# Github repo 2500-014-9697

November 29, 2025

## 1 Analysis of Rough Sleeping Data in London From 2023 - 2025

#### 1.0.1 Objectives

This notebook's aim is to do an initial exploratory analysis into the dataset for homelessness in London.

There are the categories of Nationality, age, gender, ethnicity, support needs, rough sleeping, armed forces, and accomodation outcomes set measured from 33 Area Councils.

While the scope of the dataset is wide, this notebook will look at the total numbers and gender by Area councils over 8 quarters.

**Initialising the Environment**

[6]:
```
import warnings
warnings.filterwarnings("ignore")

import numpy as np
import pandas as pd
import io
import requests
```

#### 1.0.2 Data Sources

The data comes from Combined Homelessness and Information Network (CHAIN) and describes the numbers of rough sleepers in London, etc. https://data.london.gov.uk/dataset/rough-sleeping-in-london-chain-reports-2n88x/

**Loading the Data**

[23]:
```
# Load the CSV data from the provided URL

# Loading Age of Rough Sleepers CSV
age_url = 'https://raw.githubusercontent.com/dreamsmartins/ec-assignments/
↪09691eac4bb562df9039a9058a2d56cca2057882/
↪Age%20of%20people%20seen%20Rough%20Sleeping%20LDN%2023%20Q3%20-%2025%20Q%202.
↪csv'
resp = requests.get(age_url, timeout=10)
resp.raise_for_status()
age_rs = pd.read_csv(io.StringIO(resp.content.decode('utf-8')))
```

```python
# Loading Ethnicity of Rough Sleepers CSV
ethn_url = 'https://raw.githubusercontent.com/dreamsmartins/ec-assignments/09691eac4bb562df9039a9058a2d56cca2057882/Ethnicity%20of%20People%20Seen%20Rough%20Sleeping%20LDN%2023%20Q3%20-%2025%20Q2.csv'
resp = requests.get(ethn_url, timeout=10)
resp.raise_for_status()
ethn_rs = pd.read_csv(io.StringIO(resp.content.decode('utf-8')))

# Loading Gender of Rough Sleepers CSV
# Note: remove the 'blob/' segment from raw.githubusercontent.com URLs
gen_url = 'https://raw.githubusercontent.com/dreamsmartins/ec-assignments/09691eac4bb562df9039a9058a2d56cca2057882/Gender%20of%20People%20seen%20rough%20sleeping%20LDN%2023%20Q3%20-%2025%20Q2.csv'
resp = requests.get(gen_url, timeout=10)
resp.raise_for_status()
gen_rs = pd.read_csv(io.StringIO(resp.content.decode('utf-8')))

# Load Total Number of People Seen Rough Sleeping CSV
tot_url = 'https://raw.githubusercontent.com/dreamsmartins/ec-assignments/09691eac4bb562df9039a9058a2d56cca2057882/Number%20of%20People%20Seen%20Rough%20Sleeping%20in%20LDN%2023%20Q3%20-%2025%20Q2.csv'
resp = requests.get(tot_url, timeout=10)
resp.raise_for_status()
tot_rs = pd.read_csv(io.StringIO(resp.content.decode('utf-8')))

# Load Support Needs of People Seen Rough Sleeping CSV
sup_url = 'https://raw.githubusercontent.com/dreamsmartins/ec-assignments/09691eac4bb562df9039a9058a2d56cca2057882/Support%20needs%20combo%20of%20Rough%20Sleeper%20LDN%2023%20Q3%20-%2025%20Q2.csv'
resp = requests.get(sup_url, timeout=10)
resp.raise_for_status()
sup_rs = pd.read_csv(io.StringIO(resp.content.decode('utf-8')))


print("Age Shape", age_rs.shape)

print("Ethnicity Frame Shape", ethn_rs.shape)

print("Gender Data Shape", gen_rs.shape)

print("Total Numbers", tot_rs.shape)
```

```
print("Support needs Data Frame",sup_rs.shape)
```

```
Age Shape (47, 58)
Ethnicity Frame Shape (47, 178)
Gender Data Shape (47, 42)
Total Numbers (46, 10)
Support needs Data Frame (48, 130)
```

### 1.0.3 Data inventory

Out of one spreadsheet workbook with 15 worksheets of separate data spanning records from quarter 3 of 2023 to quarter to of 2025.

I extracted 5 sheets to narrow down and focus on: - Total number, - Age, - Gender, - Ethnicity, - and Support needs of rough sleepers.

The data has eight (8) quarterly periods, this spans over 2 years…

[26]:
```
# Describe the Total Numbers of Rough Sleepers dataset
print("\nTotal Numbers Data Description:\n", tot_rs.describe(include='all'))
```

```
Total Data Description:
                              Area GSS Code   2023-24 Q3   2023-24 Q4  \
count                           44       37    37.000000    37.000000
unique                          44       35          NaN          NaN
top     Greater London Authority                    NaN          NaN
freq                             1        3          NaN          NaN
mean                           NaN      NaN   242.756757   227.918919
std                            NaN      NaN   712.930860   673.030063
min                            NaN      NaN     4.000000     1.000000
25%                            NaN      NaN    46.000000    34.000000
50%                            NaN      NaN    99.000000    84.000000
75%                            NaN      NaN   156.000000   157.000000
max                            NaN      NaN  4389.000000  4118.000000

         2024-25 Q1   2024-25 Q2   2024-25 Q3   2024-25 Q4   2025-26 Q1  \
count     37.000000    37.000000    37.000000    37.000000    37.000000
unique          NaN          NaN          NaN          NaN          NaN
top             NaN          NaN          NaN          NaN          NaN
freq            NaN          NaN          NaN          NaN          NaN
mean     232.756757   263.000000   254.027027   244.540541   242.864865
std      686.744349   780.527706   753.544973   726.132280   716.540615
min        0.000000     2.000000     2.000000     1.000000     1.000000
25%       41.000000    52.000000    49.000000    44.000000    44.000000
50%       81.000000   106.000000   107.000000    96.000000    98.000000
75%      162.000000   197.000000   155.000000   164.000000   178.000000
max     4223.000000  4780.000000  4612.000000  4427.000000  4392.000000

         2025-26 Q2
```

```
count      37.000000
unique           NaN
top              NaN
freq             NaN
mean      259.162162
std       771.828151
min         4.000000
25%        51.000000
50%       106.000000
75%       171.000000
max      4711.000000
```

[37]:
```
# Total Numbers of Rough Sleepers dataset
tot_rs.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 46 entries, 0 to 45
Data columns (total 10 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   Area         44 non-null     object
 1   GSS Code     37 non-null     object
 2   2023-24 Q3   37 non-null     float64
 3   2023-24 Q4   37 non-null     float64
 4   2024-25 Q1   37 non-null     float64
 5   2024-25 Q2   37 non-null     float64
 6   2024-25 Q3   37 non-null     float64
 7   2024-25 Q4   37 non-null     float64
 8   2025-26 Q1   37 non-null     float64
 9   2025-26 Q2   37 non-null     float64
dtypes: float64(8), object(2)
memory usage: 3.7+ KB
```

[40]:
```
# Total Numbers of Rough Sleepers dataset
tot_rs.head()
```

[40]:
```
                        Area   GSS Code  2023-24 Q3  2023-24 Q4  2024-25 Q1  \
0  Greater London Authority  E12000007      4389.0      4118.0      4223.0
1        Barking & Dagenham  E09000002        46.0        28.0        41.0
2                    Barnet  E09000003        54.0        72.0        70.0
3                    Bexley  E09000004        49.0        34.0        39.0
4                     Brent  E09000005       143.0       158.0       158.0

   2024-25 Q2  2024-25 Q3  2024-25 Q4  2025-26 Q1  2025-26 Q2
0      4780.0      4612.0      4427.0      4392.0      4711.0
1        40.0        48.0        44.0        41.0        51.0
2        85.0        52.0        52.0        57.0        62.0
3        52.0        49.0        48.0        44.0        58.0
```

```
4       215.0       149.0       187.0       184.0       176.0
```

