213208MuteebLabTask4

March 7, 2025

[3]:
```
!pip install matplotlib seaborn scikit-learn
```

```
Defaulting to user installation because normal site-packages is not writeable
Collecting matplotlib
  Downloading
matplotlib-3.10.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl
(8.6 MB)

8.6/8.6 MB 77.0 kB/s eta 0:00:00m eta
0:00:01[36m0:00:03m
Collecting seaborn
  Downloading seaborn-0.13.2-py3-none-any.whl (294 kB)

294.9/294.9 KB 252.2 kB/s eta 0:00:001m249.1 kB/s
eta 0:00:01
Requirement already satisfied: scikit-learn in
/usr/local/lib/python3.10/dist-packages (1.5.0)
Requirement already satisfied: pyparsing>=2.3.1 in /usr/lib/python3/dist-
packages (from matplotlib) (2.4.7)
Collecting contourpy>=1.0.1
  Downloading
contourpy-1.3.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (324
kB)

325.0/325.0 KB 314.8 kB/s eta 0:00:00[36m0:00:01m
eta 0:00:01
Requirement already satisfied: numpy>=1.23 in
/usr/local/lib/python3.10/dist-packages (from matplotlib) (1.26.4)
Collecting cycler>=0.10
  Downloading cycler-0.12.1-py3-none-any.whl (8.3 kB)
Requirement already satisfied: fonttools>=4.22.0 in
/usr/local/lib/python3.10/dist-packages (from matplotlib) (4.54.1)
Collecting kiwisolver>=1.3.1
  Downloading
kiwisolver-1.4.8-cp310-cp310-manylinux_2_12_x86_64.manylinux2010_x86_64.whl (1.6
MB)

1.6/1.6 MB 135.9 kB/s eta 0:00:00m eta
```

```
0:00:01[36m0:00:01
Requirement already satisfied: python-dateutil>=2.7 in
/usr/local/lib/python3.10/dist-packages (from matplotlib) (2.9.0.post0)
Requirement already satisfied: pillow>=8 in /usr/local/lib/python3.10/dist-
packages (from matplotlib) (10.4.0)
Requirement already satisfied: packaging>=20.0 in ./.local/lib/python3.10/site-
packages (from matplotlib) (24.1)
Requirement already satisfied: pandas>=1.2 in /usr/local/lib/python3.10/dist-
packages (from seaborn) (2.2.2)
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn) (1.4.2)
Requirement already satisfied: scipy>=1.6.0 in /usr/local/lib/python3.10/dist-
packages (from scikit-learn) (1.13.1)
Requirement already satisfied: threadpoolctl>=3.1.0 in
/usr/local/lib/python3.10/dist-packages (from scikit-learn) (3.5.0)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.10/dist-
packages (from pandas>=1.2->seaborn) (2024.1)
Requirement already satisfied: pytz>=2020.1 in /usr/lib/python3/dist-packages
(from pandas>=1.2->seaborn) (2022.1)
Requirement already satisfied: six>=1.5 in /usr/lib/python3/dist-packages (from
python-dateutil>=2.7->matplotlib) (1.16.0)
Installing collected packages: kiwisolver, cycler, contourpy, matplotlib,
seaborn
Successfully installed contourpy-1.3.1 cycler-0.12.1 kiwisolver-1.4.8
matplotlib-3.10.1 seaborn-0.13.2
```

[34]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import sqlite3
from sklearn.preprocessing import LabelEncoder, MinMaxScaler, StandardScaler
```

[5]:
```python
# Task 1: Importing and Handling Data
```

[8]:
```python
# Load Titanic dataset (CSV)
titanic_df = pd.read_csv("Titanic.csv")

# Load Housing dataset (CSV)
housing_df = pd.read_csv("housing.csv")

# Display first five rows of each dataset
print("Titanic Dataset:")
print(titanic_df.head())
print("\nHousing Dataset:")
print(housing_df.head())
```

```
Titanic Dataset:
```

```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked
0      0         A/5 21171   7.2500   NaN        S
1      0          PC 17599  71.2833   C85        C
2      0  STON/O2. 3101282   7.9250   NaN        S
3      0            113803  53.1000  C123        S
4      0            373450   8.0500   NaN        S

Housing Dataset:
   longitude  latitude  housing_median_age  total_rooms  total_bedrooms  \
0    -122.23     37.88                  41          880           129.0
1    -122.22     37.86                  21         7099          1106.0
2    -122.24     37.85                  52         1467           190.0
3    -122.25     37.85                  52         1274           235.0
4    -122.25     37.85                  52         1627           280.0

   population  households  median_income  median_house_value ocean_proximity
0         322         126         8.3252              452600        NEAR BAY
1        2401        1138         8.3014              358500        NEAR BAY
2         496         177         7.2574              352100        NEAR BAY
3         558         219         5.6431              341300        NEAR BAY
4         565         259         3.8462              342200        NEAR BAY
```

[9]:

```python
# Dataset information
titanic_df.info()
print(titanic_df.describe())
housing_df.info()
print(housing_df.describe())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  891 non-null    int64
```

```
 1   Survived     891 non-null    int64
 2   Pclass       891 non-null    int64
 3   Name         891 non-null    object
 4   Sex          891 non-null    object
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64
 7   Parch        891 non-null    int64
 8   Ticket       891 non-null    object
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object
 11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
       PassengerId    Survived      Pclass         Age       SibSp  \
count   891.000000  891.000000  891.000000  714.000000  891.000000
mean    446.000000    0.383838    2.308642   29.699118    0.523008
std     257.353842    0.486592    0.836071   14.526497    1.102743
min       1.000000    0.000000    1.000000    0.420000    0.000000
25%     223.500000    0.000000    2.000000   20.125000    0.000000
50%     446.000000    0.000000    3.000000   28.000000    0.000000
75%     668.500000    1.000000    3.000000   38.000000    1.000000
max     891.000000    1.000000    3.000000   80.000000    8.000000

            Parch        Fare
count  891.000000  891.000000
mean     0.381594   32.204208
std      0.806057   49.693429
min      0.000000    0.000000
25%      0.000000    7.910400
50%      0.000000   14.454200
75%      0.000000   31.000000
max      6.000000  512.329200
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20640 entries, 0 to 20639
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   longitude           20640 non-null  float64
 1   latitude            20640 non-null  float64
 2   housing_median_age  20640 non-null  int64
 3   total_rooms         20640 non-null  int64
 4   total_bedrooms      20433 non-null  float64
 5   population          20640 non-null  int64
 6   households          20640 non-null  int64
 7   median_income       20640 non-null  float64
 8   median_house_value  20640 non-null  int64
 9   ocean_proximity     20640 non-null  object
dtypes: float64(4), int64(5), object(1)
```

