Vidyavardhini's
College of Engineering & Technology
Vasai Road (W)

Department of Computer Engineering

Laboratory Manual
Student Copy

Semester	VII	Class	B.E
Course Code	CSL701		
Course Name	Machine Learning Lab		

Vidyavardhini's College of Engineering & Technology

Vision

To be a premier institution of technical education; always aiming at becoming a valuable resource for industry and society.

Mission

- To provide technologically inspiring environment for learning.
- To promote creativity, innovation and professional activities.
- To inculcate ethical and moral values.
- To cater personal, professional and societal needs through quality education.

Department Vision:

To evolve as a center of excellence in the field of Computer Engineering to cater to industrial and societal needs.

Department Mission:

- To provide quality technical education with the aid of modern resources.
- Inculcate creative thinking through innovative ideas and project development.
- To encourage life-long learning, leadership skills, entrepreneurship skills with ethical & moral values.

Program Education Objectives (PEOs):

PEO1: To facilitate learners with a sound foundation in the mathematical, scientific and engineering fundamentals to accomplish professional excellence and succeed in higher studies in Computer Engineering domain

PEO2: To enable learners to use modern tools effectively to solve real-life problems in the field of Computer Engineering.

PEO3: To equip learners with extensive education necessary to understand the impact of computer technology in a global and social context.

PEO4: To inculcate professional and ethical attitude, leadership qualities, commitment to societal responsibilities and prepare the learners for life-long learning to build up a successful career in Computer Engineering.

Program Specific Outcomes (PSOs):

PSO1: Analyze problems and design applications of database, networking, security, web technology, cloud computing, machine learning using mathematical skills, and computational tools.

PSO2: Develop computer-based systems to provide solutions for organizational, societal problems by working in multidisciplinary teams and pursue a career in the IT industry.

Program Outcomes (POs):

Engineering Graduates will be able to:

- **PO1. Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- **PO2. Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- **PO3. Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- **PO4. Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- **PO5. Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- **PO6. The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- **PO7. Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- **PO8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- **PO9. Individual and teamwork:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- **PO10. Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- **PO11. Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- **PO12. Life-long learning:** Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Course Objectives

1	To introduce the basic concepts and techniques of Machine Learning.
2	To introduce various supervised and unsupervised algorithms.
3	To introduce various ensemble techniques for combining ML models.
4	To introduce the concept of dimensionality reduction and its techniques.

Course Outcomes

CO	At the end of course students will be able to:	**Action verbs**	**Bloom's Level**
CSL701.1	Analyze the data and apply appropriate Regression Technique on the given Dataset	Analyze, Apply	Analyze (level 4)
CSL701.2	Analyze the results obtained by applying appropriate Classification Technique on the given Dataset	Analyze	Analyze (level 4)
CSL701.3	Analyze the results obtained by applying appropriate Ensemble Technique on the given Dataset	Analyze	Analyze (level 4)
CSL701.4	Apply appropriate Unsupervised Technique on the given Dataset	Apply	Apply (level 3)
CSL701.5	Analyze the results obtained by applying Dimensionality Reduction on the given Dataset	Analyze	Analyze (level 4)
CSL701.6	Build a Machine Learning Application	Create	Create (level 6)

Mapping of Experiments with Course Outcomes

List of Experiments	Course Outcomes					
	CSL701.1	CSL701.2	CSL701.3	CSL701.4	CSL701.5	CSL701.6
Analyze the Boston Housing dataset and Apply appropriate Regression Technique	3	-	-	-	-	-
Analyze the Titanic Survival Dataset and Apply appropriate Regression Technique	3	-	-	-	-	-
Apply Decision Tree Algorithm on Adult Census Income Dataset and analyze the performance of the model	-	3	-	-	-	-
Apply Random Forest Algorithm on Adult Census Income Dataset and analyze the performance of the model	-	-	3	-	-	-
Apply appropriate Unsupervised Learning Technique on the Wholesale Customers Dataset	-	-	-	3	-	-
Apply Dimensionality Reduction on Adult Census Income Dataset and analyze the performance of the model	-	-	-	-	3	-

Mini – Project	-	-	-	-	-	3

INDEX

Sr. No.	Name of Experiment	D.O.P.	D.O.C.	Page No.	Remark
1	Analyze the Boston Housing dataset and Apply appropriate Regression Technique				
2	Analyze the Titanic Survival Dataset and Apply appropriate Regression Technique				
3	Apply Decision Tree Algorithm on Adult Census Income Dataset and analyze the performance of the model				
4	Apply Random Forest Algorithm on Adult Census Income Dataset and analyze the performance of the model				
5	Apply appropriate Unsupervised Learning Technique on the Wholesale Customers Dataset				
6	Apply Dimensionality Reduction on Adult Census Income Dataset and analyze the performance of the model				

D.O.P: Date of performance

D.O.C : Date of correction

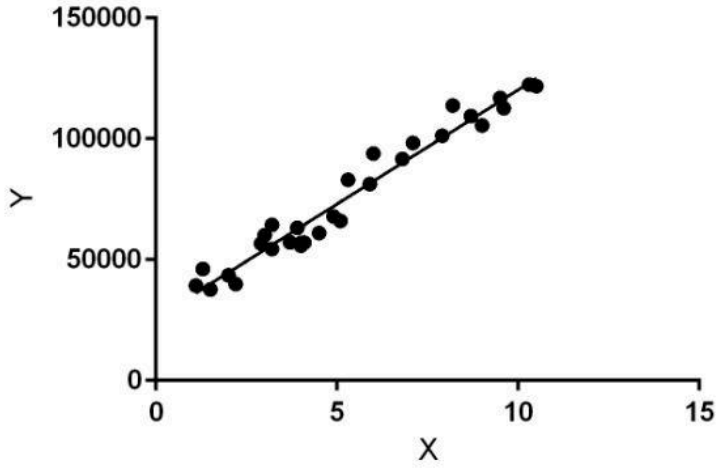
Experiment No. 1
Analyze the Boston Housing dataset and Apply appropriate Regression Technique
Date of Performance:21/07/24
Date of Submission:

Aim: Analyze the Boston Housing dataset and Apply appropriate Regression Technique.

