Predictive Analysis of Real Estate Dataset using Multiple Linear Regression Model

Gavriel Owens Vincentio

BF21DSY004

Table of Contents

1. Introduction

Real estate is property consisting of land and the buildings on it, along with its natural resources such as crops, minerals or water; immovable property of this nature; an interest vested in this (also) an item of real property, (more generally) buildings or housing in general. Real estate is different from personal property, which is not permanently attached to the land, such as vehicles, boats, jewellery, furniture, tools, and the rolling stock of a farm.

This dataset contains information collected by the U.S Census Service concerning housing in the area of Boston Mass. It was obtained from the StatLib archive (http://lib.stat.cmu.edu/datasets/boston), and has been used extensively throughout the literature to benchmark algorithms. The dataset is small in size with only 506 cases. The data was originally published by Harrison, D. and Rubinfeld, D.L. `Hedonic prices and the demand for clean air', J. Environ. Economics & Management, vol.5, 81-102, 1978.

The aim of this project is to build a prediction model to estimate the price of the real estates in Boston, USA, using multiple linear regression model by including various factors (crime rates, nitric oxides concentration, owner age, and etc.)

The dataset is split into two subsets, with a 4 : 1 ratio. One for training the model and the other for testing the model. The training sub dataset consists of 404 real estate instances and the testing dataset has data from the rest 102 real estate instances.

The model's accuracy is measured by implementing the model on the testing dataset and then comparing the predicted price with the testing dataset's actual price (MEDV). The most utilised tool in this project is python programming.

2. Descriptive Statistics

a. Data Description

Training data consists of 404 rows and 14 columns (attributes). The 14 attributes are:

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

Dependent variable (Y): *MEDV* Independent variable (X_i): *1-8,10-13*

First 5 rows from the dataset:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
119	0.14476	0.0	10.01	0	0.547	5.731	65.2	2.7592	6	432	17.8	391.50	13.61	19.3
315	0.25356	0.0	9.90	0	0.544	5.705	77.7	3.9450	4	304	18.4	396.42	11.50	16.2
430	8.49213	0.0	18.10	0	0.584	6.348	86.1	2.0527	24	666	20.2	83.45	17.64	14.5
435	11.16040	0.0	18.10	0	0.740	6.629	94.6	2.1247	24	666	20.2	109.85	23.27	13.4
395	8.71675	0.0	18.10	0	0.693	6.471	98.8	1.7257	24	666	20.2	391.98	17.12	13.1

Fig. 2.a.1

b. Descriptive Statistics for Independent and Dependent Variables

Fig 2.b.1 depicts the descriptive stats like count (number of observations), mean, standard deviation, minimum value, maximum value and the quartiles.

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
count	404.000000	404.000000	404.000000	404.000000	404.000000	404.000000	404.000000	404.000000	404.000000	404.000000	404.000000	404.000000	404.000000	404.000000
mean	3.575328	11.299505	11.092574	0.071782	0.558344	6.279943	69.100495	3.786057	9.665842	411.326733	18.430198	356.614629	12.912203	22.532178
std	7.941542	23.145229	6.726056	0.258447	0.118487	0.721350	27.737390	2.121488	8.722904	167.800579	2.262978	89.631819	7.990228	9.454097
min	0.006320	0.000000	0.460000	0.000000	0.385000	3.561000	6.000000	1.129600	1.000000	187.000000	12.600000	0.320000	1.730000	5.000000
25%	0.083672	0.000000	5.190000	0.000000	0.453000	5.873500	45.325000	2.095550	4.000000	284.000000	17.225000	374.237500	6.927500	16.775000
50%	0.324035	0.000000	9.690000	0.000000	0.538000	6.211500	77.700000	3.142300	5.000000	334.500000	18.950000	390.925000	11.395000	21.200000
75%	3.694070	12.500000	18.100000	0.000000	0.635000	6.632000	94.100000	5.117025	24.000000	666.000000	20.200000	395.690000	16.947500	25.000000
max	73.534100	100.000000	27.740000	1.000000	0.871000	8.780000	100.000000	12.126500	24.000000	711.000000	23.000000	396.900000	76.000000	67.000000

Fig. 2.b.1

Properties:

- $Mean = \mu = \frac{\Sigma X_i}{N}$
- $Standard\ Deviation = \sigma = \sqrt{\frac{\Sigma (X_i - \mu)^2}{N}}$

Where,

N is the count of each variable.

