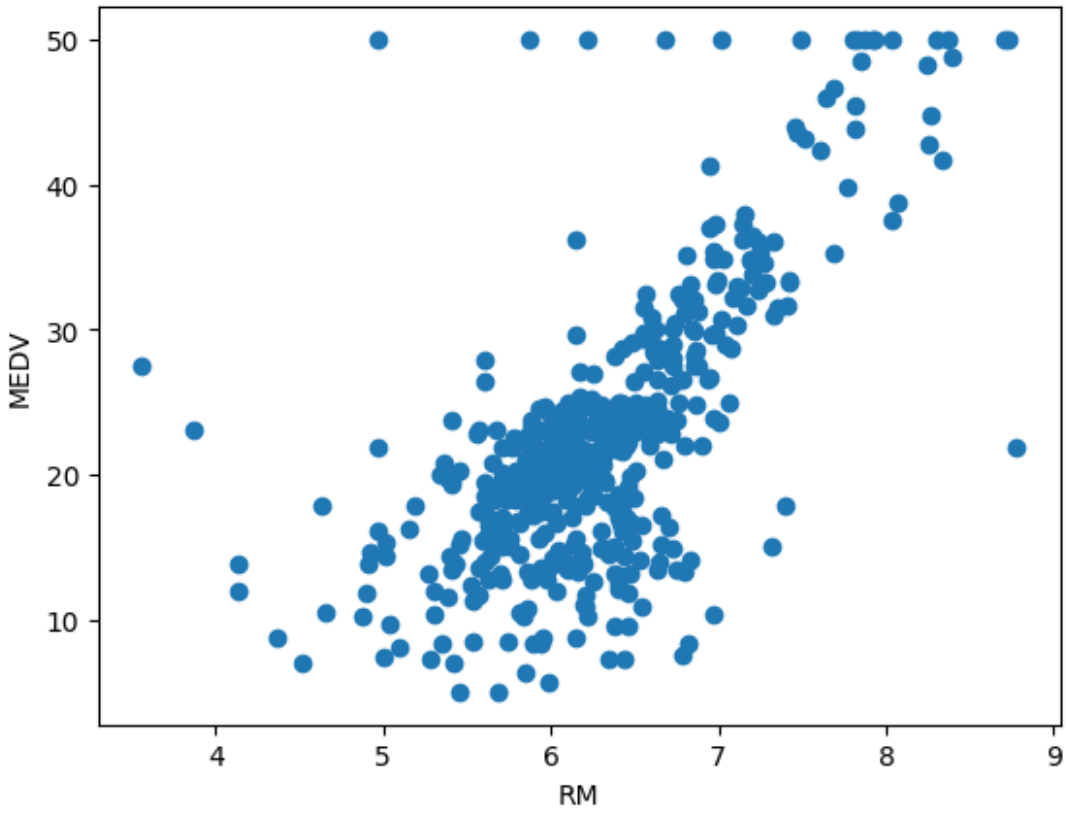
```
import matplotlib.pyplot as plt
for ele in X.columns:
  plt.scatter(X[ele],Y)
  plt.xlabel(ele)
  plt.ylabel("MEDV")
  plt.show()
```

