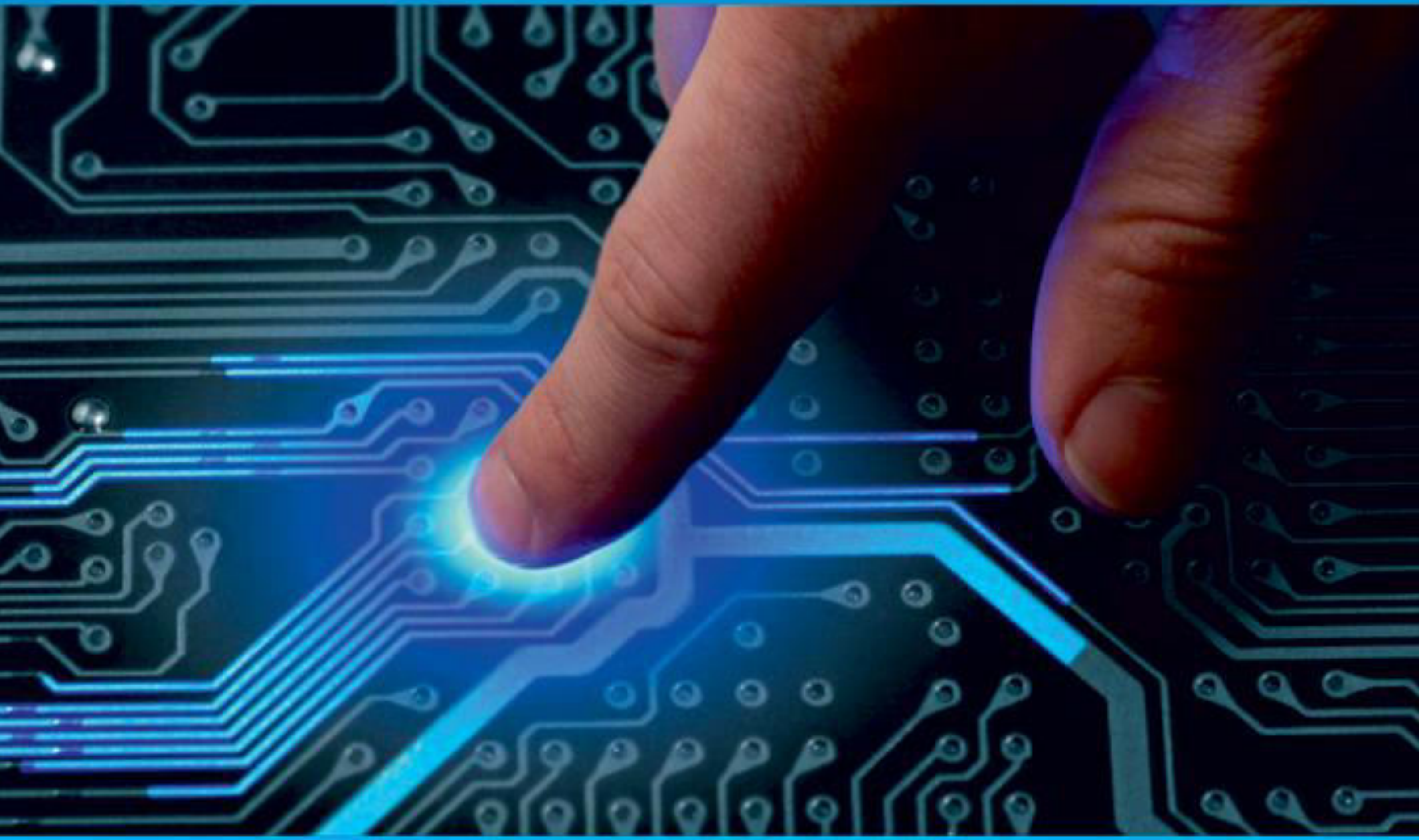
IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 10, Issue 7, July 2022

