EDA

December 17, 2025

[1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import os
```

[2]:
```
os.makedirs('figures', exist_ok = True)
sns.set_style("whitegrid")
plt.rcParams['figure.figsize'] = (12,6)
```

[3]:
```
#Unique Drugs
clean_overdose = pd.read_csv('../clean_overdose_final.csv')

if 'drug_type' in clean_overdose.columns:
    drug_types = clean_overdose['drug_type'].unique()
    print(f"Drug types in dataset: {len(drug_types)}")
    for dt in drug_types:
        print(f"  - {dt}")
```

[4]:
```
#Summary Statistics of overdose deaths per drug

#Me
for drug in drug_types:
    drug_data = clean_overdose[clean_overdose['drug_type'] == drug]
    print(f"\n{drug}:")
    print(f" Mean Rate: {drug_data['ESTIMATE'].mean():.2f}")
#median rate
    print(f" Median Rate: {drug_data['ESTIMATE'].median():.2f}")
#Standard Deviation
    print(f" Standard Deviation: {drug_data['ESTIMATE'].std():.2f}")
#Minimum
    print(f" Minimum: {drug_data['ESTIMATE'].min():.2f}")
#Maximum
    print(f" Maximum {drug_data['ESTIMATE'].max():.2f}")
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[4], line 4
```

```
      1 #Summary Statistics of overdose deaths per drug
      2
      3 #Me
----> 4 for drug in drug_types:
      5     drug_data = clean_overdose[clean_overdose['drug_type'] == drug]
      6     print(f"\n{drug}:")

NameError: name 'drug_types' is not defined
```

[ ]:
```
#Line Plot of Overall Trends

clean_overdose = pd.read_csv('../clean_overdose_final.csv')

trend_stats = clean_overdose[
    (clean_overdose['sex'] == 'All') &
    (clean_overdose['race_ethnicity'] == 'All')
    ].groupby('YEAR')['ESTIMATE'].mean().reset_index()

plt.figure(figsize=(14,7))
plt.plot(trend_stats['YEAR'], trend_stats['ESTIMATE'],
         marker='o', linewidth=3, markersize=8, color = 'darkblue')
plt.title('Drug Overdose Death Rates')
plt.xlabel('Year', fontsize=12, fontweight='bold')
plt.ylabel('Age-Adjusted Death Rate (per 100,000)', fontsize=12,␣
 ↪fontweight='bold')
plt.grid(True, alpha=0.3)
plt.tight_layout()
#plt.savefig('figures/overall_trend.png', dpi=300, bbox_inches='tight')
#print(" Saved: figures/overall_trend.png")
#plt.close()
```

[ ]:
```
#Trend by Drug Type

clean_overdose = pd.read_csv('../clean_overdose_final.csv')

if 'drug_type' in clean_overdose.columns:
    plt.figure(figsize=(14,8))

all_drugs = clean_overdose['drug_type'].unique()

for drug in all_drugs:
    drug_trend = clean_overdose[
        (clean_overdose['drug_type'] == drug) &
        (clean_overdose['sex'] == 'All') &
        (clean_overdose['race_ethnicity'] == 'All')
```

```
        ].groupby('YEAR')['ESTIMATE'].mean().reset_index()

    if len(drug_trend) > 0:
        plt.plot(drug_trend['YEAR'], drug_trend['ESTIMATE'],
                 marker='o', linewidth=2, label=drug, markersize=4)

plt.title('Drug Overdose Death Rates by Drug Type',
          fontsize=16, fontweight='bold', pad=20)
plt.xlabel('Year', fontsize=12, fontweight='bold')
plt.ylabel('Age-Adjusted Death Rate (per 100,000)', fontsize=12,␣
 ↪fontweight='bold')
plt.legend(bbox_to_anchor=(1.05, 1), loc='upper left', fontsize=9)
plt.grid(True, alpha=0.3)
plt.tight_layout()

