Code analysis  
  
Code analysis  
  
import pandas as pd

import matplotlib.pyplot as plt # Ensure plt is defined for plotting

import scipy.stats as stats

import seaborn as sns

crime\_data = pd.read\_csv('Crime\_Data\_from\_2020\_to\_Present (2).csv')

count\_before\_cleaning = crime\_data.shape[0]

print(f'Record count before cleaning: {count\_before\_cleaning}')

print(crime\_data.columns)

crime\_data\_clean = crime\_data.dropna(subset=['Vict Age', 'Vict Sex', 'DATE OCC', 'Date Rptd'])

crime\_data\_clean = crime\_data\_clean.drop\_duplicates()

crime\_data\_clean = crime\_data\_clean[crime\_data\_clean['Vict Age'] <= 120]

count\_after\_cleaning = crime\_data\_clean.shape[0]

print(f'Record count after cleaning: {count\_after\_cleaning}')  
  
  
crime\_data.describe()  
  
crime\_data.info()  
  
crime\_data = crime\_data.set\_index('DR\_NO')  
  
crime\_data.isnull().sum()

print("Data Before Cleaning:")

print(crime\_data.info()) # Show the structure and missing values before cleaning

crime\_data\_clean = crime\_data.copy()

crime\_data\_clean['Vict Age'] = crime\_data\_clean['Vict Age'].fillna(crime\_data\_clean['Vict Age'].mean())

crime\_data\_clean['Vict Sex'] = crime\_data\_clean['Vict Sex'].fillna(crime\_data\_clean['Vict Sex'].mode()[0])

crime\_data\_clean = crime\_data\_clean.drop\_duplicates()

crime\_data\_clean = crime\_data\_clean[crime\_data\_clean['Vict Age'] <= 120]

crime\_data\_clean['DATE OCC'] = pd.to\_datetime(crime\_data\_clean['DATE OCC'], errors='coerce')

crime\_data\_clean['Date Rptd'] = pd.to\_datetime(crime\_data\_clean['Date Rptd'], errors='coerce')

print("\nData After Cleaning:")

print(crime\_data\_clean.info()) # Show the structure and missing values after cleaning

print(f"\nRecord count before cleaning: {crime\_data.shape[0]}")

print(f"Record count after cleaning: {crime\_data\_clean.shape[0]}")  
**Variation in Weapon Use Across Crime Types and Geographical Areas – Descriptive Analysis**

weapon\_category\_mapping = {

'PISTOL': 'Firearm',

'REVOLVER': 'Firearm',

'SHOTGUN': 'Firearm',

'RIFLE': 'Firearm',

'HAND GUN': 'Firearm',

'AIR PISTOL': 'Firearm',

'ASSAULT WEAPON': 'Firearm',

'BAT': 'Blunt Instrument',

'CLUB': 'Blunt Instrument',

'HAMMER': 'Blunt Instrument',

'BLUNT INSTRUMENT': 'Blunt Instrument',

'KNIFE': 'Sharp Object',

'AXE': 'Sharp Object',

'MACHETE': 'Sharp Object',

'RAZOR': 'Sharp Object',

'BOMB': 'Explosive',

'DYNAMITE': 'Explosive',

'EXPLOSIVE DEVICE': 'Explosive',

'CAUSTIC CHEMICAL': 'Chemical/Poison',

'POISON': 'Chemical/Poison',

'TASER': 'Other Weapon',

'STUN GUN': 'Other Weapon',

'ROPE': 'Other Weapon',

'VERBAL THREAT': 'Verbal Threat',

'UNKNOWN FIREARM': 'Unknown',

'UNKNOWN WEAPON': 'Unknown',

}

crime\_data\_clean['Weapon Category'] = crime\_data\_clean['Weapon Desc'].map(weapon\_category\_mapping)

weapon\_usage\_by\_category = crime\_data\_clean.groupby(['AREA NAME', 'Weapon Category']).size().unstack(fill\_value=0)

weapon\_usage\_by\_category.plot(kind='bar', stacked=True, figsize=(12, 7), colormap='plasma')

plt.title('Weapon Use Across Different Geographical Areas (Based by Weapon Category)')

plt.xlabel('Geographical Area')

plt.ylabel('Number of Incidents')

plt.xticks(rotation=45)

plt.legend(title='Weapon Category', bbox\_to\_anchor=(1.05, 1), loc='upper left')