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.165

Linear Regression Using Boston Housing Dataset

Dr.Pradeep N[1] , Shashank Sharanabasavaraj[2], Shashank Shubash[3]

Professor, Department of Computer Science and Engineering, Bapuji Institute of Engineering and Technology, Davangere, Karnataka, India[1]

B.E Student, Department of Computer Science and Engineering, Bapuji Institute of Engineering and Technology, Davangere, Karnataka, India[2]

B.E Student, Department of Computer Science and Engineering, Bapuji Institute of Engineering and Technology, Davangere, Karnataka, India[3]

ABSTRACT: To do pattern recognition, this dataset is often used. It contains data about the various residences in Boston based on factors such as crime, taxation, and the amount of rooms. We'll make use of the Housing dataset, which contains data on various Boston-area residences. The UCI Machine Learning Repository no longer contains this data, which was formerly available there. Alternatively, we may utilise the scikit-learn library. There are 506 examples in this dataset, each with a unique collection of 13 variables. The objective is to estimate the house's value based on the information you have. The value of one variable is predicted based on the value of another variable using linear regression analysis. House prices are predicted based on this concept.

KEYWORDS: Pattern Recognition, UCI Machine Learning Repository, Scikit-Learn Library.

I. INTRODUCTION

Artificial intelligence is a method that lets machines act like humans by simulating their roles and types. Artificial Intelligence tries to figure out how machines can learn new skills through training. The machines change how they act based on the new information they receive. They do this by transforming large amounts of information and looking for patterns in them. Machine learning and deep learning are parts of artificial intelligence that use statistical methods to help machines learn and get better. Machine learning tells computers how to learn beyond what they were designed to do. Deep Learning is a branch of Machine Learning that makes it possible to use computers to work with neural networks with many layers.

Classification: To analyse and then predict, it's important to put the data into groups. The training data are used to create a model that sorts data into classes that have already been set up. This is called being "supervised" to learn. Supervised Learning is the process of putting a set of input variables (X) and a set of output variables (Y) together (Y). This algorithm is used to figure out the function that maps the input to the output.

Regression and classification are the two models of supervised learning that already exist. A company that looks at real estate has information about how much apartments in Boston cost. This information includes things like the crime rate, the number of people, how easy it is to get to, etc. The company decides how much the new apartments will cost based on the data it gets. A linear regression model can be used to figure this out. Regression: Unlike classification, regression gives you numeric data with values that don't change. Using Regression analysis, it looks for the best structure to find labels that are the same as those in a multi-label classification. A variable that needs to be predicted or explained is called a "dependent variable." An Independent Variable is a variable that has nothing to do with the dependent variable. Regression is figuring out what a number will be based on what you give it. Regression has been tried out in different fields of use.

II. LITERATURE SURVEY

Classification and making predictions are the most common and difficult ways to get accurate results from class labels. Deep learning makes it possible to use higher layers of computation in neural networks in the real world. This is a very powerful thing. Deep Learning works by using a process of detection with a nonlinear relationship between dependent and independent variables [2]. Normal data mining systems, such as Neural Network, are another real-world method used in educational data mining. The best thing about the neural network is that it can show all possible connections between predictor variables [3]. This was thought to be the best way to make a prediction.

When people are looking to buy a house, they are very interested in how the price of houses goes up and down. Even though it takes a long time to look at real estate, most buyers are sometimes unwilling to pay high prices. To figure out and explain these kinds of problems, systematic prediction information has been given about rates of inconsistency. Changes in the way technology is used, how to predict them, and how to find them anywhere in the world can all be done through research while sitting at home and not using any business apps [4]. The most important thing that the fundamental search of housing price looks at is how housing prices affect price levels and growth rates [5]. The time series method also affects the tendency of house price predictions from a statistical and analytical point of view.

