```
# Crime Statistics Data Preprocessing

# Import necessary libraries
import pandas as pd
import numpy as np
from sklearn.preprocessing import StandardScaler
from sklearn.impute import SimpleImputer
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import seaborn as sns

# Step 1: Load the dataset
# Here we load the crime statistics dataset and assign appropriate
column names.
url =
'https://archive.ics.uci.edu/ml/machine-learning-databases/communities
/communities.data'
column_names_url = 'https://archive.ics.uci.edu/ml/machine-learning-
databases/communities/communities.names'

# Load the dataset
data = pd.read_csv(url, header=None, na_values='?')

# Load the column names
column_names = [
    'state', 'county', 'community', 'communityname', 'fold',
    'population', 'householdsize', 'racepctblack', 'racePctWhite',
'racePctAsian',
    'racePctHisp', 'agePct12t21', 'agePct12t29', 'agePct16t24',
'agePct65up',
    'numbUrban', 'pctUrban', 'medIncome', 'pctWWage', 'pctWFarmSelf',
'pctWInvInc',
    'pctWSocSec', 'pctWPubAsst', 'pctWRetire', 'medFamInc',
'perCapInc', 'whitePerCap',
    'blackPerCap', 'indianPerCap', 'AsianPerCap', 'OtherPerCap',
'HispPerCap', 'NumUnderPov',
    'PctPopUnderPov', 'PctLess9thGrade', 'PctNotHSGrad',
'PctBSorMore', 'PctUnemployed',
    'PctEmploy', 'PctEmplManu', 'PctEmplProfServ', 'PctOccupManu',
'PctOccupMgmtProf',
    'MalePctDivorce', 'MalePctNevMarr', 'FemalePctDiv', 'TotalPctDiv',
'PersPerFam',
    'PctFam2Par', 'PctKids2Par', 'PctYoungKids2Par', 'PctTeen2Par',
'PctWorkMomYoungKids',
    'PctWorkMom', 'NumIlleg', 'PctIlleg', 'NumImmig',
'PctImmigRecent', 'PctImmigRec5',
    'PctImmigRec8', 'PctImmigRec10', 'PctRecentImmig', 'PctRecImmig5',
'PctRecImmig8',
    'PctRecImmig10', 'PctSpeakEnglOnly', 'PctNotSpeakEnglWell',
'PctLargHouseFam',
    'PctLargHouseOccup', 'PersPerOccupHous', 'PersPerOwnOccHous',
'PersPerRentOccHous',
    'PctPersOwnOccup', 'PctPersDenseHous', 'PctHousLess3BR',
'MedNumBR', 'HousVacant',
    'PctHousOccup', 'PctHousOwnOcc', 'PctVacantBoarded',
'PctVacMore6Mos', 'MedYrHousBuilt',
    'PctHousNoPhone', 'PctWOFullPlumb', 'OwnOccLowQuart',
'OwnOccMedVal', 'OwnOccHiQuart',
    'RentLowQ', 'RentMedian', 'RentHighQ', 'MedRent',
'MedRentPctHousInc', 'MedOwnCostPctInc',
    'MedOwnCostPctIncNoMtg', 'NumInShelters', 'NumStreet',
'PctForeignBorn', 'PctBornSameState',
    'PctSameHouse85', 'PctSameCity85', 'PctSameState85',
'LemasSwornFT', 'LemasSwFTPerPop',
    'LemasSwFTFieldOps', 'LemasSwFTFieldPerPop', 'LemasTotalReq',
'LemasTotReqPerPop',
    'PolicReqPerOffic', 'PolicPerPop', 'RacialMatchCommPol',
'PctPolicWhite', 'PctPolicBlack',
    'PctPolicHisp', 'PctPolicAsian', 'PctPolicMinor',
'OfficAssgnDrugUnits', 'NumKindsDrugsSeiz',
    'PolicAveOTWorked', 'LandArea', 'PopDens', 'PctUsePubTrans',
'PolicCars', 'PolicOperBudg',
    'LemasPctPolicOnPatr', 'LemasGangUnitDeploy',
'LemasPctOfficDrugUn', 'PolicBudgPerPop',
    'ViolentCrimesPerPop'
]

# Assign column names to the dataframe
data.columns = column_names

# Step 2: Initial Data Exploration (Done previously)
# Viewing basic information, first few rows, and summary statistics

# Basic Information
print("Basic Information:")
print(data.info())

print("\nFirst few rows of the dataset:")
print(data.head())

print("\nSummary Statistics for Numerical Features:")
print(data.describe())

print("\nSummary Statistics for Categorical Features:")
print(data.describe(include=[object]))
```

```
Basic Information:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1994 entries, 0 to 1993
Columns: 128 entries, state to ViolentCrimesPerPop
```

```
dtypes: float64(125), int64(2), object(1)
memory usage: 1.9+ MB
None

First few rows of the dataset:
   state  county  community        communityname  fold  population  \
0      8     NaN        NaN         Lakewoodcity     1        0.19
1     53     NaN        NaN          Tukwilacity     1        0.00
2     24     NaN        NaN         Aberdeentown     1        0.00
3     34     5.0    81440.0  Willingborotownship     1        0.04
4     42    95.0     6096.0    Bethlehemtownship     1        0.01

   householdsize  racepctblack  racePctWhite  racePctAsian  ...