[39]:
```python
# Total Numbers of Rough Sleepers dataset
# Finding missing values
print("\nMissing Values in Total Numbers Data:\n", tot_rs.isnull().sum())
```

```
Missing Values in Total Numbers Data:
 Area           2
GSS Code      9
2023-24 Q3    9
2023-24 Q4    9
2024-25 Q1    9
2024-25 Q2    9
2024-25 Q3    9
2024-25 Q4    9
2025-26 Q1    9
2025-26 Q2    9
dtype: int64
```

[42]:
```python
# Total Numbers of Rough Sleepers dataset
# Find duplicates
duplicates = tot_rs.duplicated()
print("\nNumber of duplicate rows in Total Numbers Data:", duplicates.sum())
```

```
Number of duplicate rows in Total Numbers Data: 1
```

[43]:
```python
# Total Numbers of Rough Sleepers dataset
# Show the datatypes of each column
print("\nTotal Numbers Data Types:\n", tot_rs.dtypes)
```

```
Total Numbers Data Types:
 Area           object
GSS Code       object
2023-24 Q3    float64
2023-24 Q4    float64
2024-25 Q1    float64
2024-25 Q2    float64
2024-25 Q3    float64
2024-25 Q4    float64
2025-26 Q1    float64
2025-26 Q2    float64
dtype: object
```

[46]:
```python
# Total Numbers of Rough Sleepers dataset
# Printing columns and index name
print("Columns:", tot_rs.columns.tolist())
```

```
print("Index name:", tot_rs.index.name)
print(tot_rs.head())
```

```
Columns: ['Area', 'GSS Code', '2023-24 Q3', '2023-24 Q4', '2024-25 Q1', '2024-25
Q2', '2024-25 Q3', '2024-25 Q4', '2025-26 Q1', '2025-26 Q2']
Index name: None
                         Area   GSS Code  2023-24 Q3  2023-24 Q4  2024-25 Q1  \
0  Greater London Authority  E12000007      4389.0      4118.0      4223.0
1         Barking & Dagenham  E09000002        46.0        28.0        41.0
2                     Barnet  E09000003        54.0        72.0        70.0
3                     Bexley  E09000004        49.0        34.0        39.0
4                      Brent  E09000005       143.0       158.0       158.0

   2024-25 Q2  2024-25 Q3  2024-25 Q4  2025-26 Q1  2025-26 Q2
0      4780.0      4612.0      4427.0      4392.0      4711.0
1        40.0        48.0        44.0        41.0        51.0
2        85.0        52.0        52.0        57.0        62.0
3        52.0        49.0        48.0        44.0        58.0
4       215.0       149.0       187.0       184.0       176.0
```

[54]:

```
# Sum of second row to the 37th row (Area councils and Transit Hub Counts)
row_sum = tot_rs.iloc[1:38].sum(numeric_only=True)
print("\nSum of rows 2 to 37:\n", row_sum)

#Sum of the second row to the 34th row (Area councils only)
row_sum_2 = tot_rs.iloc[1:35].sum(numeric_only=True)
print("\nSum of rows 2 to 34:\n", row_sum_2)

# Greater London Area total (first row, excluding area councils)
GLA_total = tot_rs.iloc[0, 1:]
print("\nGreater London Area Total (excluding area name):\n", GLA_total)

#Subract the row_sum from Greater London Area total
adjusted_total = GLA_total - row_sum
print("\nAdjusted Greater London Area Total (after subtracting sub-areas):\n",␣
↪adjusted_total)

#Subract the row_sum2 from Greater London Area total
adjusted_total_2 = GLA_total - row_sum_2
print("\nAdjusted Greater London Area Total (after subtracting sub-areas up to␣
↪row 34):\n", adjusted_total_2)
```

```
Sum of rows 2 to 37:
 2023-24 Q3    4593.0
2023-24 Q4    4315.0
2024-25 Q1    4389.0
2024-25 Q2    4951.0
```

```
2024-25 Q3    4787.0
2024-25 Q4    4621.0
2025-26 Q1    4594.0
2025-26 Q2    4878.0
dtype: float64

Sum of rows 2 to 34:
 2023-24 Q3    4497.0
2023-24 Q4    4230.0
2024-25 Q1    4312.0
2024-25 Q2    4882.0
2024-25 Q3    4705.0
2024-25 Q4    4524.0
2025-26 Q1    4535.0
2025-26 Q2    4832.0
dtype: float64

Greater London Area Total (excluding area name):
 GSS Code      E12000007
2023-24 Q3       4389.0
2023-24 Q4       4118.0
2024-25 Q1       4223.0
2024-25 Q2       4780.0
2024-25 Q3       4612.0
2024-25 Q4       4427.0
2025-26 Q1       4392.0
2025-26 Q2       4711.0
Name: 0, dtype: object

Adjusted Greater London Area Total (after subtracting sub-areas):
 2023-24 Q3   -204.0
2023-24 Q4   -197.0
2024-25 Q1   -166.0
2024-25 Q2   -171.0
2024-25 Q3   -175.0
2024-25 Q4   -194.0
2025-26 Q1   -202.0
2025-26 Q2   -167.0
GSS Code        NaN
dtype: object

Adjusted Greater London Area Total (after subtracting sub-areas up to row 34):
 2023-24 Q3   -108.0
2023-24 Q4   -112.0
2024-25 Q1    -89.0
2024-25 Q2   -102.0
2024-25 Q3    -93.0
2024-25 Q4    -97.0
```

```
2025-26 Q1   -143.0
2025-26 Q2   -121.0
GSS Code        NaN
dtype: object
```

### 1.0.4 Dataset Inventory Summary

The data appears not to be arranged for direct analysis. For example, the "Area" column contains figures for Greater London Authority and a supposed break down for 33 Area councils, Bus route, tube line, and Heathrow.

- The column "GSS Code" is not useful for statistical description or analysis, and will be removed.
- Oddly, there is a discrepancy between the GLA total figure and the sum of either the Area councils alone or the Area councils including Bus route, tube line, and Heathrow. The GLA total figures are smaller than any combination of the Area council and transit figues.
- It seems strange to have a special category for rough sleepers seen in and on transit hubs separate from the Area councils they are situated without reconciling how they differ.
- In the case of cleansing, tranist hubs; bus routes, tube line, and Heathrow will be disregard from any inference.
- More importantly, there is confusion as to which figure to use for the totals, because, the sum of Area councils don't match with the total of Greater London Authority figure.
- Using the sum of the Area councils might be a better choice, because at preesent there is a more transparency to that total.

### 1.0.5 Data Cleansing for the Total Numbers of Rough Sleepers

[67]:
```
tot_rs_new = tot_rs.copy()

# Drop row 1
tot_rs_new = tot_rs_new.drop(index=0)
tot_rs_new.reset_index(drop=True, inplace=True)
tot_rs_new.head()

# Drop GSS Code column
tot_rs_new = tot_rs_new.drop(columns=['GSS Code'])
tot_rs_new.head()

# Drop rows 34 to 45 (inclusive) (Transit hubs and extraneous information in
↪the source data)
tot_rs_new = tot_rs_new.drop(index=range(33, 45))
tot_rs_new
```

