Summer Internship Programme

Henry Harvin Education India LLP

Sector-2, Noida, U.P.-201306

Project Title – Housing

Mentor Name: Ms. Pooja Gupta (Designation)

Name: Tusar kumar samal

Course: Summer Internship Programme (SIP) Python

Batch: 12th Jun to 28th Jul 2019

Job: Business Analyst Associate (Intern)

Institute: Lovely professional university.

DECLARATION

I hereby declare that the project report entitled "**Housing**" submitted by me to **HENRY HARVIN EDUCATION INDIA** is a record of Bonafide project work carried out by me under the guidance of MS. POOJA GUPTA. This project is an original report with references taken from websites and help from mentors and teachers.

TABLE OF CONTENTS

Abstract:

Bhubaneswar is the smart city in our country. As it is rapidly developing the construction in the city is very costly. Economic point of view if the building is constructed at a far distance from the city it will be cheaper and residents can live peaceful without any external polluted sources. Having peaceful surroundings the main point of view of most of the people in today's lifestyle.

INTRODUCTION

The strategy provides a framework for government and the community to address the multiple factors that influence the supply and demand of housing. In this project we are predicting the price of the house based on the area and influencing factors like CRIM, ZN, INDUS, CHAS, NOX, RM, AGE, DIS, RAD, TAX, PTRATIO, B, LSTAT, MEDV

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 the tract bounds river; 0 otherwise)
NOX	nitric oxides concentration (parts per 10 million)
RM	an average number of rooms per dwelling
AGE	proportion of owner-occupied units built prior to 1940
DIS	weighted distances to five Boston employment centers
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	percent lower status of the population
MEDV	Median value of owner-occupied homes in 1000's

DATA

- **Pandas**
- **Numpy**
- **Matplotlib**
- **Model fit**
- **Linear Regression**

Importing Data with read_csv():

The first step to any data science project is to import your data. Often, you'll work with data in Comma Separated Value (CSV) files and run into problems at the very start of your workflow.

```
In [59]: df = pd.read_csv("E:\\pin2\\New folder\\PYdata\\Housing.txt", delim_whitespace=True, header = -1)
         df.columns = col_name
```

```
In [60]: df.head()
```

Out[60]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.01	18.00	2.31	0	0.54	6.58	65.20	4.09	1	296.00	15.30	396.90	4.98	24.00
1	0.03	0.00	7.07	0	0.47	6.42	78.90	4.97	2	242.00	17.80	396.90	9.14	21.60
2	0.03	0.00	7.07	0	0.47	7.18	61.10	4.97	2	242.00	17.80	392.83	4.03	34.70
3	0.03	0.00	2.18	0	0.46	7.00	45.80	6.06	3	222.00	18.70	394.63	2.94	33.40
4	0.07	0.00	2.18	0	0.46	7.15	54.20	6.06	3	222.00	18.70	396.90	5.33	36.20

Exploratory Data Analysis:

Exploratory Data Analysis (EDA) helps to answer all these questions, ensuring the best outcomes for the project. It is an approach for summarizing, visualizing, and becoming intimately familiar with the important characteristics of a data set.

```
df.describe()
```

Out[61]:

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
count	338.00	338.00	338.00	338.00	338.00	338.00	338.00	338.00	338.00	338.00	338.00	338.00	337.00	337.00
mean	0.42	14.76	8.63	0.08	0.51	6.40	61.47	4.32	4.52	310.08	17.68	380.09	10.51	25.18
std	0.65	25.21	6.12	0.27	0.10	0.68	28.66	1.92	1.60	67.78	2.22	41.14	5.97	8.56
min	0.01	0.00	0.46	0.00	0.39	4.90	2.90	1.32	1.00	188.00	12.60	70.80	1.73	11.80
25%	0.07	0.00	4.05	0.00	0.44	5.94	36.60	2.73	4.00	264.00	16.10	383.31	6.15	19.60
50%	0.14	0.00	6.91	0.00	0.49	6.25	64.80	4.02	4.00	304.00	17.90	392.72	9.45	22.90
75%	0.44	21.75	10.45	0.00	0.54	6.74	88.75	5.68	5.00	358.00	19.10	396.17	13.45	29.00
max	4.10	100.00	25.65	1.00	0.87	8.72	100.00	9.22	8.00	469.00	21.20	396.90	34.41	50.00

df.info():

This method prints information about a Data Frame including the index dtype and column dtypes, non-null values and memory usage.