Objective: Ablility to perform various feature engineering tasks, apply linear regression on the given dataset and minimise the error.

Theory:

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models a target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting. Different regression models differ based on – the kind of relationship between dependent and independent variables they are considering, and the number of independent variables getting used.



Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x). So, this regression technique finds out a linear relationship between x (input) and y(output). Hence, the name is Linear Regression.
In the figure above, X (input) is the work experience and Y (output) is the salary of a person. The regression line is the best fit line for our model.

Dataset:

The Boston Housing Dataset

The Boston Housing Dataset is a derived from information collected by the U.S. Census Service concerning housing in the area of Boston MA. The following describes the dataset columns:

CRIM - per capita crime rate by town
ZN - proportion of residential land zoned for lots over 25,000 sq.ft.
INDUS - proportion of non-retail business acres per town.
CHAS - Charles River dummy variable (1 if tract bounds river; 0 otherwise)
NOX - nitric oxides concentration (parts per 10 million)
RM - average number of rooms per dwelling
AGE - proportion of owner-occupied units built prior to 1940
DIS - weighted distances to five Boston employment centres
RAD - index of accessibility to radial highways
TAX - full-value property-tax rate per $10,000
PTRATIO - pupil-teacher ratio by town
B - 1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town
LSTAT - % lower status of the population
MEDV - Median value of owner-occupied homes in $1000's

CODE:

```
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())
```

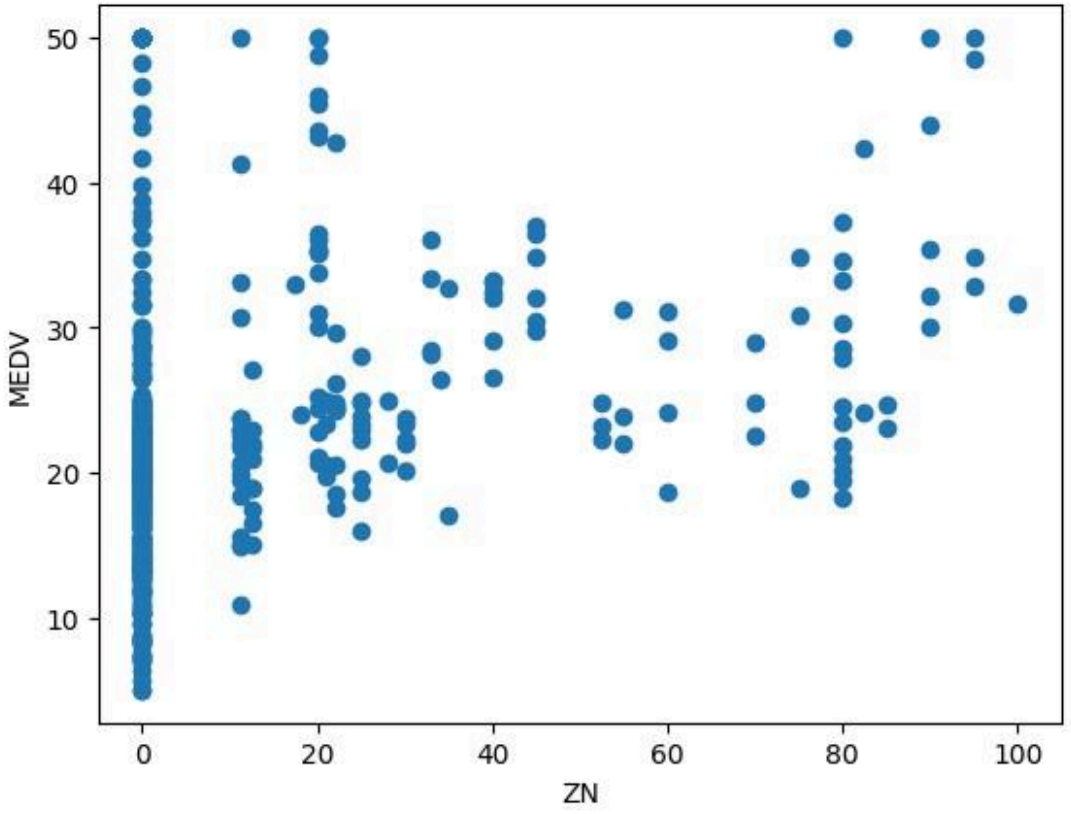
```
CRIM        0
ZN          0
INDUS       0
CHAS       20
NOX         0
RM          0
AGE         0
DIS         0
RAD         0
TAX         0
PTRATIO     0
B           0
LSTAT       0
```

```
MEDV        0 dtype:
int64
```

