```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.cluster import KMeans
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
# Provide the full path to your dataset
df = pd.read_csv("/content/netflix_titles.csv", encoding='latin-1') # Changed the encoding to 'latin-1'")
```

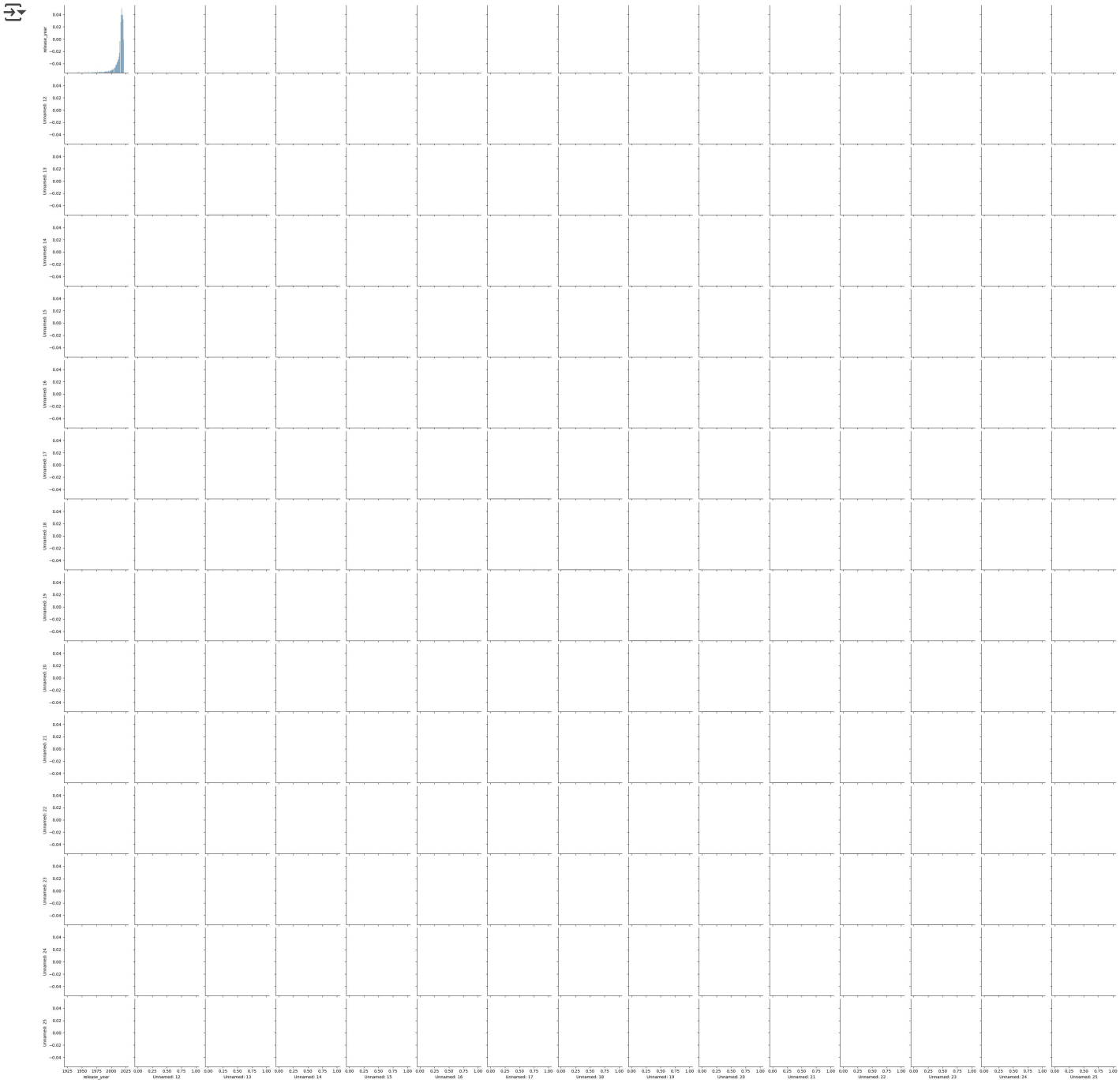
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8809 entries, 0 to 8808
Data columns (total 26 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   show_id       8809 non-null   object
 1   type          8809 non-null   object
 2   title         8809 non-null   object
 3   director      6175 non-null   object
 4   cast          7984 non-null   object
 5   country       7978 non-null   object
 6   date_added    8799 non-null   object
 7   release_year  8809 non-null   int64
 8   rating        8805 non-null   object
 9   duration      8806 non-null   object
 10  listed_in     8809 non-null   object
