BOSTON HOUSING PRICE PREDICTION

A PROJECT REPORT
Submitted By

Ankit Sharma
24/AFI/05

Submitted in partial fulfillment of the requirements for the degree of Masters of Technology in Artificial Intelligence

to

Prof. Anil Singh Parihar
Department of Computer Science & Engineering

DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Bawana Road, Delhi - 110042

Sep 15, 2024

Certificate

This is to certify that project report entitled **"BOSTON HOUSING PRICE PREDICTION"** submitted by **Ankit Sharma (24/AFI/05)** for partial fulfillment of the requiremnt for the award of degree Master of Technology (Artificial Intelligence) is a record of the candidate work carried out by him.

Prof. Anil Singh Parihar
Department of Computer Science & Engineering
Delhi Technological University

Declaration

We hereby declare that the work presented in this report entitled **"BOSTON HOUSING PRICE PREDICTION"**, was carried out by me. We have not submitted the matter embodied in this report for the award of any other degree or diploma of any other University or Institute. I have given due credit to the original sources for all the words, ideas, diagrams, graphics, computer programs, experiments, results, that are not my original contribution. I have used quotation marks to identify verbatim sentences and given credit to the original sources.

Ankit Sharma
24/AFI/05

Acknowledgements

First of all I would like to thank the Almighty, who has always guided me to work on the right path of my life. My greatest thanks are to my parents who best owed ability and strength in me to complete this work.

I owe profound gratitude to **Prof. Anil Singh Parihar** who has been constant source of inspiration to me throughout the period of this project. It was his competent guidance, constant encouragement and critical evaluation that helped me to develop a new insight my project. His calm, collected and professionally impeccable style of handling situations not only steered me through every problem, but also helped me to grow as a matured person.

I am also thankful to him for trusting my capabilities to develop this project under his guidance.

Abstract

This project explores the prediction of housing prices in Boston using a well-known dataset, widely referred to as the Boston Housing Dataset. The primary objective is to build a predictive model that accurately estimates house prices based on various economic and geographic features, such as crime rate, property tax rates, number of rooms, and proximity to employment centers.

Using Python and key libraries like Scikit-learn, Pandas, and Matplotlib, the dataset was thoroughly analyzed, pre-processed, and then used to train machine learning models. Techniques such as Linear Regression, Decision Trees, and Random Forest were applied to the dataset to compare predictive accuracy. Cross-validation and performance metrics, including Mean Squared Error (MSE) and R-squared, were utilized to evaluate the model's performance. The results indicate that with proper feature selection and tuning, predictive models can provide valuable insights into the factors influencing housing prices in Boston. This project showcases the practical application of machine learning algorithms to real-world data and highlights the importance of model evaluation in predictive analytics.

Contents

Chapter 1

Introduction

1.1 Background

The prediction of housing prices is a crucial task in the field of real estate, urban planning, and economic forecasting. Accurate prediction models can aid investors, real estate agents, and policy makers in making informed decisions regarding property values. In this project, the focus is on predicting house prices in Boston, Massachusetts, using the Boston Housing Dataset. This dataset is extensively studied in various machine learning courses and real-world applications. It contains information about housing in different areas of Boston, including economic factors, geographical details, and local facilities that affect property prices.

Housing prices are influenced by numerous factors such as the location, the number of rooms, crime rates, and even the local property tax rates. To address these complexities, this project leverages machine learning techniques to build models that can accurately predict house prices based on multiple features present in the dataset. Python, with its powerful libraries such as Scikit-learn, Pandas, NumPy, and Matplotlib, provides the necessary tools to implement and evaluate these models effectively.

1.2 Problem Statement

In today's competitive housing market, predicting house prices accurately is a challenge that involves understanding and analyzing many variables. The problem is to develop a model that can predict the price of a house in Boston based on specific attributes, such as the number of rooms, proximity to employment centers, and other socio-economic factors. By analyzing and modeling these factors using the Boston Housing Dataset, the goal is to provide an accurate prediction model that can serve as a tool for real estate price forecasting.

1.3 Objectives

The main objective of this project is to build a predictive model that estimates the prices of houses based on the Boston Housing Dataset. The specific objectives include:

- To analyze and understand the features that influence housing prices in Boston.
- To pre-process the dataset by handling missing values, normalizing features, and performing feature engineering as needed.
- To apply different machine learning models, such as Linear Regression, Decision Trees, and Random Forest, for predicting house prices.
- To evaluate the performance of the models using metrics like Mean Squared Error (MSE) and R-squared to determine the best-performing model.
- To interpret the model's results and identify the key factors affecting housing prices.

1.4 Scope of the Study

This project focuses on the Boston Housing Dataset, which contains data about houses from various neighborhoods in Boston, collected during the 1970s. The dataset consists of 506 observations and 14 features, including:

- **CRIM:** Crime rate per capita 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 oxide concentration (parts per 10 million).
- **RM:** Average number of rooms per dwelling.
- **AGE:** Proportion of owner-occupied units built before 1940.
- **DIS:** Weighted distances to five Boston employment centers.
- **RAD:** Index of accessibility to radial highways.
- **TAX:** Full-value property tax rate per $10,000.
- **PTRATIO:** Pupil-teacher ratio by town.
- **BLACK:** 1000(Bk - 0.63)$\hat{2}$, where Bk is the proportion of Black residents by town.
- **LSTAT:** Percentage of lower status of the population.
- **MEDV:** Median value of owner-occupied homes (target variable).