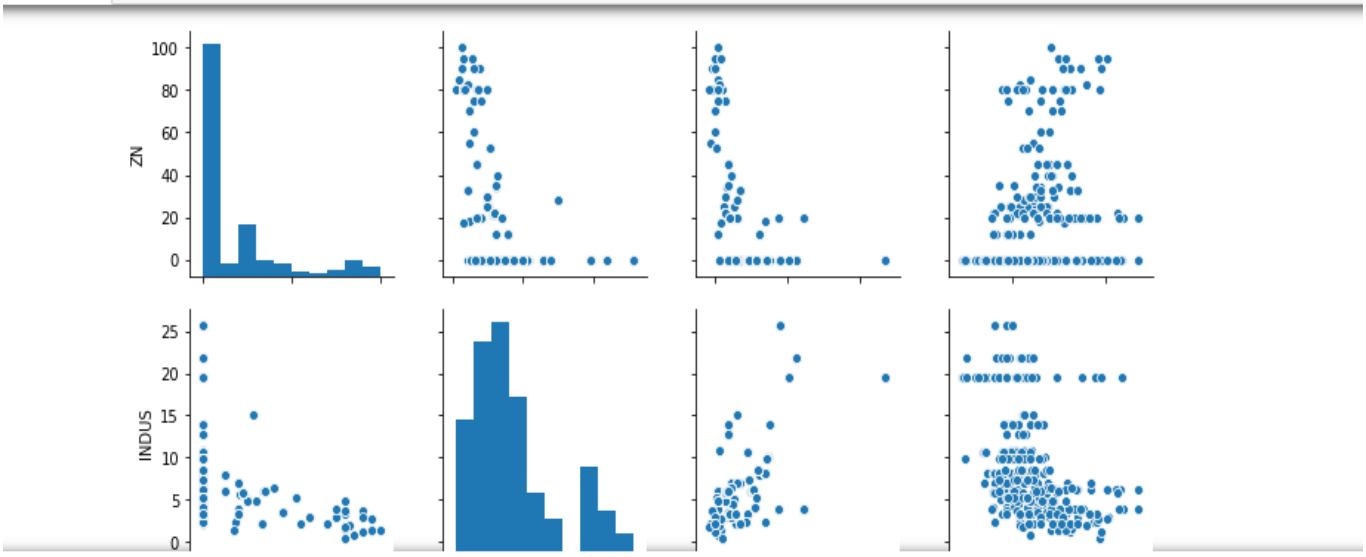
```
In [63]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 338 entries, 0 to 337
Data columns (total 14 columns):
CRIM        338 non-null float64
ZN          338 non-null float64
INDUS       338 non-null float64
CHAS        338 non-null int64
NOX         338 non-null float64
RM          338 non-null float64
AGE         338 non-null float64
DIS         338 non-null float64
RAD         338 non-null int64
TAX         338 non-null float64
PTRATIO     338 non-null float64
B           338 non-null float64
LSTAT       338 non-null float64
MEDV        338 non-null float64
dtypes: float64(12), int64(2)
memory usage: 37.0 KB
```

