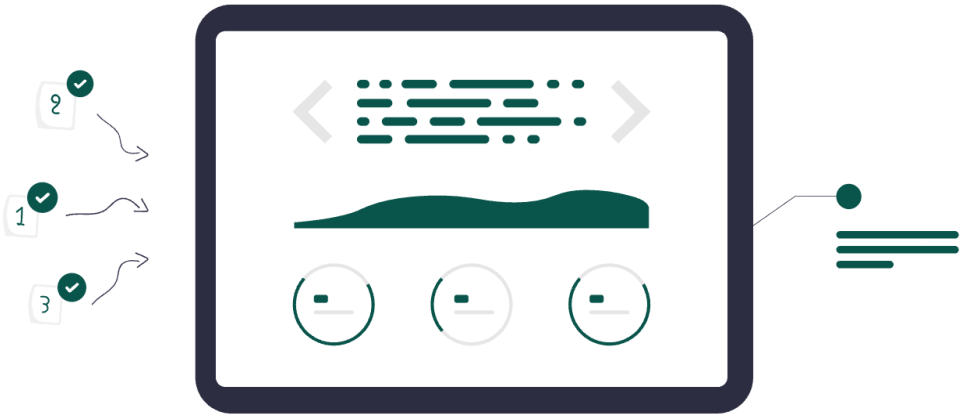
Assignment of CSE303

Exploratory Data Analysis on Boston Housing Dataset



Submitted By: Team 4

Samia Rahman	2022-3-60-087	
Suraiya Akther Maisha	2022-3-60-071	
MD. Mehedi Hasan	2022-3-60-119	
Aklhak Hossain	2022-3-60-057	https://akhlak.dev

Submitted To:

Dipayan Bhadra
Adjunct Faculty
Department of CSE
East West University

Overview

The Boston Housing dataset consists of data collected by the U.S. Census Service on housing in the Boston metropolitan area. It involves a number of features describing residential properties, socioeconomic indicators, and environmental variables within different neighborhoods.

The dataset is typically used to examine influences on housing prices.

There is one row for each neighborhood or housing district. The features have both numerical and categorical data, and these include:

- Crime rate per town
- Proportion of residential area zoned for large lots
- Mean number of rooms per unit
- Pupil-teacher ratio by town
- Access to highways
- Property tax rate
- Median home value (MEDV), the target variable

The data can be utilized for exploratory analysis, prediction modeling, and studying the association between house prices and location variables.

Data Types of Columns

Column	Python Type	Qualitative / Quantitative	Nominal / Ordinal	Discrete / Continuous	Description
crim	float64	Quantitative	–	Continuous	Crime rate per capita
zn	float64	Quantitative	–	Continuous	Proportion of residential land zoned for large lots
indus	float64	Quantitative	–	Continuous	Proportion of non-retail business acres
chas	int64	Qualitative	Nominal	Discrete	Charles River dummy variable (1 if tract bounds river, 0 otherwise)
nox	float64	Quantitative	–	Continuous	Nitric oxide concentration (pollution level)
rm	float64	Quantitative	–	Continuous	Average number of rooms per dwelling
age	float64	Quantitative	–	Continuous	Proportion of owner-occupied units built before 1940
dis	float64	Quantitative	–	Continuous	Distance to employment centers
rad	int64	Quantitative	Ordinal	Discrete	Index of accessibility to radial highways
tax	int64	Quantitative	–	Discrete	Property tax rate per $10,000
ptratio	float64	Quantitative	–	Continuous	Pupil-teacher ratio by town
black	object	Quantitative (stored as object)	–	Continuous	1000(Bk - 0.63)^2 where Bk is the proportion of Black residents
lstat	float64	Quantitative	–	Continuous	% of lower status population
medv	float64	Quantitative (Target)	–	Continuous	Median value of owner-occupied homes (in $1000s)

Data Cleaning

Check for missing values:

We can check the missing data values using pandas `isnull()` function
We can get count and ration using the function and mathematics

```
print("\nMissing Data %:")
print(df.isnull().mean() * 100)
print("\nMissing Data Count:")
print(df.isnull().sum())
```

Which will provide us with the informations of the columns as below:

```
Missing Data %:
crim       0.000000
zn         0.000000
indus      0.395257
chas       0.000000
nox        0.197628
rm         0.000000
age        0.000000
dis        0.197628
rad        0.000000
tax        0.000000
ptratio    0.197628
black      0.197628
lstat      0.197628
medv       0.000000
dtype: float64
```

```
Missing Data Count:
crim       0
zn         0
indus      2
chas       0
nox        1
rm         0
age        0
dis        1
rad        0
tax        0
ptratio    1
black      1
lstat      1
medv       0
dtype: int64
```

Based on the data type and missing values we cleaned up the data by deciding the appropriate methods as shown in the below table:

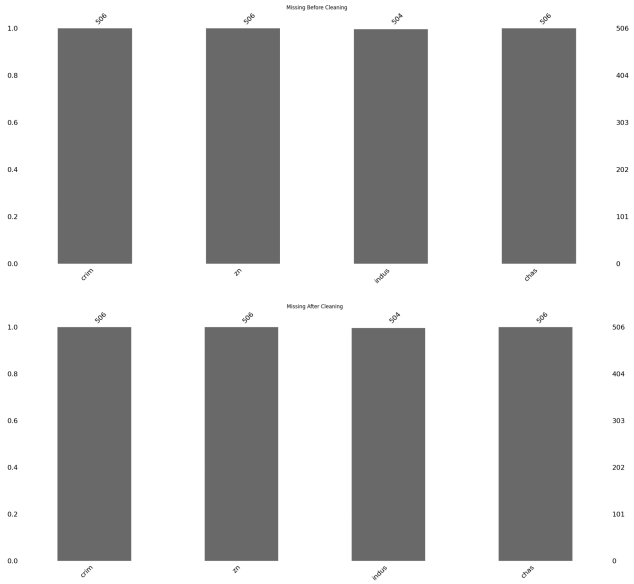
Column	Missing Count	% Missing	Type	Action
indus	2	0.39%	Continuous	Fill with **median**
nox	1	0.20%	Continuous	Fill with **median**
dis	1	0.20%	Continuous	Fill with **median**
ptratio	1	0.20%	Continuous	Fill with **median**
black	1	0.20%	Continuous, but object type	Convert to numeric and fill with **median**
lstat	1	0.20%	Continuous	Fill with **median**

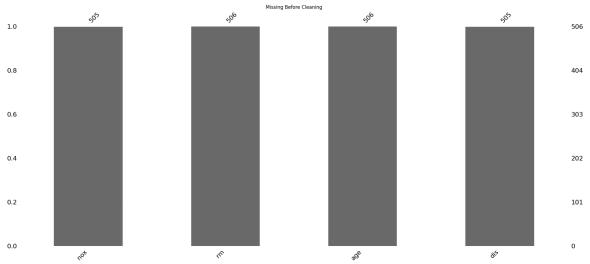
Data Cleanup

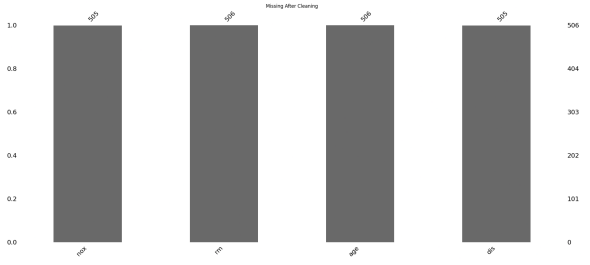
We can run a loop and choose a better data cleanup method based on the data type it self, we can write the below code as such, but before that we should convert the data types that are not numeric like the “black” data field

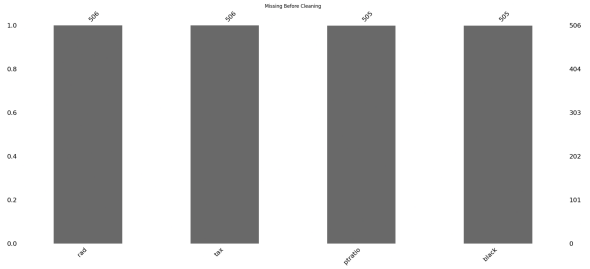
```
df_cp['black'] = pd.to_numeric(df_cp['black'], errors='coerce')

for col in df:
    if df[col].dtype == 'object':
        df[col].fillna(df[col].mode()[0], inplace=True)
    else:
        df[col].fillna(df[col].median() if df[col].skew() > 1 else
df[col].mean(), inplace=True)
```

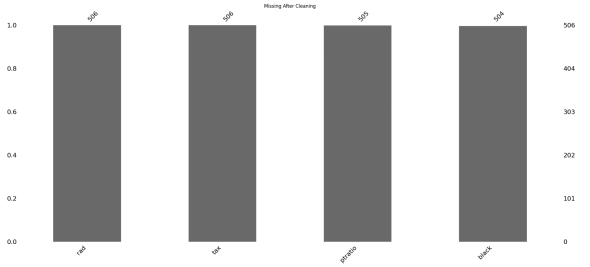
Which will fill the missing data based on the data type.
The barchart before and after data cleanup

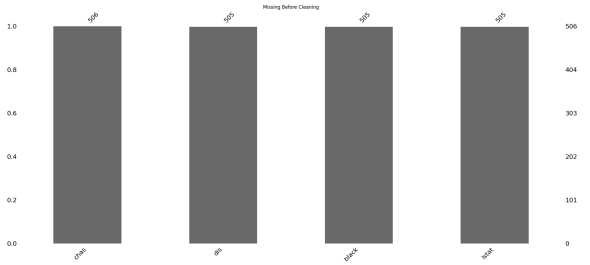


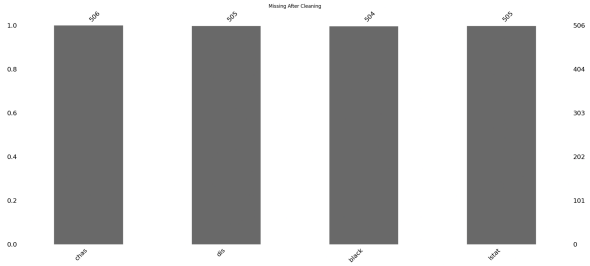

Missing Before Cleaning
505
506
506
505
1.0
0.8
0.6
0.4
0.2
0.0
506
404
303
202
101
0
nox
rm
age
dis


Missing After Cleaning
505
506
506
505
1.0
0.8
0.6
0.4
0.2
0.0
506
404
303
202
101
0
nox
rm
age
dis


Missing Before Cleaning
506
506
505
505
1.0
0.8
0.6
0.4
0.2
0.0
506
404
303
202
101
0
real
tax
ptratio
black


Missing After Cleaning
506
506
505
504
1.0
0.8
0.6
0.4
0.2
0.0
506
404
303
202
101
0
real
tax
ptratio
black

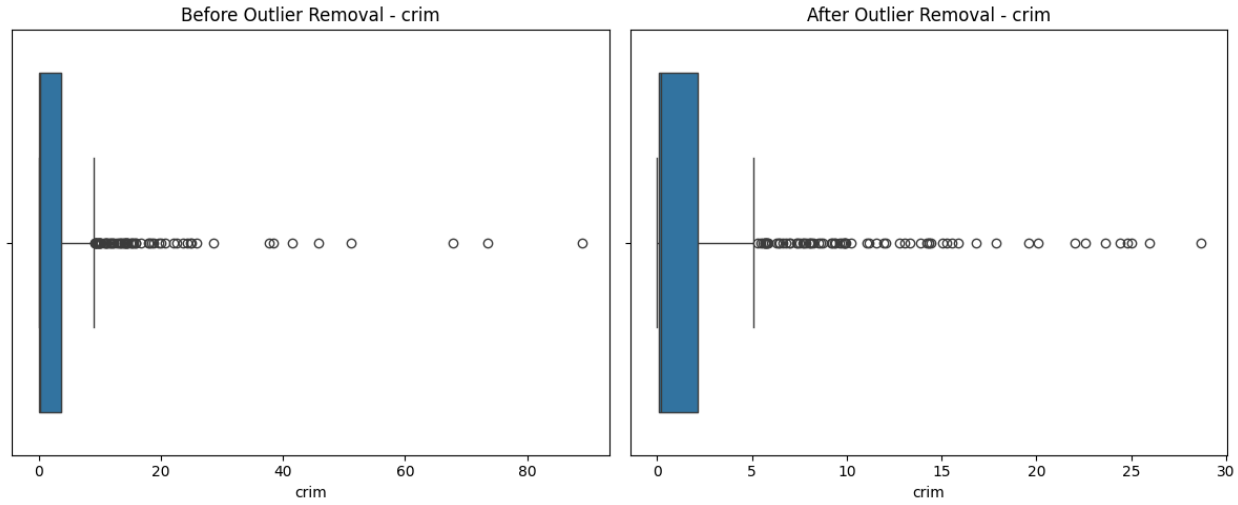

Missing Before Cleaning
506
505
505
505
1.0
0.8
0.6
0.4
0.2
0.0
506
404
303
202
101
0
chas
dis
black
stat

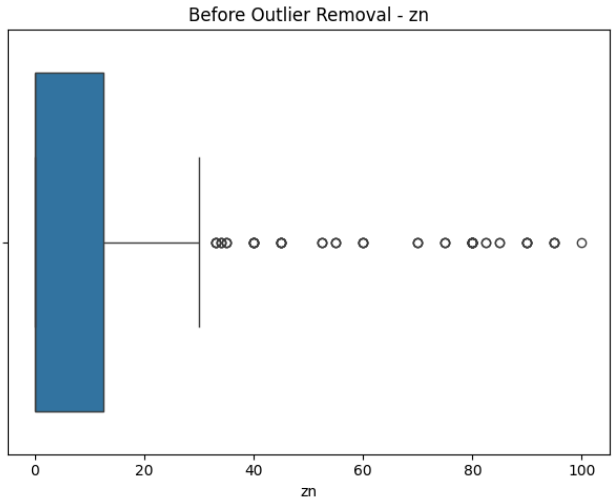

Missing After Cleaning
506
505
504
505
1.0
0.8
0.6
0.4
0.2
0.0
506
404
303
202
101
0
chas
dis
black
stat

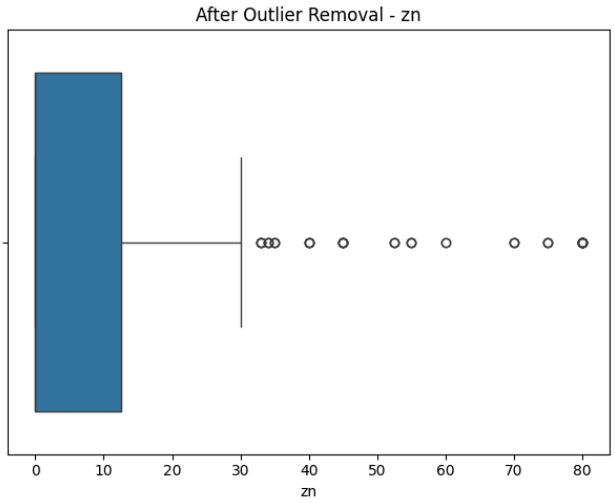
Outlier Detection

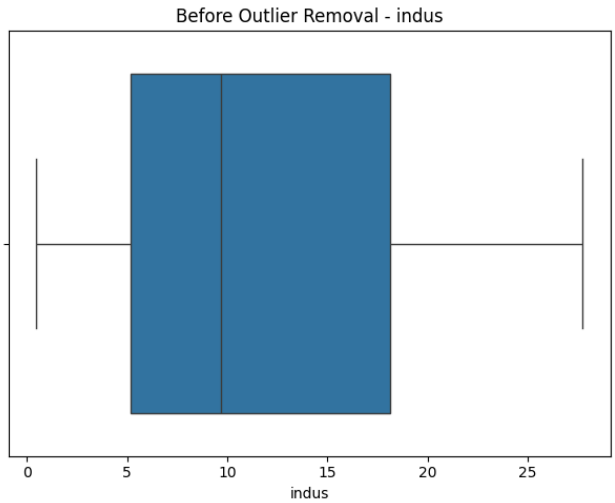
We can write a function to detect the outliers of the data and handle the outliers

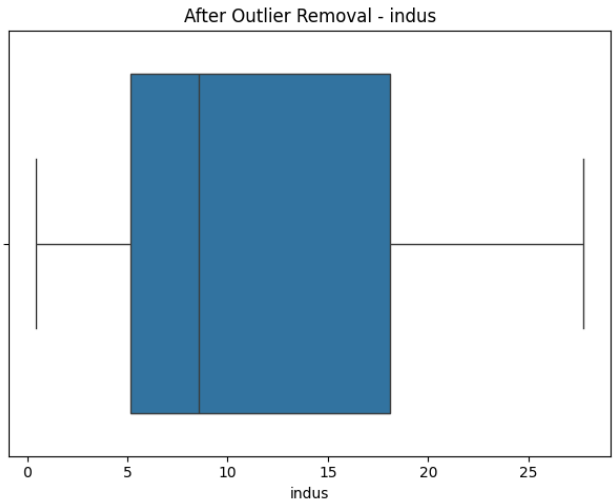
```
# Detect outliers using Z-score
z = np.abs(stats.zscore(df_cp))
df_cp_no_outliers = df_cp[(z < 3).all(axis=1)]

# Plot side-by-side boxplots
for col in df_cp:
    fig, axes = plt.subplots(1, 2, figsize=(12, 5))

    sns.boxplot(x=df_cp[col], ax=axes[0])
    axes[0].set_title(f'Before Outlier Removal - {col}')

    sns.boxplot(x=df_cp_no_outliers[col], ax=axes[1])
    axes[1].set_title(f'After Outlier Removal - {col}')

    plt.tight_layout()
    plt.show()
```

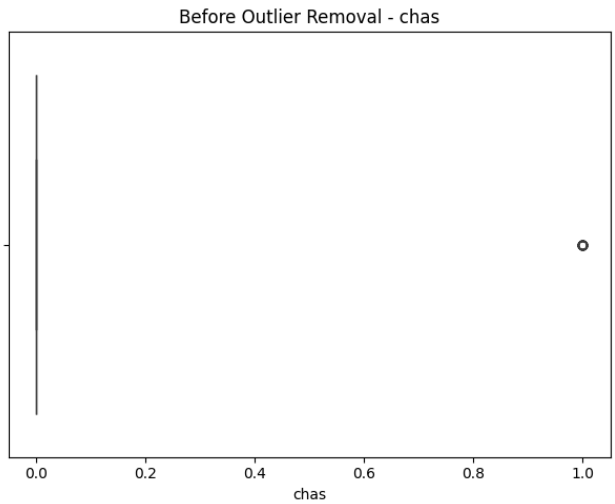


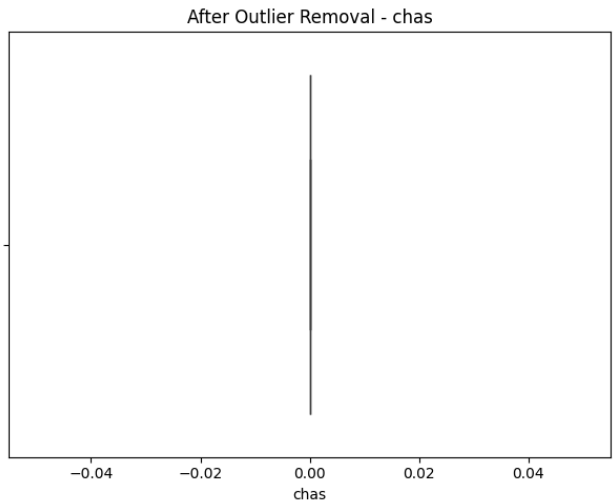

Before Outlier Removal - zn
0
20
40
60
80
100
zn

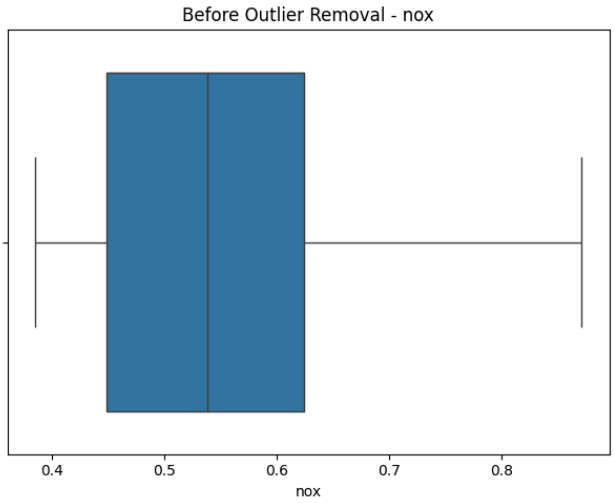

After Outlier Removal - zn
0
10
20
30
40
50
60
70
80
zn

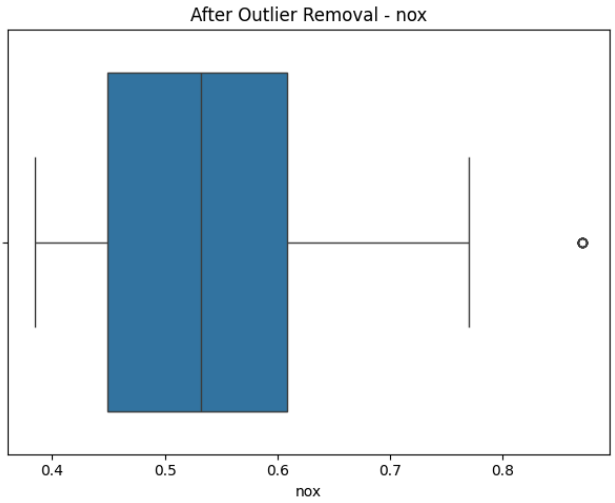

Before Outlier Removal - indus
0
5
10
15
20
25
indus

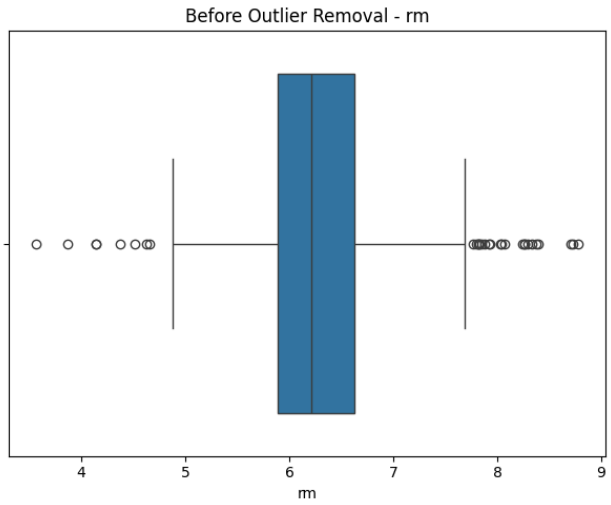

After Outlier Removal - indus
0
5
10
15
20
25
indus

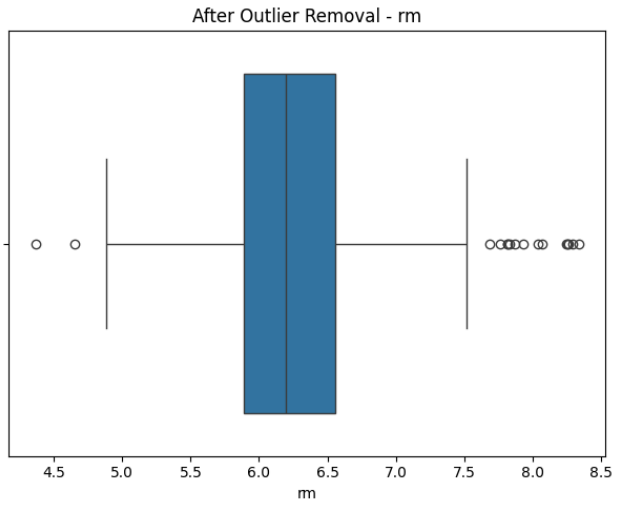

Before Outlier Removal - chas
0.0
0.2
0.4
0.6
0.8
1.0
chas

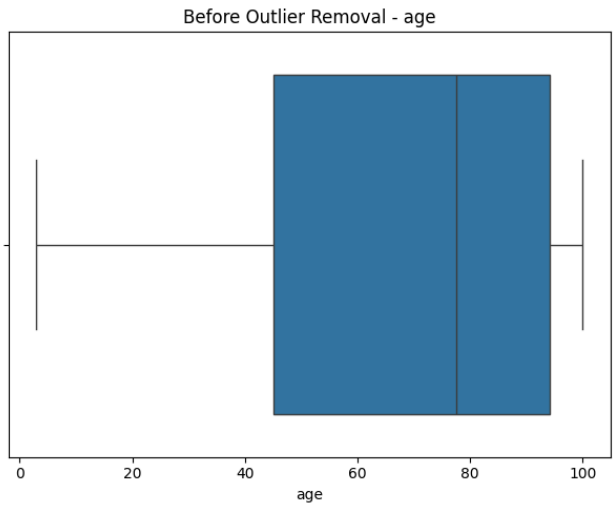

After Outlier Removal - chas
−0.04
−0.02
0.00
0.02
0.04
chas

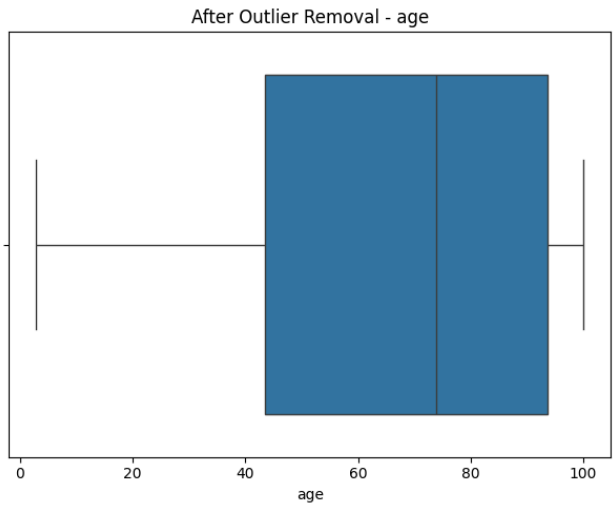

Before Outlier Removal - nox
0.4
0.5
0.6
0.7
0.8
nox

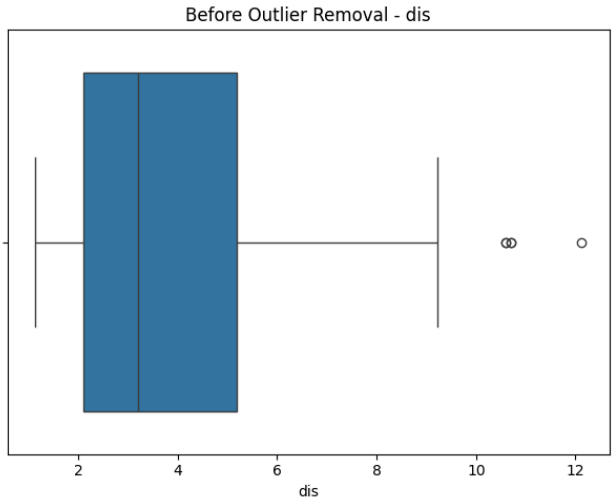

After Outlier Removal - nox
0.4
0.5
0.6
0.7
0.8
nox

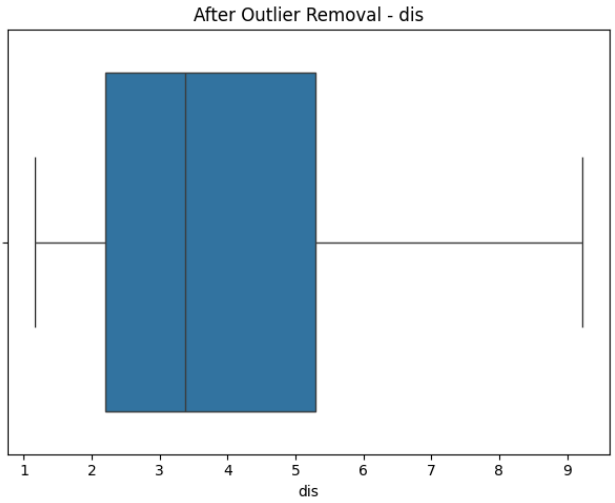

Before Outlier Removal - rm
4
5
6
7
8
9
rm

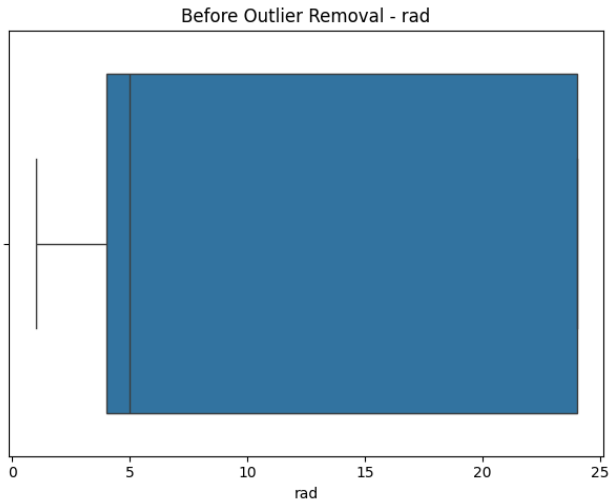

After Outlier Removal - rm
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5
rm

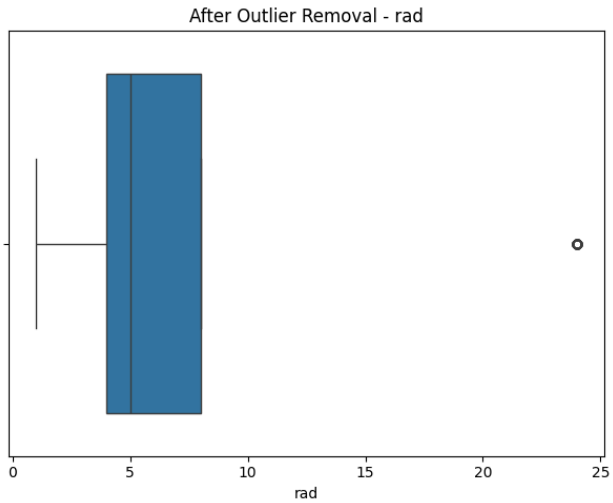

Before Outlier Removal - age
0
20
40
60
80
100
age

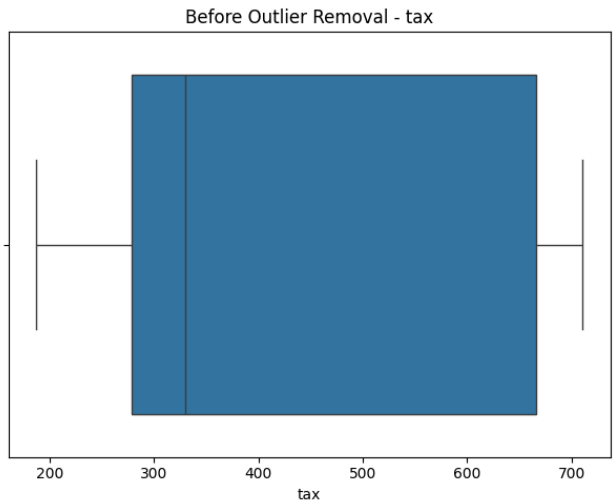

After Outlier Removal - age
0
20
40
60
80
100
age

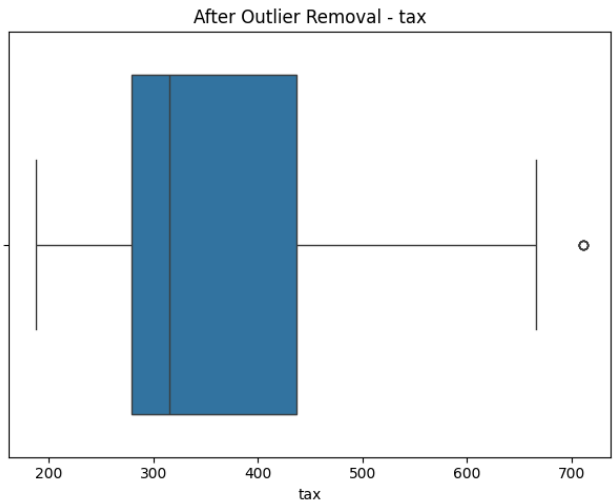

Before Outlier Removal - dis
2
4
6
8
10
12
dis

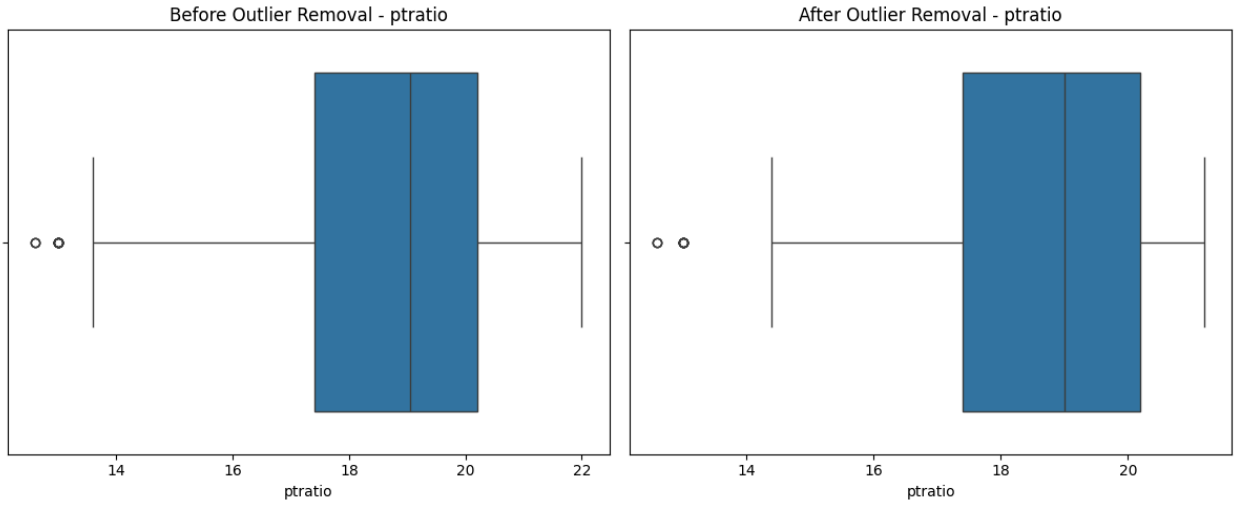

After Outlier Removal - dis
1
2
3
4
5
6
7
8
9
dis


Before Outlier Removal - rad
0
5
10
15
20
25
rad

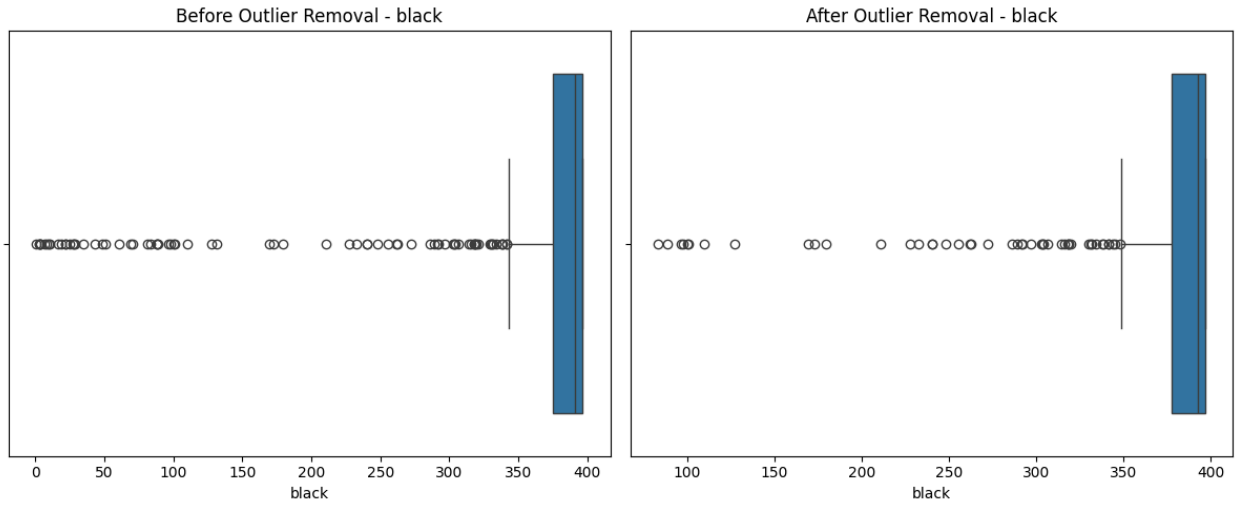

After Outlier Removal - rad
0
5
10
15
20
25
rad


Before Outlier Removal - tax
200
300
400
500
600
700
tax

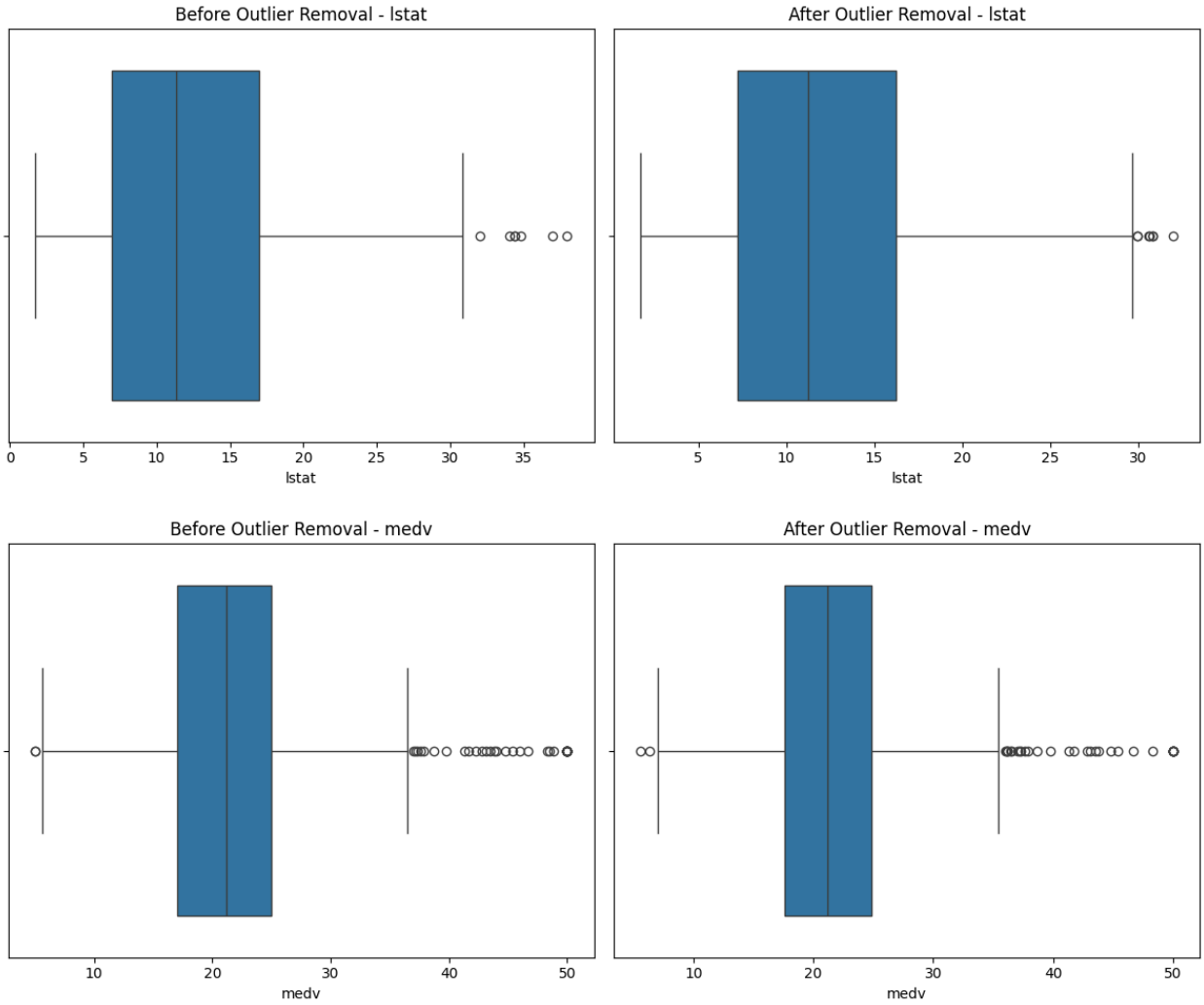

After Outlier Removal - tax
200
300
400
500
600
700
tax


Before Outlier Removal - ptratio
14
16
18
20
22
ptratio

After Outlier Removal - ptratio
14
16
18
20
ptratio


Before Outlier Removal - black
0
50
100
150
200
250
300
350
400
black

After Outlier Removal - black
100
150
200
250
300
350
400
black

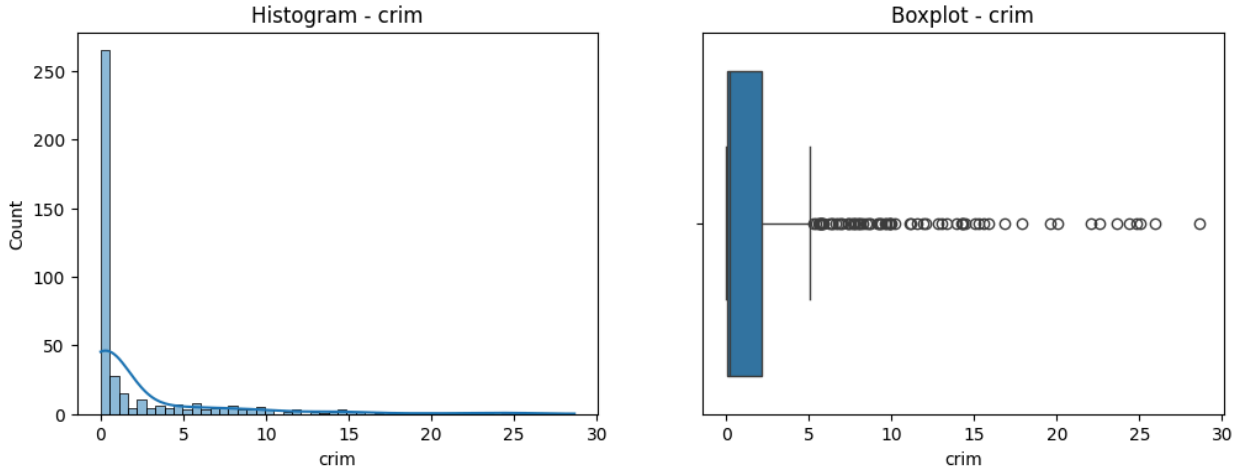

Before Outlier Removal - lstat
0
5
10
15
20
25
30
35
lstat
After Outlier Removal - lstat
5
10
15
20
25
30
lstat
Before Outlier Removal - medv
10
20
30
40
50
medv
After Outlier Removal - medv
10
20
30
40
50
medv

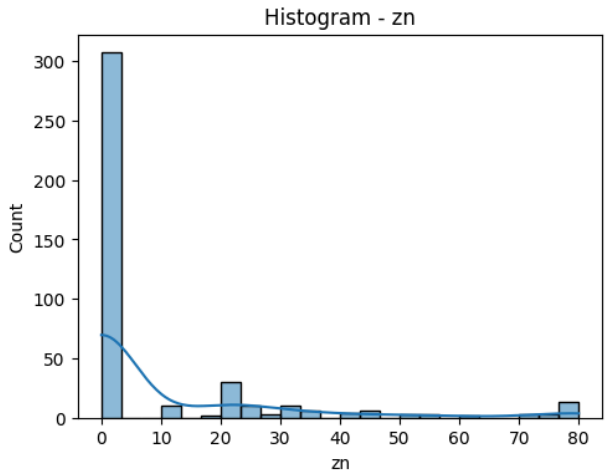
Univariate Analysis:

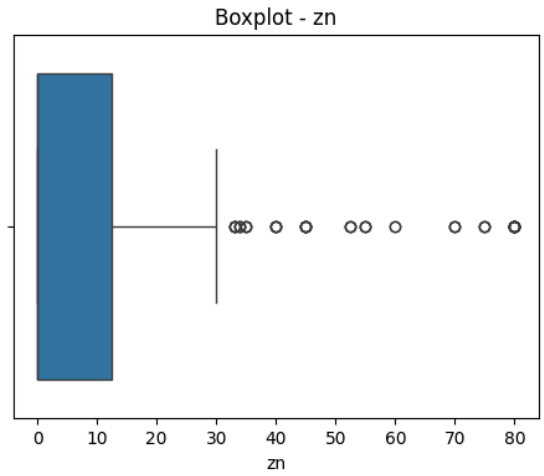
We can write below code for It:

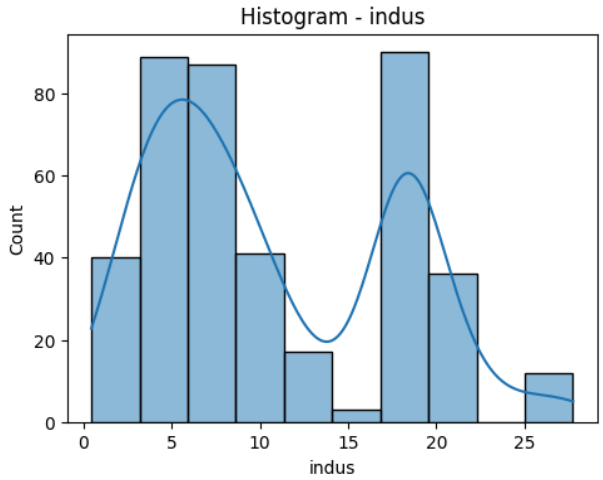
```
for col in df_cp:
    df_cp_no_outliers[col] = pd.to_numeric(df_cp_no_outliers[col],
errors='coerce')

    df_cp_no_outliers.dropna(subset=[col], inplace=True)

    fig, axes = plt.subplots(1, 2, figsize=(12, 4))
    sns.histplot(df_cp_no_outliers[col], kde=True, ax=axes[0])
    sns.boxplot(x=df_cp_no_outliers[col], ax=axes[1])
    axes[0].set_title(f'Histogram - {col}')
    axes[1].set_title(f'Boxplot - {col}')
    plt.show()

    stat, p = stats.shapiro(df_cp_no_outliers[col])
    print(f"{col} - Shapiro-Wilk Test p-value: {p:.4f}")
    print("Probably Normal" if p > 0.05 else "Probably Not Normal")
```

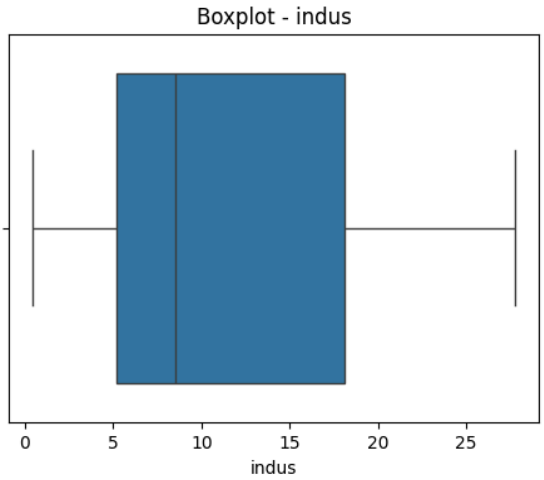
Which will provide us

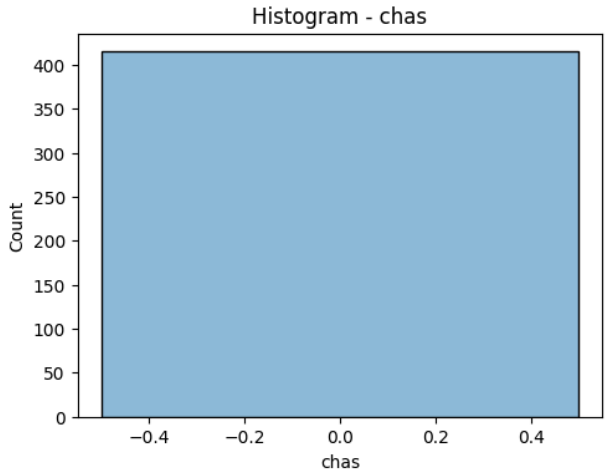


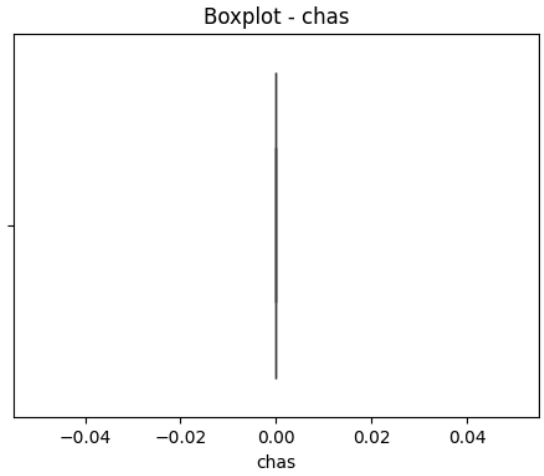

Histogram - zn
Count
zn

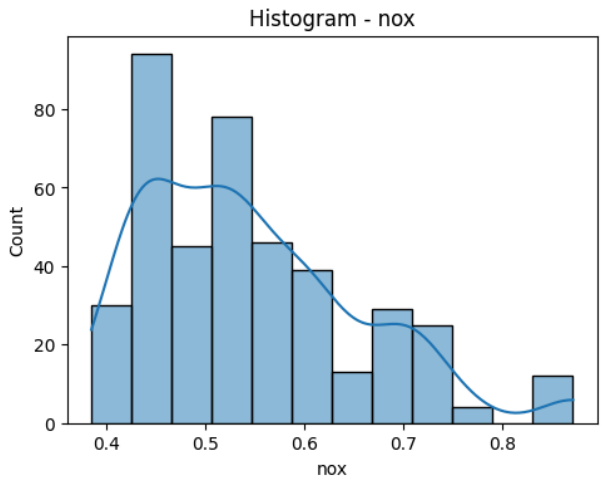

Boxplot - zn
zn

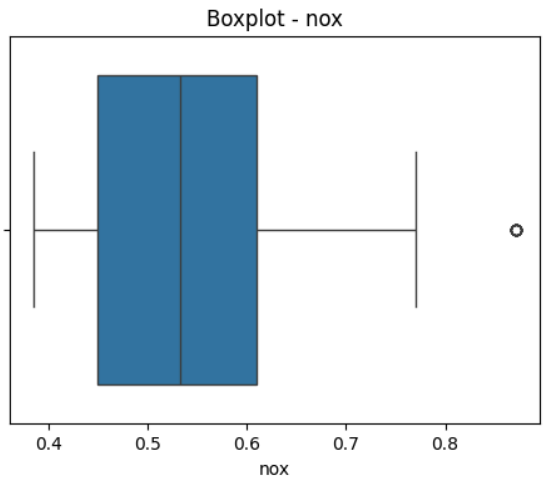

Histogram - indus
Count
indus

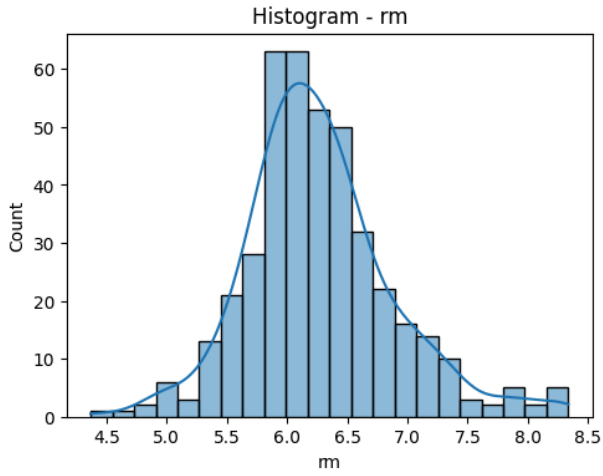

Boxplot - indus
indus

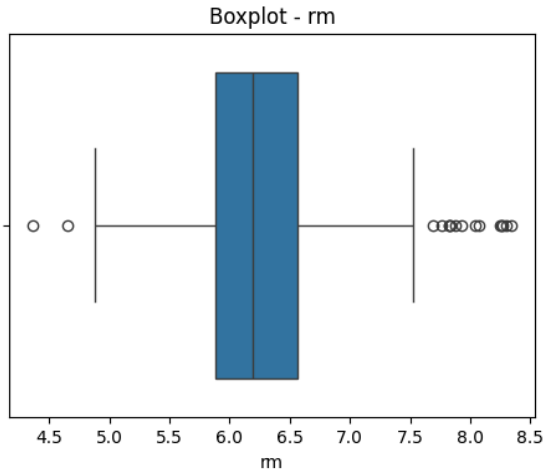

Histogram - chas
Count
chas

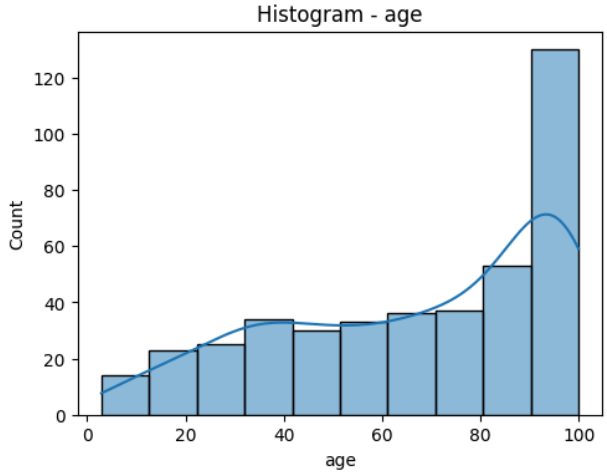

Boxplot - chas
chas

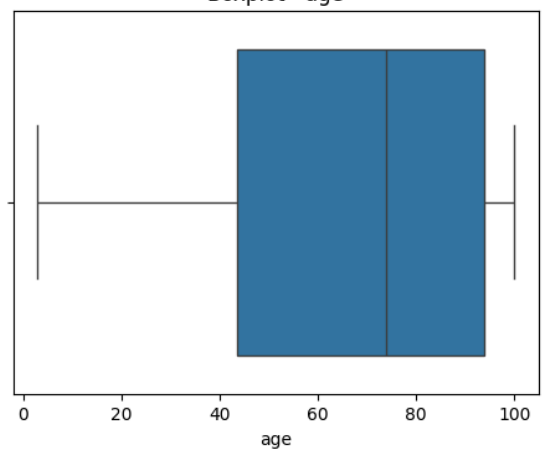

Histogram - nox
Count
nox
0
20
40
60
80
0.4
0.5
0.6
0.7
0.8

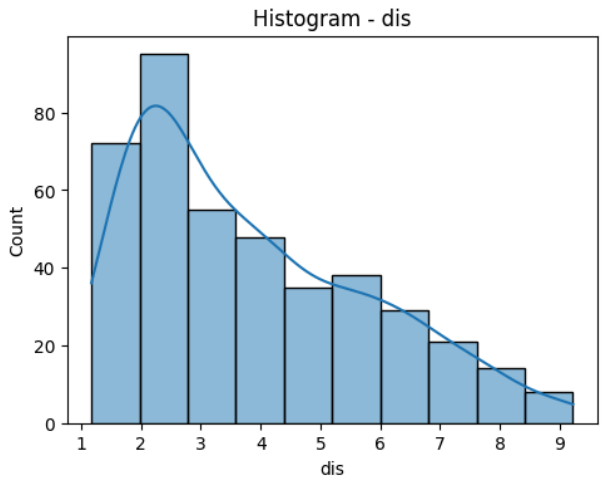

Boxplot - nox
nox
0.4
0.5
0.6
0.7
0.8

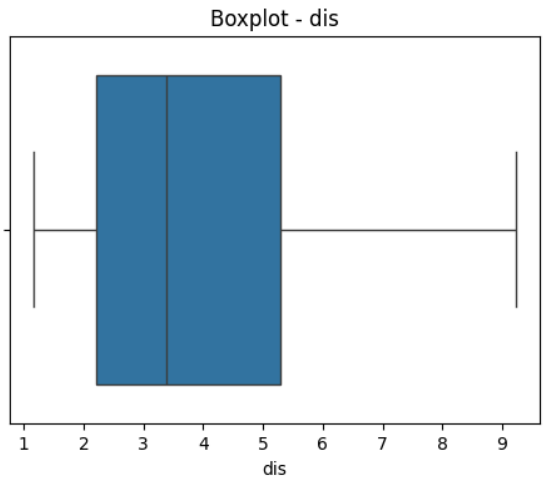

Histogram - rm
Count
rm
0
10
20
30
40
50
60
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5

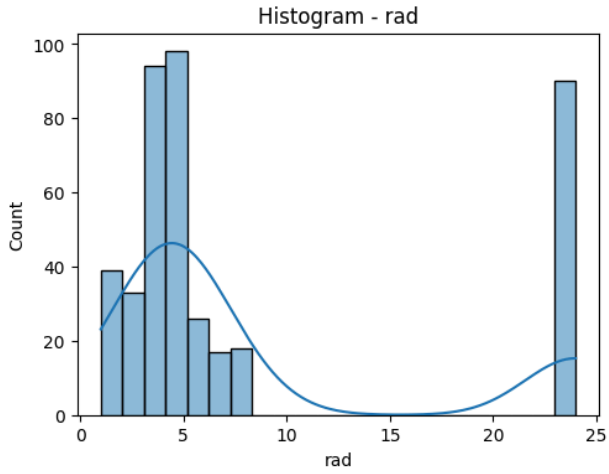

Boxplot - rm
rm
4.5
5.0
5.5
6.0
6.5
7.0
7.5
8.0
8.5

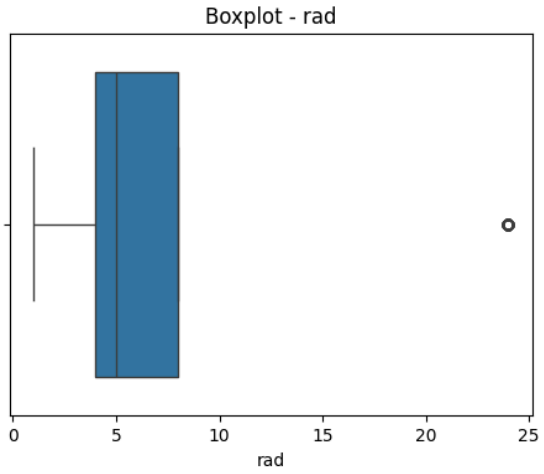

Histogram - age
Count
age
0
20
40
60
80
100
120

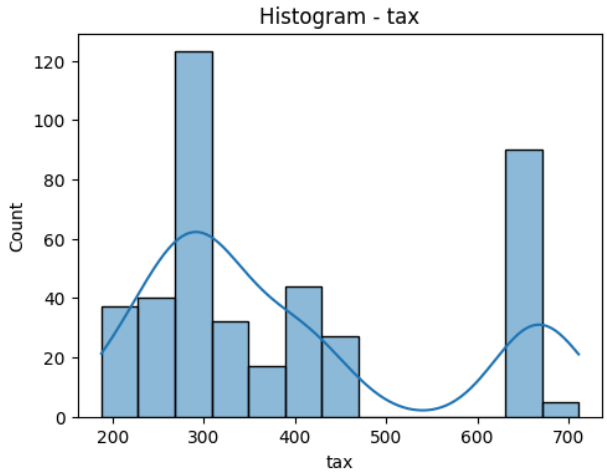

Boxplot - age
age
0
20
40
60
80
100

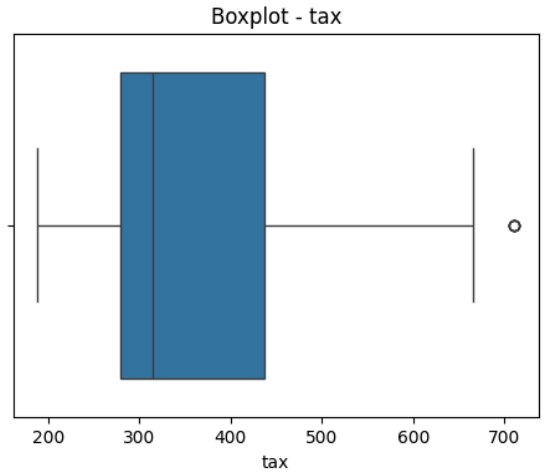

Histogram - dis
Count
dis

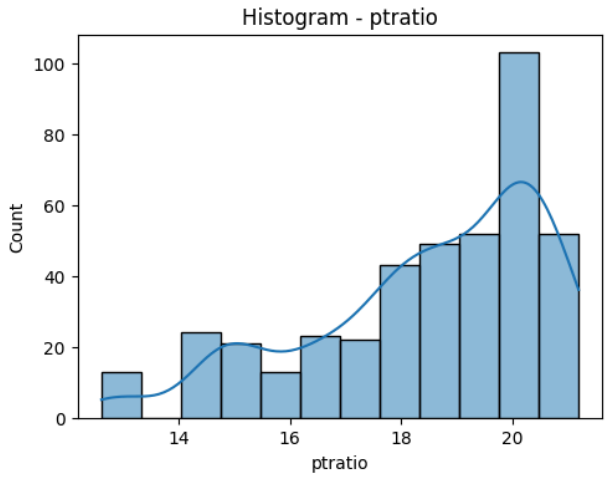

Boxplot - dis
dis

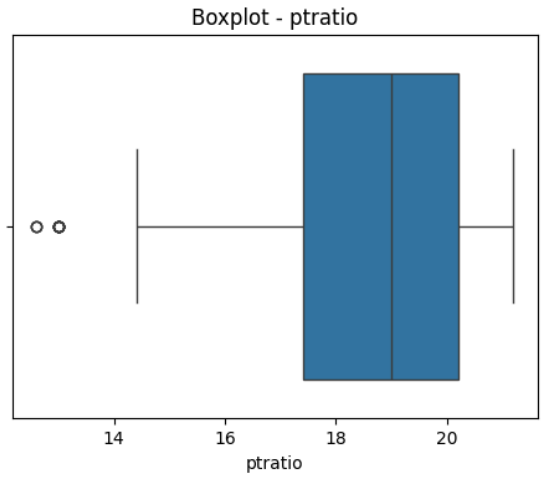

Histogram - rad
Count
rad

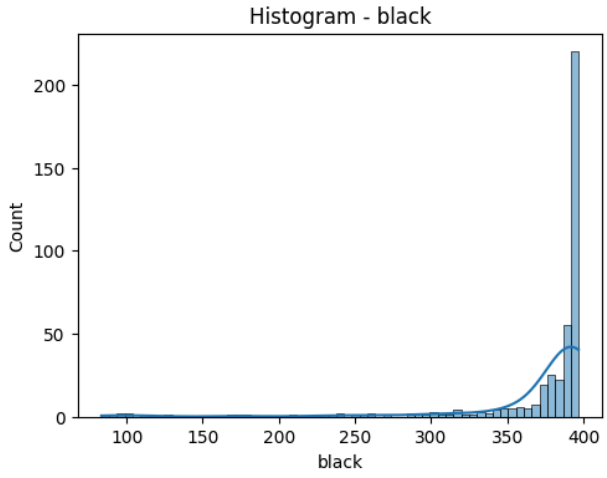

Boxplot - rad
rad

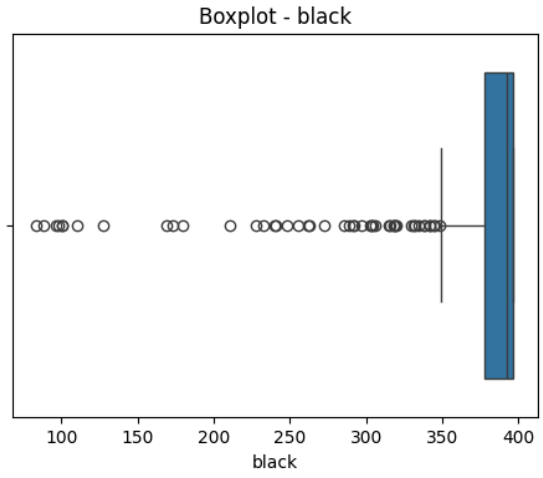

Histogram - tax
Count
tax

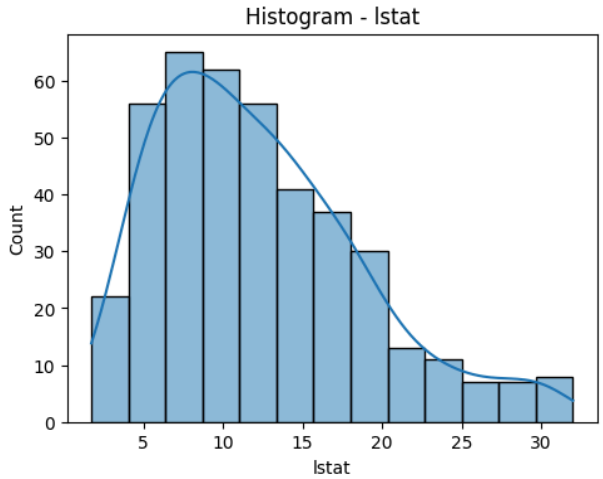

Boxplot - tax
tax

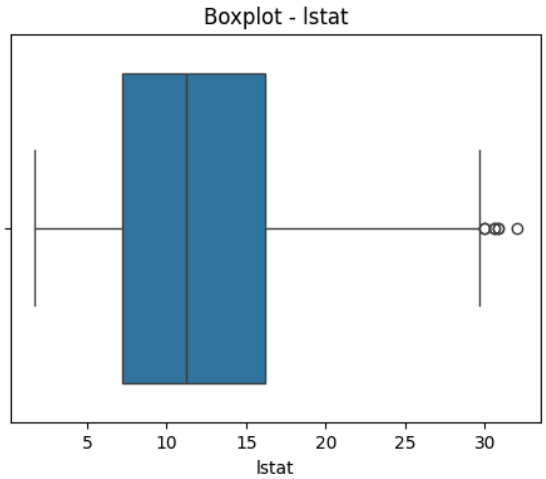

Histogram - ptratio
Count
ptratio
0
20
40
60
80
100
14
16
18
20

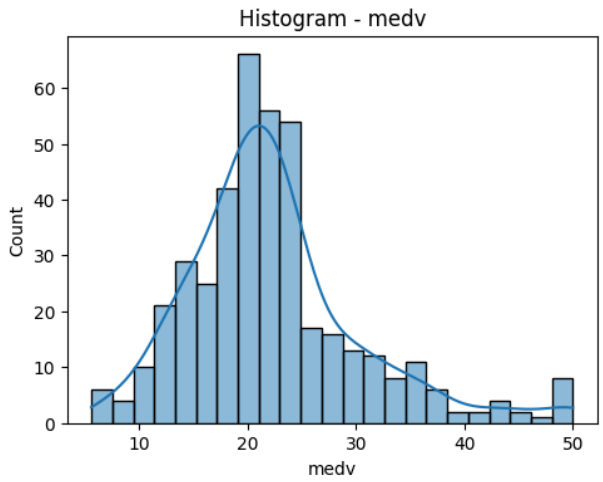

Boxplot - ptratio
ptratio
14
16
18
20

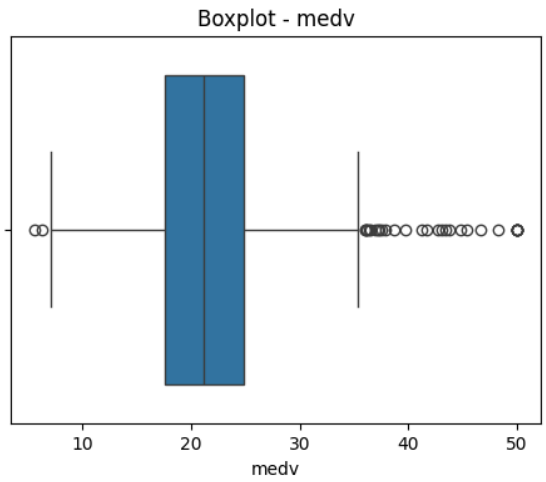

Histogram - black
Count
black
0
50
100
150
200
100
150
200
250
300
350
400


Boxplot - black
black
100
150
200
250
300
350
400


Histogram - lstat
Count
lstat
0
10
20
30
40
50
60
5
10
15
20
25
30


Boxplot - lstat
lstat
5
10
15
20
25
30


Histogram - medv
Count
medv
0
10
20
30
40
50
60
10
20
30
40
50

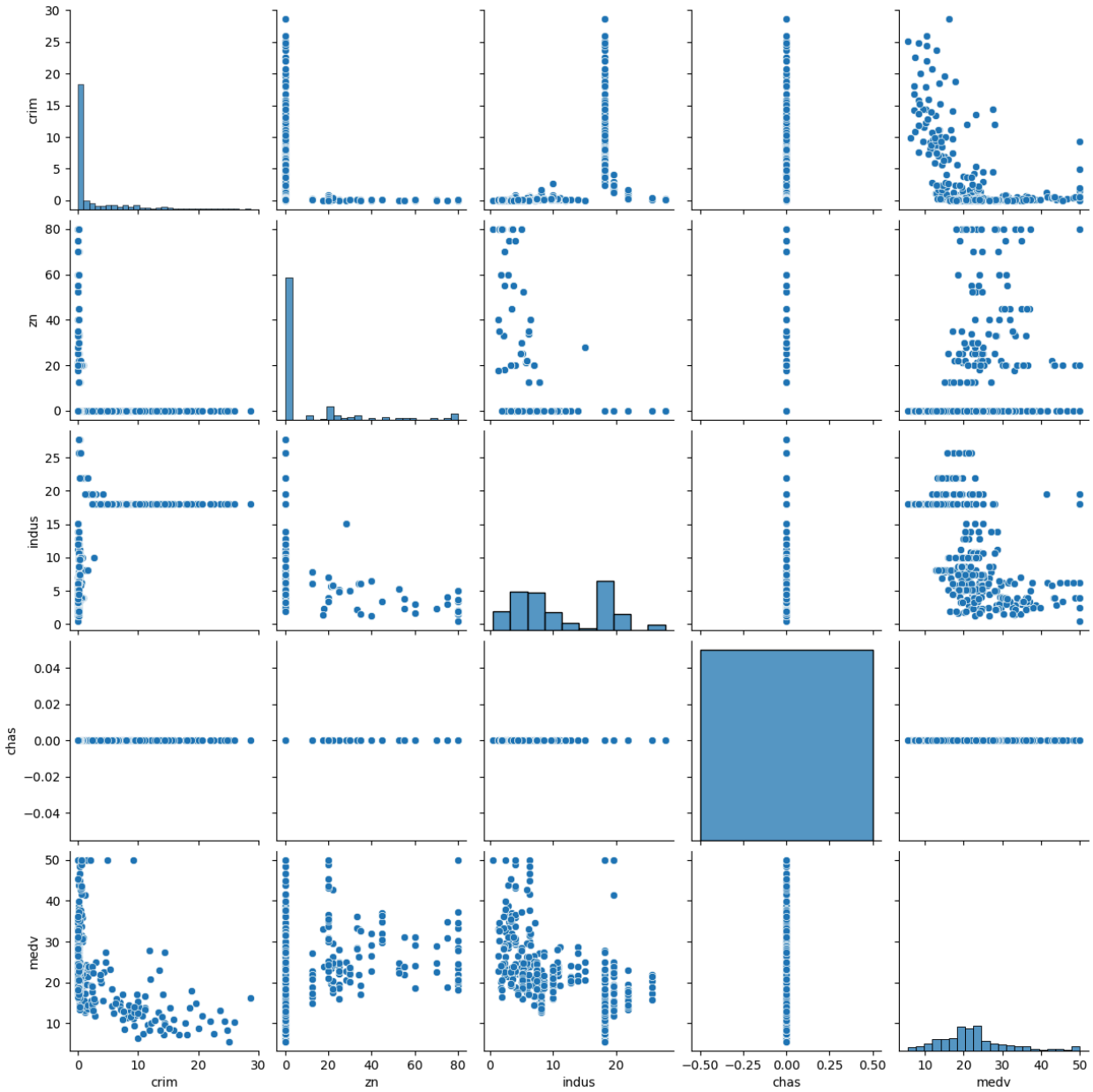

Boxplot - medv
medv
10
20
30
40
50

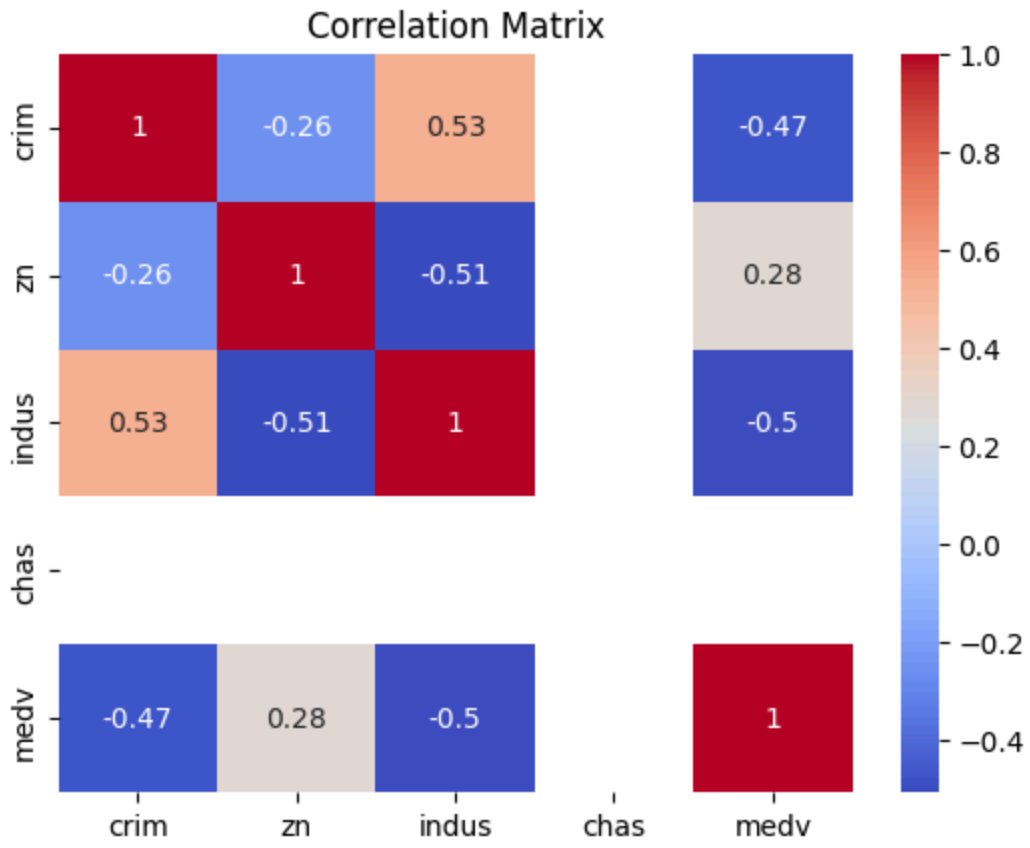
Multivariate Analysis:

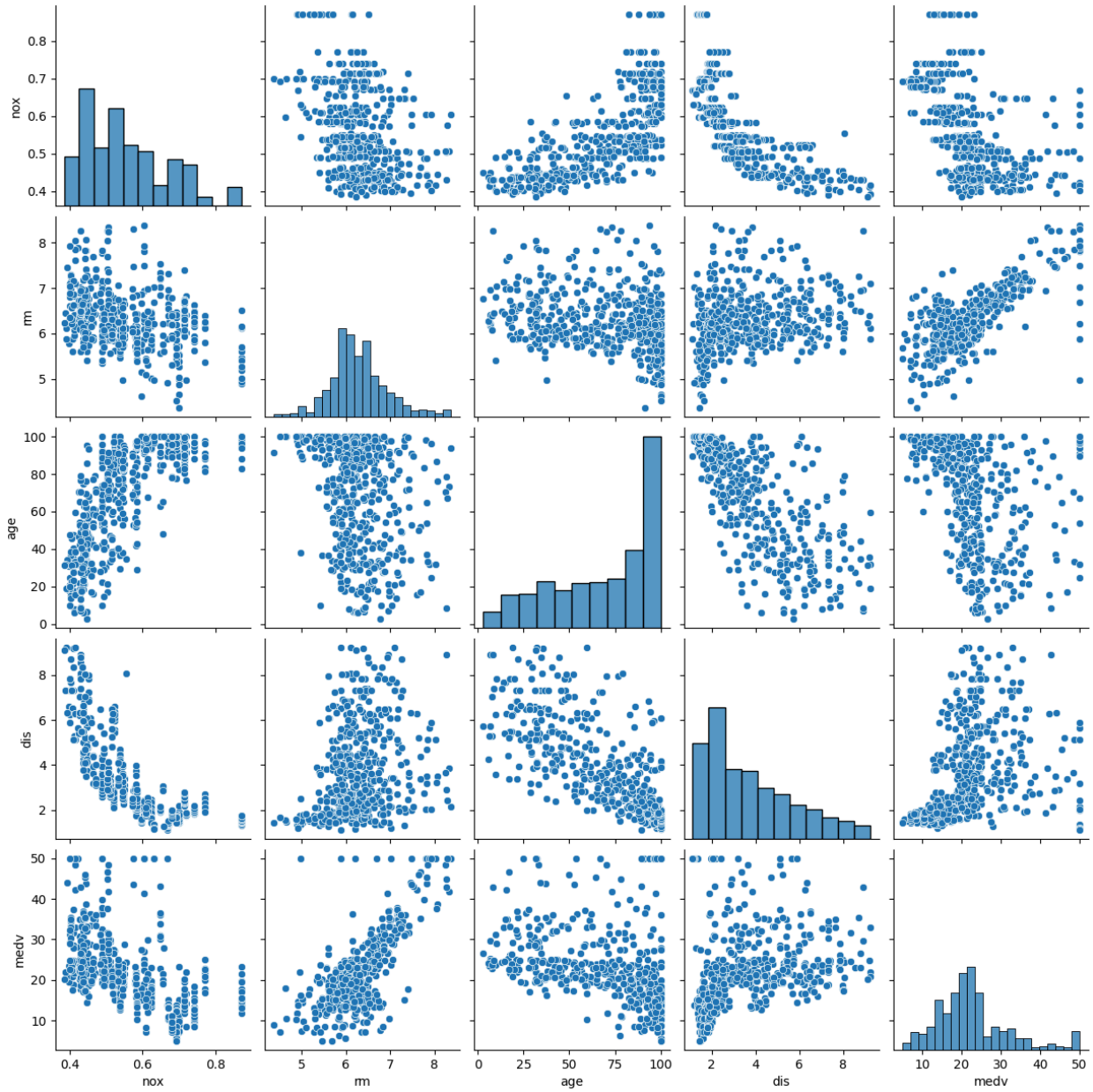
We can write below code for It

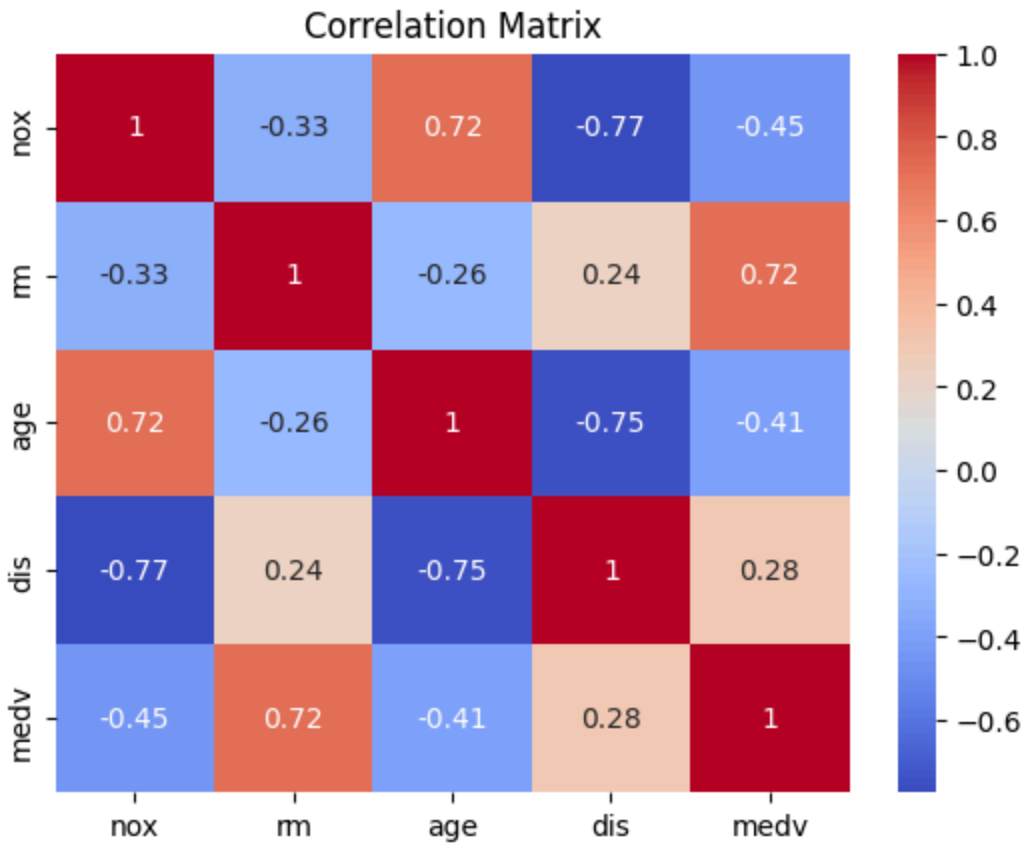
```
if target_column in df.columns:
    df_cp_corr = df_cp_no_outliers.copy()
    df_cp_corr[target_column] = df[target_column][:len(df_cp_corr)]

    numeric_columns = df_cp_corr.select_dtypes(include=np.number).columns

    numeric_columns = numeric_columns.drop(['rad']) if 'rad' in
numeric_columns else numeric_columns
    df_cp_corr[numeric_columns] =
df_cp_corr[numeric_columns].apply(pd.to_numeric, errors='coerce')

    sns.pairplot(df_cp_corr[df_cp + [target_column]])
    plt.show()

    sns.heatmap(df_cp_corr[numeric_columns].corr(), annot=True,
cmap='coolwarm')
    plt.title("Correlation Matrix")
    plt.show()
```

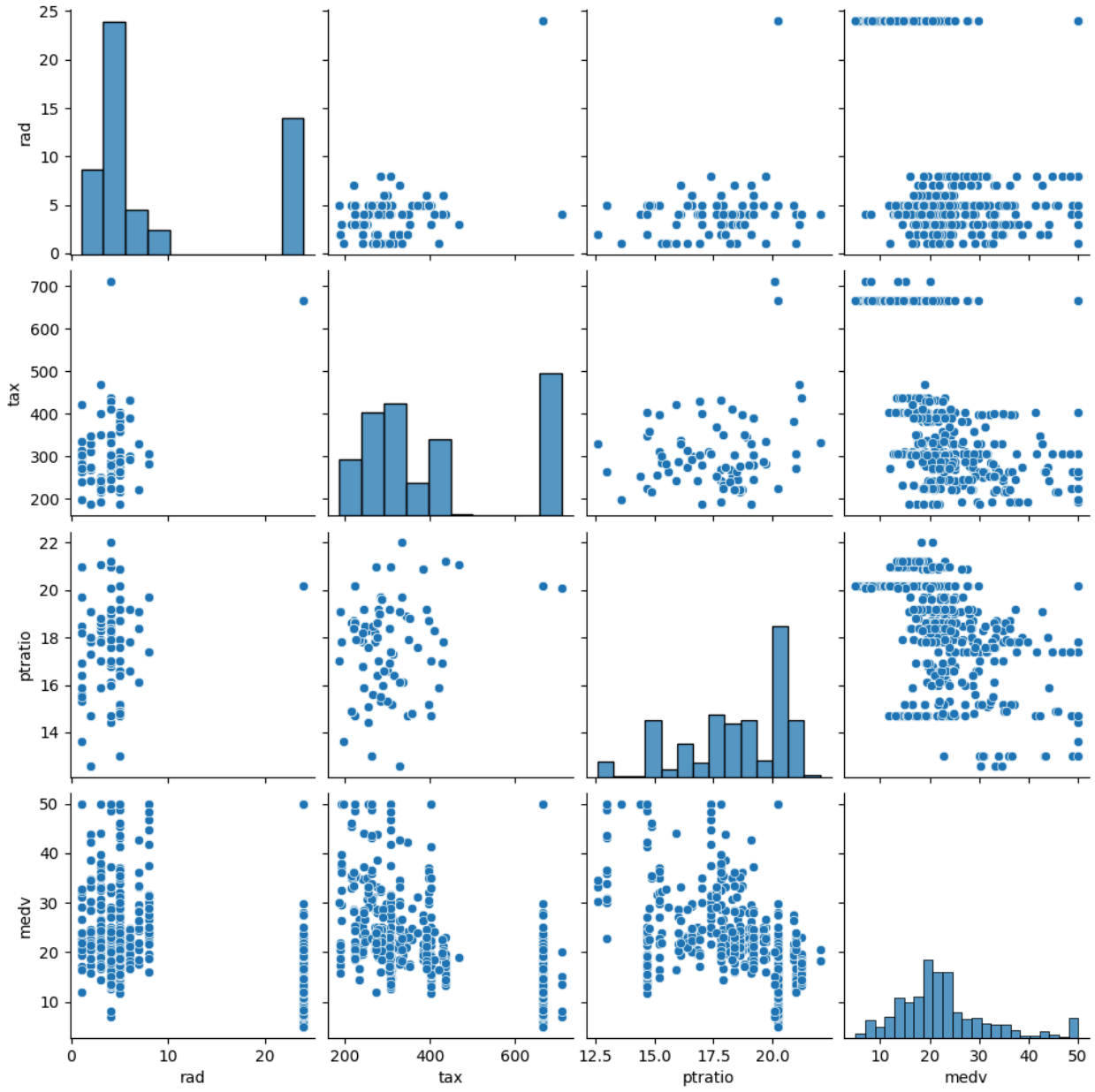
We get the output like:

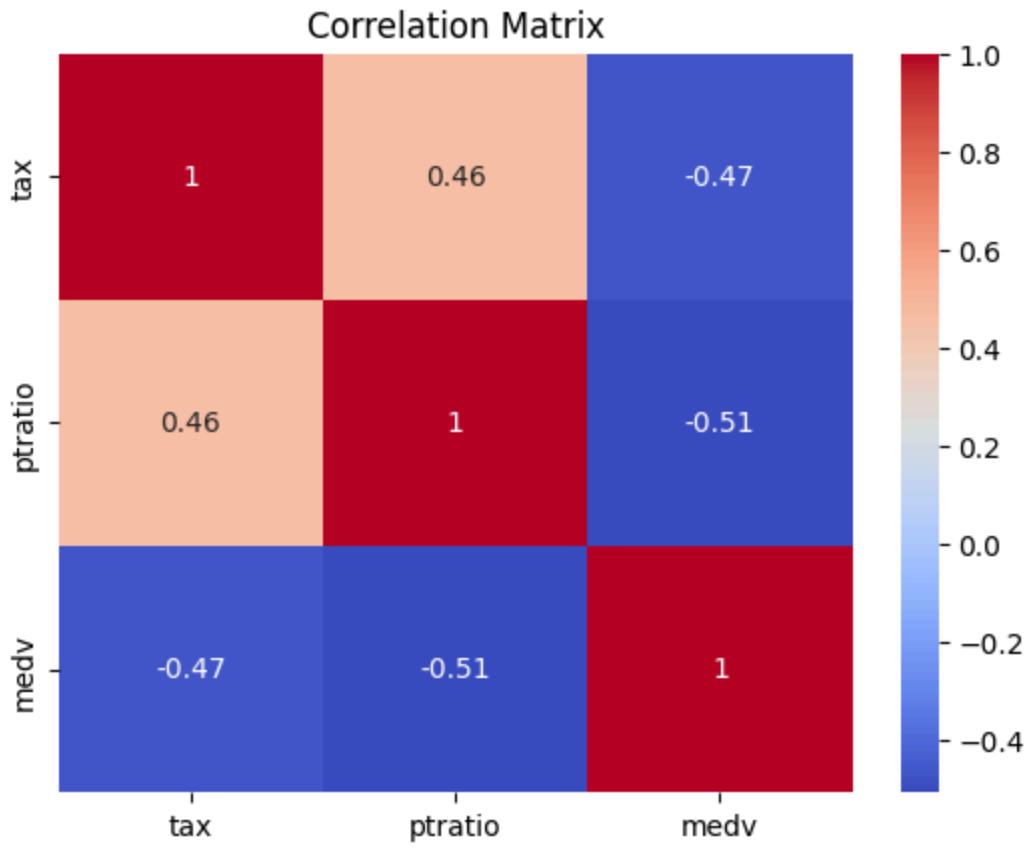


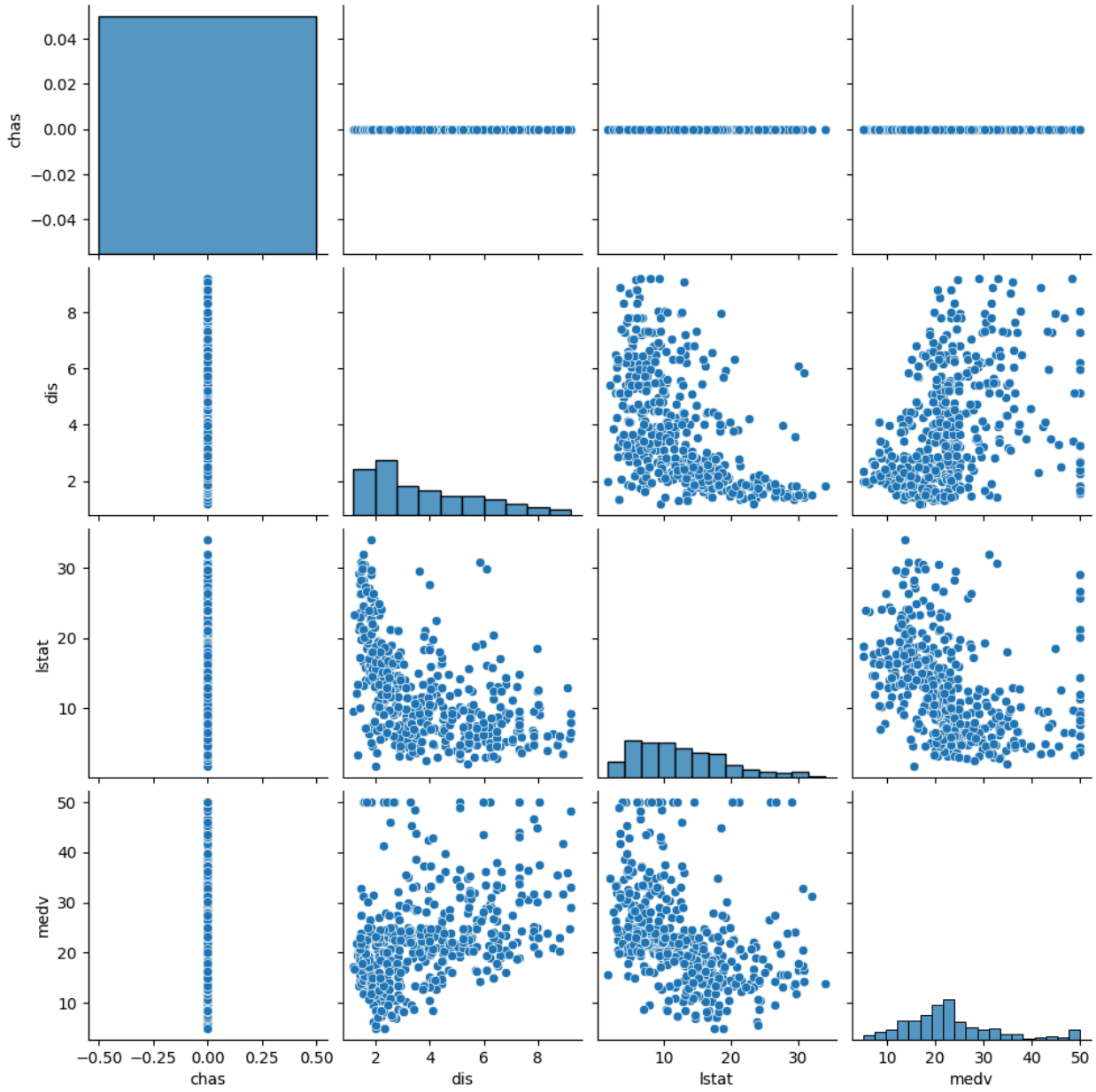

Correlation Matrix
crim
zn
indus
chas
medv
1
-0.26
0.53
-0.47
-0.26
1
-0.51
0.28
0.53
-0.51
1
-0.5
-0.47
0.28
-0.5
1
crim
zn
indus
chas
medv
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4

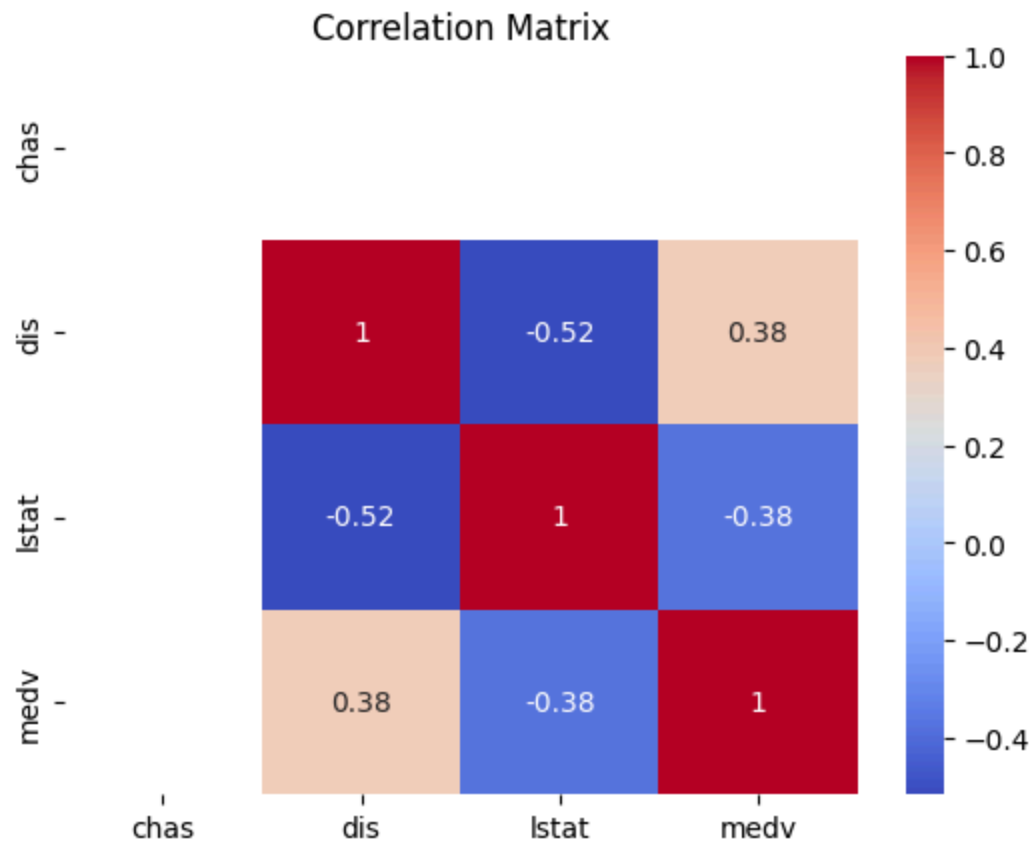

nox
rm
age
dis
medv
0.4
0.5
0.6
0.7
0.8
5
6
7
8
0
25
50
75
100
2
4
6
8
10
20
30
40
50


Correlation Matrix
nox
rm
age
dis
medv
1
-0.33
0.72
-0.77
-0.45
-0.33
1
-0.26
0.24
0.72
0.72
-0.26
1
-0.75
-0.41
-0.77
0.24
-0.75
1
0.28
-0.45
0.72
-0.41
0.28
1
nox
rm
age
dis
medv
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6


rad
tax
ptratio
medv
0
5
10
15
20
25
200
300
400
500
600
700
14
16
18
20
22
10
20
30
40
50
0
10
20
200
400
600
12.5
15.0
17.5
20.0
10
20
30
40
50


Correlation Matrix
tax
ptratio
medv
1
0.46
-0.47
0.46
1
-0.51
-0.47
-0.51
1
tax
ptratio
medv
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4


chas
dis
lstat
medv
0.04
0.02
0.00
−0.02
−0.04
8
6
4
2
30
20
10
50
40
30
20
10
−0.50
−0.25
0.00
0.25
0.50
2
4
6
8
10
20
30
10
20
30
40
50


Correlation Matrix
chas
dis
lstat
medv
1
-0.52
0.38
-0.52
1
-0.38
0.38
-0.38
1
chas
dis
lstat
medv
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4

Comment on Data

SAMIA

1. Among the assigned columns ('crim', 'zn', 'indus', 'chas'), indus had 0.395257% missing values. Since indus is skewed, we used median imputation. Others had no missing values. The missing value bar plot confirms that all missing values were successfully handled.
2. **Outliers:**
   In the **crim** column (per capita crime rate), several extreme values beyond 30 were observed before cleaning. After outlier removal (Z-score > 3), a significant number of these extreme values were removed, and the data distribution became more concentrated within a reasonable range.
   The **zn** column (proportion of residential land zoned for large lots) had scattered high outliers exceeding 50, and some beyond 80. Post-cleaning, the main body of the data remained unchanged, indicating fewer extreme values were removed, but the visible outliers were successfully addressed.
3. **Univariate:** Only the **chas** column was approximately normally distributed (Shapiro-Wilk Test p-value: 1.0000). Histograms and boxplots support this observation.
4. **Multivariate:**
   The correlation matrix illustrates the relationships between several features and the target variable, MEDV (Median value of owner-occupied homes). Key insights are:
   CRIM shows a moderate negative correlation with MEDV (r = -0.47), indicating that higher crime rates are associated with lower house prices.
   ZN has a weak positive correlation with MEDV (r = 0.28), suggesting that areas with more residential zoning tend to have slightly higher house prices.
   INDUS has a moderate negative correlation with MEDV (r = -0.50), implying that more industrial areas are associated with lower house values.
   CHAS does not have a clear correlation shown, likely due to it being a categorical variable (0 or 1), which often isn't strongly correlated in linear terms.
   From the matrix, we can interpret that lower crime and industrial presence and higher residential zoning are associated with higher house values, aligning with typical real estate expectations.

SURAIYA

1. Among the assigned columns (nox, rm, age, dis), both nox and dis had 0.1976% missing values. Based on their skewness, we used mean imputation for dis (less skewed) and nox (closer to symmetric), ensuring consistency and preserving central tendencies. rm and age had no missing data. Visual validation using the missingno barplot confirmed successful imputation.
2. Outliers:
   Boxplots revealed strong outliers in nox (values above 0.9), age (some values near 100), and dis (high peaks beyond 10). After applying Z-score filtering ($Z > 3$), the extreme values were removed. The boxplots after outlier removal showed distributions more compact and centered, indicating successful cleansing.
3. Univariate:
   Among the selected columns (nox, rm, age, dis), only rm was close to a normal distribution (Shapiro-Wilk Test p-value was relatively high). The other columns like nox, age, and dis showed skewness in their distribution. These results were clearly visible in the histograms and boxplots, where rm had a bell-shaped curve, while the others were more uneven.
4. Multivariate Analysis:
   The correlation matrix and pairplots revealed the following relationships with medv:
   rm had a strong positive correlation ($r \approx 0.70$), making it a key predictor.
   nox had a strong negative correlation ($r \approx -0.43$), indicating higher pollution lowers house prices.
   dis had a moderate positive correlation ($r \approx 0.25$), suggesting that homes slightly farther from industrial zones may be valued higher.
   age showed a weak negative correlation ($r \approx -0.38$), implying that newer homes slightly trend toward higher values.

Assumptions & Limitations:

- Z-score assumes normal distribution, which is not strictly valid for all variables (e.g., age, dis).
- Imputation based on skewness could be revisited using KNN or regression for better accuracy.
- medv used only for exploratory insight, not prediction at this stage.

MEHEDI

- Among the assigned columns (rad, tax, ptratio, black), none had missing values initially, as confirmed by the missing value bar plots. Thus, no imputation was required. Outlier analysis using Z-score method revealed significant outliers in tax (values >600), rad (discrete jumps due to low unique values), and black (values close to 0). These were successfully removed, resulting in more symmetrical and cleaner distributions. In terms of normality (Shapiro-Wilk test), none of the variables followed a normal distribution ($p < 0.05$), as supported by the histogram and boxplot visualizations. Most distributions were skewed, especially black, which had a strong left skew.

AKHLAK

1. ***Data Cleaning:*** There is no missing data for the chas column. There were 1 missing values for dis, black, and lstat columns, which is 0.197628% of the dataset. As all three are numerical and mildly skewed, I used median imputation to fill in the missing values in order to avoid the effect of extreme values.
2. ***Outlier Detection:**
    - **chas:** Binary column (either 0 or 1). No real outliers expected. After cleaning, only value 0 was retained behind which indicates over-removal. This attribute may lose its importance if value 1 is completely removed.
    - **dis:** Before removal, had clear outliers with values more than 10. After cleaning, the spread is more centralized and narrower. Boxplot clearly has a cleaner spread.
    - **black:** Boxplot showed wide spread and many individual outliers. Post-removal, shape was still almost same—meaning outliers were not significantly influential.
    - **lstat:** Had some outlier values above 30. Post-removal, distribution became tighter, reducing noise and improving modeling quality.
3. ***Univariate Analysis:***
    - **chas:** p-value = 1.0000 → likely normal (binary variable).
    - **dis:** p-value = 0.0000 → not normal, right-skewed.
    - **black:** p-value = 0.0000 → not normal, very right-skewed.
    - **lstat:** p-value = 0.0000 → not normal, right-skewed. Histogram and boxplot plots confirm these distributions.
4. ***Multivariate Analysis:***
    - **chas vs medv:** No visible correlation. Because it's categorical, it doesn't linearly affect price.
    - **dis vs medv:** Negative weak correlation (r = -0.38). Houses farther away from workplaces are more costly.
    - **black vs medv:** Weak loose correlation, perhaps biased by outliers.
    - **lstat vs medv:** Negative strong correlation (r = -0.74). Neighborhoods with greater proportions of lower status residents have cheaper houses.
5. ***Insights:***
    - lstat is an effective predictor for house price.
    - dis has moderate effect.
    - chas and black have unknown or bad effect in linear terms.

Conclusion

The Boston Housing dataset presents valuable information on the determinants of the median house value in the Boston region. The dataset contains attributes about various socioeconomic factors and housing features that are anticipated to influence house prices. A thorough exploratory data analysis (EDA) was carried out for unveiling the relationship between these features and the target attribute, median house value (MEDV).

Crime Rate and Its Effect on Housing Values (CRIM)

Another of the strongest findings of the correlation analysis is the inverse relationship between crime rate (CRIM) and MEDV. The crime rate per capita in a town is moderately negatively related to the median house value with a correlation coefficient of -0.47. This may be explained to imply that the neighborhoods with high crime rates will have low house values. In reality, those places with more crime are more undesirable, and property values fall as a consequence. This is an expected outcome because crime lower areas are safer to inhabit and therefore more desirable for future homebuyers.

Residential Zoning and Property Prices (ZN)

The percentage of residential land zoned for large lots (ZN) has a mild positive correlation (0.28) with MEDV. While the correlation is not strong, it would suggest that those regions with a higher percentage of residential zoning have a fractionally higher median house price. Residential zoning would tend to be more planned, upscale neighborhoods, which could lead to more costly housing due to lower density and bigger houses. Yet the low correlation indicates that other factors, such as access to amenities, transport and infrastructure, also play a significant role in housing prices.

Industrialization and Its Effect on Housing Prices (INDUS)

The proportion of non-retail business acres (INDUS) has a moderate negative correlation (-0.50) with MEDV, indicating that those areas with more industrial zones have lower house prices. This is likely because industrialized areas are noisier, more polluted, and have a lower standard of living. Areas with greater industrial or commercial use are less desirable for residential buyers due to environmental degradation, reduced aesthetics, and health and safety concerns. The negative correlation between INDUS and MEDV suggests that home buyers prefer areas less industrialized, and this raises the price of housing in more residential or recreational neighborhoods.

Socioeconomic Factors and Housing Prices (LSTAT)

One of the most influential predictors of house prices in the Boston Housing data is LSTAT (lower status population), with a high negative correlation (-0.74) with MEDV. This implies that the neighborhoods with a higher percentage of lower-status citizens have lower median housing

values. LSTAT is the percentage of the population in a neighborhood that is lower in socioeconomic status, e.g., lower-income individuals. The correlation between LSTAT and MEDV confirms that housing in more affluent areas costs more, which is in line with the overall trends in real estate markets where more affluent neighborhoods tend to have higher property values. The strong negative correlation of MEDV with LSTAT shows that housing prices are very responsive to socioeconomic status, and the presence of a greater percentage of lower-class residents can lower property values significantly.

Other Variables

RAD (access to radial highways) is weakly positively related to MEDV. That means that neighborhoods with more access to highways can have relatively higher house prices. Proximity to major transportation routes tends to make a neighborhood more attractive since it is more convenient for residents to commute.
TAX (property tax rate) has no powerful correlation with housing prices but may be a contributing factor to affordability. More expensive housing in an area may be brought about by higher property tax rates, although other factors such as location and neighborhood amenities generally overshadow tax rates.
AGE (proportion of units built before 1940) and DIS (distance to employment centers) are less strongly correlated with MEDV, indicating that while these variables certainly have an influence, it is not as overwhelming as variables like CRIM, INDUS, and LSTAT.
Overall Insights

The most emphatic conclusions from the analysis are that crime rate, industrial zoning, and socioeconomic status are the most influential variables that drive housing prices. Neighborhoods that have lower crime rates, less industry, and lower poverty rates have more valuable houses. The findings refer to the importance of both socioeconomic and environmental determinants of housing values. While factors such as proximity to highways and tax rates are also important, they are less powerful than the general ones of crime, industrialization, and social class. It is crucial to know these relationships for urban planning, real estate investment, and policy-making aimed at improving neighborhood quality and housing affordability.

Concisely, the analysis confirms it that more pleasant neighborhoods with a better standard of living, measured in terms of lower crime rates and higher residential zoning, will be, and are, more expensive. Socioeconomic factors, represented by the proportion of lower-status occupants, also exert a primary impact on the value of properties. This sort of observation would be useful for property owners and investors who need to assess the potential of the properties in the different neighborhoods in Boston.

Reference Links

Github Url:
https://github.com/Akhlak-Hossain-Jim/Learning-CSE-at-EWU/blob/main/Semester-8/CSE303/Lab/Section%206/Assignment%201