```
memory usage: 1.6+ MB
          longitude      latitude  housing_median_age   total_rooms  \
count  20640.000000  20640.000000        20640.000000  20640.000000
mean    -119.569704     35.631861           28.639486   2635.763081
std        2.003532      2.135952           12.585558   2181.615252
min     -124.350000     32.540000            1.000000      2.000000
25%     -121.800000     33.930000           18.000000   1447.750000
50%     -118.490000     34.260000           29.000000   2127.000000
75%     -118.010000     37.710000           37.000000   3148.000000
max     -114.310000     41.950000           52.000000  39320.000000

       total_bedrooms    population    households  median_income  \
count    20433.000000  20640.000000  20640.000000   20640.000000
mean       537.870553   1425.476744    499.539680       3.870671
std        421.385070   1132.462122    382.329753       1.899822
min          1.000000      3.000000      1.000000       0.499900
25%        296.000000    787.000000    280.000000       2.563400
50%        435.000000   1166.000000    409.000000       3.534800
75%        647.000000   1725.000000    605.000000       4.743250
max       6445.000000  35682.000000   6082.000000      15.000100

       median_house_value
count        20640.000000
mean        206855.816909
std         115395.615874
min          14999.000000
25%         119600.000000
50%         179700.000000
75%         264725.000000
max         500001.000000
```

[10]:

```
# Task 2: Exploratory Data Analysis (EDA)
```

[11]:

```
# Check for missing values
print("\nMissing values in Titanic:")
print(titanic_df.isnull().sum())
print("\nMissing values in Housing:")
print(housing_df.isnull().sum())
```

```

Missing values in Titanic:
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
```

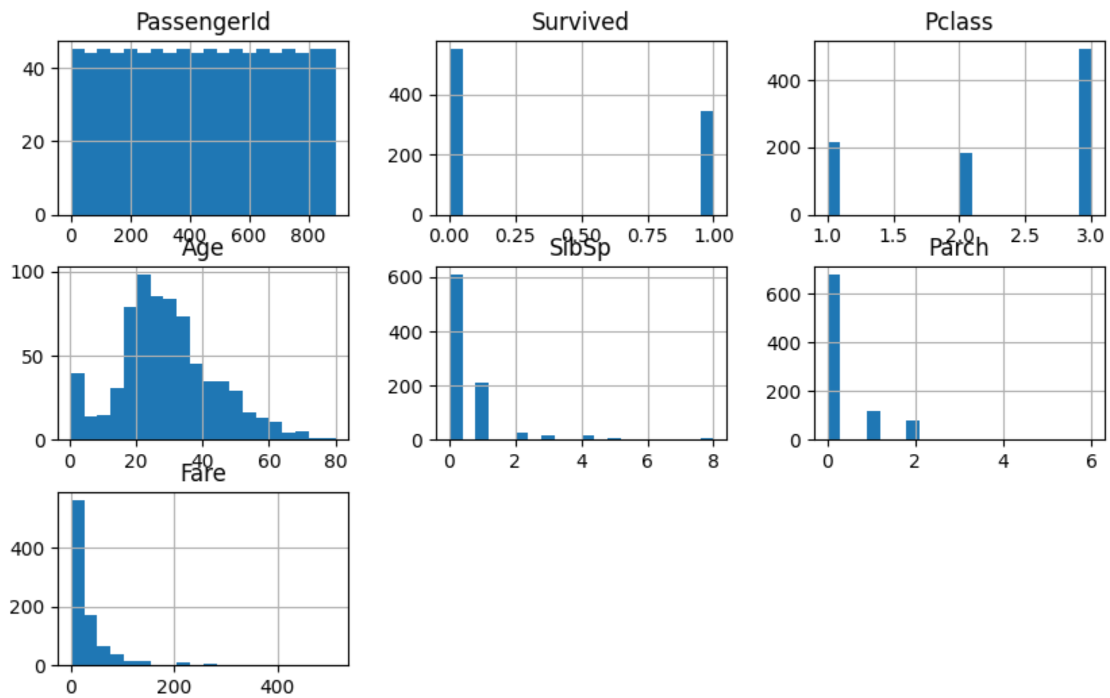
```
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64

Missing values in Housing:
longitude               0
latitude                0
housing_median_age      0
total_rooms             0
total_bedrooms        207
population              0
households              0
median_income           0
median_house_value      0
ocean_proximity         0
dtype: int64
```

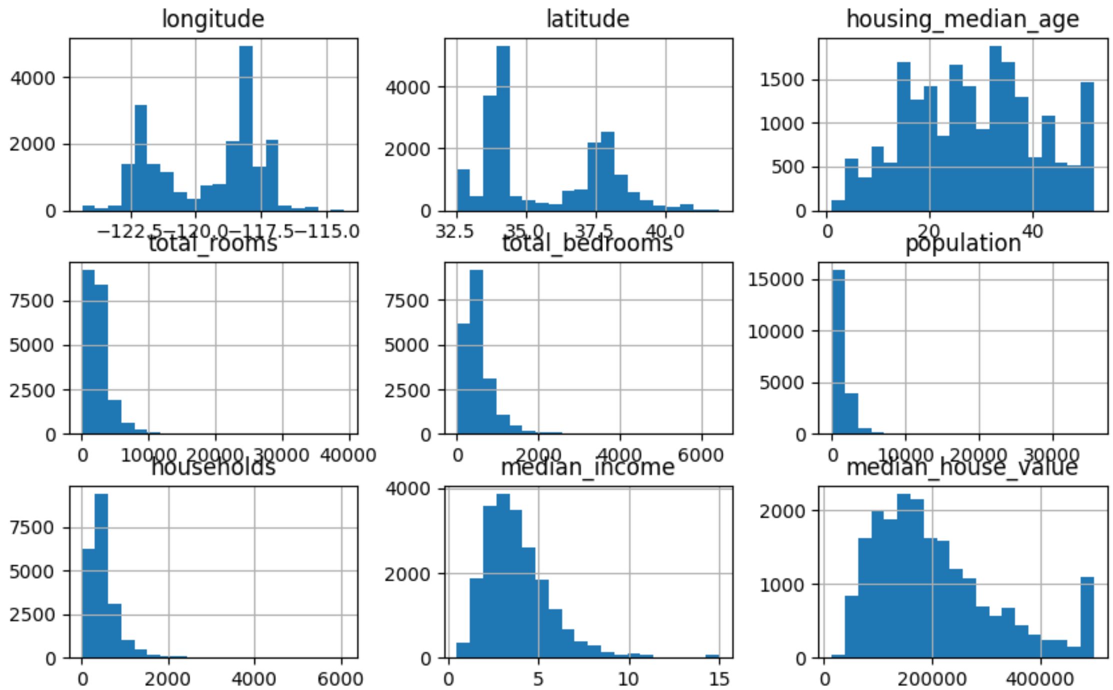
[12]:
```
# Identify numerical and categorical features
titanic_num_features = titanic_df.select_dtypes(include=[np.number]).columns.
 ↪tolist()
titanic_cat_features = titanic_df.select_dtypes(exclude=[np.number]).columns.
 ↪tolist()

housing_num_features = housing_df.select_dtypes(include=[np.number]).columns.
 ↪tolist()
housing_cat_features = housing_df.select_dtypes(exclude=[np.number]).columns.
 ↪tolist()
```

[13]:
```
print("\nTitanic Numerical Features:", titanic_num_features)
print("Titanic Categorical Features:", titanic_cat_features)
print("\nHousing Numerical Features:", housing_num_features)
print("Housing Categorical Features:", housing_cat_features)
```