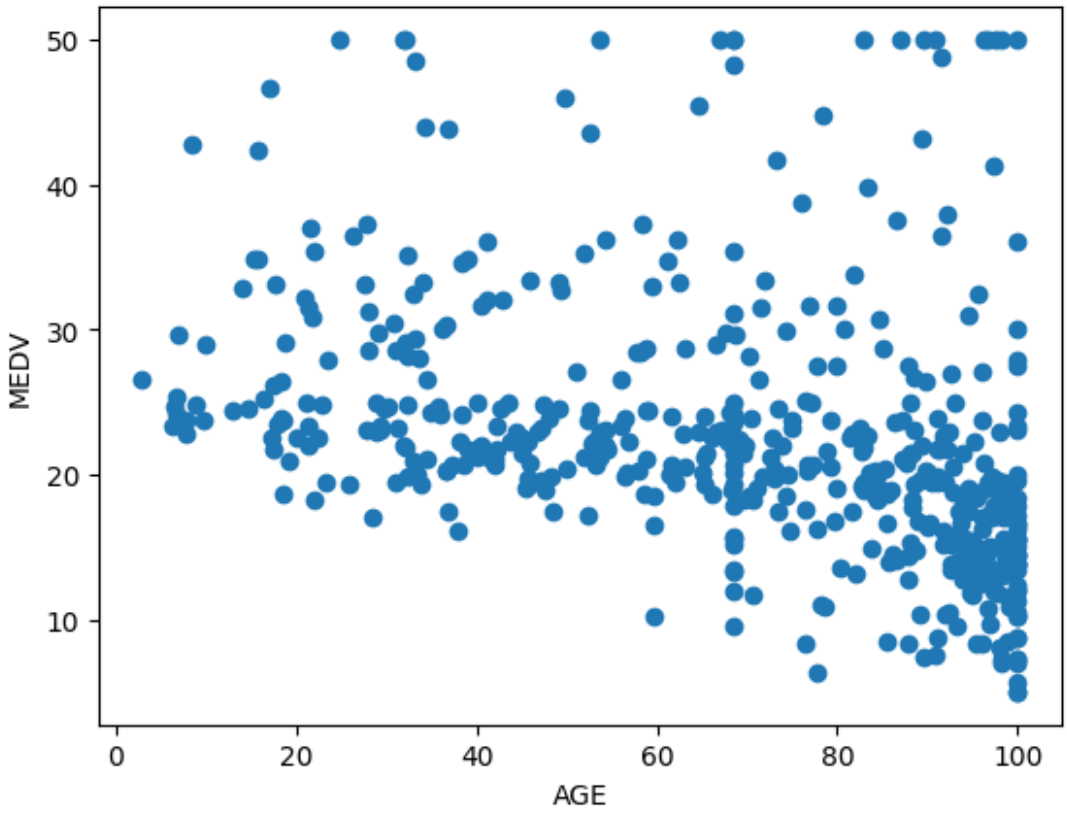

MEDV
CRIM
50
40
30
20
10
0
20
40
60
80

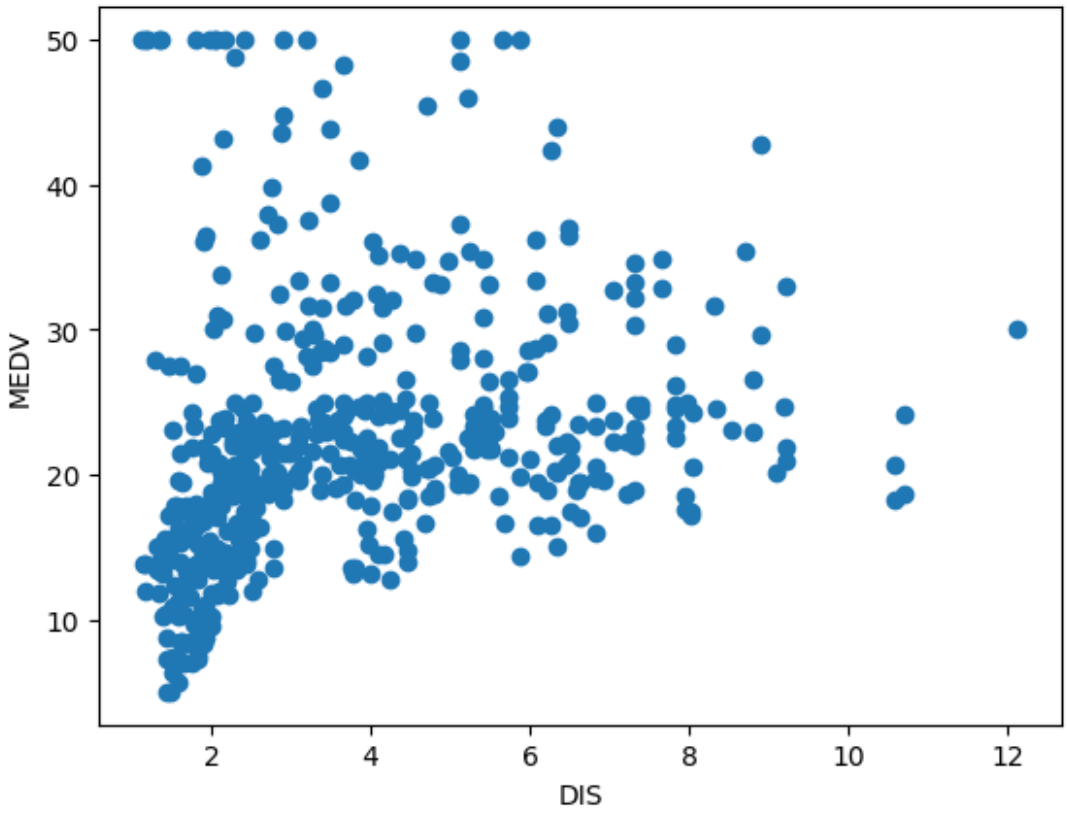

MEDV
ZN
50
40
30
20
10
0
20
40
60
80
100

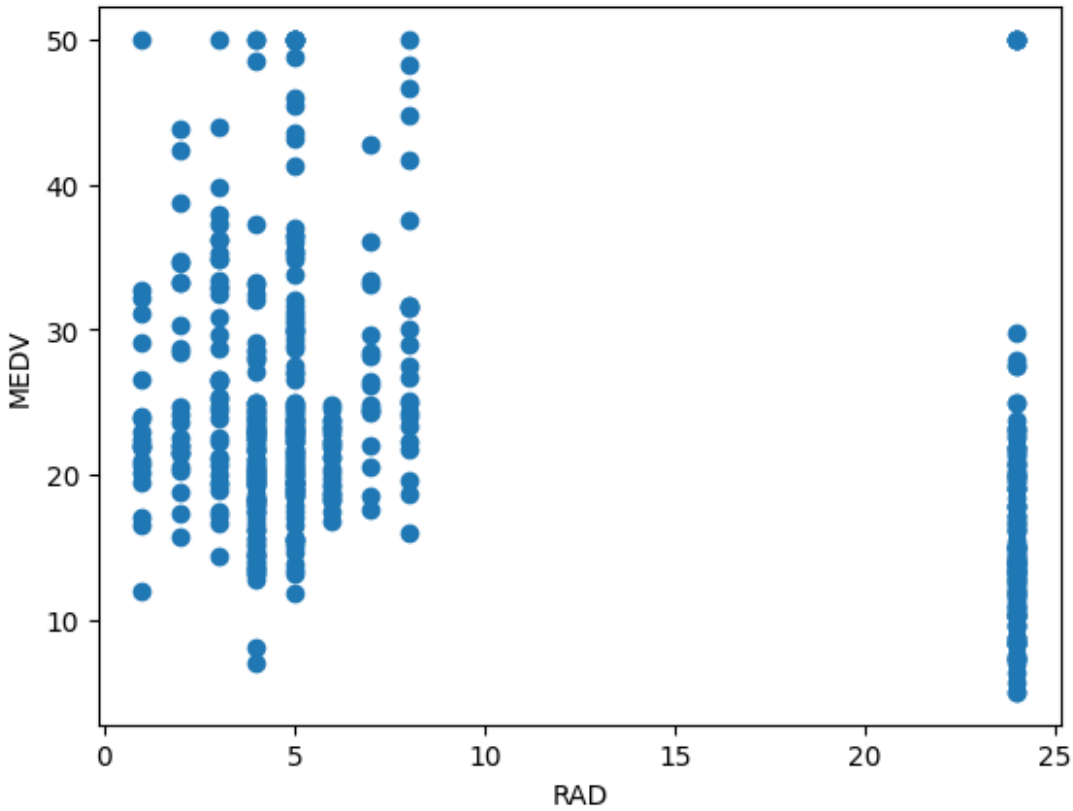

MEDV
INDUS
50
40
30
20
10
0
5
10
15