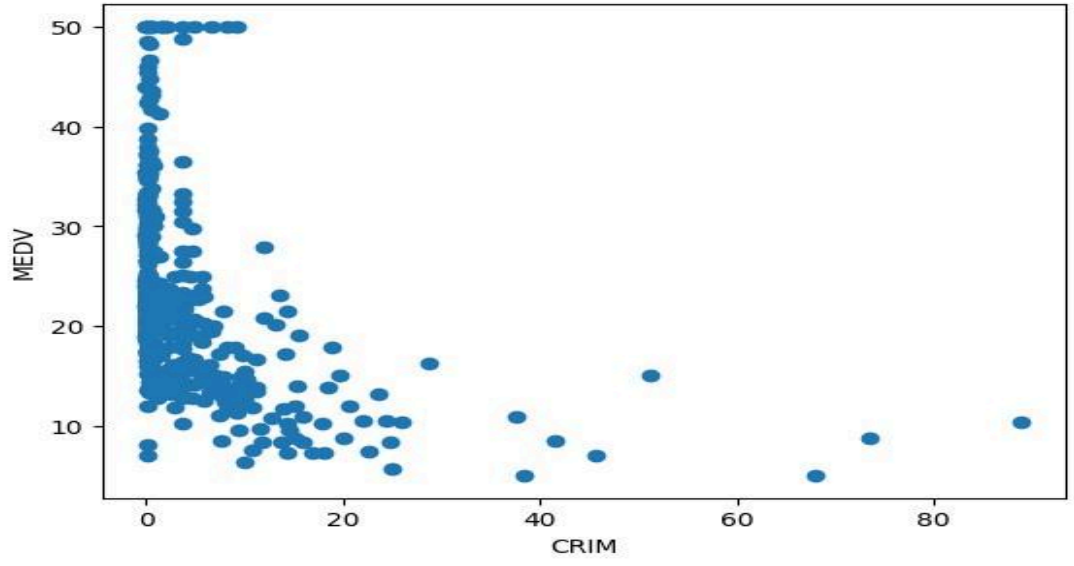
```
X = data.drop(["MEDV","CHAS"],axis='columns') Y
= data["MEDV"]
```

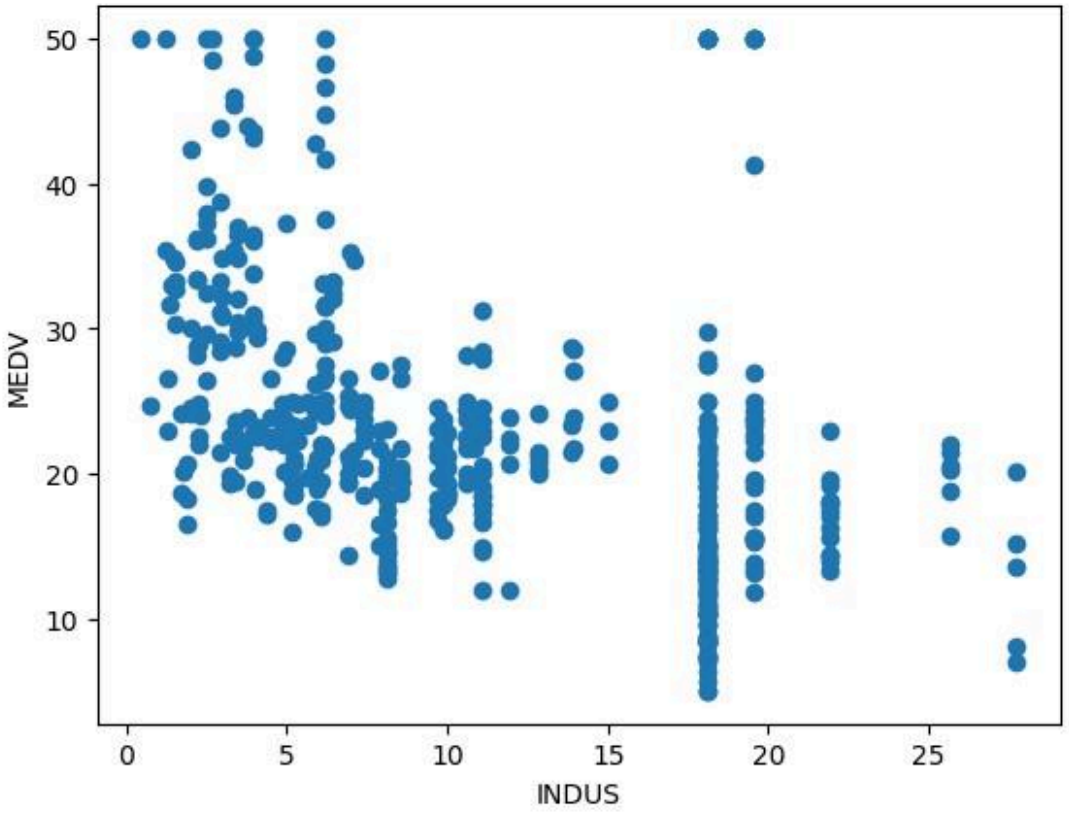
```
model = LinearRegression()
from sklearn.model_selection import train_test_split
X_train,X_test,Y_train,Y_test =
train_test_split(X,Y,test_size=0.2,random_state=20)
model.fit(X_train,Y_train)
from sklearn.metrics import mean_squared_error
```