LandArea  \
0           0.33          0.02          0.90          0.12  ...
0.12
1           0.16          0.12          0.74          0.45  ...
0.02
2           0.42          0.49          0.56          0.17  ...
0.01
3           0.77          1.00          0.08          0.12  ...
0.02
4           0.55          0.02          0.95          0.09  ...
0.04

   PopDens  PctUsePubTrans  PolicCars  PolicOperBudg
LemasPctPolicOnPatr  \
0     0.26            0.20       0.06           0.04
0.9
1     0.12            0.45        NaN            NaN
NaN
2     0.21            0.02        NaN            NaN
NaN
3     0.39            0.28        NaN            NaN
NaN
4     0.09            0.02        NaN            NaN
NaN

   LemasGangUnitDeploy  LemasPctOfficDrugUn  PolicBudgPerPop  \
0                  0.5                 0.32             0.14
1                  NaN                 0.00              NaN
2                  NaN                 0.00              NaN
3                  NaN                 0.00              NaN
4                  NaN                 0.00              NaN

   ViolentCrimesPerPop
0                 0.20
1                 0.67
2                 0.43
3                 0.12
```

```
4                 0.03

[5 rows x 128 columns]

Summary Statistics for Numerical Features:
             state      county     community         fold   population  
\
count  1994.000000  820.000000    817.000000  1994.000000  1994.000000

mean     28.683551   58.826829  46188.336597     5.493982     0.057593

std      16.397553  126.420560  25299.726569     2.873694     0.126906

min       1.000000    1.000000     70.000000     1.000000     0.000000

25%      12.000000    9.000000  25065.000000     3.000000     0.010000

50%      34.000000   23.000000  48090.000000     5.000000     0.020000

75%      42.000000   59.500000  66660.000000     8.000000     0.050000

max      56.000000  840.000000  94597.000000    10.000000     1.000000


       householdsize  racepctblack  racePctWhite  racePctAsian  
racePctHisp  \
count    1994.000000   1994.000000   1994.000000   1994.000000  
1994.000000
mean        0.463395      0.179629      0.753716      0.153681  
0.144022
std         0.163717      0.253442      0.244039      0.208877  
0.232492
min         0.000000      0.000000      0.000000      0.000000  
0.000000
25%         0.350000      0.020000      0.630000      0.040000  
0.010000
50%         0.440000      0.060000      0.850000      0.070000  
0.040000
75%         0.540000      0.230000      0.940000      0.170000  
0.160000
max         1.000000      1.000000      1.000000      1.000000  
1.000000

       ...     LandArea      PopDens  PctUsePubTrans   PolicCars  \
count  ...  1994.000000  1994.000000     1994.000000  319.000000  
mean   ...     0.065231     0.232854        0.161685    0.163103  
std    ...     0.109459     0.203092        0.229055    0.214778  
min    ...     0.000000     0.000000        0.000000    0.000000  
25%    ...     0.020000     0.100000        0.020000    0.040000  
50%    ...     0.040000     0.170000        0.070000    0.080000  
```

```
75%    ...     0.070000     0.280000        0.190000    0.195000
max    ...     1.000000     1.000000        1.000000    1.000000

       PolicOperBudg  LemasPctPolicOnPatr  LemasGangUnitDeploy  \
count     319.000000           319.000000           319.000000
mean        0.076708             0.698589             0.440439
std         0.140207             0.213944             0.405808
min         0.000000             0.000000             0.000000
25%         0.020000             0.620000             0.000000
50%         0.030000             0.750000             0.500000
75%         0.060000             0.840000             1.000000
max         1.000000             1.000000             1.000000

       LemasPctOfficDrugUn  PolicBudgPerPop  ViolentCrimesPerPop
count          1994.000000       319.000000          1994.000000
mean              0.094052         0.195078             0.237979
std               0.240328         0.164718             0.232985
min               0.000000         0.000000             0.000000
25%               0.000000         0.110000             0.070000
50%               0.000000         0.150000             0.150000
75%               0.000000         0.220000             0.330000
max               1.000000         1.000000             1.000000

[8 rows x 127 columns]

Summary Statistics for Categorical Features:
         communityname
count             1994
unique            1828
top     Greenvillecity
freq                 5

# Step 3: Handling Missing Values
# Separate numerical and categorical features
numerical_features =
data.select_dtypes(include=[np.number]).columns.tolist()
categorical_features =
data.select_dtypes(include=[object]).columns.tolist()

# Impute missing values for numerical features
imputer_num = SimpleImputer(strategy='mean')
data[numerical_features] =
imputer_num.fit_transform(data[numerical_features])

# Explanation:
# Missing values can cause errors in data analysis and machine
learning models.