 11  description   8809 non-null   object
 12  Unnamed: 12   0 non-null      float64
 13  Unnamed: 13   0 non-null      float64
 14  Unnamed: 14   0 non-null      float64
 15  Unnamed: 15   0 non-null      float64
 16  Unnamed: 16   0 non-null      float64
 17  Unnamed: 17   0 non-null      float64
 18  Unnamed: 18   0 non-null      float64
 19  Unnamed: 19   0 non-null      float64
 20  Unnamed: 20   0 non-null      float64
 21  Unnamed: 21   0 non-null      float64
 22  Unnamed: 22   0 non-null      float64
 23  Unnamed: 23   0 non-null      float64
 24  Unnamed: 24   0 non-null      float64
 25  Unnamed: 25   0 non-null      float64
dtypes: float64(14), int64(1), object(11)
memory usage: 1.7+ MB
```

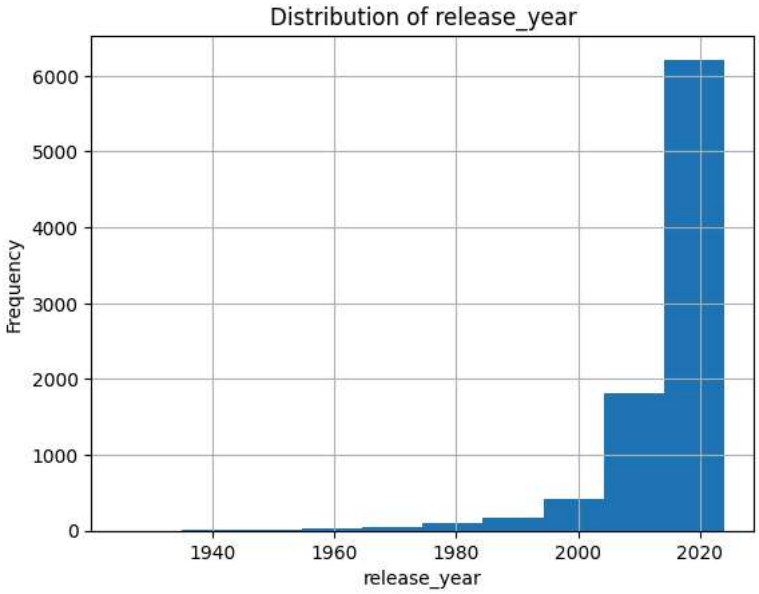
```
df.describe()
```

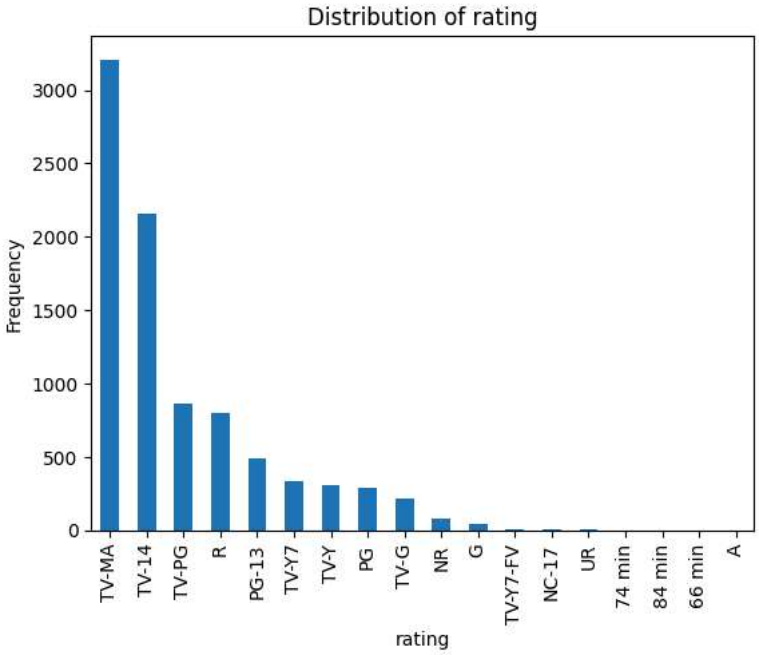
	release_year	Unnamed: 12	Unnamed: 13	Unnamed: 14	Unnamed: 15	Unnamed: 16	Unnamed: 17	Unnamed: 18	Unnamed: 19	Unnamed: 20	Unnamed: 21	Unnamed: 22	Unname