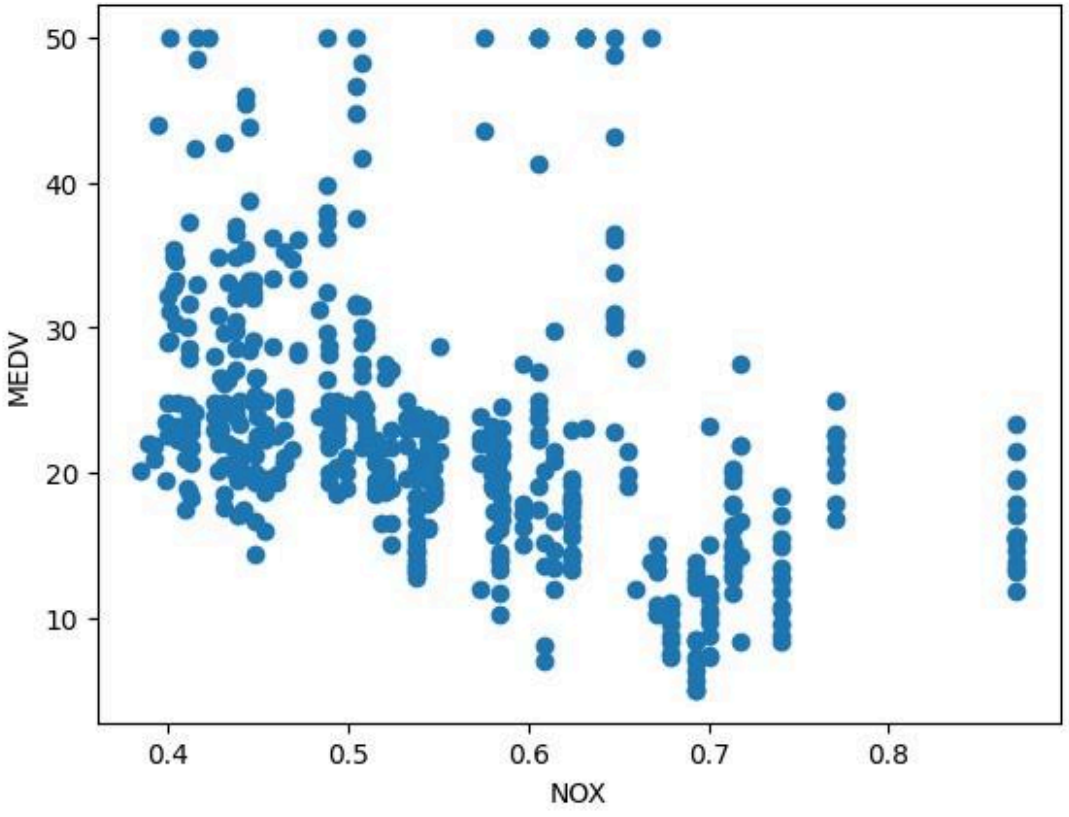
```
model.predict(X_test)
print(mean_squared_error(Y_test,model.predict(X_test)))
model.score(X_test,Y_test)
17.375922218514397
0.7302103217000235
import matplotlib.pyplot as plt
for ele in X.columns:
plt.scatter(X[ele],Y)
plt.xlabel(ele)
plt.ylabel("MEDV")   plt.show()
```