System Design

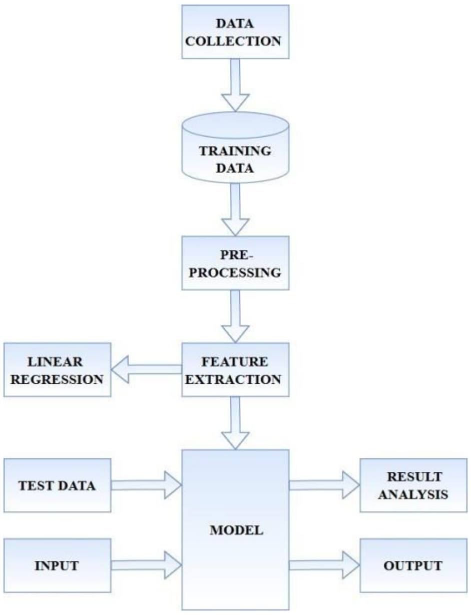


Fig:FlowDiagram

The project's flow chart is shown in the picture above. The steps are briefly explained here:

Data Collection

Getting data for a project depends on what kind of project it is. For ML projects, real-time data is used. The data set can come from many different places, like a file, a database, a sensor, or another source. Some free data sets can also be found on the internet and used. The most popular places to get data for Machine Learning models are Kaggle and UCI Machine Learning Repository. Kaggle is one of the most popular places where data sets are collected.

Pre-processing

Data pre-processing is the process of cleaning the raw data, which means that the data is collected in the real world and turned into a clean data set. Data pre-processing is the part of the process where steps are taken to turn the data into a small, clean set of data that can be used for analysis.

Most real-world information is messy, such as:

- MissingData
- NoisyData
- Inconsistent Data

Some basic pre-processing methods that can be used to turn raw data into usable information are:

- ConversionofData
- Ignoringthemissingvalues
- Fillingthemissingvalues
- Detectionofoutliers

FeatureExtraction

Data that is too large for an algorithm to analyse may be broken down into smaller characteristics, making it easier to process and more efficient. "Feature selection" refers to the process of deciding which features to include in the first place. It is important that all essential information from the input data be included in the chosen features so that the intended job may be completed without having to use the whole original data set. To explain a large dataset, feature extraction reduces the amount of resources required. The quantity of variables involved in sophisticated data analysis is one of the largest challenges. Memory and computing power are often required for complex data analysis. It may also lead to an overfitting of a classification system to the training samples, resulting in poor performance on fresh data.

ModelSelection

Selecting a final machine learning model from among a collection of candidate models is called model selection. With various kinds of models and with models of the same type that contain varying model hyper parameters, model selection may be employed.

There are different kinds of classification models:

- K-NearestNeighbor
- NaiveBayes
- DecisionTrees/RandomForest
- SupportVectorMachine
- LinearRegression

Trainand Test Data

When training a model, we first divide it into two parts called "Training data" and "Testing data." The "training data set" is used to teach the classifier how to work, and the "test data set" is used to see how well it works.

Training set: The training set is what a computer uses to learn how to handle information. The training part of machine learning is done by algorithms.

- The training data set is used to learn and adjust the classifier's settings.
- Test set:Aset ofunseendata usedonlyto assess the performance ofa fully-specified classifier.

Evaluation

Model evaluation is a key part of the process of making a model. It helps figure out which model best fits the data and how well the model chosen will work in the future. You can tune the model's hyperparameters and improve its accuracy to make it better. By adding more true positives and true negatives, the confusion matrix can be used to make things better. The output is predicted by looking at both the input and output test data. The output is then shown.

AlgorithmsIdentified

According to the instructions for our paper, we chose the following classification model to get correct results.