# Imputing with the mean is a common strategy for numerical features
to maintain data consistency.
```

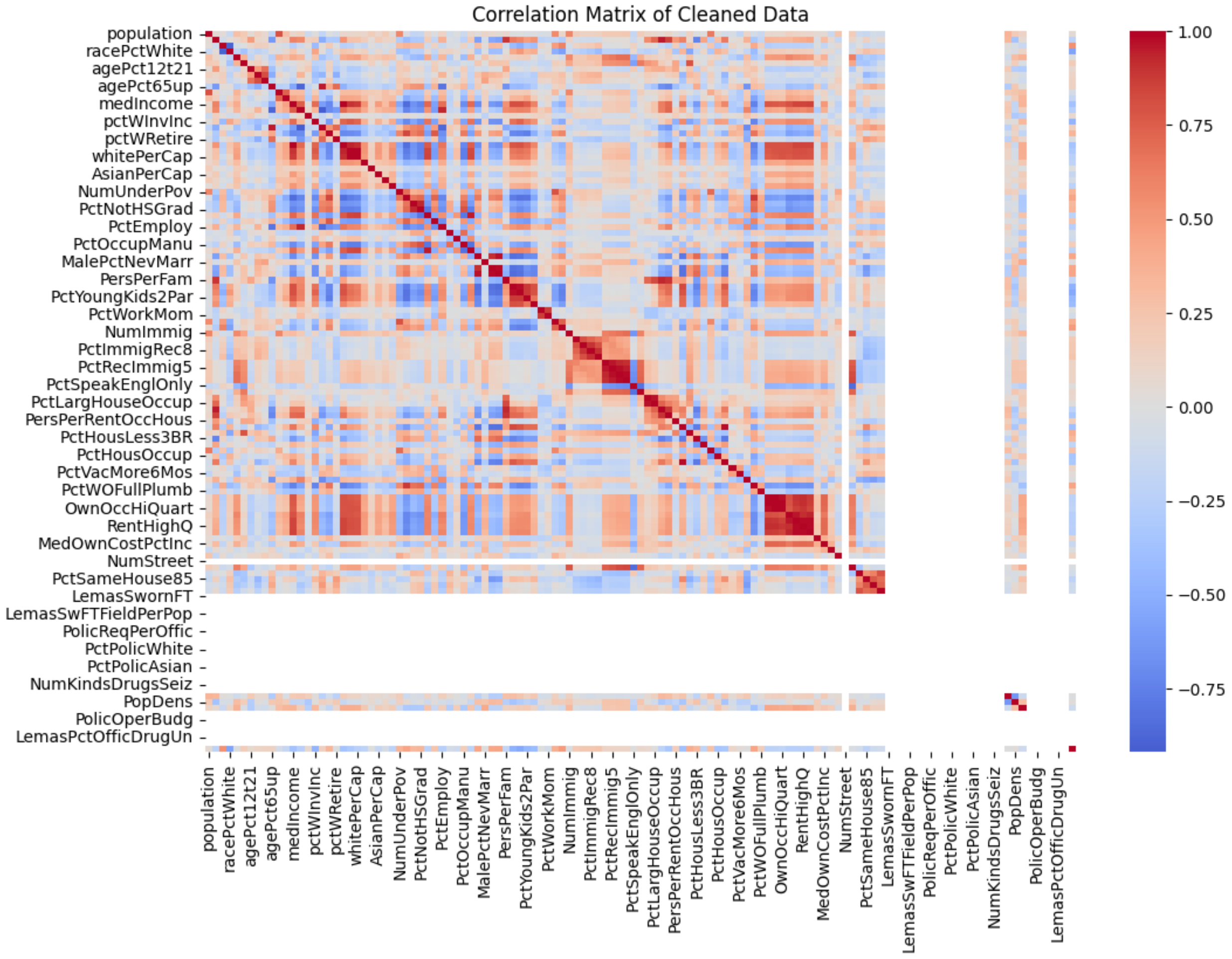
```python
# Step 4: Encoding Categorical Variables
# Drop non-essential categorical columns (communityname, state,
county, community, fold)
data = data.drop(columns=['communityname', 'state', 'county',
'community', 'fold'])

# Explanation:
# Categorical variables like 'communityname' are often non-essential
for analysis and can introduce noise.