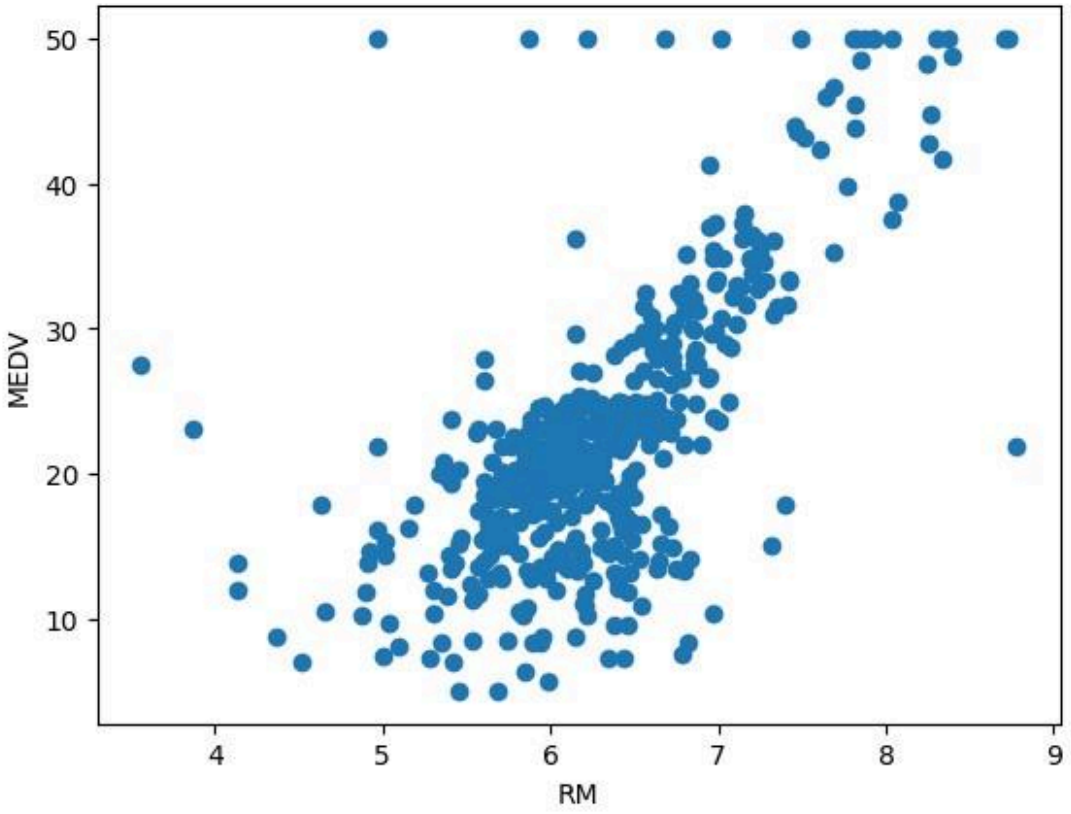

MEDV
ZN
50
40
30
20
10
0
20
40
60
80
100

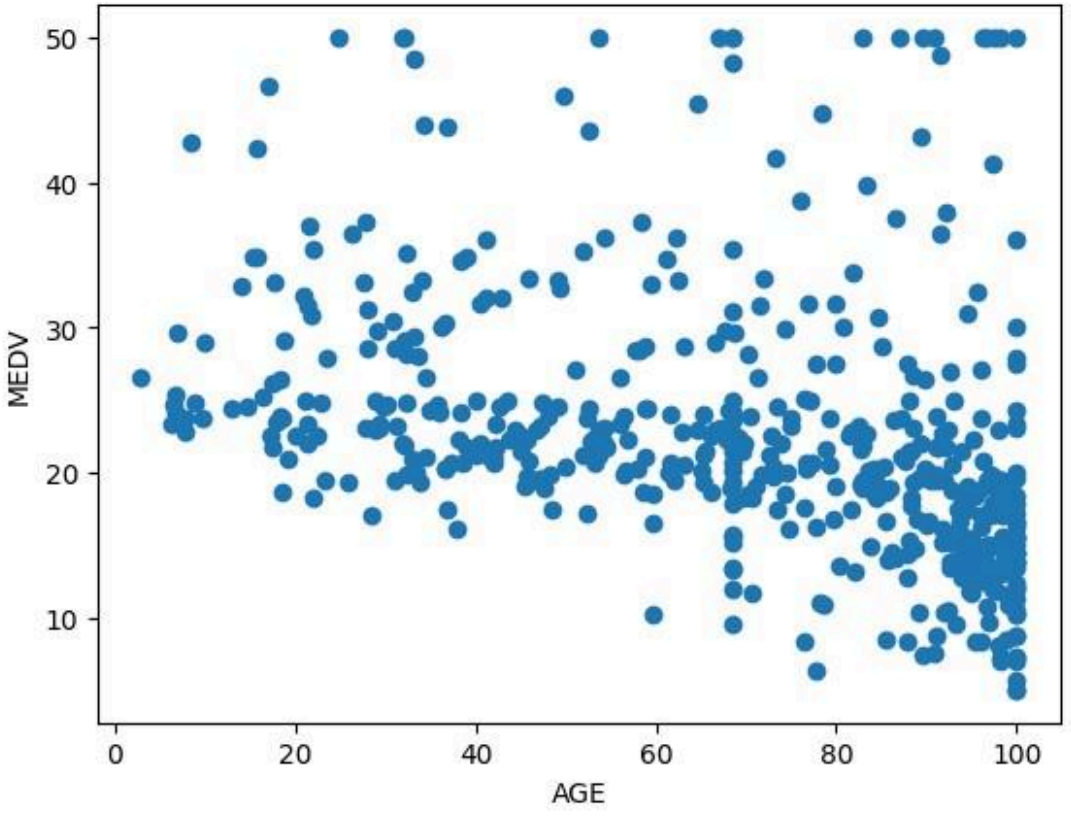
Conclusion:

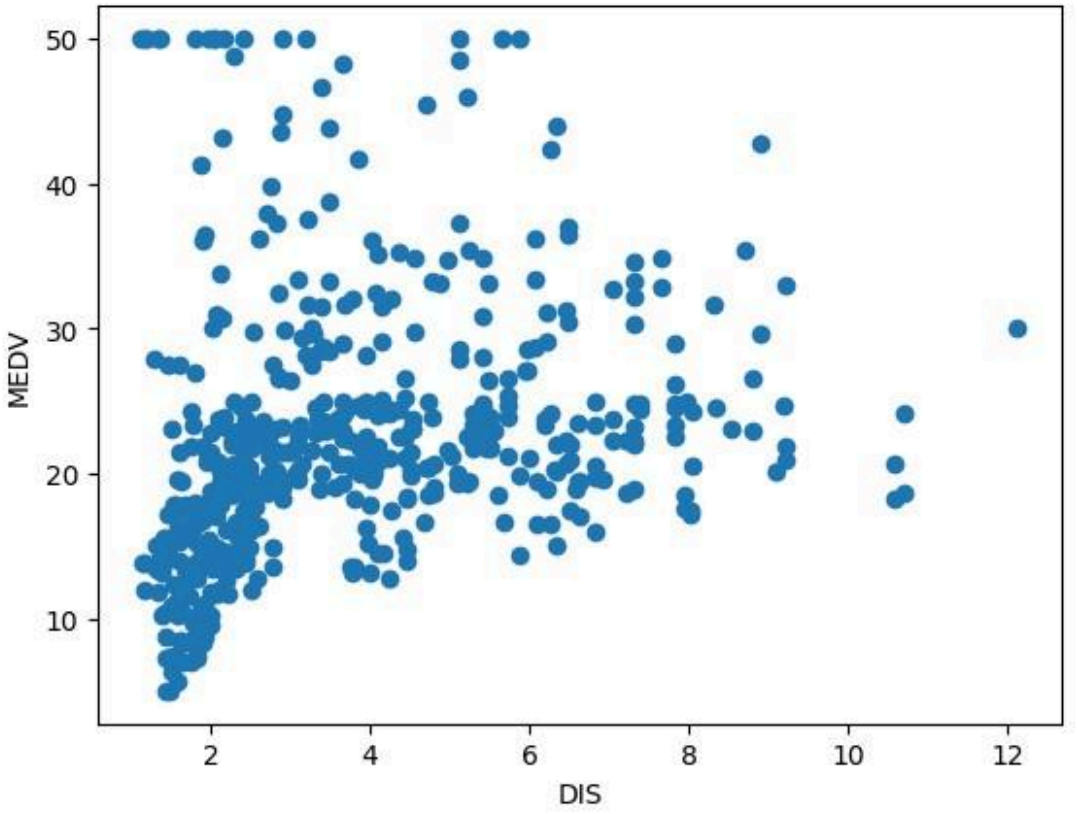

MEDV
CRIM
50
40
30
20
10
0
20
40
60
80

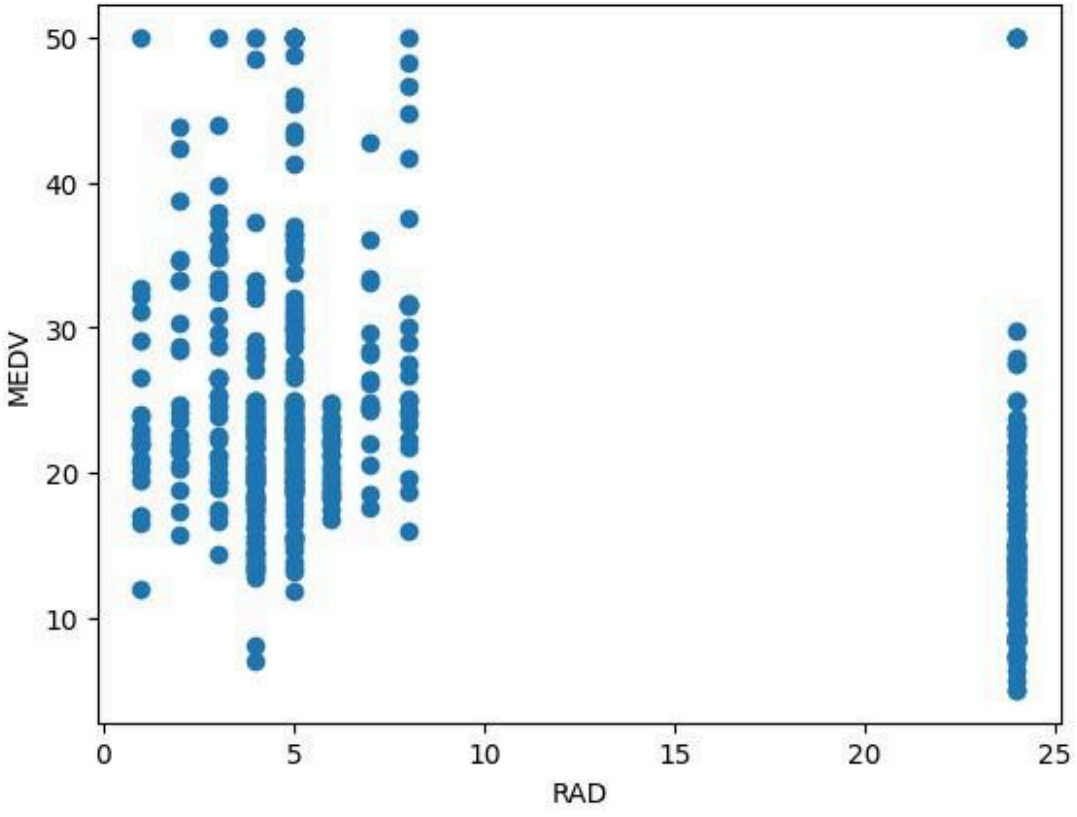

MEDV
INDUS
50
40
30
20
10
0
5
10
15
20
25

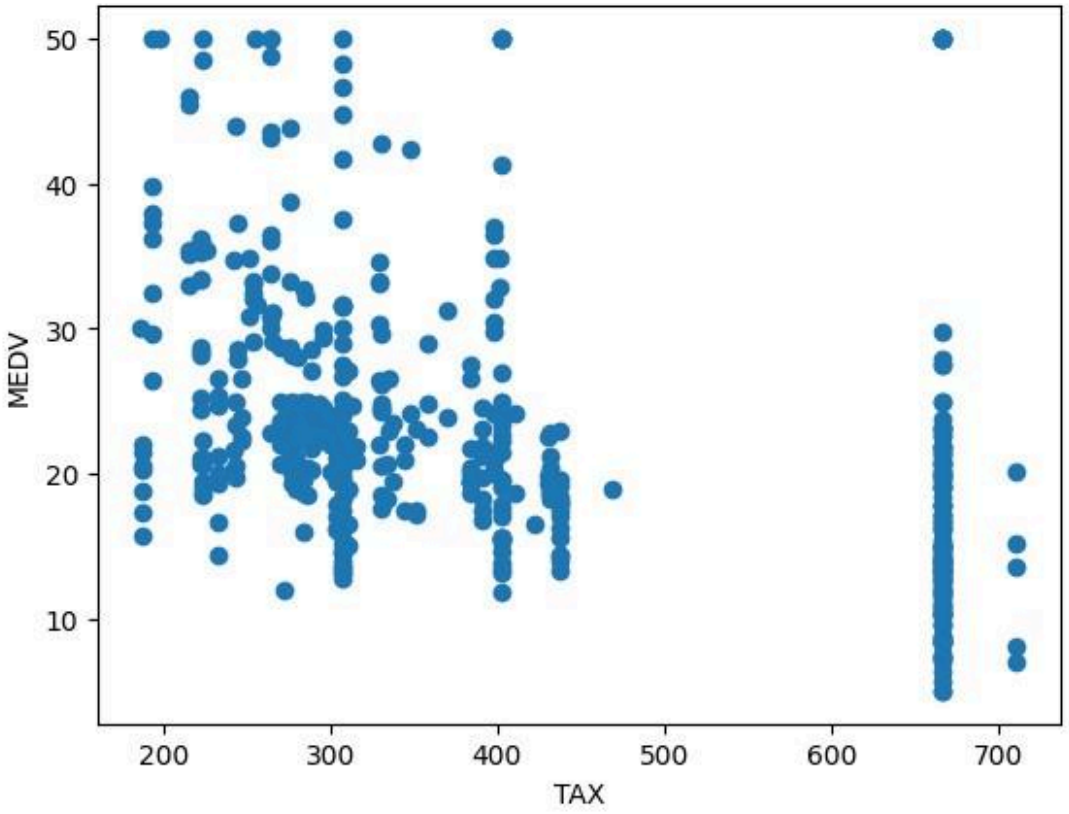

MEDV
NOX
50
40
30
20
10
0.4
0.5
0.6
0.7
0.8

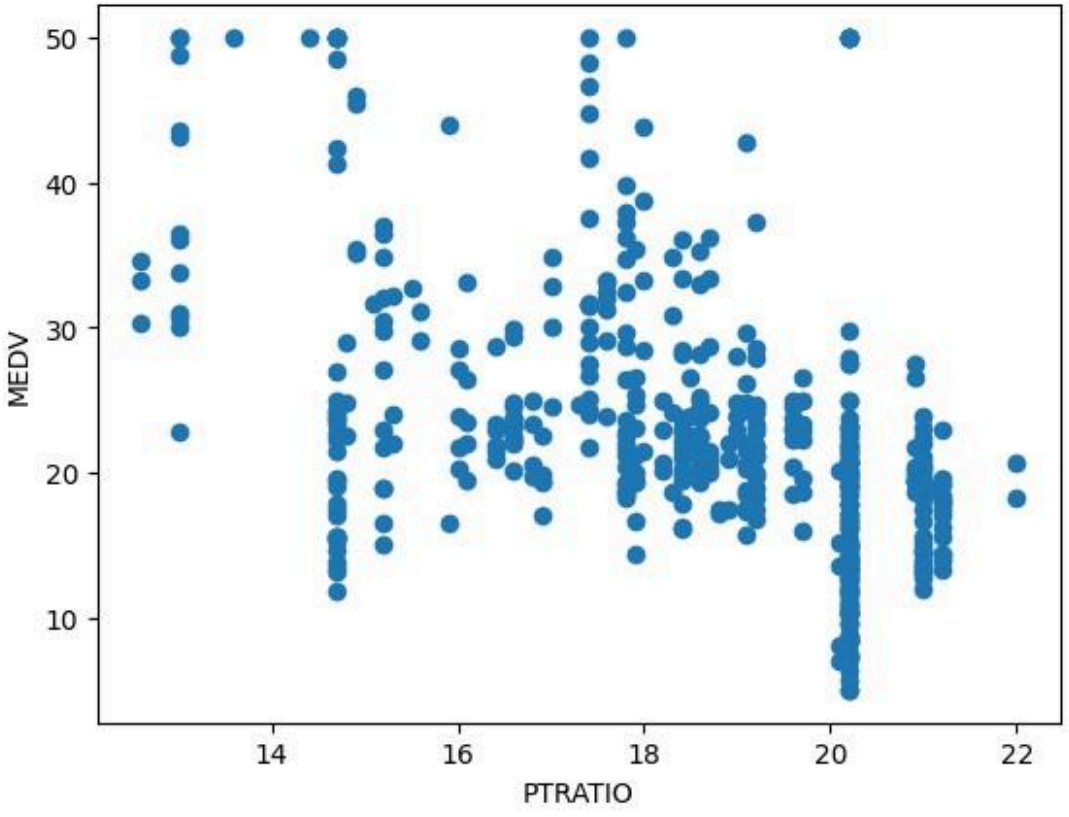

MEDV
RM
50
40
30
20
10
4
5
6
7
8
9

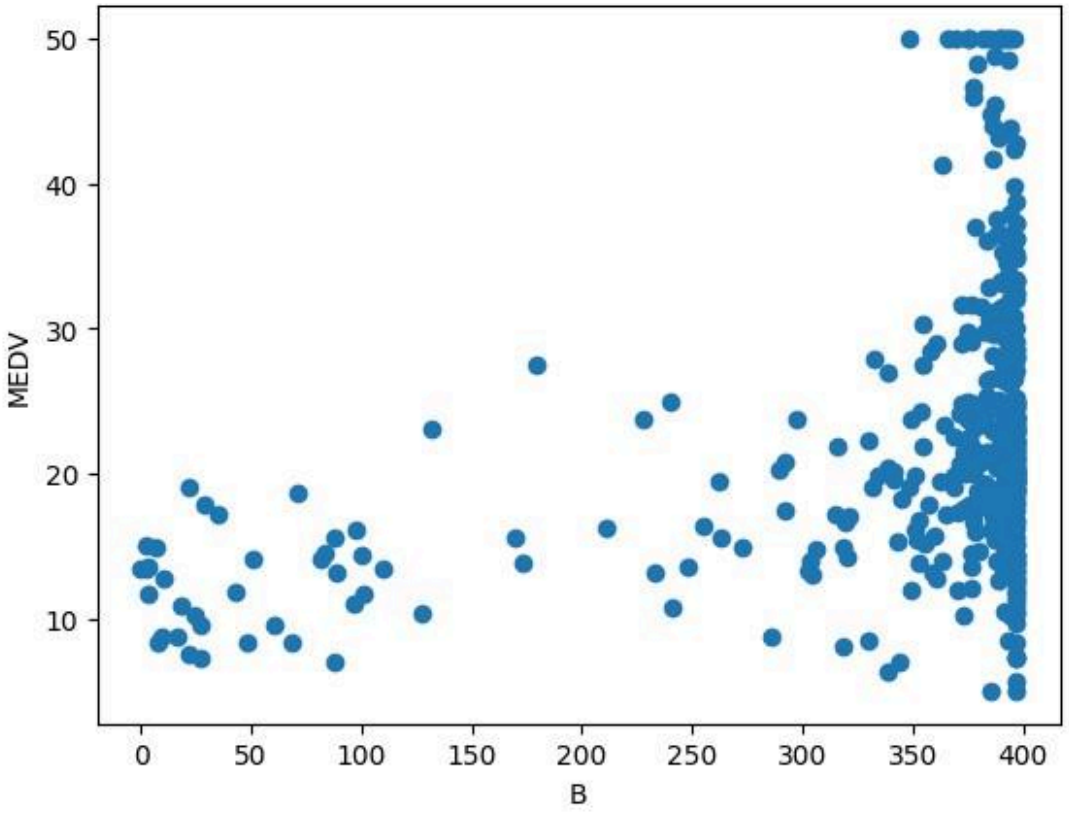

MEDV
AGE
50
40
30
20
10
0
20
40
60