LinearRegression

The algorithm for machine learning In Linear Regression, supervised learning is the foundation. It performs a process known as regression. Regression models a goal prediction value using the independent variables. It is mostly used for inferring relationships and making predictions between several variables. Various aspects of regression modelling, such as the kind of connection between dependent and independent variables and the number of independent variables, are taken into consideration while developing a model.[7]

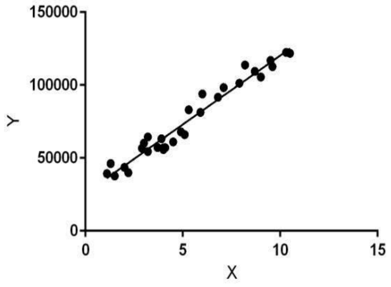


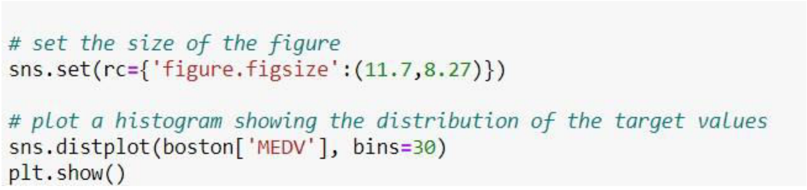
Fig: Linear Regression Sample[7]

In the picture above, X is the person's work experience and Y is their salary. Our model fits best with the regression line.

III. RESULTS AND DISCUSSION

Prior to training the model, exploratory data analysis is crucial. In this part, we'll utilise images to illustrate the relationship between the target variable and its neighbours.

Let's first take a look at the distribution of the target variable MEDV. We will utilise the distplot function of the seaborn library.



```
# set the size of the figure
sns.set(rc={'figure.figsize':(11.7,8.27)})

# plot a histogram showing the distribution of the target values
sns.distplot(boston['MEDV'], bins=30)
plt.show()
```

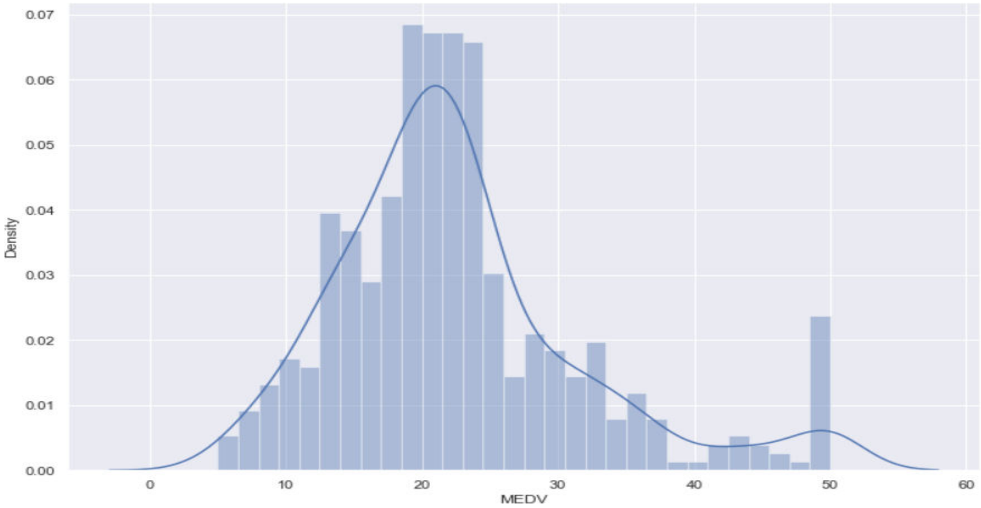
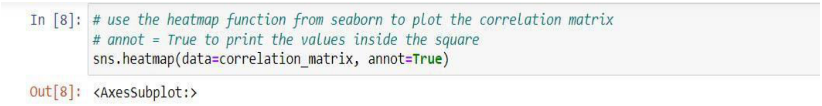
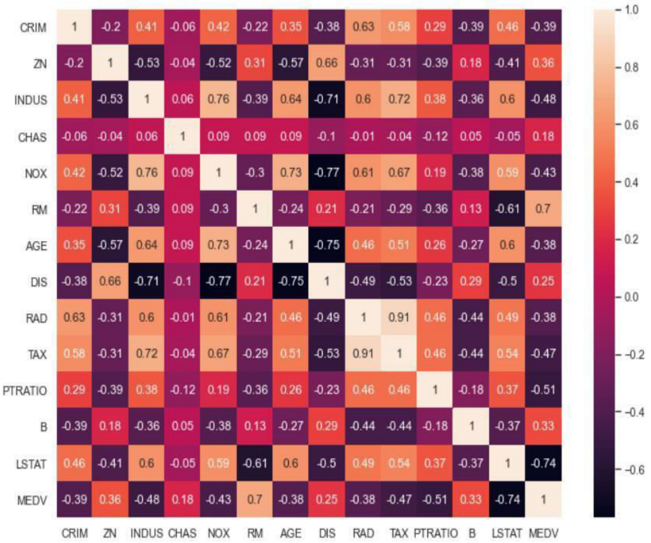