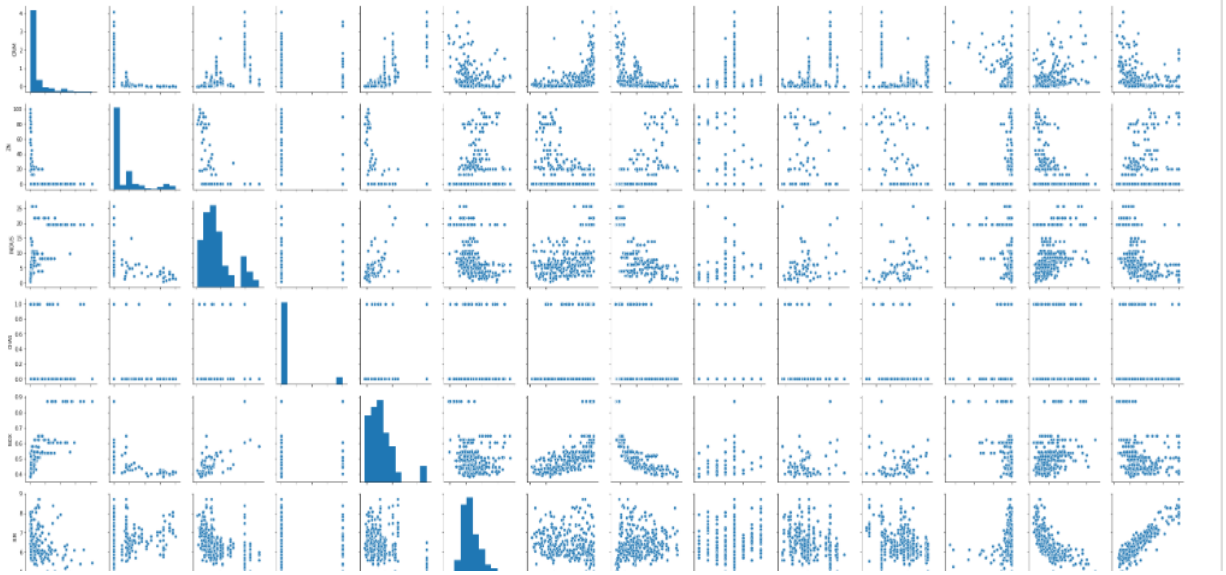
Pair plot:

To get started we need to know what data we have. We can load in the socioeconomic data as a pandas dataframe and look at the columns:

```
In [64]: sns.pairplot(df[col_study])
         plt.show()
```




```
In [65]: sns.pairplot(df)
Out[65]: <seaborn.axisgrid.PairGrid at 0xff7a90f940>
```



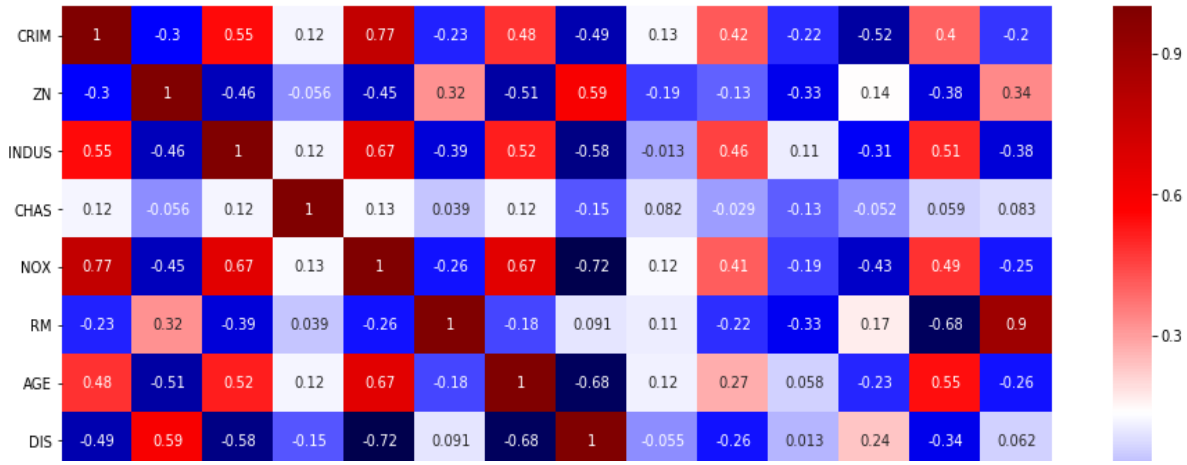
Correlation Analysis and Feature Selection:

The **Correlation Analysis** is the statistical tool used to study the closeness of the relationship between two or more variables. The variables are said to be correlated when the movement of one variable is accompanied by the movement of another variable.