count	8809.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	(
mean	2014.181292	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
std	8.818932	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
min	1925.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
25%	2013.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
50%	2017.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
75%	2019.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na
max	2024.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Na

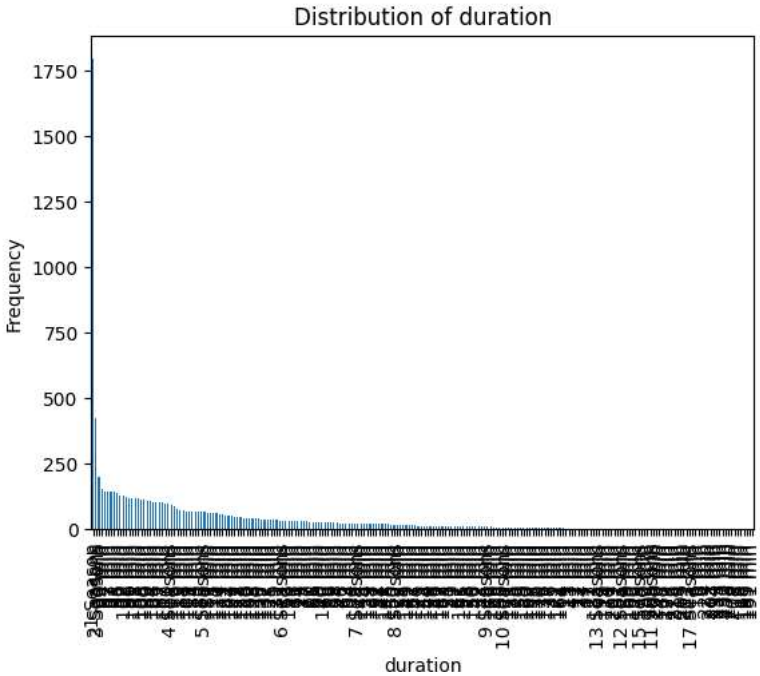
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Assuming 'df' is your dataframe

# Select only numerical columns for scatter plots
numerical_cols = df.select_dtypes(include=['number']).columns

# Create scatter plots for all pairs of numerical features
sns.pairplot(df[numerical_cols])
plt.show()
```