Fig: Histogramshowingthedistributionoftargetvalues

Thefigureaboveshowsthedistributionoftargetvaluesusinghistogram.Medianvalueofthe house and the population density are considered here.



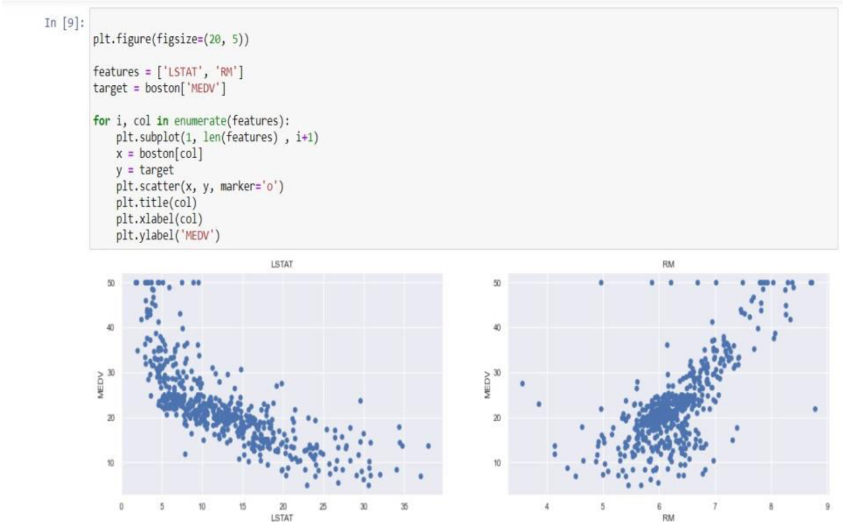
```
In [8]: # use the heatmap function from seaborn to plot the correlation matrix
        # annot = True to print the values inside the square
        sns.heatmap(data=correlation_matrix, annot=True)

Out[8]: <AxesSubplot:>
```



	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
CRIM	1	-0.2	0.41	-0.06	0.42	-0.22	0.35	-0.38	0.63	0.58	0.29	-0.39	0.46	-0.39
ZN	-0.2	1	-0.53	-0.04	-0.52	0.31	-0.57	0.66	-0.31	-0.31	-0.39	0.18	-0.41	0.36
INDUS	0.41	-0.53	1	0.06	0.76	-0.39	0.64	-0.71	0.6	0.72	0.38	-0.36	0.6	-0.48
CHAS	-0.06	-0.04	0.06	1	0.09	0.09	0.09	-0.1	-0.01	-0.04	-0.12	0.05	-0.05	0.18
NOX	0.42	-0.52	0.76	0.09	1	-0.3	0.73	-0.77	0.61	0.67	0.19	-0.38	0.59	-0.43
RM	-0.22	0.31	-0.39	0.09	-0.3	1	-0.24	0.21	-0.21	-0.29	-0.36	0.13	-0.61	0.7
AGE	0.35	-0.57	0.64	0.09	0.73	-0.24	1	-0.75	0.46	0.51	0.26	-0.27	0.6	-0.38
DIS	-0.38	0.66	-0.71	-0.1	-0.77	0.21	-0.75	1	-0.49	-0.53	-0.23	0.29	-0.5	0.25
RAD	0.63	-0.31	0.6	-0.01	0.61	-0.21	0.46	-0.49	1	0.91	0.46	-0.44	0.49	-0.38
TAX	0.58	-0.31	0.72	-0.04	0.67	-0.29	0.51	-0.53	0.91	1	0.46	-0.44	0.54	-0.47
PTRATIO	0.29	-0.39	0.38	-0.12	0.19	-0.36	0.26	-0.23	0.46	0.46	1	-0.18	0.37	-0.51
B	-0.39	0.18	-0.36	0.05	-0.38	0.13	-0.27	0.29	-0.44	-0.44	-0.18	1	-0.37	0.33
LSTAT	0.46	-0.41	0.6	-0.05	0.59	-0.61	0.6	-0.5	0.49	0.54	0.37	-0.37	1	-0.74
MEDV	-0.39	0.36	-0.48	0.18	-0.43	0.7	-0.38	0.25	-0.38	-0.47	-0.51	0.33	-0.74	1