#plt.savefig('figures/trend_by_drug_type.png', dpi=300, bbox_inches='tight')
#print(" Saved: figures/trend_by_drug_type.png")
#plt.close()
```

[ ]:
```
#Trend by Sex

sex_trend = clean_overdose[
    (clean_overdose['sex'].isin(['Male', 'Female'])) &
    (clean_overdose['race_ethnicity'] == 'All')
].groupby(['YEAR', 'sex'])['ESTIMATE'].mean().reset_index()

plt.figure(figsize=(14, 7))
for sex in ['Male', 'Female']:
    data = sex_trend[sex_trend['sex'] == sex]
    color = 'darkblue' if sex == 'Male' else 'yellow'
    plt.plot(data['YEAR'], data['ESTIMATE'],
             marker='o', linewidth=2.5, label=sex, markersize=6, color=color)

plt.title('Drug Overdose Death Rates by Sex',
          fontsize=16, fontweight='bold', pad=20)
plt.xlabel('Year', fontsize=12, fontweight='bold')
plt.ylabel('Age-Adjusted Death Rate (per 100,000)', fontsize=12,␣
 ↪fontweight='bold')
plt.legend(fontsize=12)
plt.grid(True, alpha=0.3)
plt.tight_layout()
#plt.savefig('figures/trend_by_sex.png', dpi=300, bbox_inches='tight')
#print(" Saved: figures/trend_by_sex.png")
#plt.close()
```

[ ]:
```
#Bar Charts Comparing Demographic Groups

most_recent_year = clean_overdose['YEAR'].max()
recent_data = clean_overdose[clean_overdose['YEAR'] == most_recent_year]

#Sex Comparison
sex_comparison = recent_data[
    (recent_data['sex'].isin(['Male', 'Female'])) &
    (recent_data['race_ethnicity'] == 'All')
].groupby('sex')['ESTIMATE'].mean().sort_values(ascending=False)

plt.figure(figsize=(10, 6))
bars = plt.bar(sex_comparison.index, sex_comparison.values,
               color=['darkblue', 'yellow'], width=0.6, edgecolor='black',␣
 ↪linewidth=1.5)
plt.title(f'Drug Overdose Death Rates by Sex ({most_recent_year})',
          fontsize=16, fontweight='bold', pad=20)
plt.ylabel('Age-Adjusted Death Rate (per 100,000)', fontsize=12,␣
 ↪fontweight='bold')
plt.xlabel('Sex', fontsize=12, fontweight='bold')

for bar in bars:
    height = bar.get_height()
    plt.text(bar.get_x() + bar.get_width()/2., height,
            f'{height:.1f}', ha='center', va='bottom', fontsize=12,␣
 ↪fontweight='bold')

plt.tight_layout()
#plt.savefig('figures/comparison_by_sex.png', dpi=300, bbox_inches='tight')
#print(" Saved: figures/comparison_by_sex.png")
#plt.close()
```

[ ]:
```
# Bar chart by race/ethnicity
race_comp = recent_data[
    (recent_data['race_ethnicity'] != 'All')
].groupby('race_ethnicity')['ESTIMATE'].mean().sort_values(ascending=False)

if len(race_comp) > 0:
    plt.figure(figsize=(14, 7))
    n_colors = len(race_comp)
    colors = sns.color_palette("YlGnBu_r", n_colors)
    bars = plt.bar(range(len(race_comp)), race_comp.values,
                   color=colors, edgecolor='black', linewidth=1.5)
    plt.xticks(range(len(race_comp)), race_comp.index,
               rotation=45, ha='right', fontsize=10)
    plt.title(f'Drug OD Death Rates by Race/Ethnicity ({most_recent_year})',
              fontsize=16, fontweight='bold', pad=20)
```

```
    plt.ylabel('Age-Adjusted Death Rate (per 100,000)',
               fontsize=12, fontweight='bold')
    for i, bar in enumerate(bars):
        height = bar.get_height()
        plt.text(bar.get_x() + bar.get_width()/2., height,
                 f'{height:.1f}', ha='center', va='bottom', fontsize=10,
                 fontweight='bold')

    plt.tight_layout()
    #plt.savefig('figures/comparison_by_race.png', dpi=300, bbox_inches='tight')
    #print(" Saved: figures/comparison_by_race.png")
    #plt.close()
```

[ ]:
```
summ_table = recent_data.groupby(['sex', 'race_ethnicity'])['ESTIMATE'].agg([
    ('Mean Rate', 'mean'),
    ('Count', 'count')
]).round(2).sort_values('Mean Rate', ascending=False)

print(summ_table.head(20))

summ_table.to_csv('figures/summary_statistics.csv')
print("\n Saved: figures/summary_statistics.csv")
```

[ ]:
```

```