```
Titanic Numerical Features: ['PassengerId', 'Survived', 'Pclass', 'Age',
'SibSp', 'Parch', 'Fare']
Titanic Categorical Features: ['Name', 'Sex', 'Ticket', 'Cabin', 'Embarked']

Housing Numerical Features: ['longitude', 'latitude', 'housing_median_age',
'total_rooms', 'total_bedrooms', 'population', 'households', 'median_income',
'median_house_value']
Housing Categorical Features: ['ocean_proximity']
```

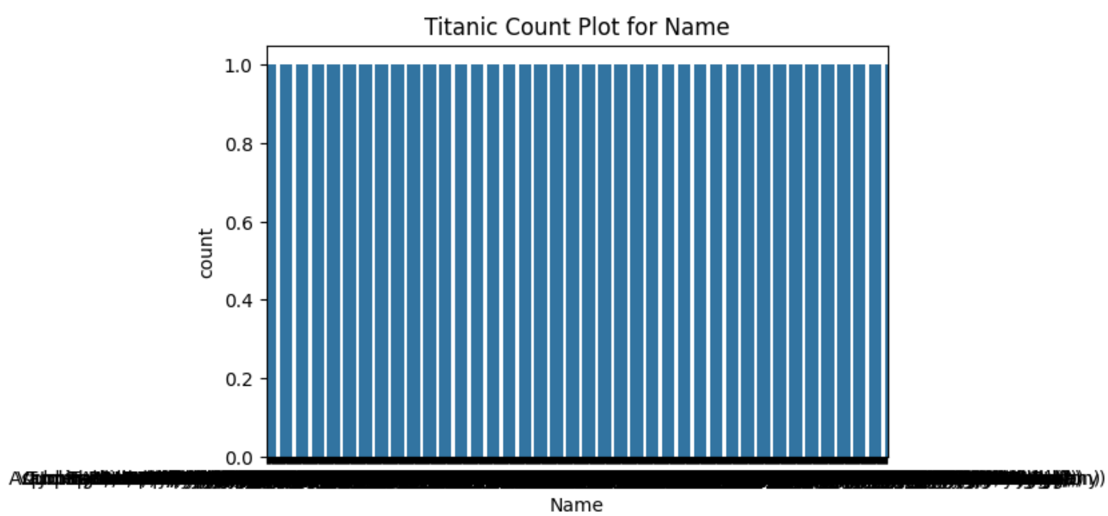
[14]:
```
# Histograms for numerical features
titanic_df[titanic_num_features].hist(figsize=(10, 6), bins=20)
plt.show()
housing_df[housing_num_features].hist(figsize=(10, 6), bins=20)
plt.show()