X_i are the variables.

c. Correlation Chart

Correlation Coefficient (r) matrix of 14 numeric variables:

$$r = \frac{\sum(X-\overline{X})(Y-\overline{Y})}{\sqrt{\sum(X-\overline{X})^2}\sqrt{(Y-\overline{Y})^2}}$$

Where, $\overline{X}$ = mean of X variable

$\overline{Y}$ = mean of Y variable

<matplotlib.axes._subplots.AxesSubplot at 0x7fa9ef3137d0>

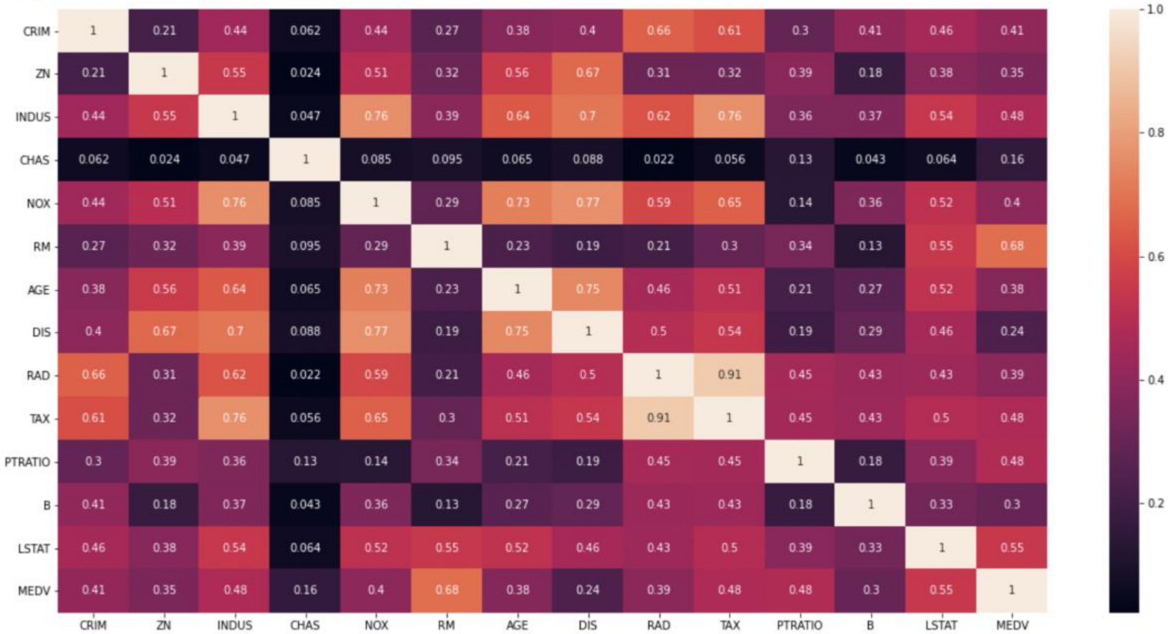
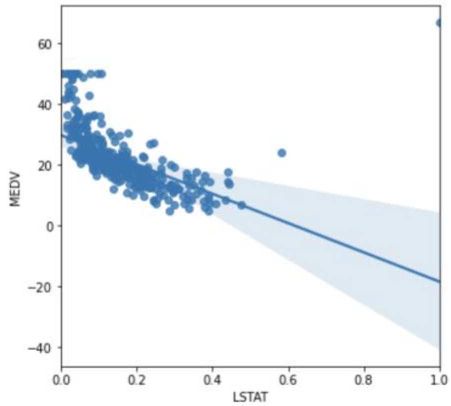


Fig. 2.c.1

Correlation Coefficient (r) provides a measure of linear relationship between X and Y. From fig. 2.c.1, we can see that LSTAT and RM have the strongest correlation with MEDV equal to 0,55 and 0,68. Other variables aren't that strongly correlated but for many of them it is about 0,4-0,45 which is not that low. The strongest relation for all x variables exists for RAD and TAX, and is equal over 90% which is strong enough to call it collinearity. Due to that as TAX variable is higher correlated with Y variable it will remain for the future analysis and RAD will be removed.