# Dropping them helps in simplifying the dataset.

# Update numerical_features list after dropping columns
numerical_features = [col for col in numerical_features if col in
data.columns]

# Step 5: Outlier Removal
# Outlier removal using the IQR method
Q1 = data.quantile(0.25)
Q3 = data.quantile(0.75)
IQR = Q3 - Q1
lower_bound = Q1 - 1.5 * IQR
upper_bound = Q3 + 1.5 * IQR
data = data[~((data < lower_bound) | (data >
upper_bound)).any(axis=1)]

# Explanation:
# Outliers can skew the results of data analysis and modeling.
# The IQR method is used to identify and remove outliers, ensuring a
more robust analysis.

# Step 6: Normalizing Numerical Features
scaler = StandardScaler()
data[numerical_features] =
scaler.fit_transform(data[numerical_features])

# Explanation:
# Normalizing numerical features ensures that all features contribute
equally to the analysis and models,
# preventing features with larger scales from dominating.

# Step 7: Splitting the Dataset into Train and Test Sets
X = data.drop(columns=['ViolentCrimesPerPop'])
y = data['ViolentCrimesPerPop']
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)

# Explanation:
# Splitting the data into training and testing sets allows for the
evaluation of model performance on unseen data,
# helping to prevent overfitting.
# Display basic information about the processed data
print("Basic Information after Preprocessing:")
print(data.info())

print("\nFirst few rows of the processed dataset:")
print(data.head())

print("\nSummary Statistics of the processed dataset:")
print(data.describe())
```

```
Basic Information after Preprocessing:
<class 'pandas.core.frame.DataFrame'>
Index: 454 entries, 2 to 1989
Columns: 123 entries, population to ViolentCrimesPerPop
dtypes: float64(123)
memory usage: 439.8 KB
None

First few rows of the processed dataset:
    population  householdsize  racepctblack  racePctWhite
racePctAsian  \
2    -0.969413      -0.073191      3.671661     -3.497654
1.441620
8     0.543005      -0.828384      1.065110     -0.603358     -
0.803391
13   -0.465274       0.115608     -0.732511      0.843790     -
0.504056
19   -0.969413      -0.167590     -0.283106      0.637055     -
0.953058
27   -0.465274      -0.545187     -0.642630      0.947158     -
0.803391

    racePctHisp  agePct12t21  agePct12t29  agePct16t24
agePct65up  ...  \
2     -0.134399     0.130566     0.290507    -0.014925   -
0.699234  ...
8     -0.721101     0.001983    -0.005875    -0.014925
0.306755  ...
13    -0.574425    -0.383769     1.179656     0.630414   -
1.265103  ...
19    -0.574425    -0.126601    -1.191407    -0.660265
0.746875  ...
27    -0.574425    -0.383769    -1.043216    -0.821600
0.998372  ...