```

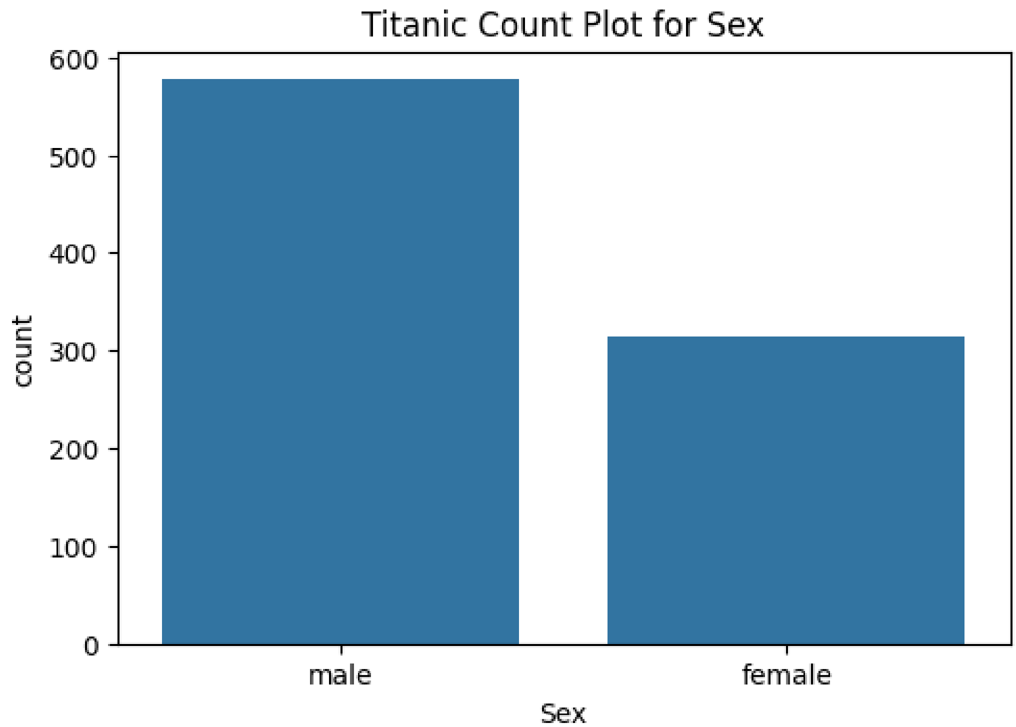


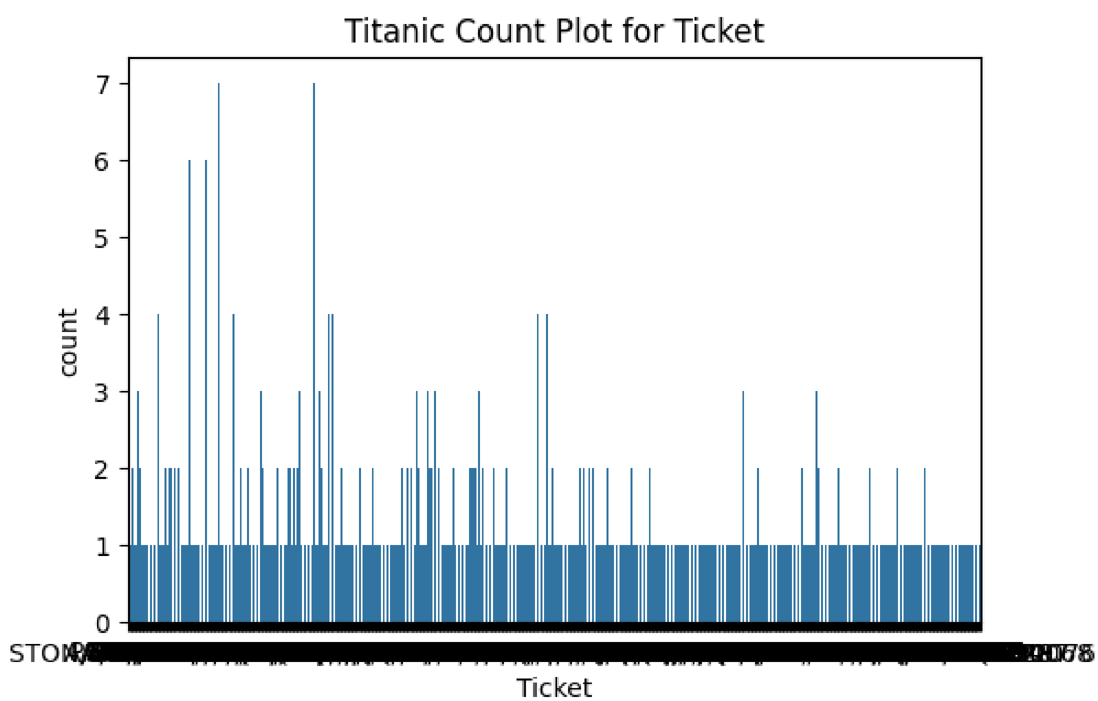


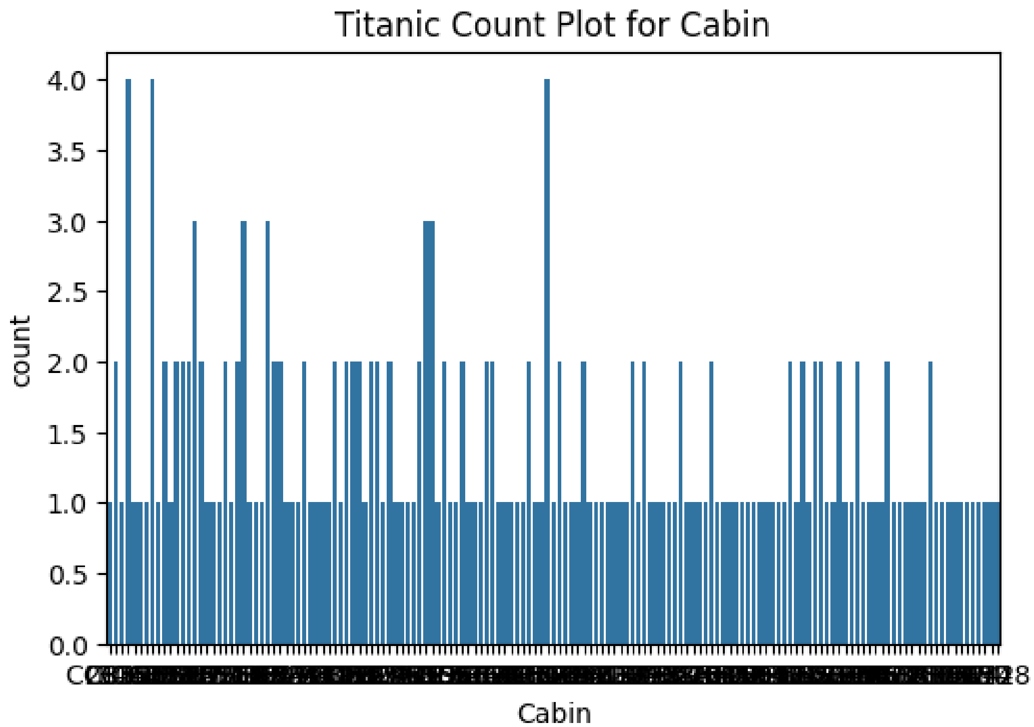
[15]:

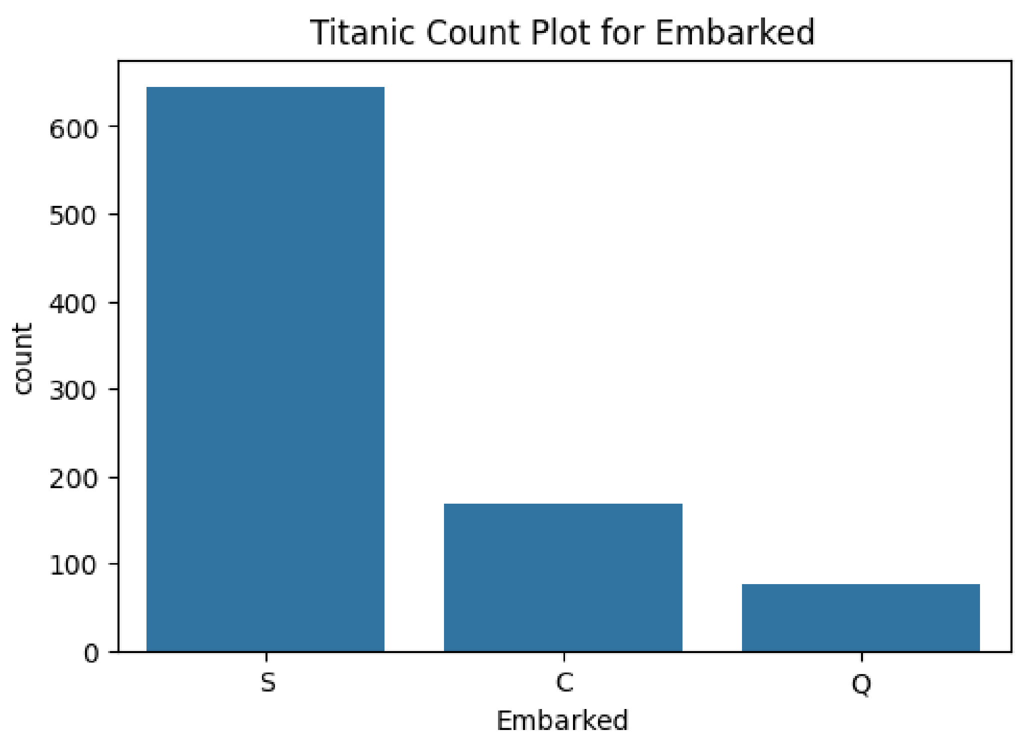
```
# Count plots for categorical features
for col in titanic_cat_features:
    plt.figure(figsize=(6, 4))
    sns.countplot(x=titanic_df[col])
    plt.title(f'Titanic Count Plot for {col}')
    plt.show()
```



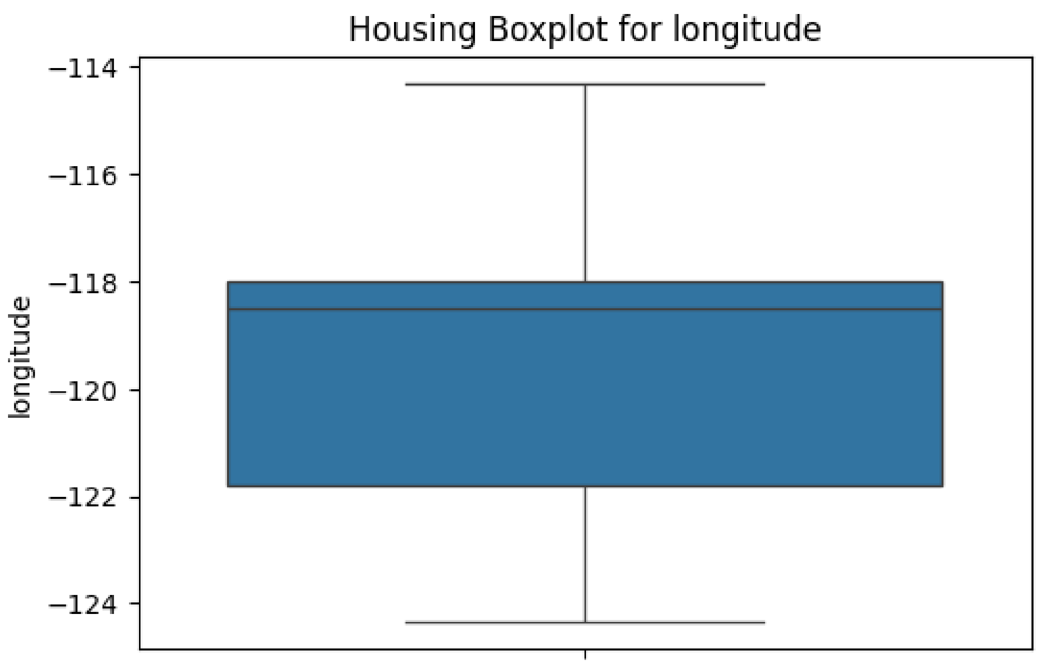

Titanic Count Plot for Sex
count
600
500
400
300
200
100
0
male
female
Sex

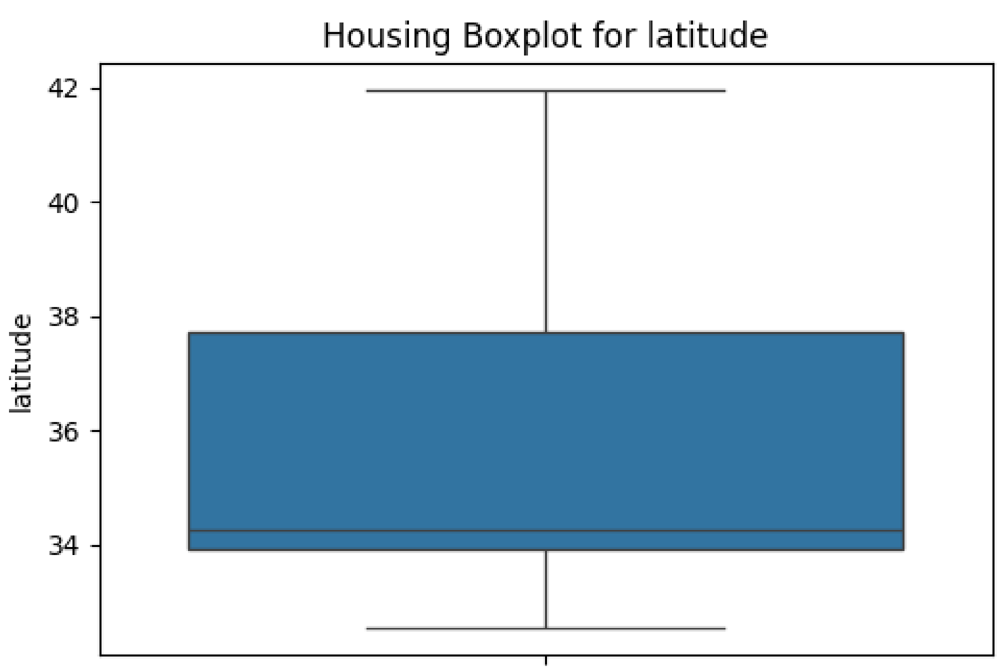

Titanic Count Plot for Ticket
count
7
6
5
4
3
2
1
0
Ticket

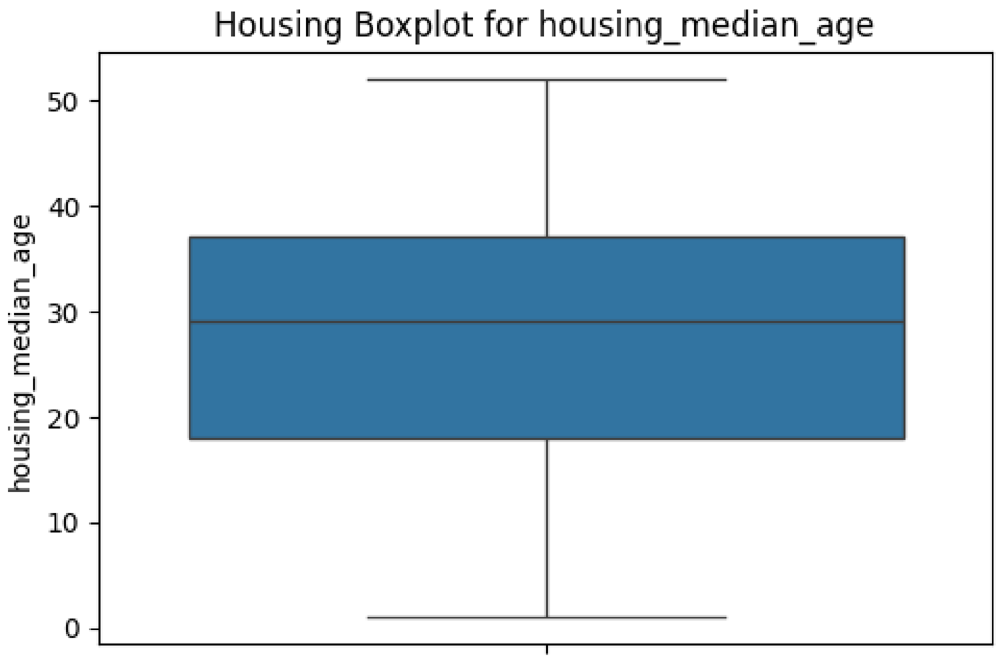