80
100

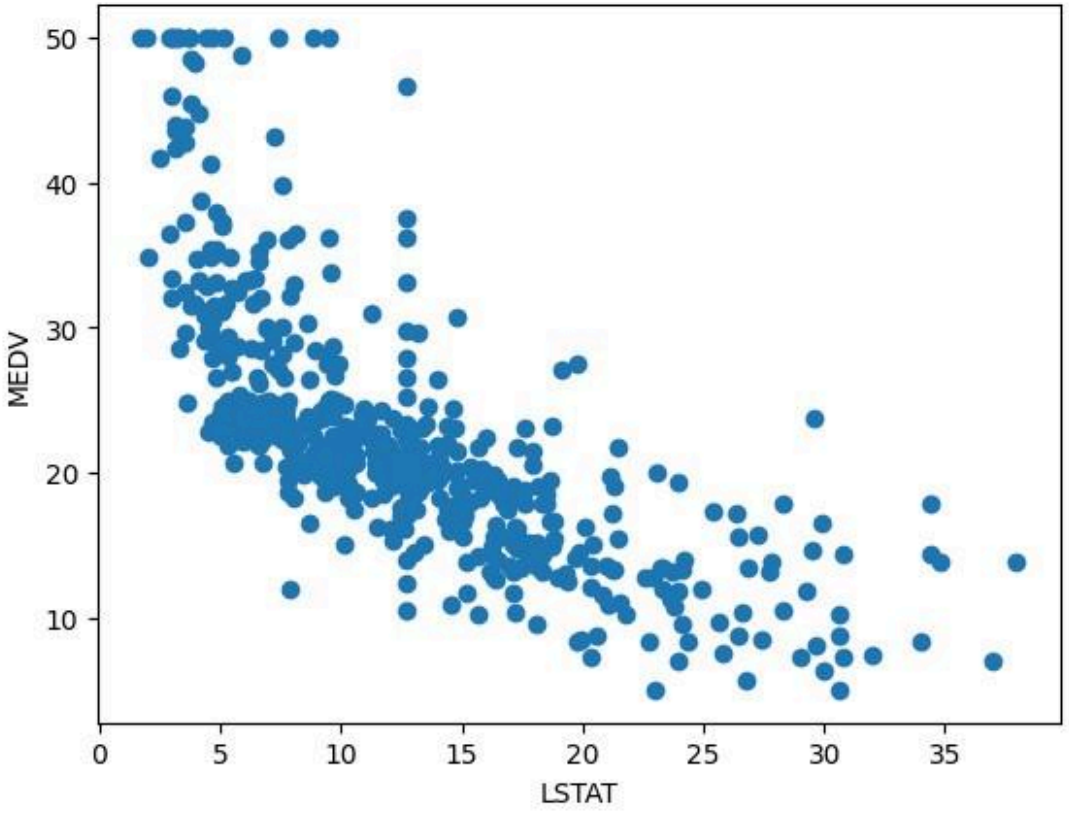

MEDV
DIS
50
40
30
20
10
2
4
6
8
10
12


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


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


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


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
```

Conclusion:

In this experiment, a linear regression model was developed to predict housing prices (MEDV) using various features from the dataset. The dataset initially contained missing values, which were handled by imputing the mean values for the respective columns. After preprocessing, the model was trained and evaluated, achieving a Mean Squared Error (MSE) of 17.38 and a Mean Absolute Error (MAE) of 3.11. The model's R-squared score of 0.73 indicates a reasonable level of predictive accuracy.