20
25

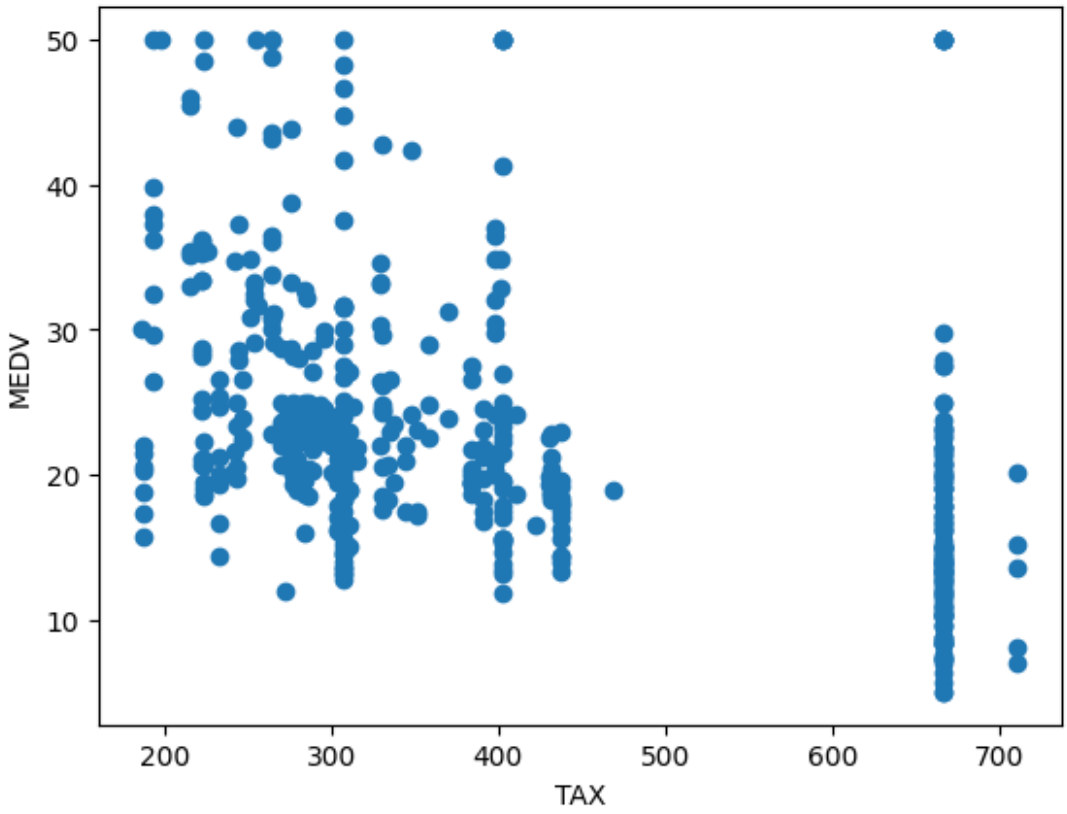

MEDV
NOX
50
40
30
20
10
0.4
0.5
0.6
0.7
0.8

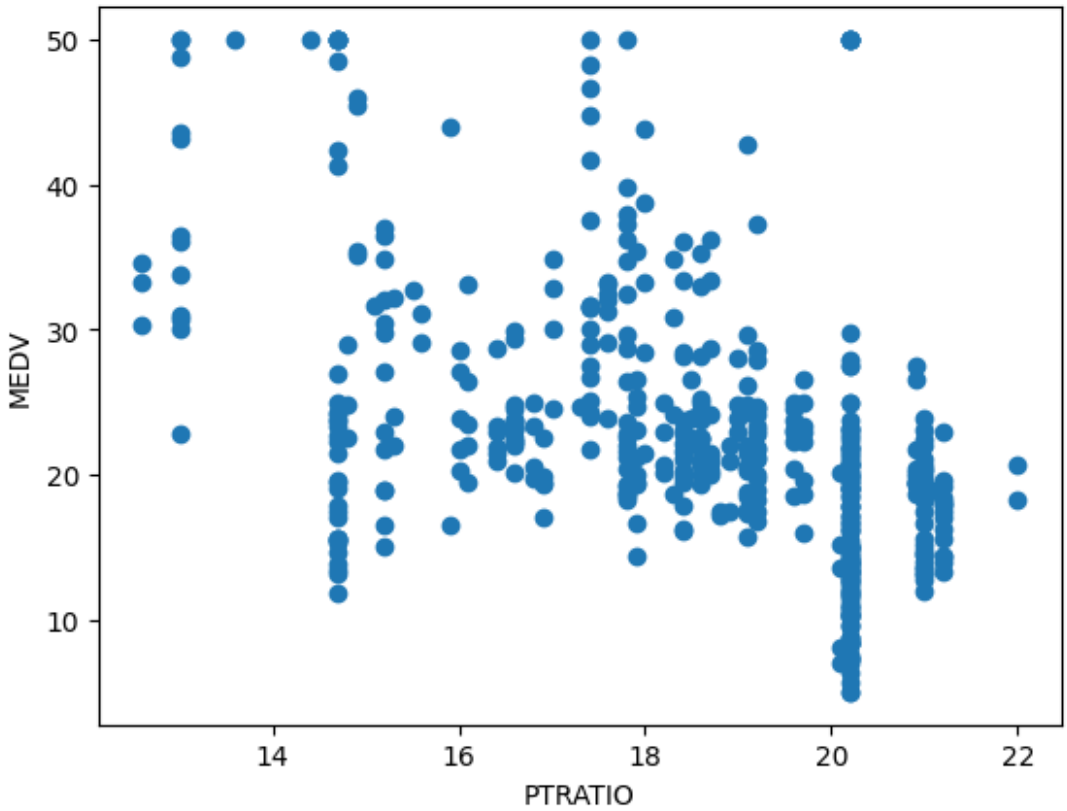

MEDV
RM
50
40
30
20
10
4
5
6
7
8
9

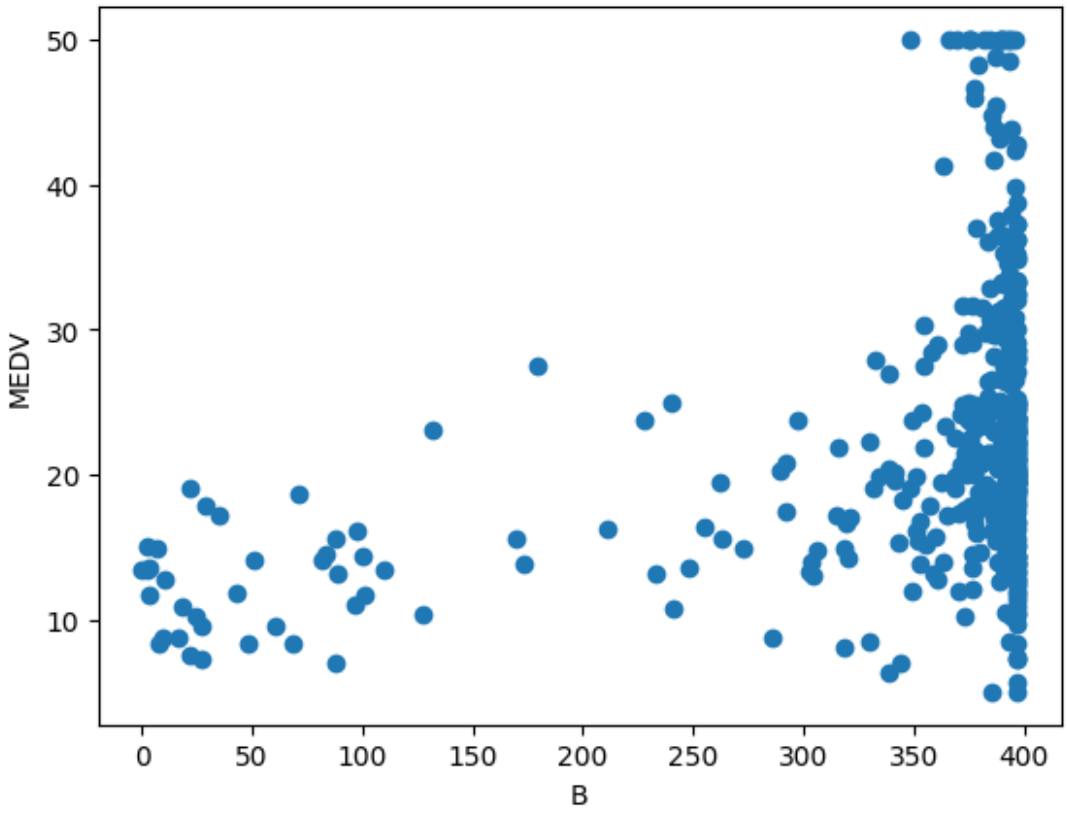

MEDV
AGE
50
40
30
20
10
0
20