Titanic Count Plot for Cabin
count
Cabin

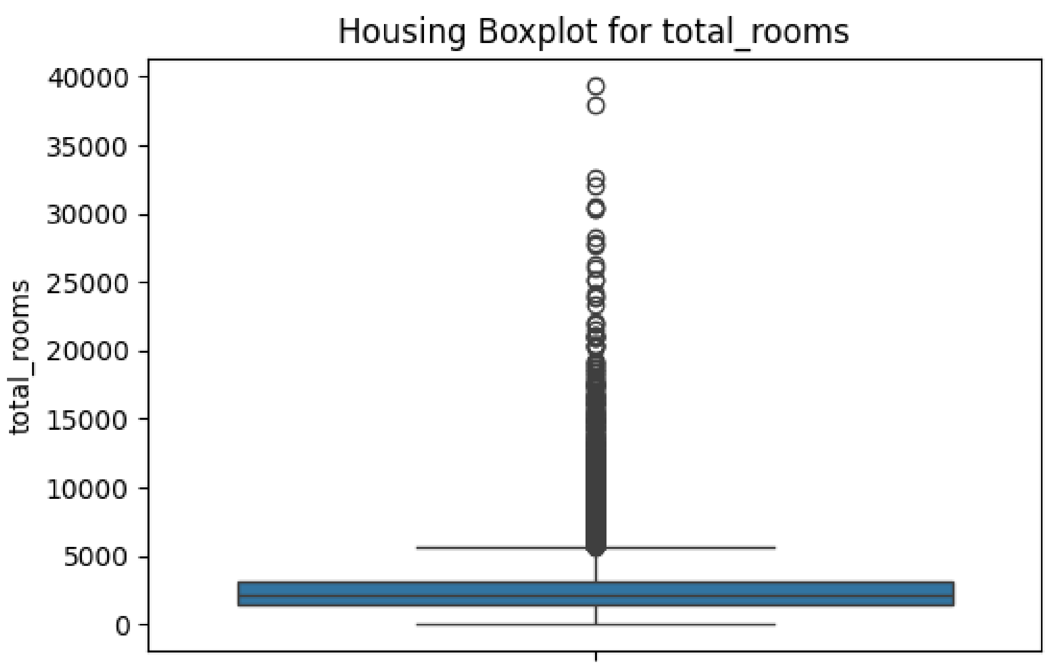

Titanic Count Plot for Embarked
count
S
C
Q
Embarked

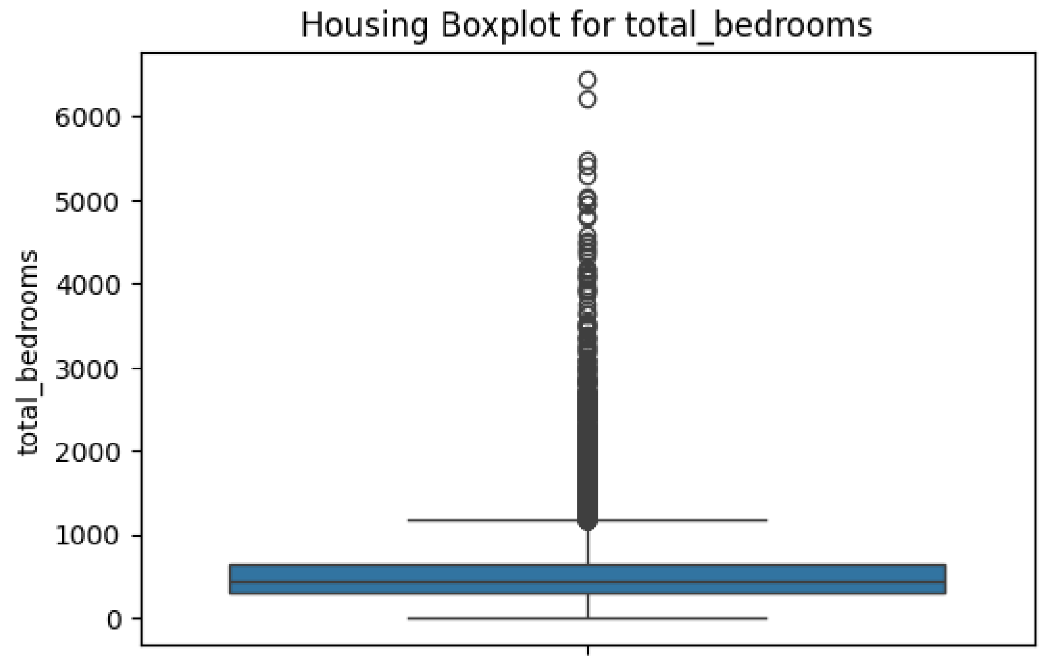
[16]:
```
# Boxplots to check for outliers
for col in housing_num_features:
    plt.figure(figsize=(6, 4))
    sns.boxplot(y=housing_df[col])
    plt.title(f'Housing Boxplot for {col}')
    plt.show()
```



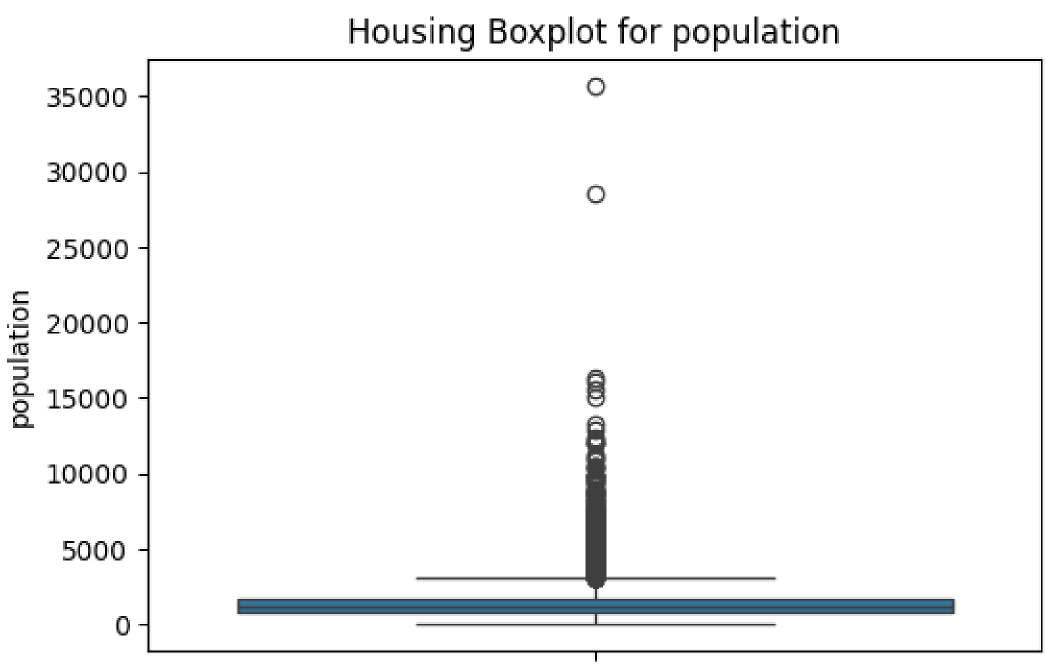



Housing Boxplot for housing_median_age
housing_median_age
50
40
30
20
10
0

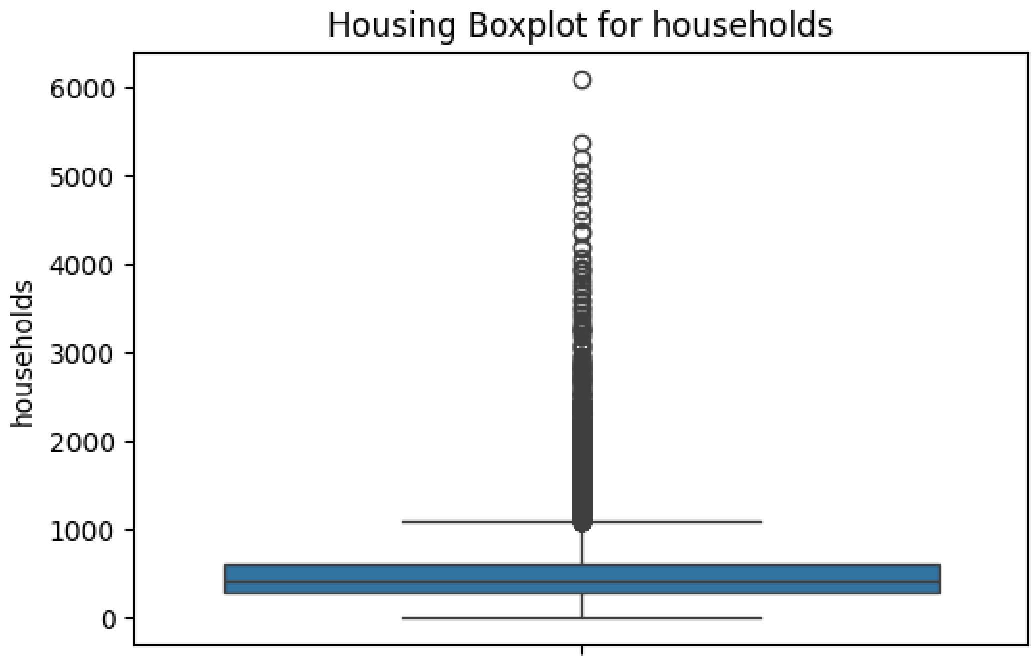

Housing Boxplot for total_rooms
total_rooms
40000
35000
30000
25000
20000
15000
10000