Fig: Heat map function from sea born to plot correlation matrix

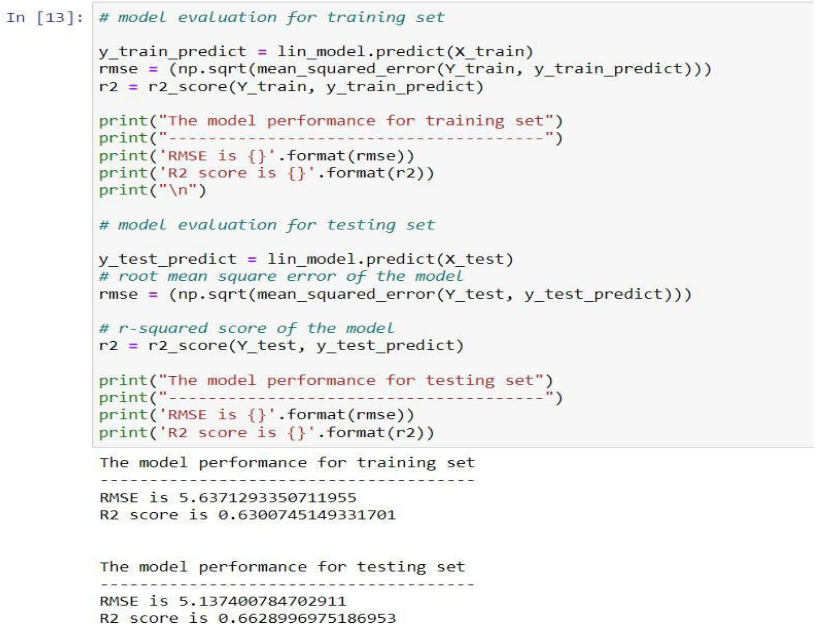
The heatmap function is shown in the picture above. For each value to be plotted, a heatmap has values that show different shades of the same colour. Most of the time, the darker colours on a chart show values that are higher than the lighter colours. For a very different value, you can also use a different colour. We can see that the values of MEDV aren't too far out of the norm. Based on what we've seen so far, our features will be RM and LSTAT. Let's see how these things change with MEDV by making a scatter plot.



```
In [9]:
plt.figure(figsize=(20, 5))

features = ['LSTAT', 'RM']
target = boston['MEDV']

for i, col in enumerate(features):
    plt.subplot(1, len(features) , i+1)
    x = boston[col]
    y = target
    plt.scatter(x, y, marker='o')
    plt.title(col)
    plt.xlabel(col)
    plt.ylabel('MEDV')
```