40
60
80
100

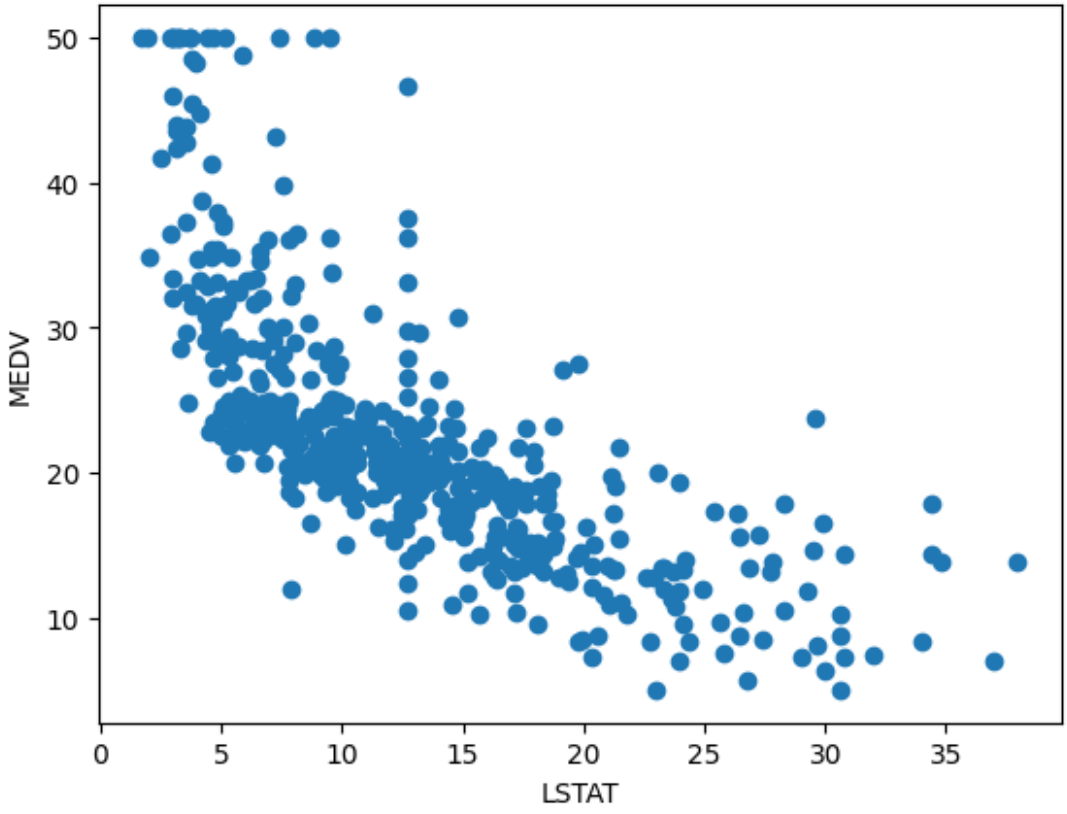

MEDV
DIS
50
40
30
20
10
2
4
6
8
10
12


MEDV
RAD
50
40
30
20
10
0
5
10
15
20
25


MEDV
50
40
30
20
10
200
300
400
500
600
700
TAX


MEDV
50
40
30
20
10
14
16
18
20
22
PTRATIO


MEDV
B
50
40
30
20
10
0
50
100
150
200
250
300
350
400


MEDV
LSTAT
50
40
30
20
10
0
5
10
15
20
25
30
35

```
y_pred = model.predict(X_test)

from sklearn.metrics import mean_squared_error, mean_absolute_error

mse = mean_squared_error(Y_test, y_pred)
print(f"Mean Squared Error: {mse}")

# Calculating Mean Absolute Error (MAE)
mae = mean_absolute_error(Y_test, y_pred)
print(f"Mean Absolute Error: {mae}")
```

```
Mean Squared Error: 17.375922218514397
Mean Absolute Error: 3.1050903865274027
```