[67]:
```
                    Area  2023-24 Q3  2023-24 Q4  2024-25 Q1  2024-25 Q2  \
0     Barking & Dagenham        46.0        28.0        41.0        40.0
1                 Barnet        54.0        72.0        70.0        85.0
```

| | | | | | |
|---|---|---|---|---|---|
| 2 | Bexley | 49.0 | 34.0 | 39.0 | 52.0 |
| 3 | Brent | 143.0 | 158.0 | 158.0 | 215.0 |
| 4 | Bromley | 34.0 | 39.0 | 51.0 | 52.0 |
| 5 | Camden | 330.0 | 341.0 | 310.0 | 298.0 |
| 6 | City of London | 279.0 | 260.0 | 298.0 | 263.0 |
| 7 | Croydon | 143.0 | 124.0 | 137.0 | 134.0 |
| 8 | Ealing | 295.0 | 259.0 | 239.0 | 265.0 |
| 9 | Enfield | 54.0 | 63.0 | 71.0 | 83.0 |
| 10 | Greenwich | 135.0 | 134.0 | 136.0 | 162.0 |
| 11 | Hackney | 81.0 | 61.0 | 88.0 | 113.0 |
| 12 | Hammersmith & Fulham | 168.0 | 113.0 | 165.0 | 167.0 |
| 13 | Haringey | 156.0 | 157.0 | 166.0 | 131.0 |
| 14 | Harrow | 40.0 | 34.0 | 19.0 | 25.0 |
| 15 | Havering | 16.0 | 17.0 | 21.0 | 22.0 |
| 16 | Hillingdon | 104.0 | 118.0 | 118.0 | 140.0 |
| 17 | Hounslow | 144.0 | 63.0 | 77.0 | 106.0 |
| 18 | Islington | 146.0 | 143.0 | 162.0 | 197.0 |
| 19 | Kensington & Chelsea | 81.0 | 75.0 | 65.0 | 67.0 |
| 20 | Kingston upon Thames | 41.0 | 33.0 | 38.0 | 60.0 |
| 21 | Lambeth | 242.0 | 163.0 | 144.0 | 165.0 |
| 22 | Lewisham | 107.0 | 99.0 | 102.0 | 113.0 |
| 23 | Merton | 20.0 | 22.0 | 25.0 | 22.0 |
| 24 | Newham | 176.0 | 163.0 | 176.0 | 197.0 |
| 25 | Redbridge | 99.0 | 89.0 | 81.0 | 95.0 |
| 26 | Richmond | 33.0 | 29.0 | 29.0 | 36.0 |
| 27 | Southwark | 218.0 | 162.0 | 196.0 | 215.0 |
| 28 | Sutton | 9.0 | 12.0 | 19.0 | 13.0 |
| 29 | Tower Hamlets | 147.0 | 155.0 | 147.0 | 198.0 |
| 30 | Waltham Forest | 71.0 | 65.0 | 78.0 | 68.0 |
| 31 | Wandsworth | 72.0 | 65.0 | 74.0 | 75.0 |
| 32 | Westminster | 740.0 | 850.0 | 752.0 | 988.0 |

| | 2024-25 Q3 | 2024-25 Q4 | 2025-26 Q1 | 2025-26 Q2 |
|---|---|---|---|---|
| 0 | 48.0 | 44.0 | 41.0 | 51.0 |
| 1 | 52.0 | 52.0 | 57.0 | 62.0 |
| 2 | 49.0 | 48.0 | 44.0 | 58.0 |
| 3 | 149.0 | 187.0 | 184.0 | 176.0 |
| 4 | 34.0 | 40.0 | 45.0 | 41.0 |
| 5 | 350.0 | 339.0 | 311.0 | 314.0 |
| 6 | 332.0 | 257.0 | 259.0 | 282.0 |
| 7 | 152.0 | 124.0 | 194.0 | 153.0 |
| 8 | 244.0 | 283.0 | 204.0 | 222.0 |
| 9 | 73.0 | 68.0 | 46.0 | 56.0 |
| 10 | 147.0 | 133.0 | 149.0 | 167.0 |
| 11 | 86.0 | 62.0 | 83.0 | 105.0 |
| 12 | 118.0 | 114.0 | 122.0 | 141.0 |
| 13 | 114.0 | 112.0 | 137.0 | 144.0 |

```
14        28.0       28.0       24.0       21.0
15        14.0       14.0       17.0       21.0
16       167.0      165.0      177.0      132.0
17       114.0      103.0      132.0      126.0
18       155.0      136.0      131.0      124.0
19        82.0       81.0       84.0       89.0
20        60.0       42.0       40.0       57.0
21       144.0      147.0      178.0      192.0
22       110.0       89.0      122.0      106.0
23        21.0       17.0       17.0       23.0
24       204.0      164.0      164.0      171.0
25       107.0       90.0       96.0      106.0
26        32.0       26.0       26.0       37.0
27       201.0      195.0      200.0      207.0
28         9.0        9.0        5.0       14.0
29       189.0      175.0      192.0      209.0
30        89.0       96.0       98.0       85.0
31        62.0       58.0       61.0       75.0
32       945.0     1001.0      879.0     1053.0
```

### 1.0.6 Stats of the Cleaned Data of Total Numbers of Rough Sleepers by Area Councils

Now the dataset includes only the 33 Area councils, which are categorical over time variables.

[66]:
```python
tot_rs_new.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 33 entries, 0 to 32
Data columns (total 9 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Area        33 non-null     object
 1   2023-24 Q3  33 non-null     float64
 2   2023-24 Q4  33 non-null     float64
 3   2024-25 Q1  33 non-null     float64
 4   2024-25 Q2  33 non-null     float64
 5   2024-25 Q3  33 non-null     float64
 6   2024-25 Q4  33 non-null     float64
 7   2025-26 Q1  33 non-null     float64
 8   2025-26 Q2  33 non-null     float64
dtypes: float64(8), object(1)
memory usage: 2.4+ KB
```

[64]:
```python
# Display the cleaned Total Numbers of Rough Sleepers dataset
tot_rs_new.reset_index(drop=True, inplace=True)
tot_rs_new.describe(include='all')
```

[64]:
```
                     Area  2023-24 Q3  2023-24 Q4  2024-25 Q1  2024-25 Q2  \
count                  33   33.000000   33.000000   33.000000   33.000000
unique                 33         NaN         NaN         NaN         NaN
top     Barking & Dagenham        NaN         NaN         NaN         NaN
freq                    1         NaN         NaN         NaN         NaN
mean                  NaN  135.545455  127.272727  130.060606  147.333333
std                   NaN  136.874050  150.884325  135.146203  169.712195
min                   NaN    9.000000   12.000000   19.000000   13.000000
25%                   NaN   49.000000   39.000000   51.000000   60.000000
50%                   NaN  104.000000   89.000000   88.000000  113.000000
75%                   NaN  156.000000  157.000000  162.000000  197.000000
max                   NaN  740.000000  850.000000  752.000000  988.000000

        2024-25 Q3   2024-25 Q4  2025-26 Q1   2025-26 Q2
count    33.000000    33.000000   33.000000    33.000000
unique         NaN          NaN         NaN          NaN
top            NaN          NaN         NaN          NaN
freq           NaN          NaN         NaN          NaN
mean    141.848485   136.333333  136.939394   146.060606
std     166.579959   174.348729  153.476981   179.541245
min       9.000000     9.000000    5.000000    14.000000
25%      52.000000    48.000000   45.000000    57.000000
50%     110.000000    96.000000  122.000000   106.000000
75%     155.000000   164.000000  178.000000   171.000000
max     945.000000  1001.000000  879.000000  1053.000000
```

### 1.0.7 Rough Sleepers by Gender

There are 4 categories of gender: - Female - Male - Non-binary - Not known

[27]:
```python
# Describe the Gender split of Rough Sleepers in the dataset
print("\nGender Data Description:\n", gen_rs.describe(include='all'))
```

```
Gender Data Description:
                             Area GSS Code Total seen rough sleeping  \
count                         44       37                        38
unique                        44       35                        35
top     Greater London Authority                                 54
freq                           1        3                         2
mean                         NaN      NaN                       NaN
std                          NaN      NaN                       NaN
min                          NaN      NaN                       NaN
25%                          NaN      NaN                       NaN
50%                          NaN      NaN                       NaN
75%                          NaN      NaN                       NaN
max                          NaN      NaN                       NaN
```

| | Female | Male | Non-binary | Not known | \ |
|---|---|---|---|---|---|
| count | 37.000000 | 37.000000 | 37.000000 | 37.000000 | |
| unique | NaN | NaN | NaN | NaN | |
| top | NaN | NaN | NaN | NaN | |
| freq | NaN | NaN | NaN | NaN | |
| mean | 35.405405 | 200.945946 | 0.378378 | 6.027027 | |
| std | 106.058019 | 588.324785 | 1.209944 | 18.628661 | |
| min | 0.000000 | 3.000000 | 0.000000 | 0.000000 | |
| 25% | 6.000000 | 41.000000 | 0.000000 | 0.000000 | |
| 50% | 13.000000 | 86.000000 | 0.000000 | 1.000000 | |
| 75% | 22.000000 | 135.000000 | 0.000000 | 4.000000 | |
| max | 641.000000 | 3630.000000 | 7.000000 | 111.000000 | |