release_year
Unnamed: 12
Unnamed: 13
Unnamed: 14
Unnamed: 15
Unnamed: 16
Unnamed: 17
Unnamed: 18
Unnamed: 19
Unnamed: 20
Unnamed: 21
Unnamed: 22
Unnamed: 23
Unnamed: 24
Unnamed: 25
0.04
0.02
0.00
−0.02
−0.04
1925
1950
1975
2000
2025
0.00
0.25
0.50
0.75
1.00

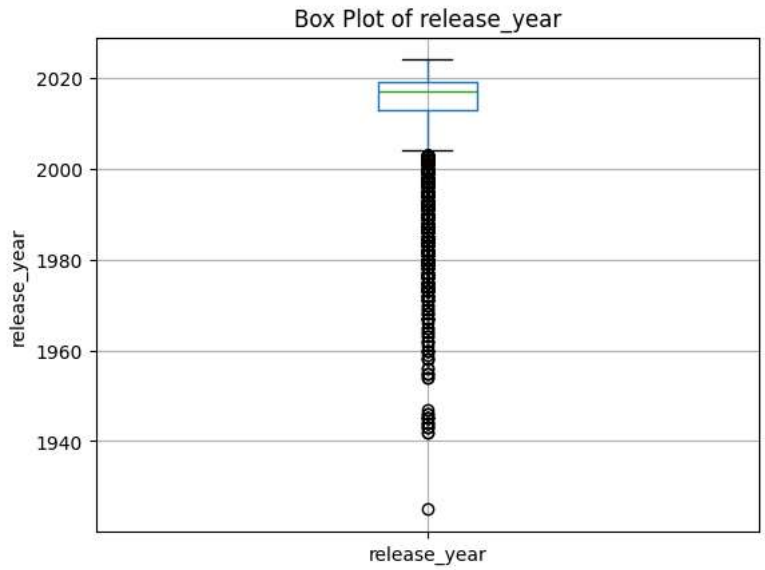
```
import pandas as pd
import matplotlib.pyplot as plt

# Assuming 'df' is your dataframe

# Specify the columns you want to create histograms for
selected_columns = ['release_year', 'rating', 'duration']

for col in selected_columns:
    plt.figure()  # Create a new figure for each histogram

    # Check if the column is numerical
    if pd.api.types.is_numeric_dtype(df[col]):
        df[col].hist()  # Generate histogram for numerical data
    else:
        df[col].value_counts().plot(kind='bar')  # Generate bar plot for categorical data

    plt.title(f'Distribution of {col}')  # Set the title
    plt.xlabel(col)  # Set the x-axis label
    plt.ylabel('Frequency')
```

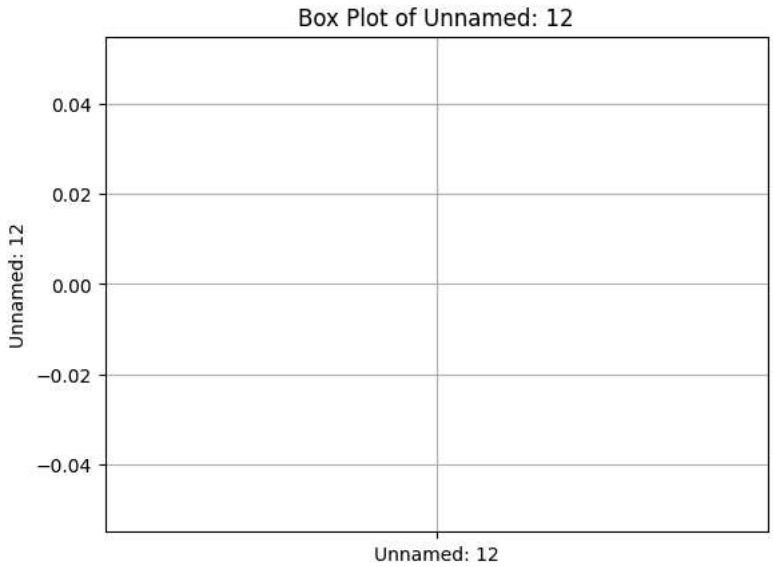

Distribution of release_year
Frequency
release_year
0
1000
2000
3000
4000
5000
6000
1940
1960
1980
2000
2020