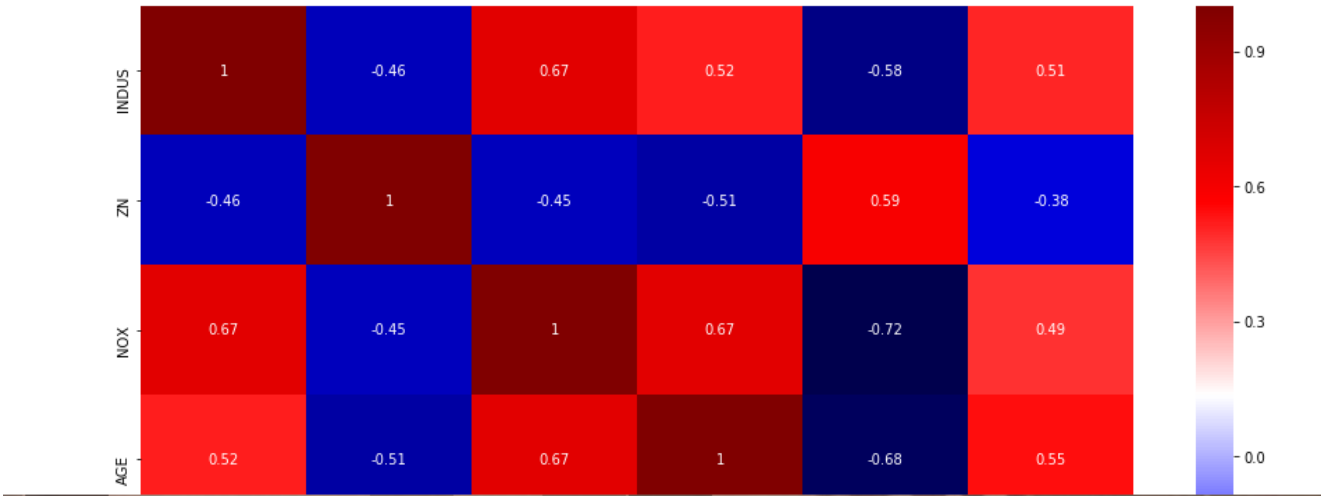
```
pd.options.display.float_format = '{:,.2f}'.format
```

[39]:
```
df.corr()
col_feature = ['INDUS', 'ZN', 'NOX', 'AGE', 'DIS', 'LSTAT']
```

[41]:
```
matrix = df.corr()
plt.figure(figsize = (16,10))
sns.heatmap(matrix, annot = True, cmap = 'seismic')
plt.show()
```



```
In [42]: plt.figure(figsize = (16,10))
         sns.heatmap(df[col_feature].corr(), annot = True, cmap = 'seismic')
         plt.show()
```

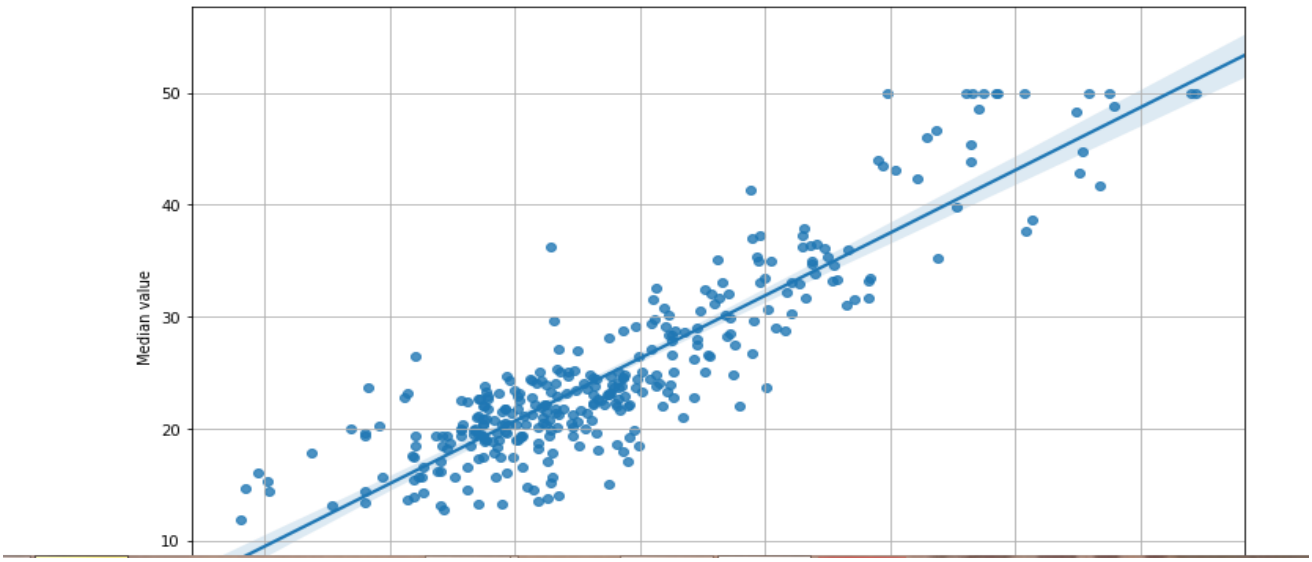


Linear Regression:

Linear regression attempts to model the relationship between two variables by fitting a linear equation to observed data. One variable is considered to be an explanatory variable, and the other is considered to be a dependent variable. A linear regression line has an equation of the form $\boldsymbol{Y = a + bX}$, where $\boldsymbol{X}$ is the explanatory variable and $\boldsymbol{Y}$ is the dependent variable. The slope of the line is $\boldsymbol{b}$, and $\boldsymbol{a}$ is the intercept (the value of $\boldsymbol{y}$ when $\boldsymbol{x} = 0$).

```
In [69]: from sklearn.linear_model import LinearRegression

In [70]: model = LinearRegression()

In [73]: model.fit(X, y)

Out[73]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
                          normalize=False)

In [74]: model.coef_

Out[74]: array([11.2036904])

In [76]: model.intercept_

Out[76]: -46.55864645633521
```

```
In [77]: plt.figure(figsize=(12, 8))
         sns.regplot(X, y)
         plt.xlabel('Avearge Rooms per dwelling')
         plt.ylabel('Median value')
         plt.grid(True)
         plt.show()
```



plt.figure(figsize = (15,8))

sns.jointplot(x = 'RM', y= 'MEDV', kind = 'reg', data = df)

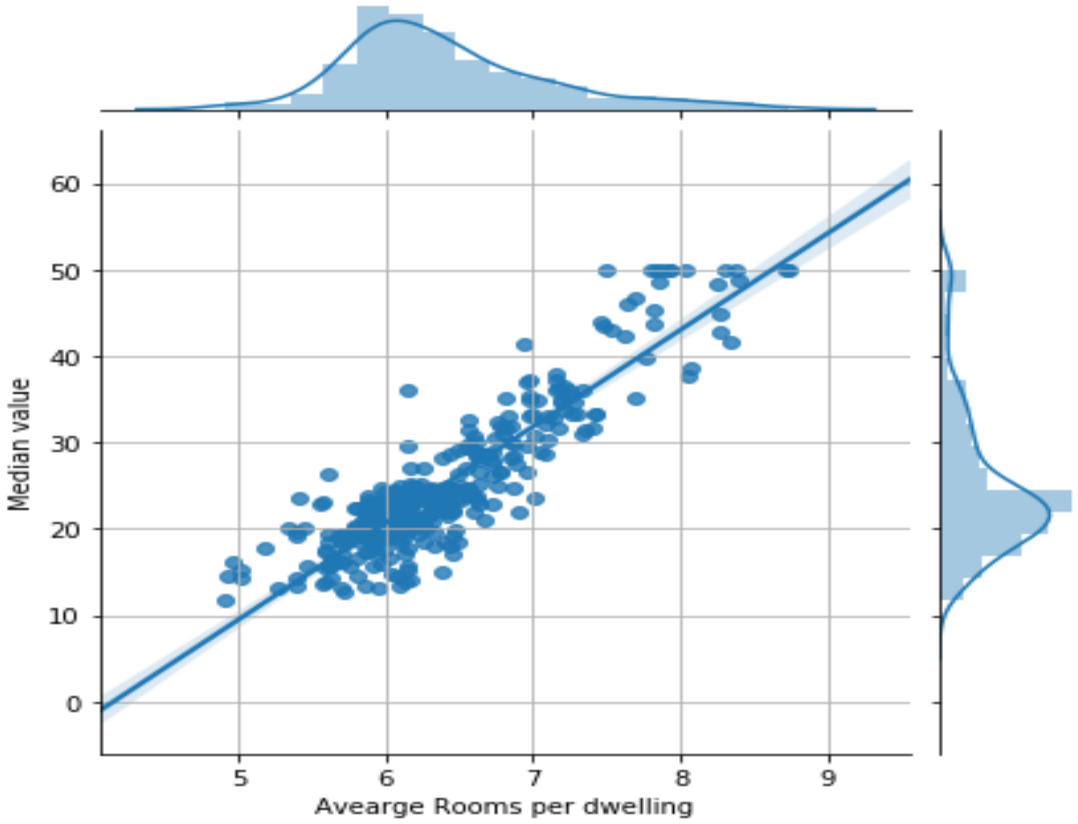
plt.xlabel('Avearge Rooms per dwelling')