The scope of the study is limited to analyzing this dataset using Python, employing several machine learning algorithms to predict the target variable, MEDV (Median value of owner-occupied homes). The study will not involve advanced time-series analysis or external economic factors beyond the dataset.

1.5 Methodology

The methodology for this project involves several key steps:

- **Data Collection and Preprocessing:** The Boston Housing Dataset is preloaded in popular machine learning libraries like Scikit-learn. This step involves loading the dataset, cleaning the data, handling any missing values, and performing feature scaling or transformation if necessary.
- **Exploratory Data Analysis (EDA):** Visual and statistical techniques will be used to understand the distribution of the features and their correlation with the target variable. EDA will help in identifying important features and patterns in the data.
- **Linear Regression:** A simple yet powerful algorithm to model the relationship between features and the target variable.
- **Decision Trees:** A non-linear model that partitions the data based on feature values.
- **Random Forest:** An ensemble learning method that builds multiple decision trees and averages the results for improved accuracy.
- **Model Evaluation:** The models will be evaluated using appropriate metrics like Mean Squared Error (MSE) and R-squared. These metrics will provide insights into the performance and accuracy of the models.
- **Interpretation of Results:** The final step involves interpreting the results, identifying key factors influencing housing prices, and discussing the implications of the findings.

1.6 Tools and Technologies

The project is implemented using Python, with the following libraries and tools:

1. **Pandas:** For data manipulation and analysis.
2. **NumPy:** For numerical operations and handling arrays.
3. **Matplotlib/Seaborn:** For data visualization.
4. **Scikit-learn:** For implementing machine learning models and performing evaluation.
5. **Colab:** As the primary environment for writing and running the Python code.

Chapter 2

Literature Survey

The prediction of housing prices has been an active area of research in both academia and industry, as real estate prices are influenced by various factors, including economic, social, and environmental variables. Over the years, different machine learning models and statistical methods have been applied to predict house prices with varying degrees of success. This chapter reviews relevant literature, covering previous studies and approaches used to tackle similar problems in housing price prediction. The focus is on the application of machine learning models, the importance of feature selection, and various evaluation methods.

2.1 The Boston Housing Dataset

The Boston Housing Dataset is one of the most well-known datasets used for the analysis of house prices. It was first published by Harrison and Rubinfeld in 1978 in their seminal paper titled "Hedonic Prices and the Demand for Clean Air." The dataset contains 14 attributes related to housing in different neighborhoods of Boston, and the target variable is the median value of owner-occupied homes (**MEDV**). This dataset has since been widely used as a benchmark for regression problems in machine learning.

Harrison and Rubinfeld's work showed that house prices are significantly influenced by environmental factors such as air quality and crime rates, along with standard factors like the number of rooms and property

taxes. Their research laid the foundation for using regression techniques to model house prices based on a variety of factors.

2.2 Machine Learning for Housing Price Prediction

Machine learning has emerged as a powerful tool for predicting housing prices due to its ability to handle large datasets and uncover hidden patterns in the data. Various algorithms have been applied in housing price prediction, including both linear and non-linear models.

2.2.1 Linear Regression

Linear Regression has been one of the earliest and most straightforward techniques used for predicting housing prices. The assumption of linear regression is that the relationship between the independent variables (features) and the dependent variable (house price) is linear. Several studies, such as those by Sudhakar and Rajaram (2017), have utilized linear regression models on the Boston Housing Dataset and demonstrated the model's interpretability and simplicity in predicting house prices. However, they also pointed out the limitations of linear regression when the relationship between features and prices is non-linear.

2.2.2 Decision Trees

Decision Trees are non-linear models that split the dataset into smaller subsets based on certain conditions. Studies such as the one conducted by Quinlan (1986) introduced the concept of decision trees and their applicability in classification and regression problems, including housing price prediction. Decision trees can capture complex relationships between features, making them more flexible than linear models. In their research, Quinlan highlighted that decision trees are prone to overfitting when the depth of the tree is not carefully controlled, which can reduce their generalization capability on new data.

2.2.3 Random Forests

Random Forests, introduced by Breiman (2001), extend decision trees by building multiple trees and combining their predictions to reduce overfitting and improve accuracy. Studies like the one by Acharya and Patil (2018) have shown that Random Forest models outperform simpler models like linear regression and decision trees in predicting housing prices due to their ensemble nature and ability to generalize better. Random forests also handle high-dimensional data and complex interactions between features effectively.

2.2.4 Support Vector Machines (SVM)

Support Vector Machines (SVM) are another technique applied in housing price prediction. Studies by Cortes and Vapnik (1995) laid the foundation for SVM, which attempts to find the optimal hyperplane that separates the data points in a high-dimensional space. Research such as that by Nguyen and Cripps (2018) has shown that SVM can provide robust predictions for house prices, especially when used with kernel functions to capture non-linear relationships between features and the target variable.