Fig: Python Codeandits Resulting Scatterplot

The figure above shows the scatterplot. A scatter plot is a diagram where each value in the data set is represented by a dot. In the above code we use for plotting 2 features LSTAT and RM.



```
In [13]: # model evaluation for training set

         y_train_predict = lin_model.predict(X_train)
         rmse = (np.sqrt(mean_squared_error(Y_train, y_train_predict)))
         r2 = r2_score(Y_train, y_train_predict)

         print("The model performance for training set")
         print("--------------------------------------")
         print('RMSE is {}'.format(rmse))
         print('R2 score is {}'.format(r2))
         print("\n")

         # model evaluation for testing set

         y_test_predict = lin_model.predict(X_test)
         # root mean square error of the model
         rmse = (np.sqrt(mean_squared_error(Y_test, y_test_predict)))

         # r-squared score of the model
         r2 = r2_score(Y_test, y_test_predict)

         print("The model performance for testing set")
         print("--------------------------------------")
         print('RMSE is {}'.format(rmse))
         print('R2 score is {}'.format(r2))
```

```
The model performance for training set
--------------------------------------
RMSE is 5.6371293350711955
R2 score is 0.6300745149331701


The model performance for testing set
--------------------------------------
RMSE is 5.137400784702911
R2 score is 0.6628996975186953
```

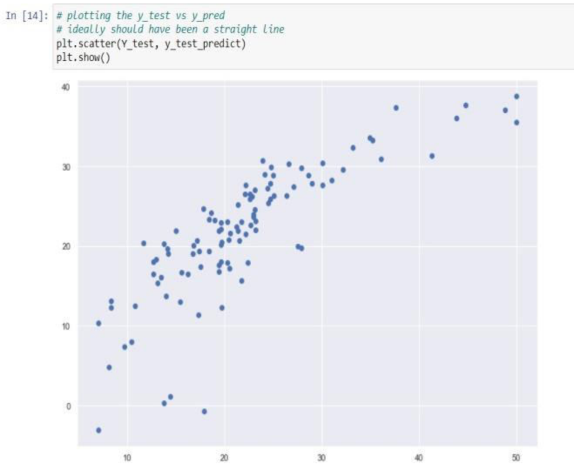
Fig: Python Code for training and testing and its Result

The figure above shows how the dataset was trained and tested. Testing and training are needed to get the process just right. RMSE shows how close the data points are to the line of best fit. R2 score shows how much of the change in the dependent variable can be predicted from the change in the independent variable (s).

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.165

|| Volume 10, Issue 7, July 2022 ||

| DOI: 10.15680/IJIRCCE.2022.1007026 |



```
In [14]: # plotting the y_test vs y_pred
         # ideally should have been a straight line
         plt.scatter(Y_test, y_test_predict)
         plt.show()
```

Fig:Plottingthey_testv/sy_pred

Fromthe figureabovewecanhavethe followingobservations:

- As the value of RM rises, so do the prices. To my knowledge, the data seems to have been restricted at 50.
- Increasing the LSTAT has the effect of lowering prices. Even if it doesn't seem to be following a linear path, it does.

IV. CONCLUSION

In this day and age of "big data," the Boston housing dataset is small. But there was a time when it was very hard to get your hands on neatly organised and labelled data, so a public dataset like this was very useful to researchers. Even though we now have a lot of datasets to choose from thanks to things like Kaggle and open government initiatives, this one is still a mainstay of machine learning practise. Regression analysis is more flexible and can be used in many different ways. In this project, we used linear regression on a set of data about homes in Boston.

REFERENCES

- Journals/Conferences

[1] Mengyu Huang, “Theoryand Implementation of linear regression”,IEEE,Chongqing, China, Jul 2020.

- Links

[2] https://en.wikipedia.org/wiki/Machine_learning

[3] http://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/

[4] https://www.javatpoint.com/applications-of-machine-learning

[5] https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)

[6] https://towardsdatascience.com/everything-you-need-to-know-about-jupyter-notebooks-10770719952b

[7] https://www.geeksforgeeks.org/ml-linear-regression/

● Textbook

[8] AlbertoBoschetti,LucaMassaron,"PythonDataScienceEssentials-ThirdEdition", PacktPublishing,ISBN:978178953786

[9] Al Sweigart,"Automate the Boring Stuff with Python-1st Edition",No Starch Press, 2015.ISBN:978-1593275990

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details