| | Total seen rough sleeping.1 | Female.1 | Male.1 | … | \ |
|---|---|---|---|---|---|
| count | 38 | 37.000000 | 37.000000 | … | |
| unique | 34 | NaN | NaN | … | |
| top | 63 | NaN | NaN | … | |
| freq | 2 | NaN | NaN | … | |
| mean | NaN | 32.675676 | 191.540541 | … | |
| std | NaN | 97.957206 | 564.210146 | … | |
| min | NaN | 0.000000 | 1.000000 | … | |
| 25% | NaN | 5.000000 | 31.000000 | … | |
| 50% | NaN | 13.000000 | 70.000000 | … | |
| 75% | NaN | 21.000000 | 130.000000 | … | |
| max | NaN | 592.000000 | 3458.000000 | … | |

| | Total seen rough sleeping.6 | Female.6 | Male.6 | Non-binary.6 | \ |
|---|---|---|---|---|---|
| count | 38 | 37.000000 | 37.000000 | 37.000000 | |
| unique | 36 | NaN | NaN | NaN | |
| top | 122 | NaN | NaN | NaN | |
| freq | 2 | NaN | NaN | NaN | |
| mean | NaN | 42.081081 | 196.945946 | 0.108108 | |
| std | NaN | 124.302583 | 580.890071 | 0.393262 | |
| min | NaN | 1.000000 | 0.000000 | 0.000000 | |
| 25% | NaN | 7.000000 | 37.000000 | 0.000000 | |
| 50% | NaN | 19.000000 | 79.000000 | 0.000000 | |
| 75% | NaN | 28.000000 | 150.000000 | 0.000000 | |
| max | NaN | 755.000000 | 3566.000000 | 2.000000 | |

| | Not known.6 | Total seen rough sleeping.7 | Female.7 | Male.7 | \ |
|---|---|---|---|---|---|
| count | 37.000000 | 38 | 37.000000 | 37.000000 | |
| unique | NaN | 36 | NaN | NaN | |
| top | NaN | 106 | NaN | NaN | |
| freq | NaN | 2 | NaN | NaN | |
| mean | 3.729730 | NaN | 46.864865 | 208.513514 | |
| std | 11.570193 | NaN | 141.551357 | 619.170171 | |
| min | 0.000000 | NaN | 1.000000 | 2.000000 | |
| 25% | 0.000000 | NaN | 9.000000 | 42.000000 | |

```
50%        0.000000                                  NaN   15.000000    83.000000
75%        2.000000                                  NaN   30.000000   136.000000
max       69.000000                                  NaN  853.000000  3788.000000

        Non-binary.7  Not known.7
count      37.000000    37.000000
unique           NaN          NaN
top              NaN          NaN
freq             NaN          NaN
mean        0.378378     3.405405
std         1.276961    10.491953
min         0.000000     0.000000
25%         0.000000     0.000000
50%         0.000000     0.000000
75%         0.000000     3.000000
max         7.000000    63.000000

[11 rows x 42 columns]
```

[ ]:
```
# Viewing the Gender dataset
gen_rs.head()
```

[ ]:
```
                        Area   GSS Code Total seen rough sleeping  Female  \
0  Greater London Authority  E12000007                       4389   641.0
1        Barking & Dagenham  E09000002                         46     4.0
2                    Barnet  E09000003                         54    13.0
3                    Bexley  E09000004                         49     5.0
4                     Brent  E09000005                        143     9.0

     Male  Non-binary  Not known Total seen rough sleeping.1  Female.1  \
0  3630.0         7.0      111.0                        4118     592.0
1    42.0         0.0        0.0                          28       5.0
2    41.0         0.0        0.0                          72       6.0
3    44.0         0.0        0.0                          34       2.0
4   132.0         0.0        2.0                         158      14.0

   Male.1  …  Total seen rough sleeping.6  Female.6  Male.6  Non-binary.6  \
0  3458.0  …                         4392     755.0  3566.0           2.0
1    23.0  …                           41       2.0    39.0           0.0
2    66.0  …                           57       2.0    55.0           0.0
3    31.0  …                           44       7.0    37.0           0.0
4   142.0  …                          184      22.0   158.0           0.0

   Not known.6  Total seen rough sleeping.7  Female.7  Male.7  Non-binary.7  \
0         69.0                         4711     853.0  3788.0           7.0
1          0.0                           51       4.0    47.0           0.0
2          0.0                           62       9.0    53.0           0.0
```

```
3          0.0                                   58        9.0     49.0            0.0
4          4.0                                  176       34.0    136.0            0.0

   Not known.7
0         63.0
1          0.0
2          0.0
3          0.0
4          6.0

[5 rows x 42 columns]
```

### 1.0.8 Data Inventory for Rough Sleeping Gender Data

[69]:
```python
# Sum of second row to the 37th row (Area councils and Transit Hub Counts)
gen_row_sum = gen_rs.iloc[1:38].sum(numeric_only=True)
print("\nSum of rows 2 to 37:\n", gen_row_sum)

#Sum of the second row to the 34th row (Area councils only)
gen_row_sum_2 = gen_rs.iloc[1:35].sum(numeric_only=True)
print("\nSum of rows 2 to 34:\n", gen_row_sum_2)

# Greater London Area total (first row, excluding area councils)
gen_GLA_total = gen_rs.iloc[0, 1:]
print("\nGreater London Area Total (excluding area name):\n", gen_GLA_total)

#Subract the row_sum from Greater London Area total
difference = gen_GLA_total - gen_row_sum
print("\nAdjusted Greater London Area Total (after subtracting sub-areas):\n",␣
↪difference)

#Subract the row_sum2 from Greater London Area total
difference_2 = gen_GLA_total - gen_row_sum_2
print("\nAdjusted Greater London Area Total (after subtracting sub-areas up to␣
↪row 34):\n", difference_2)
```

```
Sum of rows 2 to 37:
 Female          669.0
Male            3805.0
Non-binary         7.0
Not known        112.0
Female.1         617.0
Male.1          3629.0
Non-binary.1       4.0
Not known.1       65.0
Female.2         690.0
Male.2          3625.0
```

```
Non-binary.2       5.0
Not known.2       69.0
Female.3         821.0
Male.3          4021.0
Non-binary.3       6.0
Not known.3      103.0
Female.4         815.0
Male.4          3876.0
Non-binary.4       8.0
Not known.4       88.0
Female.5         757.0
Male.5          3789.0
Non-binary.5       8.0
Not known.5       67.0
Female.6         802.0
Male.6          3721.0
Non-binary.6       2.0
Not known.6       69.0
Female.7         881.0
Male.7          3927.0
Non-binary.7       7.0
Not known.7       63.0
dtype: float64

Sum of rows 2 to 34:
 Female           652.0
Male            3726.0
Non-binary         7.0
Not known        112.0
Female.1         596.0
Male.1          3565.0
Non-binary.1       4.0
Not known.1       65.0
Female.2         665.0
Male.2          3573.0
Non-binary.2       5.0
Not known.2       69.0
Female.3         801.0
Male.3          3972.0
Non-binary.3       6.0
Not known.3      103.0
Female.4         794.0
Male.4          3815.0
Non-binary.4       8.0
Not known.4       88.0
Female.5         734.0
Male.5          3715.0
Non-binary.5       8.0
```

```
Not known.5       67.0
Female.6         790.0
Male.6          3674.0
Non-binary.6       2.0
Not known.6       69.0
Female.7         867.0
Male.7          3896.0
Non-binary.7       7.0
Not known.7       62.0
dtype: float64

Greater London Area Total (excluding area name):
 GSS Code                        E12000007
Total seen rough sleeping           4389
Female                             641.0
Male                              3630.0
Non-binary                           7.0
Not known                          111.0
Total seen rough sleeping.1         4118
Female.1                           592.0
Male.1                            3458.0
Non-binary.1                         3.0
Not known.1                         65.0
Total seen rough sleeping.2         4223
Female.2                           668.0
Male.2                            3482.0
Non-binary.2                         4.0
Not known.2                         69.0
Total seen rough sleeping.3         4780
Female.3                           798.0
Male.3                            3874.0
Non-binary.3                         5.0
Not known.3                        103.0
Total seen rough sleeping.4         4612
Female.4                           779.0
Male.4                            3737.0
Non-binary.4                         8.0
Not known.4                         88.0
Total seen rough sleeping.5         4427
Female.5                           724.0
Male.5                            3628.0
Non-binary.5                         8.0
Not known.5                         67.0
Total seen rough sleeping.6         4392
Female.6                           755.0
Male.6                            3566.0
Non-binary.6                         2.0
Not known.6                         69.0
```

```
Total seen rough sleeping.7         4711
Female.7                           853.0
Male.7                            3788.0
Non-binary.7                         7.0
Not known.7                         63.0
Name: 0, dtype: object

Adjusted Greater London Area Total (after subtracting sub-areas):
 Female                         -28.0
Female.1                        -25.0
Female.2                        -22.0
Female.3                        -23.0
Female.4                        -36.0
Female.5                        -33.0
Female.6                        -47.0
Female.7                        -28.0
GSS Code                          NaN
Male                           -175.0
Male.1                         -171.0
Male.2                         -143.0
Male.3                         -147.0
Male.4                         -139.0
Male.5                         -161.0
Male.6                         -155.0
Male.7                         -139.0
Non-binary                        0.0
Non-binary.1                     -1.0
Non-binary.2                     -1.0
Non-binary.3                     -1.0
Non-binary.4                      0.0
Non-binary.5                      0.0
Non-binary.6                      0.0
Non-binary.7                      0.0
Not known                        -1.0
Not known.1                       0.0
Not known.2                       0.0
Not known.3                       0.0
Not known.4                       0.0
Not known.5                       0.0
Not known.6                       0.0
Not known.7                       0.0
Total seen rough sleeping         NaN
Total seen rough sleeping.1       NaN
Total seen rough sleeping.2       NaN
Total seen rough sleeping.3       NaN
Total seen rough sleeping.4       NaN
Total seen rough sleeping.5       NaN
Total seen rough sleeping.6       NaN
```

```
Total seen rough sleeping.7      NaN
dtype: object

Adjusted Greater London Area Total (after subtracting sub-areas up to row 34):
 Female                         -11.0
Female.1                        -4.0
Female.2                         3.0
Female.3                        -3.0
Female.4                       -15.0
Female.5                       -10.0
Female.6                       -35.0
Female.7                       -14.0
GSS Code                         NaN
Male                           -96.0
Male.1                        -107.0
Male.2                         -91.0
Male.3                         -98.0
Male.4                         -78.0
Male.5                         -87.0
Male.6                        -108.0
Male.7                        -108.0
Non-binary                       0.0
Non-binary.1                    -1.0
Non-binary.2                    -1.0
Non-binary.3                    -1.0
Non-binary.4                     0.0
Non-binary.5                     0.0
Non-binary.6                     0.0
Non-binary.7                     0.0
Not known                       -1.0
Not known.1                      0.0
Not known.2                      0.0
Not known.3                      0.0
Not known.4                      0.0
Not known.5                      0.0
Not known.6                      0.0
Not known.7                      1.0
Total seen rough sleeping        NaN
Total seen rough sleeping.1      NaN
Total seen rough sleeping.2      NaN
Total seen rough sleeping.3      NaN
Total seen rough sleeping.4      NaN
Total seen rough sleeping.5      NaN
Total seen rough sleeping.6      NaN
Total seen rough sleeping.7      NaN
dtype: object
```

[70]:
```
gen_rs.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 47 entries, 0 to 46
Data columns (total 42 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Area                          44 non-null     object
 1   GSS Code                      37 non-null     object
 2   Total seen rough sleeping     38 non-null     object
 3   Female                        37 non-null     float64
 4   Male                          37 non-null     float64
 5   Non-binary                    37 non-null     float64
 6   Not known                     37 non-null     float64
 7   Total seen rough sleeping.1   38 non-null     object
 8   Female.1                      37 non-null     float64
 9   Male.1                        37 non-null     float64
 10  Non-binary.1                  37 non-null     float64
 11  Not known.1                   37 non-null     float64
 12  Total seen rough sleeping.2   38 non-null     object
 13  Female.2                      37 non-null     float64
 14  Male.2                        37 non-null     float64
 15  Non-binary.2                  37 non-null     float64
 16  Not known.2                   37 non-null     float64
 17  Total seen rough sleeping.3   38 non-null     object
 18  Female.3                      37 non-null     float64
 19  Male.3                        37 non-null     float64
 20  Non-binary.3                  37 non-null     float64
 21  Not known.3                   37 non-null     float64
 22  Total seen rough sleeping.4   38 non-null     object
 23  Female.4                      37 non-null     float64
 24  Male.4                        37 non-null     float64
 25  Non-binary.4                  37 non-null     float64
 26  Not known.4                   37 non-null     float64
 27  Total seen rough sleeping.5   38 non-null     object
 28  Female.5                      37 non-null     float64
 29  Male.5                        37 non-null     float64
 30  Non-binary.5                  37 non-null     float64
 31  Not known.5                   37 non-null     float64
 32  Total seen rough sleeping.6   38 non-null     object
 33  Female.6                      37 non-null     float64
 34  Male.6                        37 non-null     float64
 35  Non-binary.6                  37 non-null     float64
 36  Not known.6                   37 non-null     float64
 37  Total seen rough sleeping.7   38 non-null     object
 38  Female.7                      37 non-null     float64
 39  Male.7                        37 non-null     float64
 40  Non-binary.7                  37 non-null     float64
```

```
 41  Not known.7                  37 non-null     float64
dtypes: float64(32), object(10)
memory usage: 15.6+ KB
```

**Summary** The data structure is similar to the total rough sleeping data. Therefore, the cleansing can follow similar steps.

The Gender split of the dataset is similar to the Total RS numbers, in terms of discrepancy with sum of the Area Council and the Greater London Authority Total.

- The Gender splits into 4 categories; Female, Male, Non-binary, and Not known. And this is over 8 quarters 2023 Q3 - 2025 Q2.
- Renaming the quarter headings to be more appropriate.

### 1.0.9 Data Cleansing for the Gender Numbers of Rough Sleepers

[ ]:
```python
# Copy of Gender dataset
gen_rs_new = gen_rs.copy()

# Drop row 1
gen_rs_new = gen_rs_new.drop(index=0)
gen_rs_new.reset_index(drop=True, inplace=True)
gen_rs_new.head()

# Drop GSS Code column
gen_rs_new = gen_rs_new.drop(columns=['GSS Code'])


# Drop rows 34 to 45 (inclusive) (Transit hubs and extraneous information in␣
↪the source data)
gen_rs_new = gen_rs_new.drop(index=range(33, 46))
gen_rs_new

gen_rs_new.tail(12)

# Change column names to be more descriptive
gen_rs_new = gen_rs_new.rename(columns={'Female': 'Female_23-24_Q3', 'Male':
↪'Male_23-24_Q3', 'Non-binary': 'Non_binary_23-24_Q3', 'Not known':␣
↪'Not_known_23-24_Q3' })
gen_rs_new = gen_rs_new.rename(columns={'Female.1': 'Female_23-24_Q4', 'Male.1':
↪'Male_23-24_Q4', 'Non-binary.1': 'Non_binary_23-24_Q4', 'Not known.1':␣
↪'Not_known_23-24_Q4' })
gen_rs_new = gen_rs_new.rename(columns={'Female.2': 'Female_24-25_Q1', 'Male.2':
↪'Male_24-25_Q1', 'Non-binary.2': 'Non_binary_24-25_Q1', 'Not known.2':␣
↪'Not_known_24-25_Q1' })
```

```
gen_rs_new = gen_rs_new.rename(columns={'Female.3': 'Female_24-25_Q2', 'Male.3':
 ↪'Male_24-25_Q2', 'Non-binary.3': 'Non_binary_24-25_Q2', 'Not known.3':␣
 ↪'Not_known_24-25_Q2' })
gen_rs_new = gen_rs_new.rename(columns={'Female.4': 'Female_24-25_Q3', 'Male.4':
 ↪'Male_24-25_Q3', 'Non-binary.4': 'Non_binary_24-25_Q3', 'Not known.4':␣
 ↪'Not_known_24-25_Q3' })
gen_rs_new = gen_rs_new.rename(columns={'Female.5': 'Female_24-25_Q4', 'Male.5':
 ↪'Male_24-25_Q4', 'Non-binary.5': 'Non_binary_24-25_Q4', 'Not known.5':␣
 ↪'Not_known_24-25_Q4' })
gen_rs_new = gen_rs_new.rename(columns={'Female.6': 'Female_25-26_Q1', 'Male.6':
 ↪'Male_25-26_Q1', 'Non-binary.6': 'Non_binary_25-26_Q1', 'Not known.6':␣
 ↪'Not_known_25-26_Q1' })
gen_rs_new = gen_rs_new.rename(columns={'Female.7': 'Female_25-26_Q2', 'Male.7':
 ↪'Male_25-26_Q2', 'Non-binary.7': 'Non_binary_25-26_Q2', 'Not known.7':␣
 ↪'Not_known_25-26_Q2' })


# Drop columns with 'Total seen rough sleeping'
gen_rs_new = gen_rs_new.drop(columns=['Total seen rough sleeping', 'Total seen␣
 ↪rough sleeping.1', 'Total seen rough sleeping.2', 'Total seen rough sleeping.
 ↪3', 'Total seen rough sleeping.4', 'Total seen rough sleeping.5', 'Total␣
 ↪seen rough sleeping.6', 'Total seen rough sleeping.7'])



gen_rs_new.head()
```

[ ]:

```
                Area  Female_23-24_Q3  Male_23-24_Q3  Non_binary_23-24_Q3  \
0  Barking & Dagenham              4.0           42.0                  0.0
1             Barnet             13.0           41.0                  0.0
2             Bexley              5.0           44.0                  0.0
3              Brent              9.0          132.0                  0.0
4            Bromley              1.0           33.0                  0.0

   Not_known_23-24_Q3  Female_23-24_Q4  Male_23-24_Q4  Non_binary_23-24_Q4  \
0                 0.0              5.0           23.0                  0.0
1                 0.0              6.0           66.0                  0.0
2                 0.0              2.0           31.0                  0.0
3                 2.0             14.0          142.0                  0.0
4                 0.0              5.0           34.0                  0.0

   Not_known_23-24_Q4  Female_24-25_Q1  …  Non_binary_24-25_Q4  \
0                 0.0              3.0  …                  0.0
1                 0.0             12.0  …                  0.0
2                 1.0              8.0  …                  0.0
3                 2.0             16.0  …                  0.0
4                 0.0              5.0  …                  0.0
```

```
   Not_known_24-25_Q4  Female_25-26_Q1  Male_25-26_Q1  Non_binary_25-26_Q1  \
0                 0.0              2.0           39.0                  0.0
1                 0.0              2.0           55.0                  0.0
2                 0.0              7.0           37.0                  0.0
3                 4.0             22.0          158.0                  0.0
4                 0.0              8.0           37.0                  0.0

   Not_known_25-26_Q1  Female_25-26_Q2  Male_25-26_Q2  Non_binary_25-26_Q2  \
0                 0.0              4.0           47.0                  0.0
1                 0.0              9.0           53.0                  0.0
2                 0.0              9.0           49.0                  0.0
3                 4.0             34.0          136.0                  0.0
4                 0.0              8.0           33.0                  0.0

   Not_known_25-26_Q2
0                 0.0
1                 0.0
2                 0.0
3                 6.0
4                 0.0

[5 rows x 33 columns]
```

[83]:
```
gen_rs_new.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 33 entries, 0 to 32
Data columns (total 33 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Area                 33 non-null     object
 1   Female_23-24_Q3      33 non-null     float64
 2   Male_23-24_Q3        33 non-null     float64
 3   Non_binary_23-24_Q3  33 non-null     float64
 4   Not_known_23-24_Q3   33 non-null     float64
 5   Female_23-24_Q4      33 non-null     float64
 6   Male_23-24_Q4        33 non-null     float64
 7   Non_binary_23-24_Q4  33 non-null     float64
 8   Not_known_23-24_Q4   33 non-null     float64
 9   Female_24-25_Q1      33 non-null     float64
 10  Male_24-25_Q1        33 non-null     float64
 11  Non_binary_24-25_Q1  33 non-null     float64
 12  Not_known_24-25_Q1   33 non-null     float64
 13  Female_24-25_Q2      33 non-null     float64
 14  Male_24-25_Q2        33 non-null     float64
 15  Non_binary_24-25_Q2  33 non-null     float64
 16  Not_known_24-25_Q2   33 non-null     float64
```

```
 17  Female_24-25_Q3      33 non-null     float64
 18  Male_24-25_Q3        33 non-null     float64
 19  Non_binary_24-25_Q3  33 non-null     float64
 20  Not_known_24-25_Q3   33 non-null     float64
 21  Female_24-25_Q4      33 non-null     float64
 22  Male_24-25_Q4        33 non-null     float64
 23  Non_binary_24-25_Q4  33 non-null     float64
 24  Not_known_24-25_Q4   33 non-null     float64
 25  Female_25-26_Q1      33 non-null     float64
 26  Male_25-26_Q1        33 non-null     float64
 27  Non_binary_25-26_Q1  33 non-null     float64
 28  Not_known_25-26_Q1   33 non-null     float64
 29  Female_25-26_Q2      33 non-null     float64
 30  Male_25-26_Q2        33 non-null     float64
 31  Non_binary_25-26_Q2  33 non-null     float64
 32  Not_known_25-26_Q2   33 non-null     float64
dtypes: float64(32), object(1)
memory usage: 8.6+ KB
```

[137]:
```
gen_rs_new.describe()
```

[137]:
```
       Female_23-24_Q3  Male_23-24_Q3  Non_binary_23-24_Q3  \
count        33.000000      33.000000            33.000000
mean         19.606061     112.333333             0.212121
std          29.300106     105.342497             0.484612
min           0.000000       9.000000             0.000000
25%           7.000000      42.000000             0.000000
50%          13.000000      92.000000             0.000000
75%          22.000000     135.000000             0.000000
max         166.000000     549.000000             2.000000

       Not_known_23-24_Q3  Female_23-24_Q4  Male_23-24_Q4  \
count           33.000000        33.000000      33.000000
mean             3.393939        17.848485     107.333333
std              5.957870        27.087960     121.835100
min              0.000000         0.000000      11.000000
25%              0.000000         5.000000      34.000000
50%              1.000000        13.000000      74.000000
75%              4.000000        20.000000     130.000000
max             25.000000       156.000000     674.000000

       Non_binary_23-24_Q4  Not_known_23-24_Q4  Female_24-25_Q1  \
count            33.000000           33.000000        33.000000
mean              0.121212            1.969697        19.969697
std               0.415149            3.995973        29.158495
min               0.000000            0.000000         2.000000
25%               0.000000            0.000000         5.000000
```

```
50%                0.000000            1.000000        15.000000
75%                0.000000            1.000000        20.000000
max                2.000000           19.000000       165.000000

       Male_24-25_Q1  …  Non_binary_24-25_Q4  Not_known_24-25_Q4  \
count      33.000000  …            33.000000           33.000000
mean      107.848485  …             0.242424            2.030303
std       104.846650  …             0.560708            4.726817
min        15.000000  …             0.000000            0.000000
25%        46.000000  …             0.000000            0.000000
50%        77.000000  …             0.000000            1.000000
75%       128.000000  …             0.000000            2.000000
max       574.000000  …             2.000000           26.000000

       Female_25-26_Q1  Male_25-26_Q1  Non_binary_25-26_Q1  \
count        33.000000      33.000000            33.000000
mean         23.848485     110.939394             0.060606
std          32.033304     119.616559             0.242306
min           1.000000       4.000000             0.000000
25%           8.000000      37.000000             0.000000
50%          19.000000      96.000000             0.000000
75%          28.000000     150.000000             0.000000
max         184.000000     679.000000             1.000000

       Not_known_25-26_Q1  Female_25-26_Q2  Male_25-26_Q2  \
count           33.000000        33.000000      33.000000
mean             2.090909        26.151515     117.818182
std              3.660291        40.398114     137.173807
min              0.000000         1.000000      13.000000
25%              0.000000         9.000000      47.000000
50%              1.000000        16.000000      90.000000
75%              2.000000        30.000000     136.000000
max             15.000000       239.000000     800.000000

       Non_binary_25-26_Q2  Not_known_25-26_Q2
count            33.000000           33.000000
mean              0.212121            1.878788
std               0.649883            3.089878
min               0.000000            0.000000
25%               0.000000            0.000000
50%               0.000000            0.000000
75%               0.000000            3.000000
max               3.000000           12.000000

[8 rows x 32 columns]
```

### 1.0.10 Exploratory Data Analysis

[101]:
```
# Sum of females for all quarters
# female_sum = gen_rs_new.filter(like='Female').sum(axis=1)
# print("\nSum of females for all quarters:\n", female_sum)

# Sum of all genders for Quarter 3 2023-2024
quarter_3_sum = gen_rs_new.filter(like='23-24_Q3').sum(axis=1)
# print("\nSum of all genders for Quarter 3 2023-2024:\n", quarter_3_sum)

# Sum of all Areas for Quarter 3 2023-2024
quarter_3_sum_tot = tot_rs_new.filter(like='23-24 Q3').sum(axis=1)
# print("\nSum of all Areas for Quarter 3 2023-2024:\n", quarter_3_sum_tot)



# Display a comparison between gender totals and overall totals for Quarter 3␣
↪2023-2024
comparison_q3 = pd.DataFrame({
    'Area': gen_rs_new['Area'],
    'Total Genders Q3 2023-2024': quarter_3_sum,
    'Total Overall Q3 2023-2024': quarter_3_sum_tot
})
#print("\nComparison between gender totals and overall totals for Quarter 3␣
↪2023-2024:\n", comparison_q3)
#gen_rs.info()
display(comparison_q3.head(10))
```

| | Area | Total Genders Q3 2023-2024 | Total Overall Q3 2023-2024 |
|---|---|---|---|
| 0 | Barking & Dagenham | 46.0 | 46.0 |
| 1 | Barnet | 54.0 | 54.0 |
| 2 | Bexley | 49.0 | 49.0 |
| 3 | Brent | 143.0 | 143.0 |
| 4 | Bromley | 34.0 | 34.0 |
| 5 | Camden | 330.0 | 330.0 |
| 6 | City of London | 279.0 | 279.0 |
| 7 | Croydon | 143.0 | 143.0 |
| 8 | Ealing | 295.0 | 295.0 |
| 9 | Enfield | 54.0 | 54.0 |

The above output, shows that the sum of the data matches between the total numbers and the gender splits per quarter.

While there is a consistent discrepancy between the sum of all area councils and the [removed] figure for Greater London Authority, the sums matching up shows that the data are good quality.

### 1.0.11 Stacked Bar Chart of Gender splits across the 8 Quarters

Using Plotly graph tool.

[100]:

```
import plotly.graph_objects as go

# 1. Setup Data
quarters = ['23-24_Q3', '23-24_Q4', '24-25_Q1', '24-25_Q2', '24-25_Q3',␣
 ↪'24-25_Q4', '25-26_Q1', '25-26_Q2']

female_totals = gen_rs_new.filter(like='Female').sum()
male_totals = gen_rs_new.filter(like='Male').sum()
non_binary_totals = gen_rs_new.filter(like='Non_binary').sum()
not_known_totals = gen_rs_new.filter(like='Not_known').sum()

fig = go.Figure()

# 2. Add Traces with Labels
# We add 'text' and 'textposition' to every trace

fig.add_trace(go.Bar(
    x=quarters,
    y=female_totals,
    name='Female',
    marker_color='blue',
    opacity=1.0,
    # --- New Label Code ---
    text=female_totals,   # The values to display
    textposition='auto'   # Puts label inside the bar if there is room
))

fig.add_trace(go.Bar(
    x=quarters,
    y=male_totals,
    name='Male',
    marker_color='orange',
    opacity=0.6,
    # --- New Label Code ---
    text=male_totals,
    textposition='auto'
))

fig.add_trace(go.Bar(
    x=quarters,
    y=non_binary_totals,
    name='Non Binary',
    marker_color='green',
    opacity=0.6,
    # --- New Label Code ---
    text=non_binary_totals,
    textposition='auto'
```

```
))

fig.add_trace(go.Bar(
    x=quarters,
    y=not_known_totals,
    name='Not known',
    marker_color='red',
    opacity=0.6,
    # --- New Label Code ---
    text=not_known_totals,
    textposition='auto'
))

# 3. Update Layout
fig.update_layout(
    title='Total Number of Rough Sleepers by Gender Through the Quarters',
    xaxis_title='Quarters',
    yaxis_title='Number of Rough Sleepers',
    barmode='stack', # Stacked bars usually look best with 'auto' text position
    legend_title='Gender',
    # Optional: Uniform font size for labels
    uniformtext_minsize=8,
    uniformtext_mode='hide'
)

fig.show()
```

**Summary** The stacked bar chart above shows gender splits for rough sleepers across 8 quarters from 2023 Q3 to 2025 Q2.

- We can see a trend upwards from 2024 Q4 for Male and Female.
- We can also see a larger proportion of rough sleepers are male, with very few number for Non-binary and not known.

#### 1.0.12 Clustered Horizontal Bar Chart of Rough Sleepers Grouped by Area Council

[149]:
```
#show a clustered horizontal bar chart for the total number of rough sleepers␣
↪by area for each quarter
import plotly.express as px
fig = px.bar(
    tot_rs_new,
    x=tot_rs_new.columns[1:],  # All quarter columns
    y='Area',
    orientation='h',
    title='Total Number of Rough Sleepers by Area for Each Quarter',
    labels={'value': 'Number of Rough Sleepers', 'Area': 'Area Council'},
    barmode='group'  # Clustered bars
```

```
)
# Increase the height of the plot for better readability
import plotly.express as px
fig.update_layout(
    title='Total Number of Rough Sleepers by Area',
    xaxis_title='Quarters',
    yaxis_title='Number of Rough Sleepers',
    barmode='group',       # Clustered bars
    height=2000,            # <--- CHANGE THIS VALUE
    legend_title='Gender'
)
fig.show()
```

The clustered chart above roughly shows the distribution of rough sleepers by area council. Westminster has the highest number of rough sleepers by far over 2 years.

While Havering, Merton, and Sutton have the least, judging from this quick analysis.

**Interactive Pie Chart For Gender: Male vs Female Proportion Comparisons**

[ ]:
```
# Show the proportion of Male vs Female rough sleepers in one quarter
import plotly.express as px
fig = px.pie(gen_rs_new.melt(id_vars=['Area'], var_name='var',␣
 ↪value_name='Count').query("var.str.contains('Male|Female')"),
             names='var',
                values='Count',
                title='Proportion of Male vs Female Rough Sleepers in the Last␣
 ↪Three Quarters')
fig.show()
```

The Pie chart emphasises the proportion of genders for rough sleepers. In the case of 2025 Q2, there are 82.1% rough sleeping males compared to 17.9% females.

N.B: The pie chart is interactive and proportions can be assessed dynamically by clicking on the legend (Gender Quarter of choice).

