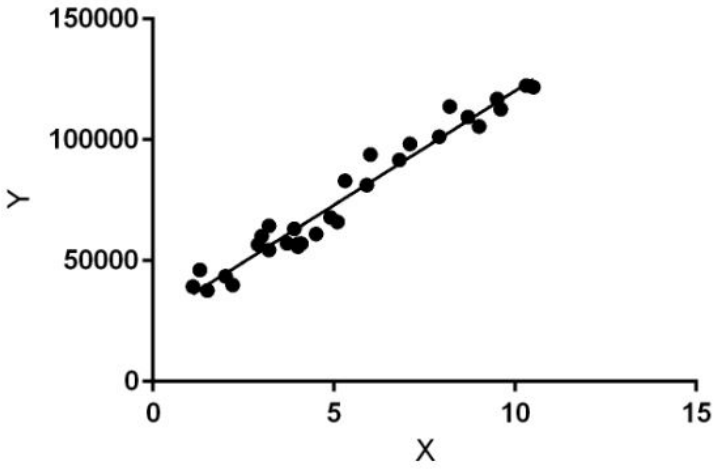
| |
|---|
| Experiment No. 1 |
| Analyze the Boston Housing dataset and apply appropriate Regression Technique |
| Date of Performance: 27/07/2023 |
| Date of Submission: 10/08/2023 |

**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
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

```
BostonTrain = pd.read_csv("/content/boston_train.csv")
```

```
BostonTrain.head()
```

| | ID | crim | zn | indus | chas | nox | rm | age | dis | rad | tax | ptratio | black |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 0.00632 | 18.0 | 2.31 | 0 | 0.538 | 6.575 | 65.2 | 4.0900 | 1 | 296 | 15.3 | 396.90 |
| **1** | 2 | 0.02731 | 0.0 | 7.07 | 0 | 0.469 | 6.421 | 78.9 | 4.9671 | 2 | 242 | 17.8 | 396.90 |
| **2** | 4 | 0.03237 | 0.0 | 2.18 | 0 | 0.458 | 6.998 | 45.8 | 6.0622 | 3 | 222 | 18.7 | 394.63 |
| **3** | 5 | 0.06905 | 0.0 | 2.18 | 0 | 0.458 | 7.147 | 54.2 | 6.0622 | 3 | 222 | 18.7 | 396.90 |
| **4** | 7 | 0.08829 | 12.5 | 7.87 | 0 | 0.524 | 6.012 | 66.6 | 5.5605 | 5 | 311 | 15.2 | 395.60 |

```
BostonTrain.info()
BostonTrain.describe()
```

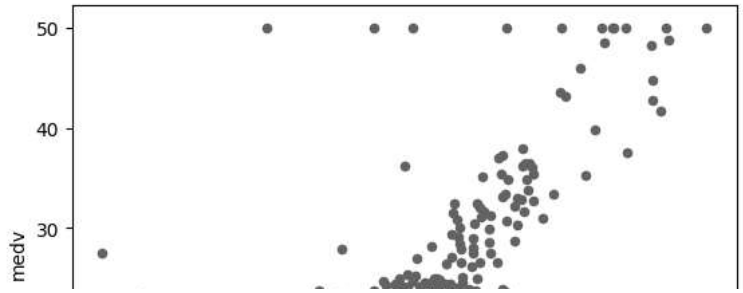
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 333 entries, 0 to 332
Data columns (total 15 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   ID       333 non-null    int64
 1   crim     333 non-null    float64
 2   zn       333 non-null    float64
 3   indus    333 non-null    float64
 4   chas     333 non-null    int64
 5   nox      333 non-null    float64
 6   rm       333 non-null    float64
 7   age      333 non-null    float64
 8   dis      333 non-null    float64
 9   rad      333 non-null    int64
 10  tax      333 non-null    int64
 11  ptratio  333 non-null    float64
 12  black    333 non-null    float64
 13  lstat    333 non-null    float64
 14  medv     333 non-null    float64
dtypes: float64(11), int64(4)
memory usage: 39.1 KB
```

| | ID | crim | zn | indus | chas | nox | r |
|---|---|---|---|---|---|---|---|
| **count** | 333.000000 | 333.000000 | 333.000000 | 333.000000 | 333.000000 | 333.000000 | 333.00000 |
| **mean** | 250.951952 | 3.360341 | 10.689189 | 11.293483 | 0.060060 | 0.557144 | 6.26561 |
| **std** | 147.859438 | 7.352272 | 22.674762 | 6.998123 | 0.237956 | 0.114955 | 0.70395 |
| **min** | 1.000000 | 0.006320 | 0.000000 | 0.740000 | 0.000000 | 0.385000 | 3.56100 |
| **25%** | 123.000000 | 0.078960 | 0.000000 | 5.130000 | 0.000000 | 0.453000 | 5.88400 |
| **50%** | 244.000000 | 0.261690 | 0.000000 | 9.900000 | 0.000000 | 0.538000 | 6.20200 |
| **75%** | 377.000000 | 3.678220 | 12.500000 | 18.100000 | 0.000000 | 0.631000 | 6.59500 |
| **max** | 506.000000 | 73.534100 | 100.000000 | 27.740000 | 1.000000 | 0.871000 | 8.72500 |