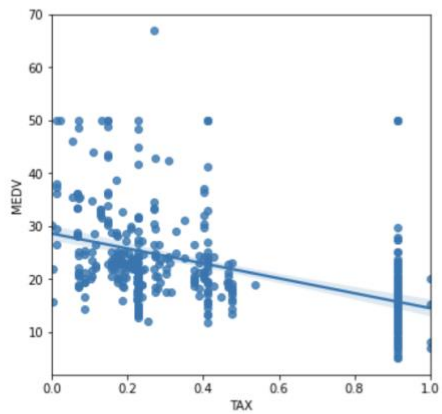
d. Graphs and Fitted Lines

Scatter plots between independent and five highest correlated dependent variables:



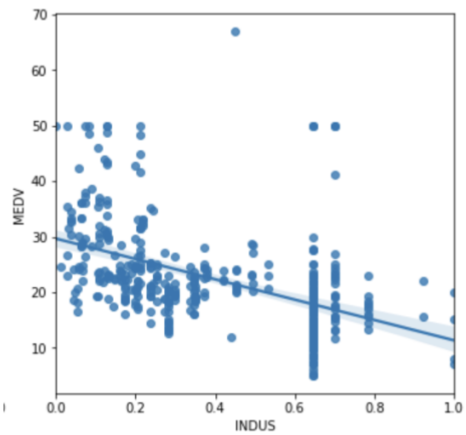
MEDV VS. LSTAT

MEDV and LSTAT appear to be weakly negatively correlated as the points seem to fall on a line. There is a strong possibility of a linear relationship.



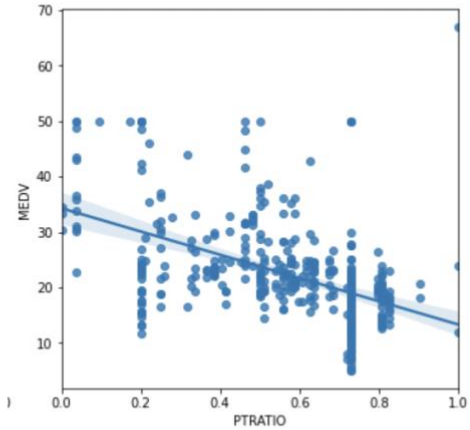
MEDV VS. TAX

MEDV and TAX appear to be weakly correlated as the points don't show any linear pattern.



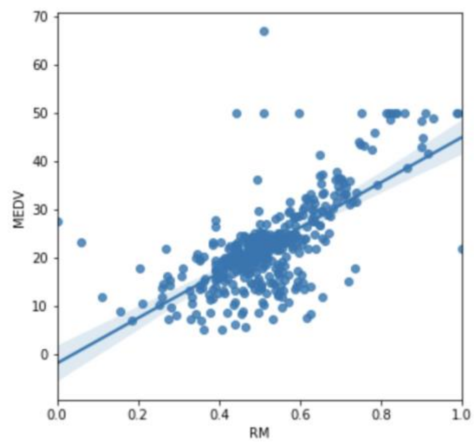
MEDV VS. INDUS

MEDV and INDUS appear to be weakly correlated as the points don't show any linear pattern.



MEDV VS. PTRATIO

MEDV and PTRATIO appear to be weakly correlated as the points don't show any linear pattern.



MEDV VS. RM

MEDV and RM appear to be weakly positively correlated as the points seem to fall on a line. There is a strong possibility of a linear relationship.

3. Multiple Regression Prediction Model

a. Model

Multiple linear regression prediction model for MEDV (Y) and X_i (CRIM, ZN, *INDUS, CHAS, NOX, RM, AGE, DIS, TAX, PTRATIO, B, LSTAT*):

$$\hat{Y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + \beta_6 X_6 + \beta_7 X_7 + \beta_8 X_8 + \beta_9 X_9 + \beta_{10} X_{10} + \beta_{11} X_{11} + \beta_{12} X_{12} \qquad (3.a.1)$$

Where,

$\hat{Y}$ is the predicted value of dependent variable Y.