### 1.0.13 Interactive Horizontal Distribution Chart of Rough Sleepers

[142]:
```
# Distribution of rough sleepers by gender
import plotly.express as px
# Melt the wide-format gender dataframe into long format
m = gen_rs_new.melt(id_vars=['Area'], var_name='var', value_name='Count')
# Ensure Count is numeric and drop missing values
m['Count'] = pd.to_numeric(m['Count'], errors='coerce')
m = m.dropna(subset=['Count'])
# Extract Gender and Quarter from the variable name (e.g. 'Female_23-24_Q3')
m[['Gender','Quarter']] = m['var'].str.extract(r'(?P<Gender>.+)_(?
 ↪P<Quarter>\d{2}-\d{2}_Q\d)')
# Create a combined label for the y-axis: Gender and Quarter
m['Gender and Quarter'] = m['Gender'].str.strip() + ' ' + m['Quarter']
```

```
# Order the quarters if present
quarter_order =␣
↪['23-24_Q3','23-24_Q4','24-25_Q1','24-25_Q2','24-25_Q3','24-25_Q4','25-26_Q1','25-26_Q2']
m['Gender and Quarter'] = pd.Categorical(m['Gender and Quarter'], categories=[g␣
↪+ ' ' + q for q in quarter_order for g in m['Gender'].unique()] ,␣
↪ordered=False)
# Create horizontal box plot: x = numeric Count, y = Gender + Quarter
fig = px.box(m, x='Count', y='Gender and Quarter', orientation='h',
            title='Distribution of Rough Sleepers by Gender and Quarter',
            labels={'Count':'Number of Rough Sleepers', 'Gender and Quarter':
↪'Gender and Quarter'},
            points='all')
fig.update_layout(yaxis={'automargin': True})
fig.show()
```

[143]:
```
# Histogram of rough sleepers by Males in all quarters
import plotly.express as px
fig = px.histogram(gen_rs_new.melt(id_vars=['Area'], var_name='var',␣
↪value_name='Count').query("var.str.contains('Male')"),
            x='Count',
            nbins=50,
            title='Histogram of Rough Sleepers by Males in All Quarters',
            labels={'Count':'Number of Rough Sleepers'})
fig.show()
```

[144]:
```
# Histogram of rough sleepers by Females in all quarters

#Label the Quarters



import plotly.express as px
fig = px.histogram(gen_rs_new.melt(id_vars=['Area'], var_name='var',␣
↪value_name='Count').query("var.str.contains('Female')"),
            x='Count',
            nbins=50,
            title='Histogram of Rough Sleepers by Females in All Quarters',
            labels={'Count':'Number of Rough Sleepers'})
fig.show()
```

#### 1.0.14 Boxplot for Male Rough Sleepers through the Quarters

[145]:
```
# Box plot for rough sleepers by Males through the quarters
import plotly.express as px
fig = px.box(gen_rs_new.melt(id_vars=['Area'], var_name='var',␣
↪value_name='Count').query("var.str.contains('Male')"),
            x='var',
            y='Count',
```

```
             title='Box Plot for Rough Sleepers by Males Through the Quarters',
             labels={'var':'Quarter', 'Count':'Number of Rough Sleepers'})
fig.show()
```

[161]:
```
# Show the median for Males in 2025 Q2
male_median_25_26_Q2 = gen_rs_new['Male_25-26_Q2'].median()
print("Median number of Male Rough Sleepers in 2025 Q2:", male_median_25_26_Q2)

# Show the Interquartile Range (IQR) for Females in 2025 Q2
male_Q2 = gen_rs_new['Male_25-26_Q2']
Q1 = male_Q2.quantile(0.25)
Q3 = male_Q2.quantile(0.75)
IQR = Q3 - Q1
print("Interquartile Range (IQR) of Male Rough Sleepers in 2025 Q2:", IQR)


# Show the boxplot numbers for Females in 2025 Q2
male_boxplot_25_26_Q2 = gen_rs_new['Male_25-26_Q2'].describe()
print("\nBoxplot statistics for Male Rough Sleepers in 2025 Q2:\n")

display(male_boxplot_25_26_Q2)
```

```
Median number of Male Rough Sleepers in 2025 Q2: 90.0
Interquartile Range (IQR) of Male Rough Sleepers in 2025 Q2: 89.0

Boxplot statistics for Male Rough Sleepers in 2025 Q2:


count     33.000000
mean     117.818182
std      137.173807
min       13.000000
25%       47.000000
50%       90.000000
75%      136.000000
max      800.000000
Name: Male_25-26_Q2, dtype: float64
```

#### 1.0.15 Boxplot for Female Rough Sleepers through the Quarters

[146]:
```
# Box plot for rough sleepers by Females through the quarters
import plotly.express as px
fig = px.box(gen_rs_new.melt(id_vars=['Area'], var_name='var',␣
 ↪value_name='Count').query("var.str.contains('Female')"),
             x='var',
             y='Count',
             title='Box Plot for Rough Sleepers by Females Through the Quarters',
             labels={'var':'Quarter', 'Count':'Number of Rough Sleepers'})
```

```
fig.show()
```

### 1.0.16 Insight

Just as a brief illustration:

For 2025 Q2, see the boxplot diagram and the statistics for Male, female, and non-binary splits.

[160]:
```
# Show the median for Females in 2025 Q2
female_median_25_26_Q2 = gen_rs_new['Female_25-26_Q2'].median()
print("Median number of Female Rough Sleepers in 2025 Q2:",␣
↪female_median_25_26_Q2)

# Show the Interquartile Range (IQR) for Females in 2025 Q2
female_Q2 = gen_rs_new['Female_25-26_Q2']
Q1 = female_Q2.quantile(0.25)
Q3 = female_Q2.quantile(0.75)
IQR = Q3 - Q1
print("Interquartile Range (IQR) of Female Rough Sleepers in 2025 Q2:", IQR)


# Show the boxplot numbers for Females in 2025 Q2
female_boxplot_25_26_Q2 = gen_rs_new['Female_25-26_Q2'].describe()

print("\nBoxplot statistics for Female Rough Sleepers in 2025 Q2:\n")
display(female_boxplot_25_26_Q2)
```

```
Median number of Female Rough Sleepers in 2025 Q2: 16.0
Interquartile Range (IQR) of Female Rough Sleepers in 2025 Q2: 21.0

Boxplot statistics for Female Rough Sleepers in 2025 Q2:


count     33.000000
mean      26.151515
std       40.398114
min        1.000000
25%        9.000000
50%       16.000000
75%       30.000000
max      239.000000
Name: Female_25-26_Q2, dtype: float64
```

### 1.0.17 Boxplot for Non-binary Rough Sleepers through the Quarters

[147]:
```
# Box plot for rough sleepers by Non-binary through the quarters
import plotly.express as px
fig = px.box(gen_rs_new.melt(id_vars=['Area'], var_name='var',␣
↪value_name='Count').query("var.str.contains('Non_binary')"),
```

```
            x='var',
            y='Count',
            title='Box Plot for Rough Sleepers by Non-binary Through the␣
 ↪Quarters',
            labels={'var':'Quarter', 'Count':'Number of Rough Sleepers'})
fig.show()
```

[164]:
```
# Show the median for Non-binary in 2025 Q2
non_binary_median_25_26_Q2 = gen_rs_new['Non_binary_25-26_Q2'].median()
print("Median number of Non-binary Rough Sleepers in 2025 Q2:",␣
 ↪non_binary_median_25_26_Q2)

# Show the Interquartile Range (IQR) for Females in 2025 Q2
non_binary_Q2 = gen_rs_new['Non_binary_25-26_Q2']
Q1 = non_binary_Q2.quantile(0.25)
Q3 = non_binary_Q2.quantile(0.75)
IQR = Q3 - Q1
print("Interquartile Range (IQR) of Non-binary Rough Sleepers in 2025 Q2:", IQR)


# Show the boxplot numbers for Females in 2025 Q2
non_binary_boxplot_25_26_Q2 = gen_rs_new['Non_binary_25-26_Q2'].describe()

print("\nBoxplot statistics for Non-Binary Rough Sleepers in 2025 Q2:\n")
display(non_binary_boxplot_25_26_Q2)
```

```
Median number of Non-binary Rough Sleepers in 2025 Q2: 0.0
Interquartile Range (IQR) of Non-binary Rough Sleepers in 2025 Q2: 0.0

Boxplot statistics for Non-Binary Rough Sleepers in 2025 Q2:


count    33.000000
mean      0.212121
std       0.649883
min       0.000000
25%       0.000000
50%       0.000000
75%       0.000000
max       3.000000
Name: Non_binary_25-26_Q2, dtype: float64
```

## 2 Next Steps

While it may seem like this analysis is small in scope, it can be useful for charities planning shelters and need to understand which demographic, in this case male or female to prioritise.

Despite a considerable larger proportion of male to female and non-binary, a charity may choose to prioritise non-binary because the numbers are small and as well as their budget.

Same can be said for females as well. It could be argued that females could be more vulnerable than males, especially if they have children and their needs prioritising.

Apart from that insight, delving into the age, nationality, ethnicity, accomodation outcomes, and support needs dataset would be the next step to make more comparisons and find relationships across the data.

Down the line, I can compare this dataset against housing, living standards, and other environmental factors to understand causal effects of homelessness and scan for resources to aid the wider problem.