5000
0

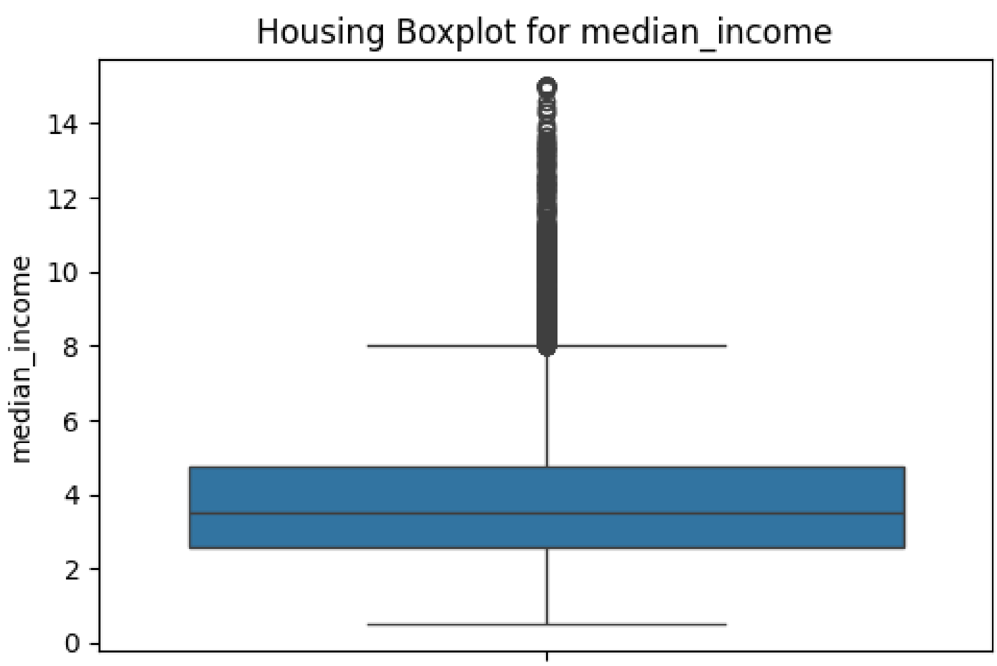


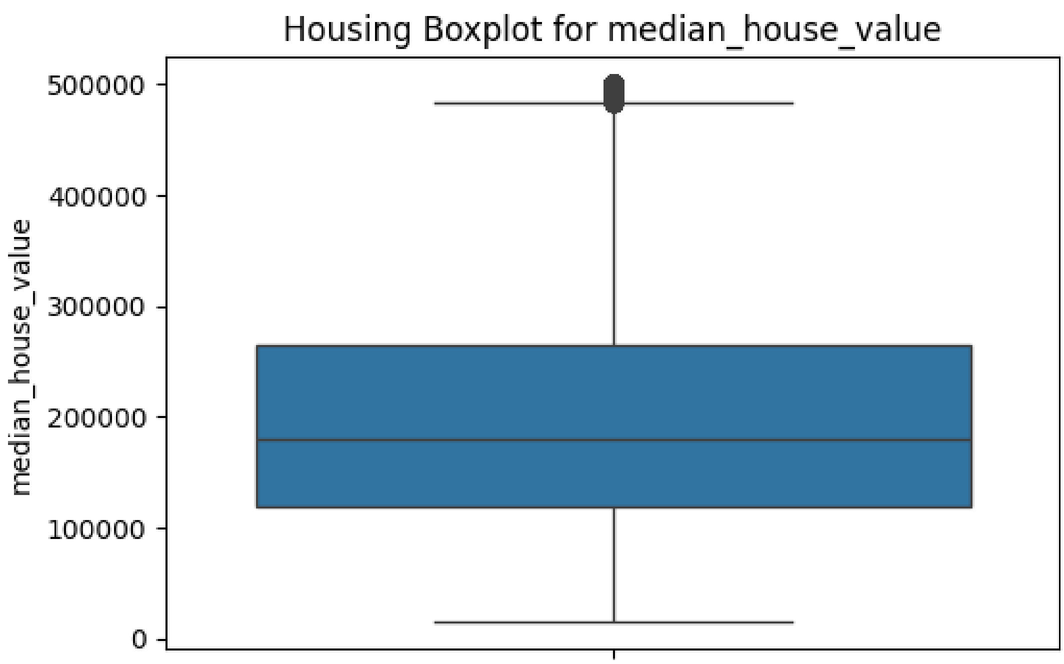
Housing Boxplot for total_bedrooms
total_bedrooms
6000
5000
4000
3000
2000
1000
0



Housing Boxplot for population
population
35000
30000
25000
20000
15000
10000
5000
0


Housing Boxplot for households
households
6000
5000
4000
3000
2000
1000
0


Housing Boxplot for median_income
median_income
14
12
10
8
6
4
2
0



[17]:
```
# Compute correlation between numerical features
print("\nTitanic Correlation Matrix:")
print(titanic_df[titanic_num_features].corr())
print("\nHousing Correlation Matrix:")
print(housing_df[housing_num_features].corr())
```