plt.ylabel('Median value')

plt.grid(True)

plt.show()

```
<Figure size 1080x576 with 0 Axes>
```



Model fit:

The **goodness of fit** of a statistical **model** describes how well it **fits** a set of observations. Measures of **goodness of fit** typically summarize the discrepancy between observed values and the values expected under the **model** in question.

In [80]:
```
model.fit(X1, y)
```

Out[80]:
```
LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
         normalize=False)
```

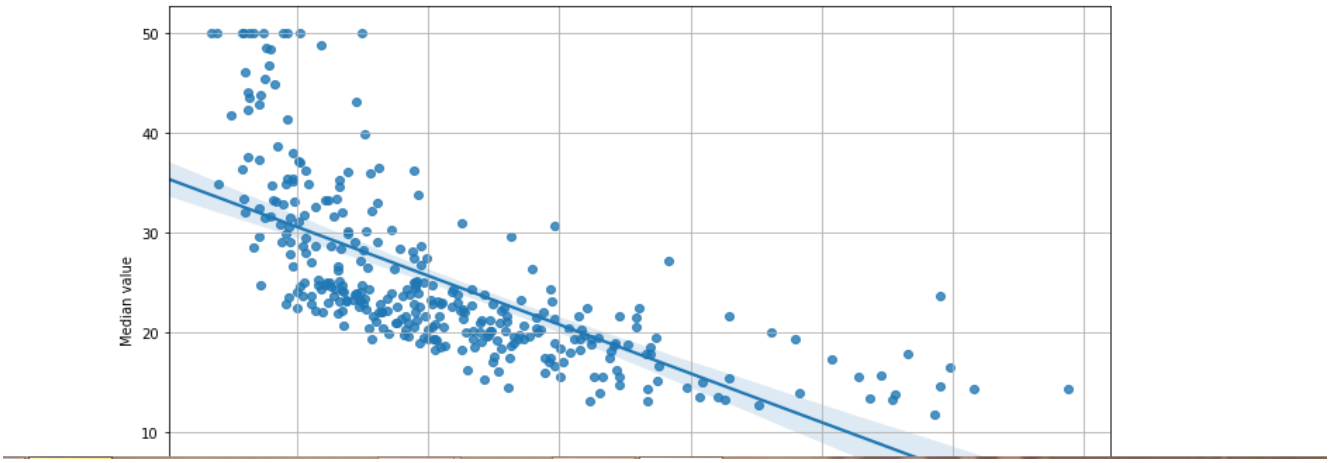
In [81]:
```
print(model.coef_)
print(model.intercept_)
```

```
[-0.97694907]
35.43267404141873
```

Figure plot:

A **plot** is a graphical technique for representing a data set, usually as a **graph** showing the relationship between two or more variables.

```
In [82]: plt.figure(figsize=(12, 8))
         sns.regplot(X1, y)
         plt.xlabel('%lower status of the population')
         plt.ylabel('Median value')
         plt.grid(True)
         plt.show()
```



plt.figure(figsize = (15,8))

sns.jointplot(x = 'LSTAT', y= 'MEDV', kind = 'reg', data = df)