X_i is the actual value of independent/explanatory variable:

X_1 : *CRIM*	X_6 : *RM*	X_{11} : *B*
X_2 : *ZN*	X_7 : *AGE*	X_{12} : *LSTAT*
X_3 : *INDUS*	X_8 : *DIS*	
X_4 : *CHAS*	X_9 : *TAX*	
X_5 : *NOX*	X_{10} : *PTRATIO*	

β_i : Regression Coefficient of respective X_i

This section is divided into 3 sections dedicated to:

1. Analysis of the regression statistic table
2. Hypothesis test to check model utility
3. Regression Coefficient table and final model

b. Regression Statistics Table

The following table is the regression statistics table. R^2 (Coefficient of Determination) is the most important factor in it.

Regression Statistics	
Multiple R	0.803
R Square	0.645
Adjusted R Square	0.634
Standard Error	5.712
Observations	404

Fig. 3.b.1

Explanation of the terms in the table:

1. Multiple R - Square Root of R^2
2. R square – Coefficient of Determination given be the formula:

$$R^2 = 1 - \frac{\text{SSResid}}{\text{SSTo}}$$

Where,

$$SSResid = \Sigma(Y - \hat{Y})^2$$

$$SSTo = \Sigma(Y - \bar{Y})^2$$

R-squared is a statistical measure of how close the data are to the fitted regression line. An R^2 value of 0.645 means that our model predicts with an accuracy of 64.5 percent.

3. Adjusted R square - Adjusted R-squared adjusts the statistic based on the number of independent variables in the model.
4. Standard error – Standard deviation of the error/residual
5. Observations – Total number of observation

The table gives the overall goodness of fit measures.

c. Hypothesis Testing

To check model utility, we hypothesize:

H_0 : $\beta_1 = \beta_2 = \beta_3 = \beta_4 = \beta_5 = \beta_6 = \beta_7 = \beta_8 = \beta_9 = \beta_{10} = \beta_{11} = \beta_{12} = 0$

There is no linear relationship between the dependent and independent variables.

H_A : $\beta_i \neq 0$ where i = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12

There is at least one independent variable which has a linear relationship with dependent variable.

ANOVA Table

Source	Sum of squares	Degree of Freedom	Mean squares	F
Treatment	SS_T	k-1	$MS_T = \frac{SS_T}{k-1}$	$F = \frac{MS_T}{MS_E}$
Error	SS_E	N-k	$MS_E = \frac{SS_E}{N-k}$	
Total	TotalSS	N-1		

Table 3.c.1

* Here, k is the number of coefficients (12) and N is the total number of observations (404).

Assuming significance level $\alpha = 0.05$ (from Table 3.c.1)

	Degree of Freedom	Sum of squares	Mean Squares	F	Significance F (p –value)
Regression	11	23229.1969	2111.7451	64.71	1.813
Residual	392	12790.9248	32.6299		
Total	403	36020.1217			

Fig. 3.c.2

F – critical (df1 = 11, df2 = 392) = 1.813; F – statistic = 64.71

Since F – statistic > F – critical ,

We reject the null hypothesis. Hence, there is at least one independent variable which has a linear relationship with the dependent variable.

d. Regression Coefficient Table and Finalised Model

The following table gives the value, standard error (SE), t statistic, p-value and confidence interval of regression coefficients:

	coef	std err	t	P>|t|	[0.025	0.975]
const	14.1108	6.216	2.270	0.024	1.891	26.331
CRIM	−0.1163	0.048	−2.410	0.016	−0.211	−0.021
ZN	0.0402	0.019	2.069	0.039	0.002	0.078
INDUS	−0.0646	0.087	−0.739	0.460	−0.237	0.107
CHAS	2.8398	1.135	2.502	0.013	0.608	5.072
NOX	−13.5754	4.864	−2.791	0.006	−23.137	−4.013
RM	6.2249	0.507	12.274	0.000	5.228	7.222
AGE	−0.0659	0.017	−3.833	0.000	−0.100	−0.032
DIS	−1.6350	0.273	−5.991	0.000	−2.172	−1.098
TAX	−0.0034	0.003	−0.991	0.322	−0.010	0.003
PTRATIO	−0.7419	0.170	−4.351	0.000	−1.077	−0.407
B	0.0100	0.004	2.732	0.007	0.003	0.017
LSTAT	−0.0295	0.053	−0.559	0.577	−0.133	0.074