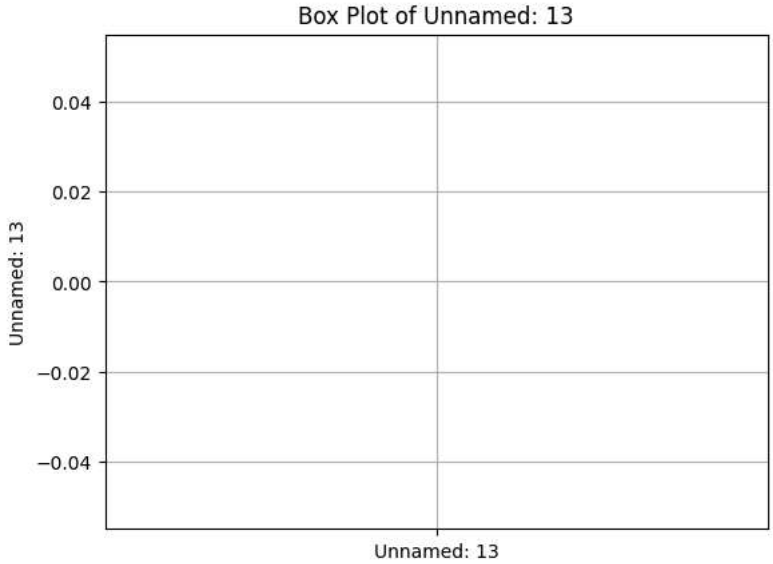

Distribution of rating
Frequency
rating
0
500
1000
1500
2000
2500
3000
TV-MA
TV-14
TV-PG
R
PG-13
TV-Y7
TV-Y
PG
TV-G
NR
G
TV-Y7-FV
NC-17
UR
74 min
84 min
66 min
A

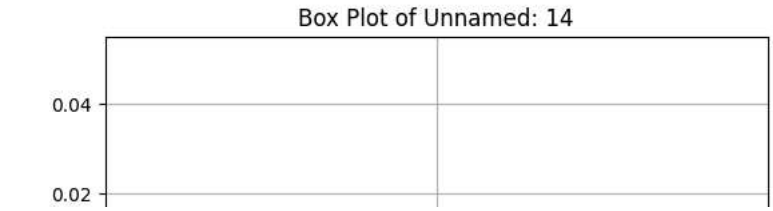

Distribution of duration
Frequency
duration
0
250
500
750
1000
1250
1500
1750

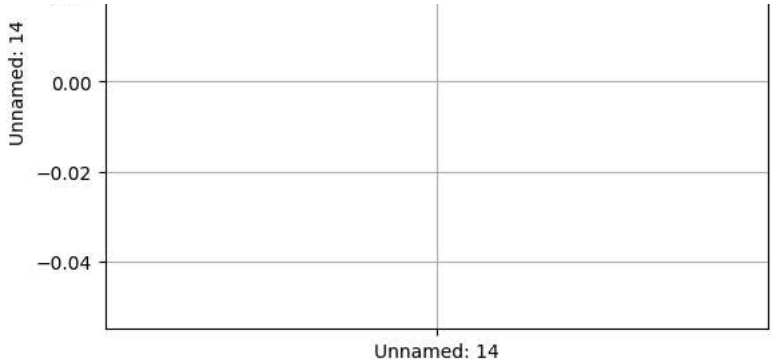
```
import pandas as pd
import matplotlib.pyplot as plt

# Assuming 'df' is your dataframe

# Iterate through all columns and create box plots
for col in df.columns:
    # Check if the column is numerical (box plots are typically for numerical data)
    if pd.api.types.is_numeric_dtype(df[col]):
        plt.figure()  # Create a new figure for each box plot
        df.boxplot(column=[col])  # Generate the box plot
        plt.title(f'Box Plot of {col}')  # Set the title
        plt.ylabel(col)  # Set the y-axis label
        plt.show()  # Display the box plot
```

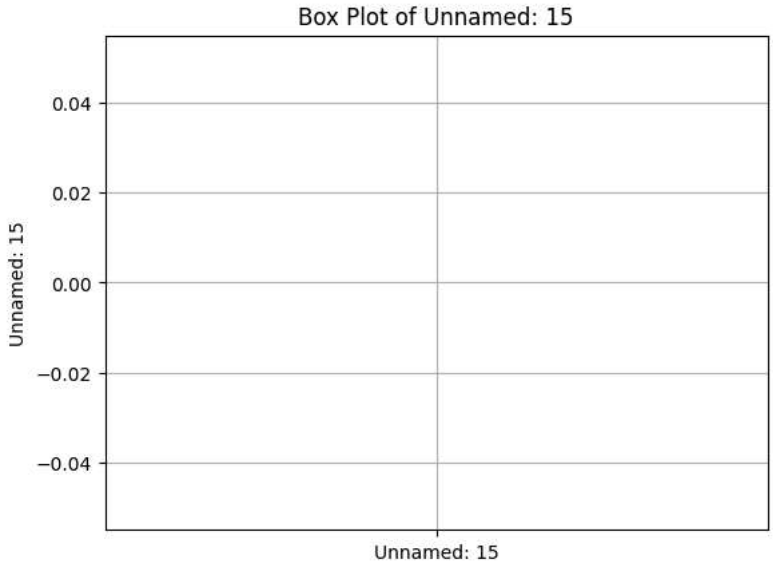

Box Plot of release_year
release_year
2020
2000
1980
1960
1940
release_year

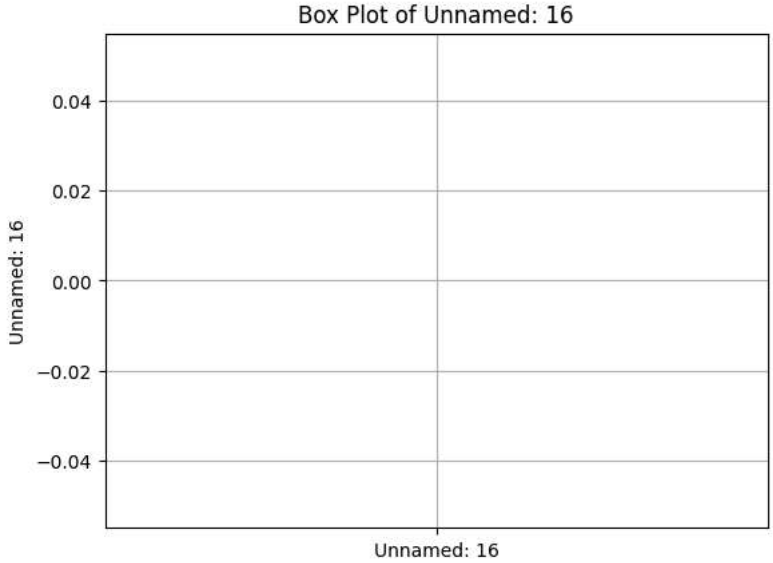

Box Plot of Unnamed: 12
Unnamed: 12
0.04
0.02
0.00
−0.02
−0.04
Unnamed: 12


Box Plot of Unnamed: 13
Unnamed: 13
0.04
0.02
0.00
−0.02
−0.04
Unnamed: 13


Box Plot of Unnamed: 14
0.04
0.02


Unnamed: 14
0.00
−0.02
−0.04
Unnamed: 14


Box Plot of Unnamed: 15
Unnamed: 15
0.04
0.02
0.00
−0.02
−0.04
Unnamed: 15


Box Plot of Unnamed: 16
Unnamed: 16
0.04
0.02
0.00
−0.02
−0.04
Unnamed: 16


Box Plot of Unnamed: 17
Unnamed: 17
0.04
0.02
0.00
−0.02