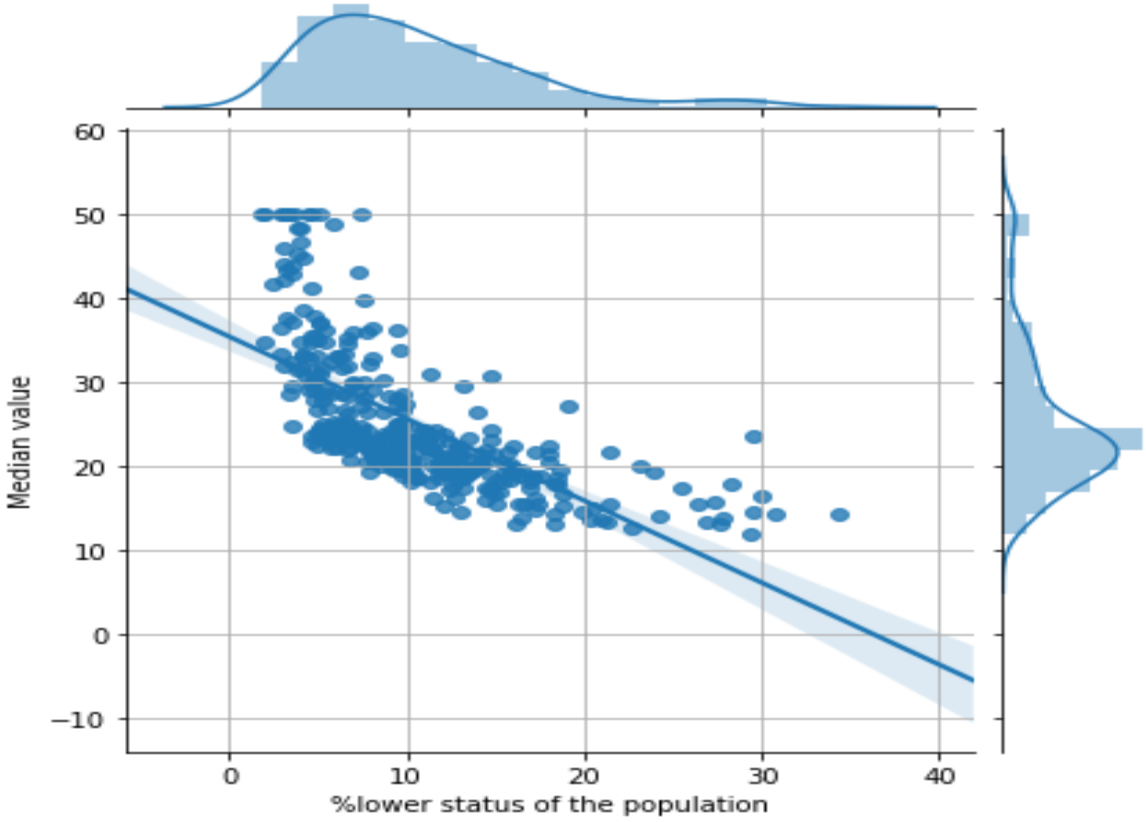
plt.xlabel('%lower status of the population')

plt.ylabel('Median value')

plt.grid(True)

plt.show()

<Figure size 1080x576 with 0 Axes>



References:

1.Thakur, Atul (25 November 2008). "33% of Indians live in less space than US prisoners". The Times of India.

2. "Reforming the Power Sector: Controlling Electricity Theft and Improving Revenue"(PDF). The World Bank. Archived from the original (PDF) on 25 February 2009.

3.*"Development Policy Review". World Bank.*

4."'Power-full' Gujarat gives 24-hour electricity". Times of India. 4 May 2012..

*5.*The Politics of Toilets, *Boloji*

6. Mumbai Slum: Dharavi, *National Geographic, May 2007*

7. *"India Signs Loan and Project Agreements with World Bank for US $100 Million for Low Income Housing Finance Project" (Press release). Press Information Bureau, Government of India. 15 August 2013. Retrieved 11 June 2014.*

8. Jump up to:[a] [b] *"Mumbai housing is the priciest in the developing world". Global Property Guide.*

9. *"Skyscrapers of Mumbai". Emporis.com. 15 June 2009. Archived from the original on 5 August 2011. Retrieved 12 August 2010.*

10. "Skyscrapers of Navi Mumbai". Emporis.com. 15 June 2009. Archived from the original on 9 May 2005. Retrieved 12 August 2010.