Fig. 3.d.1

A simple summary of the above output is that the fitted line is (By substituting coefficients in equation 3.a.1:

$$\begin{aligned}\boldsymbol{\hat{Y} = 14.11 - 0.11\,X_1 + 0.04\,X_2 - 0.06\,X_3 + 2.84\,X_4 - 13.57\,X_5} \\ \boldsymbol{+ 6.22\,X_6 - 0.06\,X_7 - 1.65\,X_8 - 0.003\,X_9 - 0.} \quad (3.d.2) \\ \boldsymbol{+ 0.01\,X_{11} - 0.03\,X_{12}}\end{aligned}$$

Residual Plot

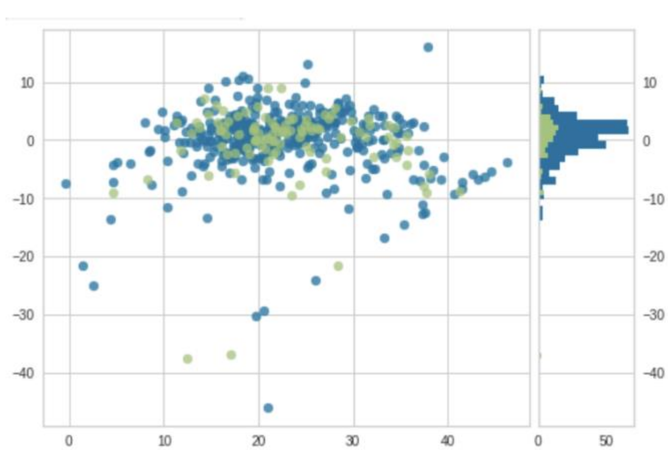


Fig. 3.d.3

4. Testing

a. Price Prediction

The prediction model (equation 3.d.2) is tested on the data from the 102 points in the testing set. We predict the accuracy of our model by comparing the mean absolute error, mean squared errors, and root mean square error of the testing and training dataset. From there we get:

	Training Data	Testing Data
MSE	31.66	45.76
MAE	3.58	3.82
RMSE	5.63	6.76

We also calculate the average value of r^2 after 10 cross validations and found that the value is 0.5414. Below is the fitted line visualization:

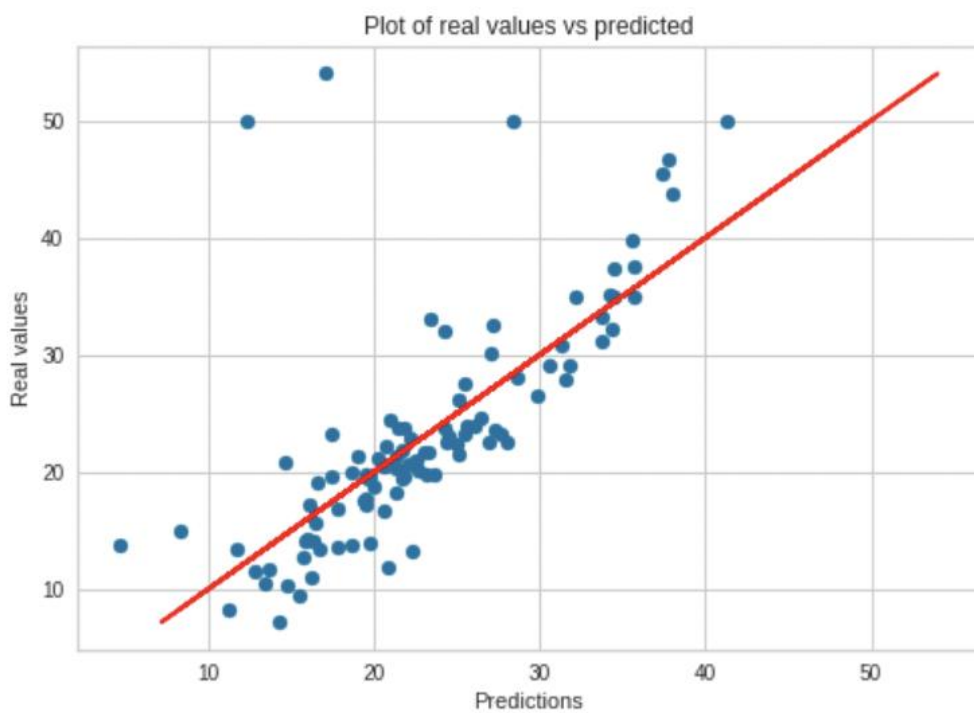


Fig. 4.a.1

5. Conclusion

In summary, we can analyse which variables are most significant and how they impact the created model. Looking at the t statistic and probability that coefficient is equal to 0, the three most significant variables are:

- RM - average number of rooms per dwelling
- DIS - weighted distances to five Boston employment centres
- AGE - proportion of owner-occupied units built prior to 1940

They can be interpreted as:

- Each additional room increase the price of home by around 6224 dollars.
- Increase of weighted distances to Boston employment centres by 1 unit decrease the price of the house by around 1635 dollars.
- Increase of age by 1 unit decrease the price of the home by around 66 dollars.

In conclusion, for the testing conduction, cross validation was applied for 10 splits and 3 repeats, and scoring method was r^2. Average value of r^2 was equal to 54.14% which is around 10.36% lower than r^2 achieved for whole training set (64.5%). Metrics chosen to measure goodness of predictions were MSE, MAE and RMSE. Difference between training and test sets are very small, MAE even shows slightly higher error for test set.

6. References

1. https://www.kaggle.com/arslanali4343/real-estate-dataset
2. https://en.wikipedia.org/wiki/Real_estate
3. http://www.cs.toronto.edu/~delve/data/boston/bostonDetail.html
4. https://dziganto.github.io/data%20science/linear%20regression/machine%20learning/python/Linear-Regression-101-Metrics/
5. https://www.scikit-yb.org/en/latest/api/regressor/residuals.html
6. https://www.danielsoper.com/statcalc/calculator.aspx?id=4

7. Appendix

Full version of python file is available in github:

https://github.com/deknared/Projects/blob/54ebd5889d4701552ff8b967ad3489d3637d955d/Stats_Final_Project.ipynb

These are some of the python codes that I used in this project:

#importing necessary libraries and reading the dataset

```
[1]  import numpy as np
     import pandas as pd
     import seaborn as sns
     import matplotlib.pyplot as plt
     from sklearn.metrics import classification_report
     from sklearn.metrics import confusion_matrix
     from sklearn.metrics import accuracy_score
```

```
[2]  df = pd.read_csv('data.csv')
```

#splitting the dataset

```
[8] from sklearn.model_selection import train_test_split

    training_data, testing_data = train_test_split(df, test_size=0.2, random_state=25)

    print(f"No. of training examples: {training_data.shape[0]}")
    print(f"No. of testing examples: {testing_data.shape[0]}")
```

```
No. of training examples: 404
No. of testing examples: 102
```

#to plot the scatter plots

```
from sklearn import preprocessing
# Let's scale the columns before plotting them against MEDV
min_max_scaler = preprocessing.MinMaxScaler()
column_sels = ['LSTAT', 'INDUS', 'PTRATIO', 'RM', 'TAX', 'DIS', 'AGE']
x = training_data.loc[:,column_sels]
y = training_data['MEDV']
x = pd.DataFrame(data=min_max_scaler.fit_transform(x), columns=column_sels)
fig, axs = plt.subplots(ncols=4, nrows=2, figsize=(20, 10))
index = 0
axs = axs.flatten()
for i, k in enumerate(column_sels):
    sns.regplot(y=y, x=x[k], ax=axs[i])
plt.tight_layout(pad=0.4, w_pad=0.5, h_pad=5.0)
```

#to plot the heatmap

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

#plotting residual plot

```
from sklearn.linear_model import LinearRegression
from yellowbrick.regressor import ResidualsPlot

# Instantiate the linear model and visualizer
model = LinearRegression()
visualizer = ResidualsPlot(model)

visualizer.fit(x_train, y_train)  # Fit the training data to the visualizer
visualizer.score(x_test, y_test)
visualizer.show()
```