```
Titanic Correlation Matrix:
             PassengerId  Survived    Pclass       Age     SibSp     Parch  \
PassengerId     1.000000 -0.005007 -0.035144  0.036847 -0.057527 -0.001652
Survived       -0.005007  1.000000 -0.338481 -0.077221 -0.035322  0.081629
Pclass         -0.035144 -0.338481  1.000000 -0.369226  0.083081  0.018443
Age             0.036847 -0.077221 -0.369226  1.000000 -0.308247 -0.189119
SibSp          -0.057527 -0.035322  0.083081 -0.308247  1.000000  0.414838
Parch          -0.001652  0.081629  0.018443 -0.189119  0.414838  1.000000
Fare            0.012658  0.257307 -0.549500  0.096067  0.159651  0.216225

                 Fare
PassengerId  0.012658
Survived     0.257307
Pclass      -0.549500
Age          0.096067
SibSp        0.159651
Parch        0.216225
Fare         1.000000
```

```
Housing Correlation Matrix:
                    longitude  latitude  housing_median_age  total_rooms  \
longitude            1.000000 -0.924664           -0.108197     0.044568
latitude            -0.924664  1.000000            0.011173    -0.036100
housing_median_age  -0.108197  0.011173            1.000000    -0.361262
total_rooms          0.044568 -0.036100           -0.361262     1.000000
total_bedrooms       0.069608 -0.066983           -0.320451     0.930380
population           0.099773 -0.108785           -0.296244     0.857126
households           0.055310 -0.071035           -0.302916     0.918484
median_income       -0.015176 -0.079809           -0.119034     0.198050
median_house_value  -0.045967 -0.144160            0.105623     0.134153