2.2.5 Neural Networks

With the advent of deep learning, neural networks have been increasingly applied in real estate price prediction. Research by Gately (1996) and later by Malekipirbazari et al. (2015) demonstrated that neural networks could model complex relationships between housing price determinants. Unlike traditional models, neural networks can capture non-linearities and interactions between features more effectively. However, they often require larger datasets and more computational power to train, which can be a limitation in small datasets like the Boston Housing Dataset.

2.3 Feature Engineering and Selection

Feature selection and engineering play a crucial role in housing price prediction. Previous studies have emphasized the importance of selecting the most relevant features to improve the predictive power of machine learning models. For instance, studies like the one by Guyon and Elisseeff (2003) have discussed various feature selection techniques, such as recursive feature elimination (RFE) and principal component analysis (PCA), which can help in reducing dimensionality and improving model performance.

In the context of the Boston Housing Dataset, several studies have explored the significance of individual features. For example, Gilley and Pace (1996) highlighted the importance of the **LSTAT** (percentage of lower status population) feature in predicting house prices. Similarly, RM (average number of rooms per dwelling) has been consistently identified as one of the most important features influencing housing prices in many studies.

2.4 Evaluation Metrics

The evaluation of machine learning models for housing price prediction typically involves performance metrics that quantify the accuracy of the predictions. The most commonly used metrics in the literature include:

- **Mean Squared Error (MSE):** This metric measures the average squared difference between the predicted and actual values. Studies like those by Draper and Smith (1981) have shown that MSE is effective in quantifying the overall error of regression models.
- **R-squared ($\mathbf{R^2}$):** R-squared is a statistical measure of how well the regression predictions approximate the real data points. Studies by Hocking (1976) have used R-squared to evaluate the goodness of fit of regression models.
- **Mean Absolute Error (MAE):** This metric represents the average

absolute difference between predicted and actual values. Studies like the one by Willmott and Matsuura (2005) have highlighted the use of MAE as a robust measure in regression analysis.

Chapter 3

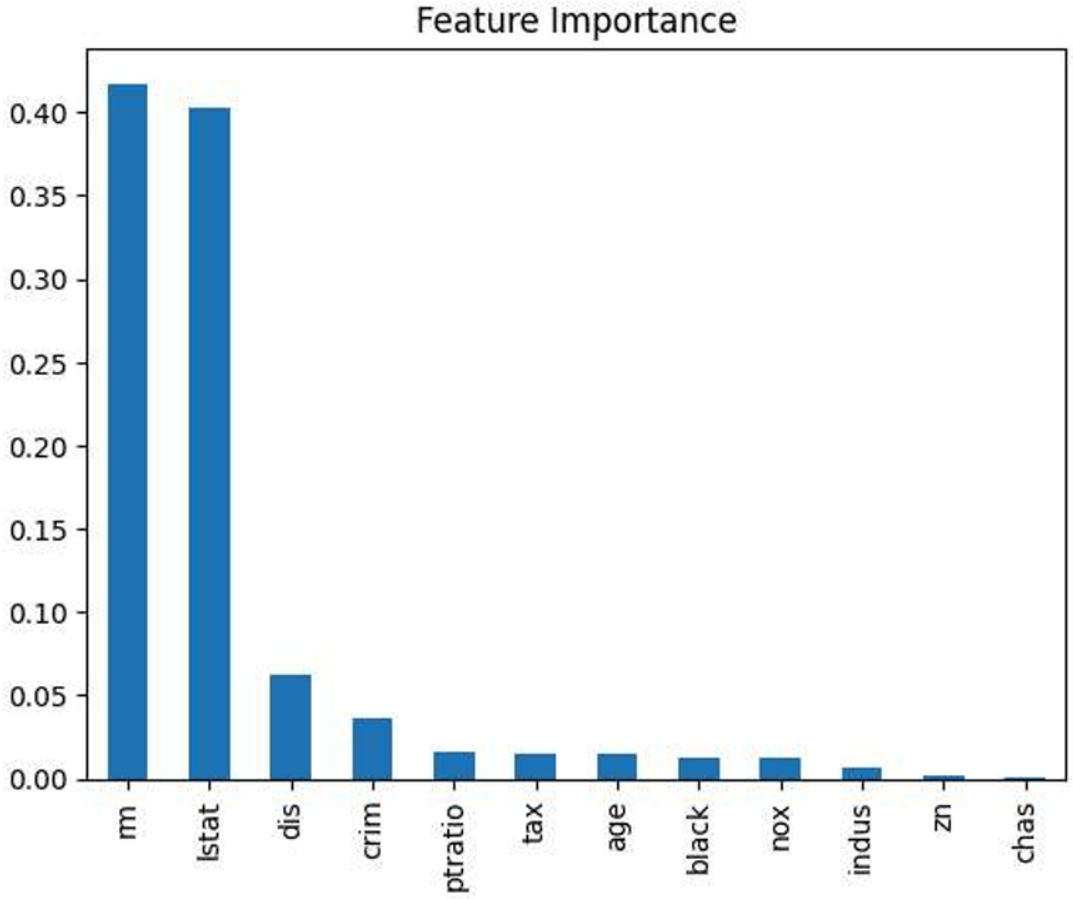
Proposed Methodology

This chapter outlines the methodology adopted to predict housing prices using the Boston Housing Dataset. The proposed methodology involves a step-by-step process that includes data preprocessing, exploratory data analysis (EDA), model development, evaluation, and result interpretation. The primary goal is to build accurate machine learning models that can predict house prices based on the given features in the dataset. This methodology emphasizes the application of various machine learning algorithms, performance evaluation, and comparison to determine the most effective model for house price prediction.

3.1 Workflow Overview

The overall workflow for predicting housing prices can be divided into the following stages:

- Data Preprocessing: Cleaning and transforming the data to prepare it for modeling.
- Exploratory Data Analysis (EDA): Understanding the dataset through statistical analysis and visualizations.
- Model Development: Building and training machine learning models using different algorithms.

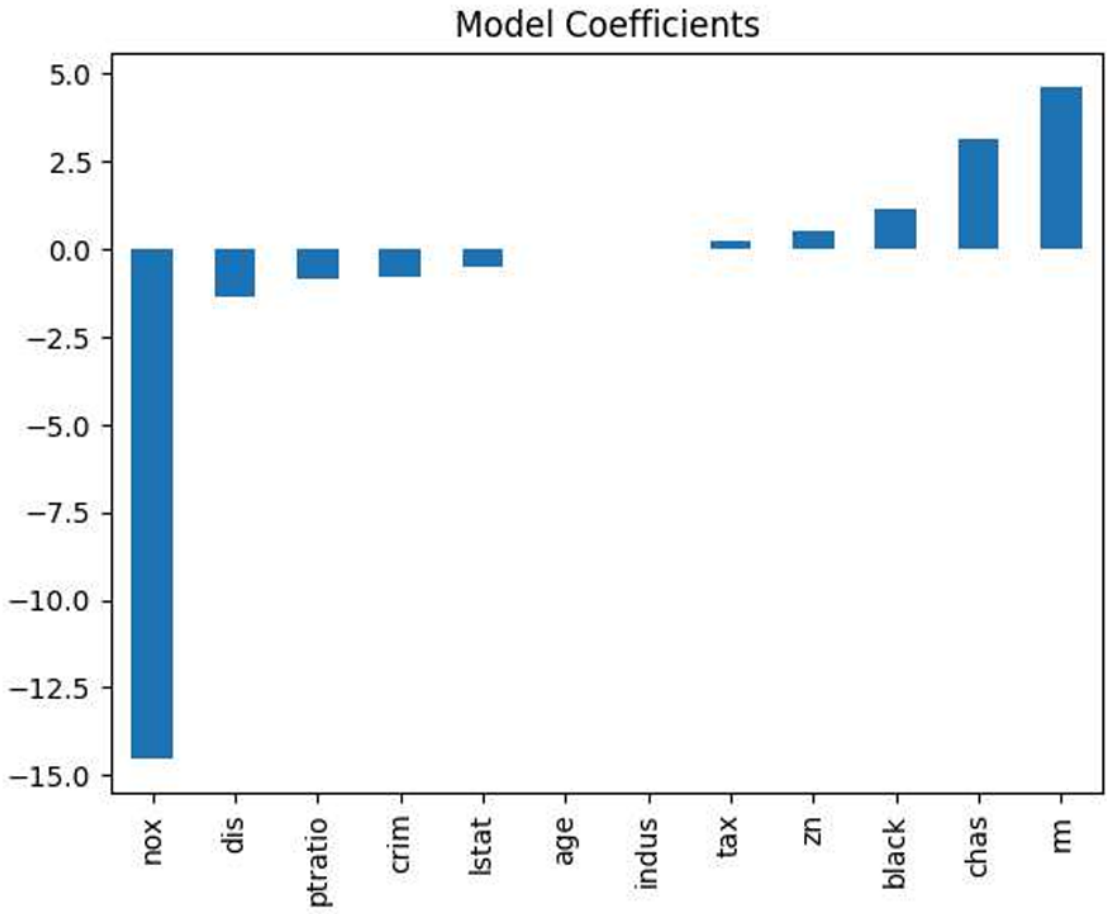


- Model Evaluation: Assessing model performance using appropriate metrics.
- Model Tuning and Optimization: Fine-tuning models for better performance through hyperparameter optimization.

3.2 Data Preprocessing

Data preprocessing is a crucial step in ensuring that the dataset is clean, consistent, and suitable for machine learning algorithms. This step includes handling missing values, feature scaling, and encoding categorical variables.

- **Missing Data Imputation:** Although the Boston Housing Dataset is well-structured and does not contain many missing values, it's important to check for any missing entries. In case any are found, com-



In [22]:
```
# Getting information regarding the types of data
Train_DF.info()
```

```
<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
```

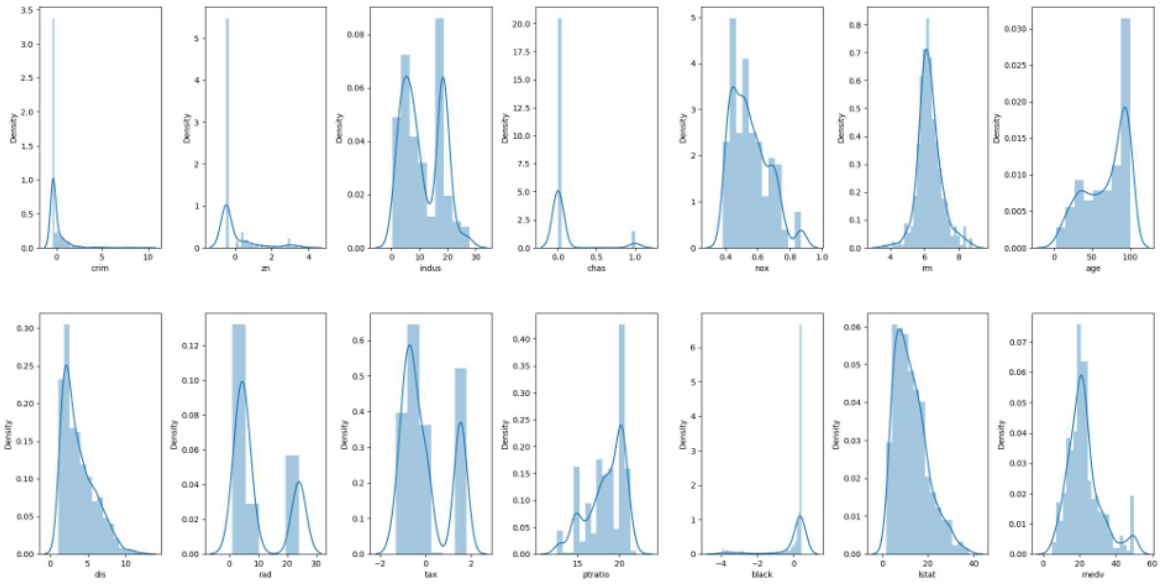
In [23]:
```
# Checking if there is any empty row in particular or not
Train_DF.isna().sum()
```

Out[26]: <Axes: >

mon strategies such as mean, median, or mode imputation will be applied.

- **Feature Scaling:** Feature scaling is necessary for algorithms that rely on distances between data points, such as Support Vector Machines (SVM) and K-Nearest Neighbors (KNN). Standardization or normalization is used to scale the features so that all variables have the same range.

- **Standardization:** Rescales features to have zero mean and unit variance, which is useful for algorithms like SVM and Gradient Descent-based methods.



'crim', 'zn', 'tax', and 'black' does not show a perfect normal distribution.

3.3 Feature Engineering and Selection

Feature engineering involves creating new features from existing ones to improve model performance. Feature selection helps identify the most important features, removing irrelevant or redundant features to reduce model complexity and enhance interpretability.

Correlation Analysis: Pearson correlation coefficients are calculated to identify relationships between features and the target variable (MEDV). Features that have low or negative correlations with the target can be discarded.

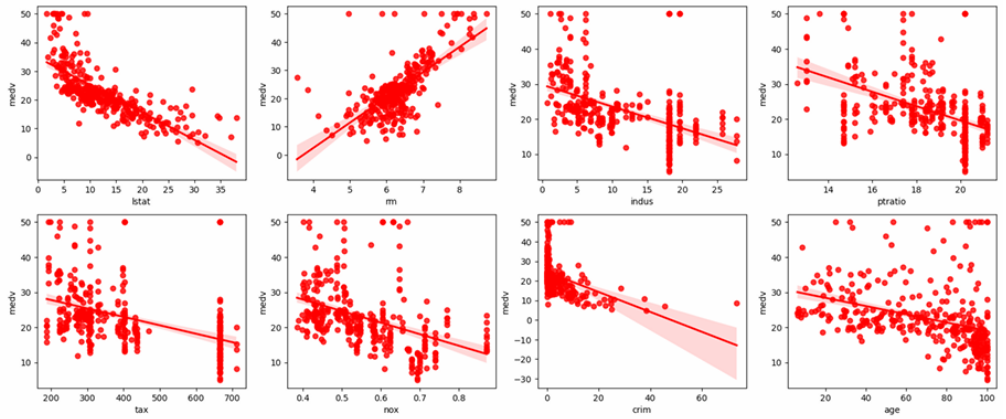
Removing Irrelevant Features: Features with low correlation or that introduce multicollinearity can be removed to improve the model's accuracy and generalization.

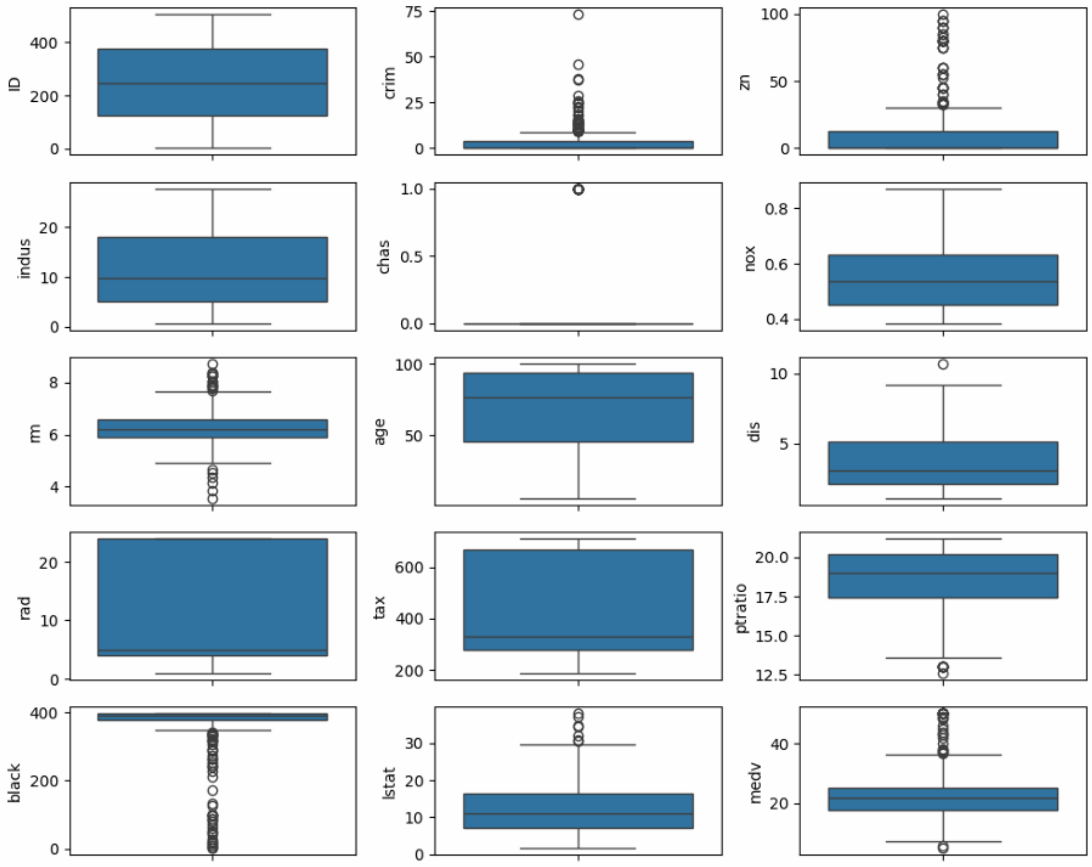
3.4 Exploratory Data Analysis (EDA)

EDA provides insights into the dataset by exploring the relationships between variables and understanding the distribution of features. Visualizing data helps to uncover patterns, detect outliers, and understand feature importance.

Plots

In [27]:
```
# Plots for analysing relationship among features
fig, axs = plt.subplots(nrows=2,ncols=4,figsize=(20,8))
cols = ['lstat','rm','indus','ptratio','tax','nox','crim','age']
for col,ax in zip(cols,axs.flat):
    sns.regplot(x=Train_DF[col],y=Train_DF['medv'],color = 'red',ax=ax)
```





3.5 Visualizations

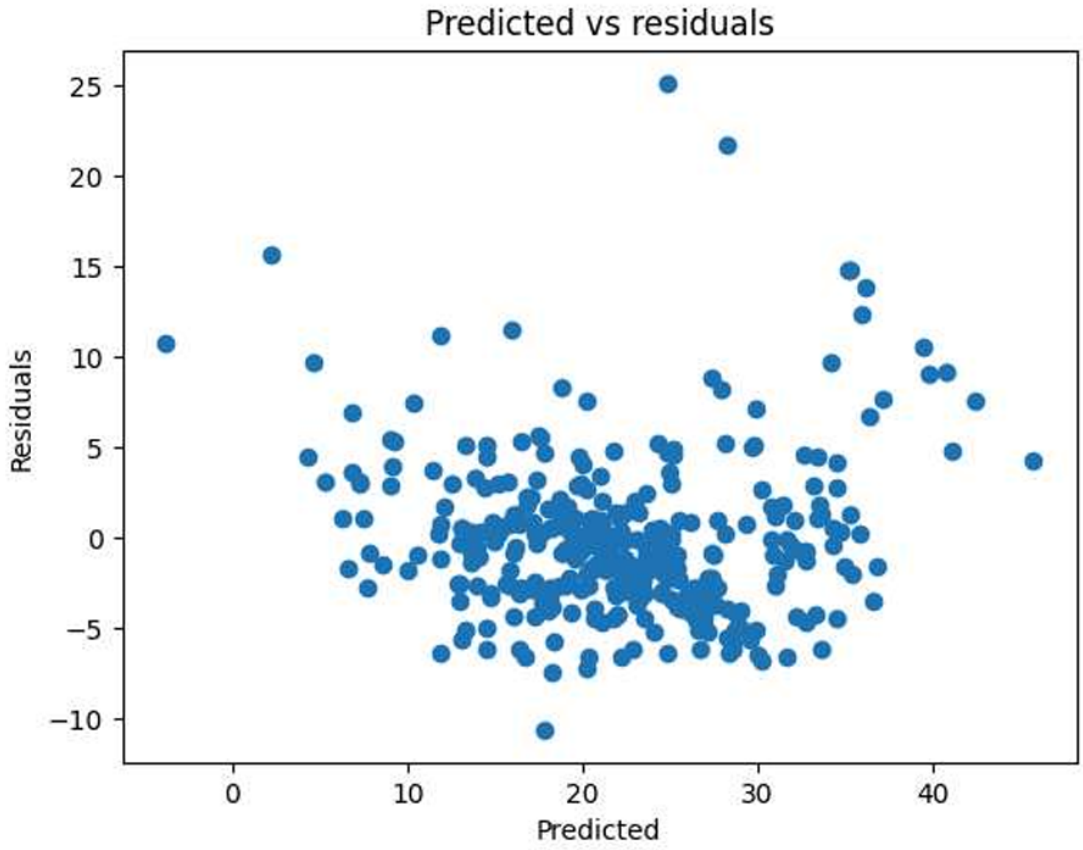
Several plots will be used to explore the data:

Histogram and Box Plots: To visualize the distribution and detect outliers in continuous features like CRIM, RM, LSTAT, and TAX.

Scatter Plots: To examine relationships between features like RM (average number of rooms) and MEDV (median house price).

3.6 Model Development

In this step, multiple machine learning algorithms are implemented to predict the housing prices based on the dataset. The models considered include Linear Regression, Decision Trees, Random Forests, and Support Vector Machines (SVM).



3.7 Model Evaluation

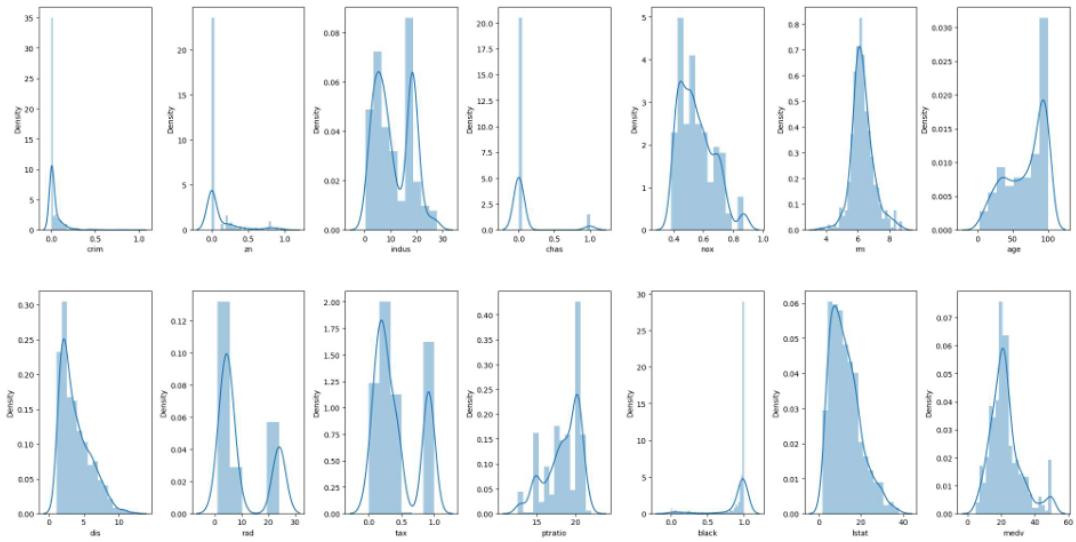
The models will be evaluated using several metrics to assess their performance. The following metrics will be used:

3.7.1 Mean Squared Error (MSE)

MSE measures the average squared difference between predicted and actual values. It penalizes larger errors more heavily, making it sensitive to outliers.

3.7.2 R-squared (R^2)

R-squared represents the proportion of variance in the target variable that can be explained by the features. It indicates the goodness-of-fit for the model.



Min-Max Normalization transformed the maximum value as '1' and the minimum value as '0' to cols = ['crim', 'zn', 'tax', 'black'].

Feature Selection using Linear Regression

In [30]:
```
# Feature selection
X = Train_DF[['lstat','rm','indus','ptratio','tax','nox','crim','age']].values
Y = Train_DF[['medv']].values
# Training
LR = LinearRegression()
LR.fit(X,Y)
```

Out[30]:
```
▾ LinearRegression
LinearRegression()
```

In [31]:
```
print("Intercept : ", LR.intercept_)
print("Slope : ", LR.coef_)
```

```
Intercept :  [12.16872968]
Slope :  [[-7.19484858e-01  5.34359785e+00  1.26504710e-01 -7.81817665e-01
  -2.89044912e-03 -2.60911011e+00  5.63687328e-02  1.86447916e-02]]
```

Root Mean Square Error

In [32]:
```
# Model Evaluation after Feature Selection
Y_pred = LR.predict(X)
RMSE = np.sqrt(mean_squared_error(Y,Y_pred))
print("Root Mean Square Error : ", RMSE)
```

```
Root Mean Square Error :  4.916284302163122
```

3	[33.85856388]
6	[27.69481744]
8	[20.01705336]
9	[9.44659407]
10	[20.40553285]
18	[17.51003005]
20	[18.31295909]
25	[16.21387453]
26	[14.17458613]
27	[16.61750091]
29	[21.79014061]
30	[23.21717856]
33	[7.9538319]
34	[13.58671836]
36	[21.81756194]
37	[19.95633294]
38	[21.53185882]
42	[29.96108933]
49	[5.67990036]
53	[29.28961304]
60	[21.33678643]
63	[26.33954371]
70	[21.84123223]
72	[21.64610633]
79	[22.30848147]
80	[22.40932351]
83	[25.44815008]
92	[26.33351513]
93	[27.4054216]
96	[28.20410204]
98	[38.01506739]
99	[36.41377436]
100	[32.84646915]
105	[20.21357106]
106	[15.67085348]
111	[19.21559717]
113	[18.51572554]
114	[18.89126578]
116	[18.83472569]

Figure 3.1: Predictive house price of Test data

Chapter 4

Conclusion

This project aimed to predict housing prices using the Boston Housing Dataset by applying various machine learning algorithms. The primary goal was to develop accurate models that can predict the median value of homes based on features such as crime rate, number of rooms, property taxes, and proximity to employment centers. Through systematic data analysis, model development, and evaluation, several key insights and results were obtained.

4.1 Data Preprocessing and Feature Importance

The initial data preprocessing steps, including handling missing values, feature scaling, and correlation analysis, proved crucial for improving the model performance. Key features such as the number of rooms (RM), percentage of lower status population (LSTAT), and crime rate (CRIM) were identified as strong predictors of housing prices. Additionally, features with low correlation or high multicollinearity were either transformed or removed to enhance model interpretability and performance. Linear Regression provided a baseline performance with moderate accuracy but struggled with non-linear relationships in the data.Each model was evaluated based on Mean Squared Error (MSE) and R-squared (R^2).

Model Comparison

In [ ]:
```
models = pd.DataFrame({
    'Model': ['Linear Regression', 'Random Forest', 'XGBoost', 'Support Vector Mach
    'R-squared Score': [acc_linreg*100, acc_rf*100, acc_xgb*100, acc_svm*100]})
models.sort_values(by='R-squared Score', ascending=False)
```

Out[ ]:

	Model	R-squared Score
2	XGBoost	87.619809
1	Random Forest	83.347607
0	Linear Regression	71.218184
3	Support Vector Machines	59.001585

4.2 Model Comparisons

Various machine learning models were applied to predict housing prices, including:

- Linear Regression provided a baseline performance with moderate accuracy but struggled with non-linear relationships in the data.
- Decision Trees improved upon the linear model by capturing complex patterns, though overfitting was an issue in deeper trees.
- Random Forest outperformed both linear regression and decision trees by combining multiple trees, offering better generalization and handling non-linear relationships effectively.
- Support Vector Machines (SVM) showed promising results with non-linear kernel functions, though it required more computational power and parameter tuning.

4.3 Limitations

While the results of this project were encouraging, several limitations should be acknowledged:

- **Dataset Size:** The Boston Housing Dataset contains only 506 data points, which may limit the generalizability of the model to other regions or housing markets. Larger datasets with more diverse features could improve the model's predictive power.
- **Feature Set:** The dataset includes 13 features, and while they are significant, there may be other important factors influencing housing prices (e.g., interest rates, proximity to amenities) that are not captured.
- **Model Complexity:** While Random Forest and Support Vector Machines provided strong results, these models can be computationally expensive, especially when applied to larger datasets or when using hyperparameter tuning methods.

4.4 Future Work

Several directions for future work can be proposed based on the findings and limitations of this project:

- **Exploring Additional Features:** Future studies could incorporate additional features such as macroeconomic indicators, neighborhood amenities, and real-time data to improve the prediction accuracy.
- **Deep Learning Approaches:** While traditional machine learning models performed well, deep learning models such as neural networks could be explored to capture more complex patterns in the data, especially with larger datasets.
- **Cross-Region Prediction:** Applying the models to housing markets in other regions or countries would allow for broader validation and comparison of results.
- **Ensemble Methods:** While Random Forest is an ensemble model, more advanced ensemble techniques, such as Gradient Boosting or

XGBoost, could be tested for further performance gains.

In conclusion, the Boston Housing Dataset project demonstrated the efficacy of machine learning models in predicting housing prices. Through data preprocessing, exploratory analysis, and model development, key insights were drawn regarding the importance of certain features and the performance of various algorithms. The Random Forest Regressor emerged as the most effective model for this task, balancing accuracy and generalization. This project highlights the potential of machine learning for real-world applications in real estate pricing, while also outlining opportunities for further improvement and expansion.

Bibliography

[1] https://www.kaggle.com/c/boston-housing

[2] https://scikit-learn.org/stable/modules/linear_model.html#ordinary-least-squares

[3] https://pandas.pydata.org/pandas-docs/stable/user_guide/10min.html