**Relationship Between Crime Type and Reporting Time – Comparative Analysis**

print("Columns in crime\_data\_clean:", crime\_data\_clean.columns)

if 'Reporting\_Delay\_Days' in crime\_data\_clean.columns:

print("Column 'Reporting\_Delay\_Days' exists. Data type:", crime\_data\_clean['Reporting\_Delay\_Days'].dtype)

print("Sample data:\n", crime\_data\_clean['Reporting\_Delay\_Days'].head())

else:

print("Column 'Reporting\_Delay\_Days' does not exist. Recalculating...")

crime\_data\_clean['DATE OCC'] = pd.to\_datetime(crime\_data\_clean['DATE OCC'], errors='coerce')

crime\_data\_clean['Date Rptd'] = pd.to\_datetime(crime\_data\_clean['Date Rptd'], errors='coerce')

crime\_data\_clean['Reporting\_Delay\_Days'] = (crime\_data\_clean['Date Rptd'] - crime\_data\_clean['DATE OCC']).dt.days

print("Recalculated 'Reporting\_Delay\_Days'. Data type:", crime\_data\_clean['Reporting\_Delay\_Days'].dtype)

if 'Reporting\_Delay\_Days' in crime\_data\_clean.columns:

import matplotlib.pyplot as plt

import seaborn as sns

plt.figure(figsize=(12, 8))

sns.boxplot(x='Reporting\_Delay\_Days', y='Crime\_Category', data=crime\_data\_clean, orient='h', palette='coolwarm')

plt.title('Distribution of Reporting Delays by Crime Category')

plt.xlabel('Reporting Delay (Days)')

plt.ylabel('Crime Category')

plt.show()

top\_10\_crime\_types = crime\_data\_clean.groupby('Crm Cd Desc')['Reporting\_Delay\_Days'].mean().nlargest(10).index

crime\_data\_top\_10 = crime\_data\_clean[crime\_data\_clean['Crm Cd Desc'].isin(top\_10\_crime\_types)]

plt.figure(figsize=(12, 8))

sns.boxplot(x='Reporting\_Delay\_Days', y='Crm Cd Desc', data=crime\_data\_top\_10, orient='h', palette='coolwarm')

plt.title('Reporting Delay for Top 10 Crime Types')

plt.xlabel('Reporting Delay (Days)')

crime\_data\_clean['DATE OCC'] = pd.to\_datetime(crime\_data\_clean['DATE OCC'], errors='coerce')

crime\_data\_clean['Date Rptd'] = pd.to\_datetime(crime\_data\_clean['Date Rptd'], errors='coerce')

crime\_data\_clean['Reporting\_Delay\_Days'] = (crime\_data\_clean['Date Rptd'] - crime\_data\_clean['DATE OCC']).dt.days

crime\_data\_clean = crime\_data\_clean.dropna(subset=['Reporting\_Delay\_Days', 'Crm Cd Desc'])

grouped\_data = [group['Reporting\_Delay\_Days'].values for name, group in crime\_data\_clean.groupby('Crm Cd Desc')]

f\_stat, p\_value = stats.f\_oneway(\*grouped\_data)

# Print the ANOVA test result

print(f"One-Way ANOVA F-statistic: {f\_stat}")

print(f"One-Way ANOVA p-value: {p\_value}")

# Interpretation of results

if p\_value < 0.05:

print("The mean reporting delay differs significantly across crime types (p < 0.05).")

else:

print("No significant difference in mean reporting delay across crime types (p >= 0.05).")