```
BostonTrain.drop('ID', axis = 1, inplace=True)
```

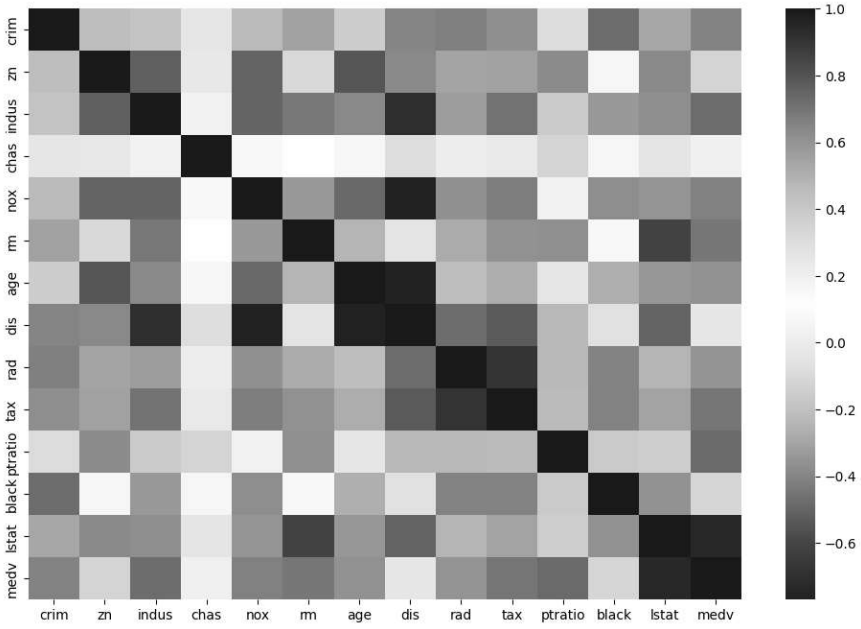
```
BostonTrain.plot.scatter('rm', 'medv')
```

```
<Axes: xlabel='rm', ylabel='medv'>
```




```
plt.subplots(figsize=(12,8))
sns.heatmap(BostonTrain.corr(), cmap = 'RdGy')
```

```
<Axes: >
```

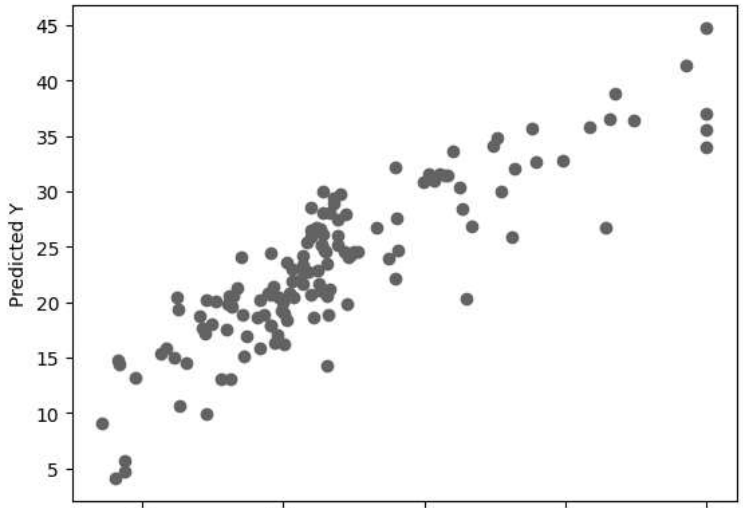



```
X = BostonTrain[['crim', 'zn', 'indus', 'chas', 'nox', 'rm', 'age', 'dis', 'rad', 'tax',
       'ptratio', 'black', 'lstat']]
y = BostonTrain['medv']


from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression


X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4)


lm = LinearRegression()
lm.fit(X_train,y_train)
```

```
▾ LinearRegression
LinearRegression()
```

```
predictions = lm.predict(X_test)


plt.scatter(y_test,predictions)
plt.xlabel('Y Test')
plt.ylabel('Predicted Y')
```

```
Text(0, 0.5, 'Predicted Y')
```




```
from sklearn import metrics

print('MAE:', metrics.mean_absolute_error(y_test, predictions))
print('MSE:', metrics.mean_squared_error(y_test, predictions))
print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, predictions)))
```

```
MAE: 3.5156447428522086
MSE: 21.75464362568027
RMSE: 4.664187348904444
```

**Conclusion:**

1. What are features have been chosen to develop the model? Justify the features chosen to estimate the price of a house.

   **CRIM (Per Capita Crime Rate):**
   The crime rate in a neighborhood can significantly affect housing prices. Higher crime rates often lead to lower property values.

   **ZN (Proportion of Residential Land Zoned for Large Lots):**
   Zoning for large residential lots can be an indicator of more spacious and upscale neighborhoods, which tend to have higher home prices.

   **INDUS (Proportion of Non-Retail Business Acres per Town):**
   The industrial composition of an area can influence home prices. Residential neighborhoods with more businesses may have lower property values.

   **RM (Average Number of Rooms per Dwelling):**
   The number of rooms is often positively correlated with the size and quality of a house. Larger homes with more rooms generally have higher prices.

   **AGE (Proportion of Owner-Occupied Units Built Before 1940):**
   Older homes may have lower prices due to the need for maintenance and renovations. Newer homes may command higher prices.

   **DIS (Weighted Distance to Employment Centers):**
   Proximity to employment centers can impact housing prices. Houses closer to job opportunities are often more expensive due to convenience.

   **TAX (Property Tax Rate):**
   Property tax rates affect the ongoing costs of homeownership. Higher tax rates can reduce the affordability of a property.

**PTRATIO (Pupil-Teacher Ratio):**

Schools play a significant role in property values. A lower pupil-teacher ratio is often associated with better schools and higher housing prices.

**B (Proportion of Residents of African American Descent):**

Demographics can affect housing prices. Neighborhoods with higher diversity and inclusivity may have more stable or appreciating property values.

**LSTAT (Percentage of Lower Status Population):**

The socioeconomic status of a neighborhood's residents can impact housing prices. Higher percentages of lower-status residents may lead to lower property values.

**RAD (Accessibility to Highways):**

Easy access to highways can be desirable, especially for commuters. Proximity to highways can positively influence housing prices.

**CHAS (Proximity to Charles River):**

Being near natural amenities like rivers can enhance the appeal of a neighborhood, potentially leading to higher property values.

2. Comment on the Mean Squared Error calculated.

An MSE (Mean Squared Error) of 4.664187348904444 indicates that, on average, the model's predictions for housing prices are off by 4.664187348904444 units squared, suggesting reasonable accuracy. Lower MSE values signify better predictions, but the interpretation depends on the scale of the housing prices. Comparing it with other metrics can provide a more comprehensive assessment of the model's performance.