    LandArea    PopDens  PctUsePubTrans     PolicCars  PolicOperBudg  \
2  -1.092247  0.449501       -0.640574  2.775558e-17   1.387779e-17
8  -0.093160  0.058743       -0.416002  2.775558e-17   1.387779e-17
13  1.238956 -1.211222       -0.865147  2.775558e-17   1.387779e-17
19  1.571984 -1.308912       -0.303716  2.775558e-17   1.387779e-17
```

```
27 -0.093160 -0.722774       -0.865147  2.775558e-17   1.387779e-17

    LemasPctPolicOnPatr  LemasGangUnitDeploy  LemasPctOfficDrugUn  \
2                   0.0         5.551115e-17                  0.0
8                   0.0         5.551115e-17                  0.0
13                  0.0         5.551115e-17                  0.0
19                  0.0         5.551115e-17                  0.0
27                  0.0         5.551115e-17                  0.0

    PolicBudgPerPop  ViolentCrimesPerPop
2     -5.551115e-17             2.559078
8     -5.551115e-17             3.414679
13    -5.551115e-17            -0.349967
19    -5.551115e-17            -0.863328
27    -5.551115e-17            -0.264407

[5 rows x 123 columns]

Summary Statistics of the processed dataset:
         population  householdsize  racepctblack  racePctWhite
racePctAsian  \
count  4.540000e+02   4.540000e+02  4.540000e+02  4.540000e+02
4.540000e+02
mean  -3.130144e-17   1.584636e-16  4.890850e-17 -4.450674e-16 -
7.042824e-17
std    1.001103e+00   1.001103e+00  1.001103e+00  1.001103e+00
1.001103e+00
min   -9.694131e-01  -2.810767e+00 -7.325109e-01 -4.014492e+00 -
1.102725e+00
25%   -9.694131e-01  -7.339849e-01 -6.426298e-01 -4.999900e-01 -
6.537233e-01
50%   -4.652739e-01  -7.319056e-02 -4.628677e-01  3.269516e-01 -
2.795549e-01
75%    5.430046e-01   5.876038e-01  1.662998e-01  7.404224e-01
2.442809e-01
max    4.071979e+00   3.136382e+00  3.761542e+00  1.050526e+00
4.135632e+00

        racePctHisp   agePct12t21   agePct12t29   agePct16t24
agePct65up  \
count  4.540000e+02  4.540000e+02  4.540000e+02  4.540000e+02
4.540000e+02
mean   1.565072e-17  3.932244e-16 -9.312179e-16 -6.260288e-17 -
2.425862e-16
std    1.001103e+00  1.001103e+00  1.001103e+00  1.001103e+00
1.001103e+00
min   -7.211009e-01 -2.826862e+00 -2.821513e+00 -2.434949e+00 -
2.459714e+00
25%   -5.744253e-01 -6.409367e-01 -5.986413e-01 -6.602649e-01 -
7.621085e-01
```

```
50%   -4.277498e-01  1.982570e-03 -5.875432e-03 -1.762602e-01
1.181317e-01
75%    1.227681e-02  6.449018e-01  5.868904e-01  4.690795e-01
7.468746e-01
max    4.852570e+00  3.345163e+00  4.143485e+00  3.857113e+00
2.695978e+00

       ...      LandArea       PopDens  PctUsePubTrans
PolicCars  \
count  ...  4.540000e+02  4.540000e+02    4.540000e+02  4.540000e+02

mean   ... -3.130144e-17 -1.408565e-16   -7.042824e-17  2.775558e-17

std    ...  1.001103e+00  1.001103e+00    1.001103e+00  0.000000e+00

min    ... -1.425276e+00 -1.406601e+00   -8.651465e-01  2.775558e-17

25%    ... -7.592179e-01 -7.227740e-01   -6.405745e-01  2.775558e-17

50%    ... -2.596745e-01 -2.343260e-01   -3.598594e-01  2.775558e-17

75%    ...  5.728977e-01  6.448806e-01    3.699998e-01  2.775558e-17

max    ...  3.237129e+00  3.380190e+00    4.075439e+00  2.775558e-17


       PolicOperBudg  LemasPctPolicOnPatr  LemasGangUnitDeploy  \
count   4.540000e+02                454.0         4.540000e+02
mean    1.387779e-17                  0.0         5.551115e-17
std     0.000000e+00                  0.0         0.000000e+00
min     1.387779e-17                  0.0         5.551115e-17
25%     1.387779e-17                  0.0         5.551115e-17
50%     1.387779e-17                  0.0         5.551115e-17
75%     1.387779e-17                  0.0         5.551115e-17
max     1.387779e-17                  0.0         5.551115e-17

       LemasPctOfficDrugUn  PolicBudgPerPop  ViolentCrimesPerPop
count                454.0     4.540000e+02         4.540000e+02
mean                   0.0    -5.551115e-17        -1.036860e-16
std                    0.0     0.000000e+00         1.001103e+00
min                    0.0    -5.551115e-17        -1.120009e+00
25%                    0.0    -5.551115e-17        -6.922080e-01
50%                    0.0    -5.551115e-17        -3.499674e-01
75%                    0.0    -5.551115e-17         3.345138e-01
max                    0.0    -5.551115e-17         4.783641e+00

[8 rows x 123 columns]

# Save the cleaned dataset for further analysis
data.to_csv('cleaned_communities_crime_data.csv', index=False)
```

```python
# Visualize the cleaned data
plt.figure(figsize=(12, 8))
sns.heatmap(data.corr(), cmap='coolwarm', center=0)
plt.title('Correlation Matrix of Cleaned Data')
plt.show()

print("Data preprocessing and cleaning complete.")
```



```
Data preprocessing and cleaning complete.
```