                    total_bedrooms  population  households  median_income  \
longitude                 0.069608    0.099773    0.055310      -0.015176
latitude                 -0.066983   -0.108785   -0.071035      -0.079809
housing_median_age       -0.320451   -0.296244   -0.302916      -0.119034
total_rooms               0.930380    0.857126    0.918484       0.198050
total_bedrooms            1.000000    0.877747    0.979728      -0.007723
population                0.877747    1.000000    0.907222       0.004834
households                0.979728    0.907222    1.000000       0.013033
median_income            -0.007723    0.004834    0.013033       1.000000
median_house_value        0.049686   -0.024650    0.065843       0.688075

                    median_house_value
longitude                    -0.045967
latitude                     -0.144160
housing_median_age            0.105623
total_rooms                   0.134153
total_bedrooms                0.049686
population                   -0.024650
households                    0.065843
median_income                 0.688075
median_house_value            1.000000
```

[18]:
```python
# Task 3: Handling Missing Values
```

[19]:
```python
# Identify missing values and count them
print("\nMissing Values Count in Titanic:")
print(titanic_df.isnull().sum())
print("\nMissing Values Count in Housing:")
print(housing_df.isnull().sum())
```

```
Missing Values Count in Titanic:
PassengerId      0
Survived         0
Pclass           0
```

```
Name            0
Sex             0
Age           177
SibSp           0
Parch           0
Ticket          0
Fare            0
Cabin         687
Embarked        2
dtype: int64

Missing Values Count in Housing:
longitude               0
latitude                0
housing_median_age      0
total_rooms             0
total_bedrooms        207
population              0
households              0
median_income           0
median_house_value      0
ocean_proximity         0
dtype: int64
```

[30]:
```
# Handling missing values (only for numerical columns)
titanic_df[titanic_num_features] = titanic_df[titanic_num_features].
  ↪fillna(titanic_df[titanic_num_features].mean())
housing_df[housing_num_features] = housing_df[housing_num_features].
  ↪fillna(housing_df[housing_num_features].mean())
```

[22]:
```
# Task 4: Handling Outliers
```

[23]:
```
# Detect outliers using IQR
Q1 = housing_df[housing_num_features].quantile(0.25)
Q3 = housing_df[housing_num_features].quantile(0.75)
IQR = Q3 - Q1
outliers = (housing_df[housing_num_features] < (Q1 - 1.5 * IQR)) |␣
  ↪(housing_df[housing_num_features] > (Q3 + 1.5 * IQR))
housing_df = housing_df[~outliers.any(axis=1)]  # Remove outliers
```

[24]:
```
# Task 5: Data Encoding
```

[35]:
```
# Ensure categorical features exist before encoding(one hot encoding)
valid_titanic_cat_features = [col for col in titanic_cat_features if col in␣
  ↪titanic_df.columns]
titanic_df = pd.get_dummies(titanic_df, columns=valid_titanic_cat_features,␣
  ↪drop_first=True)
```

```
# Label Encoding for categorical features in Housing dataset
label_encoder = LabelEncoder()
for col in housing_cat_features:
    if housing_df[col].dtype == 'object':  # Ensure column is categorical
        housing_df[col] = label_encoder.fit_transform(housing_df[col])
```

[26]:
```
# Task 6: Feature Scaling
```

[27]:
```
# Min-Max Scaling
from sklearn.preprocessing import MinMaxScaler, StandardScaler
scaler = MinMaxScaler()
housing_df[housing_num_features] = scaler.
 ↪fit_transform(housing_df[housing_num_features])
```

[28]:
```
# Standardization
std_scaler = StandardScaler()
housing_df[housing_num_features] = std_scaler.
 ↪fit_transform(housing_df[housing_num_features])
```

[29]:
```
print("\nFinal Processed Titanic Dataset:")
print(titanic_df.head())
print("\nFinal Processed Housing Dataset:")
print(housing_df.head())
```

```
Final Processed Titanic Dataset:
   PassengerId  Survived  Pclass   Age  SibSp  Parch     Fare  \
0            1         0       3  22.0      1      0   7.2500
1            2         1       1  38.0      1      0  71.2833
2            3         1       3  26.0      0      0   7.9250
3            4         1       1  35.0      1      0  53.1000
4            5         0       3  35.0      0      0   8.0500

   Name_Abbott, Mr. Rossmore Edward  Name_Abbott, Mrs. Stanton (Rosa Hunt)  \
0                             False                                  False
1                             False                                  False
2                             False                                  False
3                             False                                  False
4                             False                                  False

   Name_Abelson, Mr. Samuel  …  Cabin_F G63  Cabin_F G73  Cabin_F2  \
0                     False  …        False        False     False
1                     False  …        False        False     False
2                     False  …        False        False     False
3                     False  …        False        False     False
4                     False  …        False        False     False
```

```
   Cabin_F33  Cabin_F38  Cabin_F4  Cabin_G6  Cabin_T  Embarked_Q  Embarked_S
0      False      False     False     False    False       False        True
1      False      False     False     False    False       False       False
2      False      False     False     False    False       False        True
3      False      False     False     False    False       False        True
4      False      False     False     False    False       False        True

[5 rows x 1726 columns]

Final Processed Housing Dataset:
   longitude  latitude  housing_median_age  total_rooms  total_bedrooms  \
2  -1.314915  0.994305            1.843959    -0.622296       -1.154160
3  -1.319903  0.994305            1.843959    -0.799005       -0.951153
4  -1.319903  0.994305            1.843959    -0.475801       -0.748146
5  -1.319903  0.994305            1.843959    -1.124041       -1.050401
6  -1.319903  0.989690            1.843959     0.355557        0.194709

   population  households  median_income  median_house_value ocean_proximity
2   -1.163692   -1.166347       2.551010            1.760499        NEAR BAY
3   -1.061011   -0.962390       1.432657            1.645303        NEAR BAY
4   -1.049418   -0.768145       0.187802            1.654902        NEAR BAY
5   -1.301153   -1.088649       0.319846            0.881593        NEAR BAY
6   -0.173313    0.470165       0.058183            1.196250        NEAR BAY
```

[ ]: