COCONUT Team

Lufei Chen, Hai Guo, Jen Peng, Haolin Liu

Introduction

Drug overdose is not just a statistic—it's a devastating reality that plagues communities across the United States. Behind each number lies a story of loss, of shattered dreams, and of futures cut tragically short. Central to our exploration are three key questions:

- What are the trends in drug overdose death rates over time? Highlight any significant changes or patterns in the death rates over the years.

- What formula would you use for a metric to rank and label the concern the government should have for different drug overdose types? What would the resulting ranking be when you use this formula?

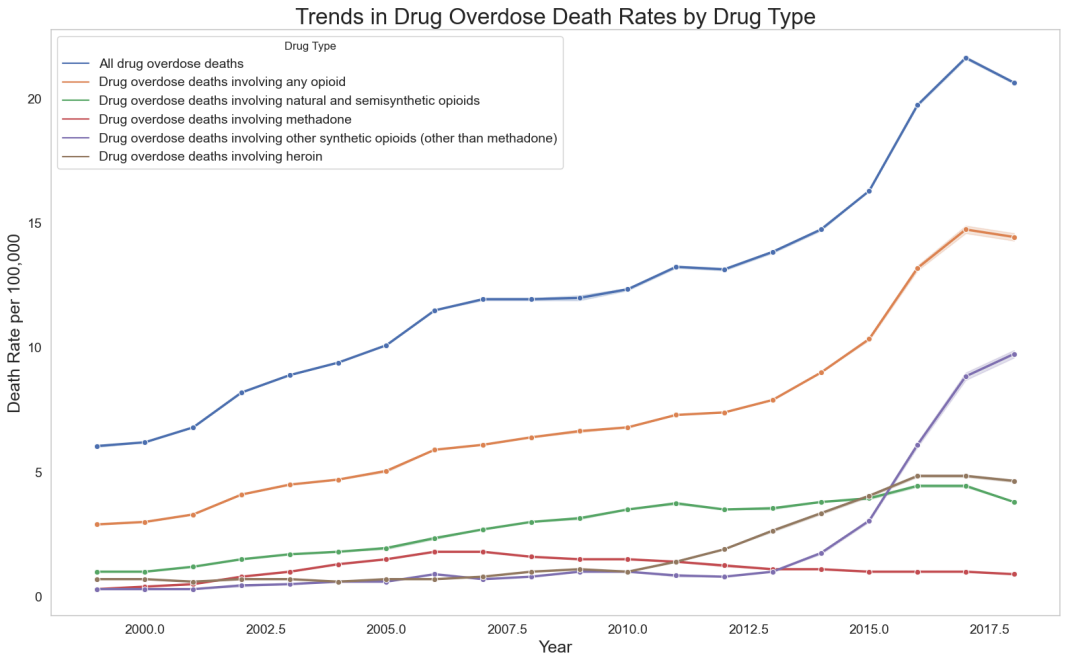
- Develop a predictive model to forecast future drug overdose death rates for each demographic group.

Given that we are dealing with real-life tragedies, we have a responsibility to protect the privacy of victims and those affected. We used a given dataset for data analysis and machine learning to address these questions.

Question 1- What are the trends in drug overdose death rates over time? Highlight any significant changes or patterns in the death rates over the years.

Drug type

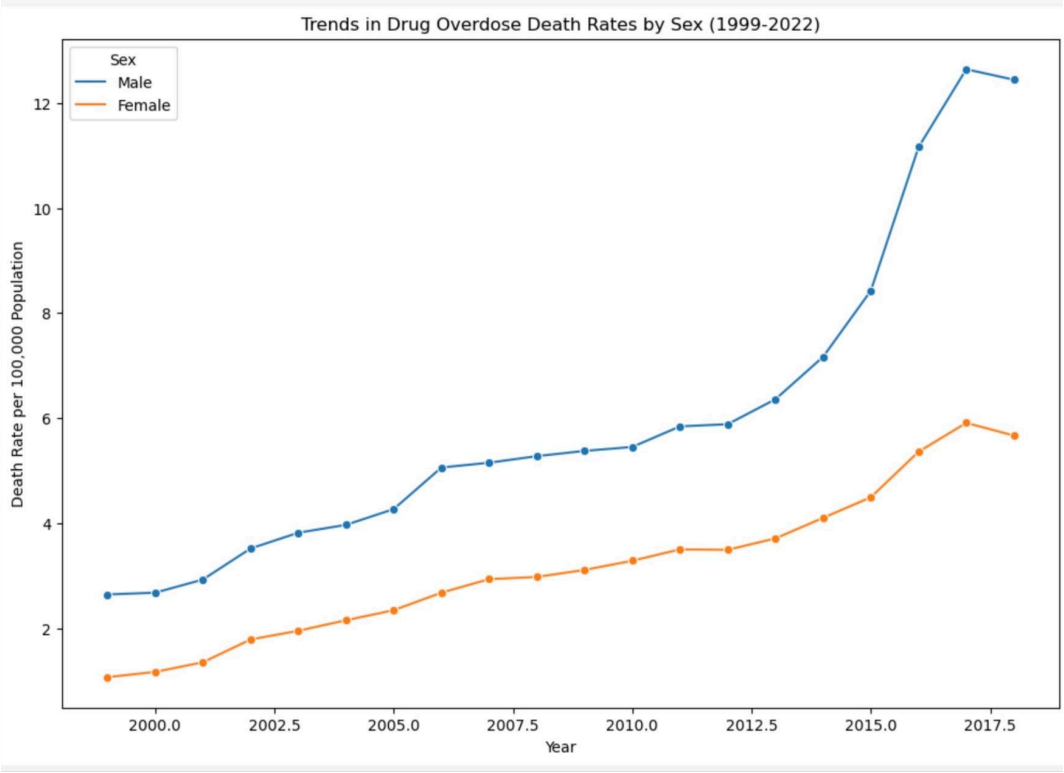
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
drug =
pd.read_csv('Drug_overdose_death_rates__by_drug_type__sex__age__race__and_Hispanic_ori
gin__United_States_20240518.csv')
drug.head()
drug_types = drug[drug['STUB_NAME'] == 'Total']
plt.figure(figsize=(16, 10))
sns.lineplot(data=drug_types, x='YEAR', y='ESTIMATE', hue='PANEL', marker='o',
linewidth=2.5)
plt.title('Trends in Drug Overdose Death Rates by Drug Type', fontsize=24)
plt.xlabel('Year', fontsize=18)
plt.ylabel('Death Rate per 100,000', fontsize=18)
plt.legend(title='Drug Type', loc='upper left', fontsize=14)
plt.grid(False)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)
plt.tight_layout()
plt.show()
```



The line plot illustrates the trends in drug overdose death rates from 1999 to 2018, broken down by various drug types. The overall trend shows an alarming increase in drug overdose deaths, particularly from opioids. First, the death rate for all drug overdose deaths has steadily risen, peaking around 2017, before showing a slight decline in 2018. Drug overdose deaths involving any opioid have shown a marked increase, reflecting the opioid crisis. This category also peaks around 2017. Also, Overdose deaths involving synthetic opioids, other than methadone, have sharply increased since 2013, with a notable spike in recent years. For Heroin, the death rates involving it have increased significantly since 2010, peaking around 2016. These trends underscore the need for targeted interventions focusing on the most affected drug types, especially synthetic opioids and heroin.

Sex:

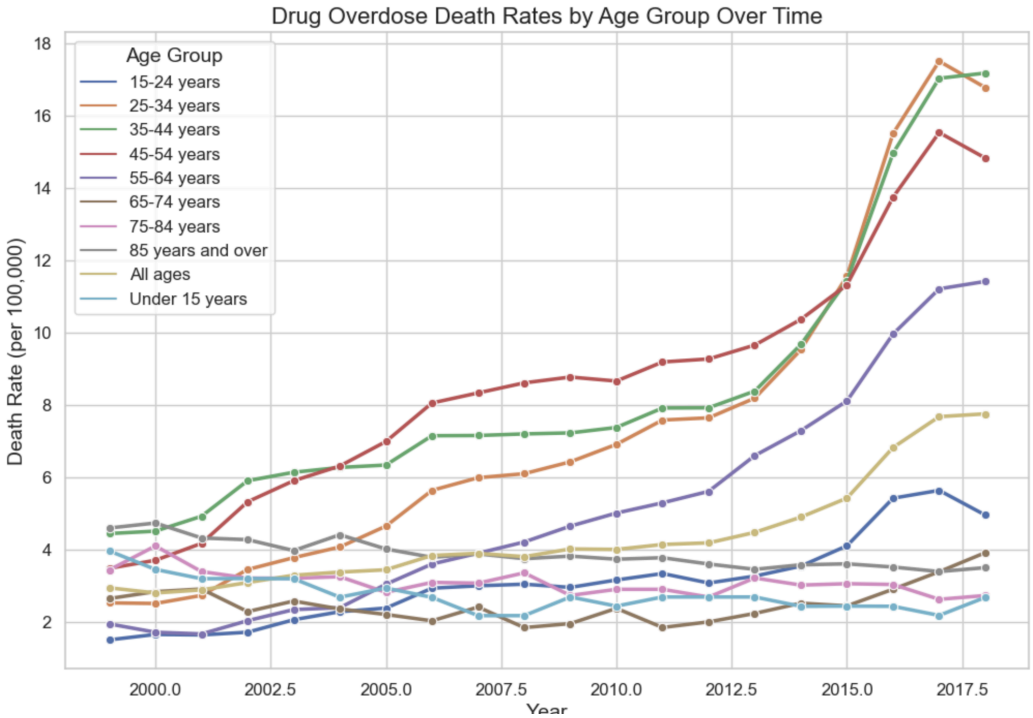
```
sex_filtered = drug[(drug['STUB_NAME'] == 'Sex') & (drug['STUB_LABEL'].isin(['Male',
'Female']))]
pivot_sex = sex_filtered.pivot_table(index='YEAR', columns='STUB_LABEL',
values='ESTIMATE').reset_index()
melted_data = pivot_sex.melt(id_vars='YEAR', value_vars=['Male', 'Female'],
var_name='Sex', value_name='Death Rate')

plt.figure(figsize=(12, 8))
sns.lineplot(data=melted_data, x='YEAR', y='Death Rate', hue='Sex', marker='o')
plt.title('Trends in Drug Overdose Death Rates by Sex (1999-2022)')
plt.xlabel('Year')
plt.ylabel('Death Rate per 100,000 Population')
plt.legend(title='Sex')
plt.show()
```



In this section we present the changes in drug overdose death rates per 100,000 population for men and women from 1999 to 2022. Both sexes exhibit a steady growth, with a more noticeable increase beginning from 2010. Throughout the time, males have far greater death rates than girls, and after 2010 the difference widened noticeably. The sharp rise in male death rates starts from 2013 and peaks at about 12 deaths per 100,000 by 2017, whereas female death rates peak at about 6 deaths per 100,000. The general increasing tendency for both sexes is unaffected by the little decline in male rates following their peak, which emphasizes the urgent need for focused efforts to deal with this growing public health issue.

Age

```
age_trend = drug.groupby(['YEAR', 'AGE'])['ESTIMATE'].mean().reset_index()
plt.figure(figsize=(12, 8))
sns.lineplot(data=age_trend, x='YEAR', y='ESTIMATE', hue='AGE', marker='o',
linewidth=2.5)
plt.title('Drug Overdose Death Rates by Age Group Over Time', fontsize=16)
plt.xlabel('Year', fontsize=14)
plt.ylabel('Death Rate (per 100,000)', fontsize=14)
plt.legend(title='Age Group', fontsize=12, title_fontsize=14)
plt.show()
```

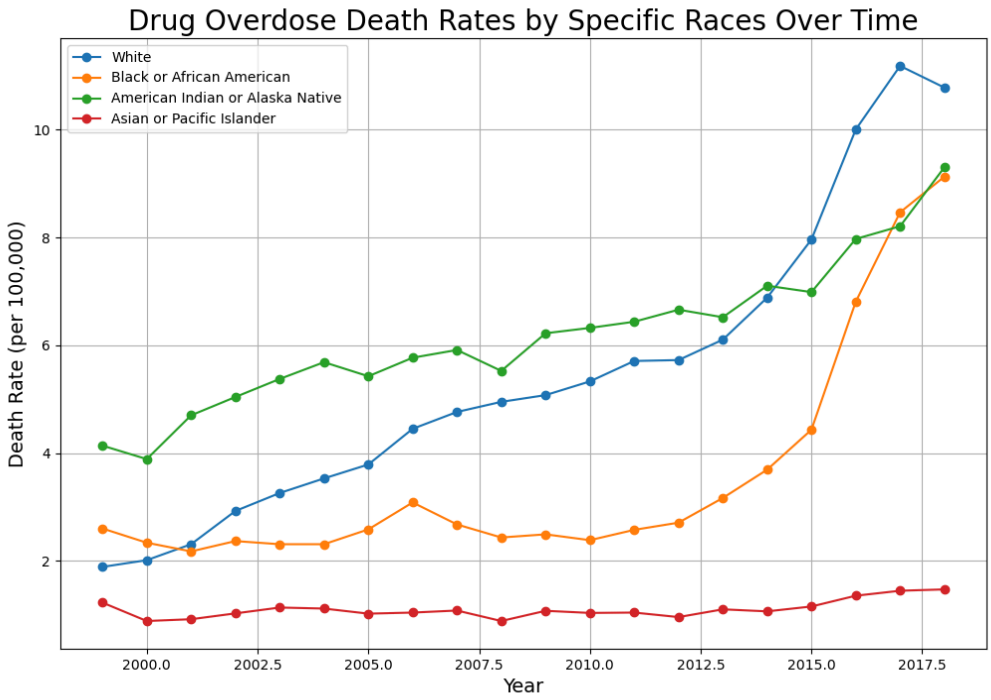


In this analysis, we examine the trend of drug overdose death rates by age group from 2000 to 2018. The graph visualizes death rates per 100,000 individuals for various age groups, revealing several critical patterns. The general trend indicates a significant increase in drug overdose death rates across most age groups over the observed period, highlighting a worsening epidemic. Notably, young adults aged 25-34 years exhibit a sharp rise in death rates, especially in the latter part of the period. By the end of the timeframe, this group shows one of the highest death rates among all age groups. Middle-aged adults (35-44, 45-54, and 55-64 years) also experience substantial increases, with the 45-54 age group particularly affected. Although older adults (65+ years) show increases, their rates remain lower compared to younger and middle-aged adults. The group under 15 years has the lowest and relatively stable death rates throughout the period. The most striking feature of the graph is the sharp rise in death rates from around 2013 onwards, affecting nearly all age groups but most pronounced in younger and middle-aged adults. This surge highlights the broader impact of the opioid crisis and other drug-related issues on younger populations. The data suggests that younger and middle-aged adults are disproportionately affected, necessitating targeted public health interventions. The increasing trends underscore the need for comprehensive strategies, including better access to treatment, prevention programs, and education on the risks of drug use. Addressing the drug overdose crisis through age-specific interventions is crucial to mitigating the rising death rates and providing support to those most at risk.

Race:

```
race_keywords = ['White', 'Black or African American', 'American Indian or Alaska
Native', 'Asian or Pacific Islander']
filtered_race_data_labels =
drug[drug['STUB_LABEL'].str.contains('|'.join(race_keywords), case=False)]
```

```
filtered_race_data_labels['Race'] =
filtered_race_data_labels['STUB_LABEL'].apply(lambda x: 'White' if 'White' in x else

'Black or African American' if 'Black or African American' in x else
'American Indian or Alaska Native' if 'American Indian or Alaska Native' in x else
'Asian or Pacific Islander' if 'Asian or Pacific Islander' in x else None)
race_avg_data = filtered_race_data_labels.groupby(['YEAR', 'Race']).agg({'ESTIMATE':
'mean'}).reset_index()
plt.figure(figsize=(12, 8))
for race in race_keywords:
   race_specific_data = race_avg_data[race_avg_data['Race'] == race]
   if not race_specific_data.empty:
       plt.plot(race_specific_data['YEAR'], race_specific_data['ESTIMATE'],
label=race, marker = 'o')
plt.xlabel('Year',fontsize = 14)
plt.ylabel('Death Rate (per 100,000)', fontsize = 14)
plt.title('Drug Overdose Death Rates by Specific Races Over Time', fontsize = 20)
```



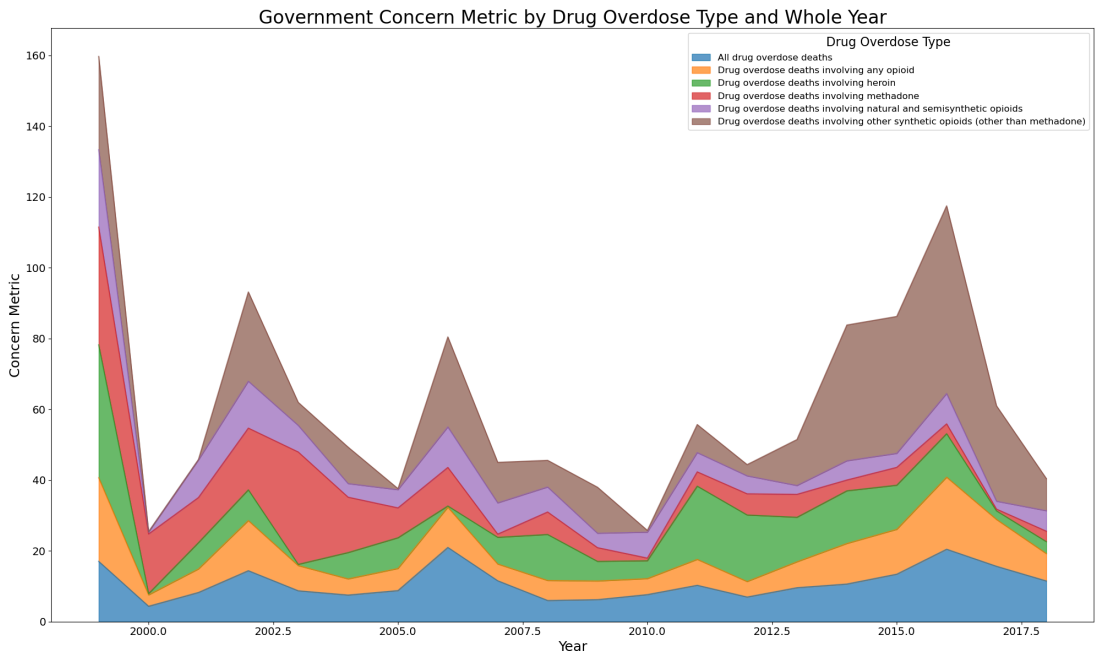
The analysis examines trends in drug overdose death rates across different races in the United States from 1999 to 2017, using data from the CDC. The data was filtered to include Whites, Black or African Americans, American Indian or Alaska Natives, and Asian or Pacific Islanders. It was then aggregated by year and race to calculate the average death rate per 100,000 population. The visualization created using line plots reveals significant racial disparities, with Whites and Black or African Americans experiencing the highest increases in death rates, particularly from 2013 onwards. This trend underscores the need for targeted public health interventions to address the specific challenges faced by these communities. The death rate of the race of Asian or Pacific Islanders remains the most stable and lowest level over time. The stability and relatively low levels of drug overdose deaths among Asian or Pacific Islanders compared to other ethnic groups can be attributed to several factors. Cultural attitudes towards drug use in

many Asian and Pacific Islander communities often emphasize avoiding drug use, which may reduce the prevalence of drug use. In addition, family and community structures in these groups can provide strong social support networks that deter substance abuse. In addition, public health initiatives and interventions targeting these communities may be more effective in preventing substance abuse and providing timely support and treatment. Overall, drug abuse death rates have roughly increased over time across all races, suggesting that governments need to be vigilant, strengthen drug abuse monitoring and communicate the right information to the public.

Question 2- What formula would you use for a metric to rank and label the concern the government should have for different drug overdose types? What would the resulting ranking be when you use this formula?

Concern Metric for Recent Year Drug Overdose Types

```
total_population_data = drug[drug['STUB_NAME'] == 'Total'].copy()

total_population_data['Growth Rate'] =
total_population_data.groupby('PANEL')['ESTIMATE'].pct_change() * 100

total_population_data['Growth Rate'].fillna(total_population_data['Growth
Rate'].mean(), inplace=True)

total_population_data['Growth Rate'] = total_population_data['Growth Rate'].abs()

w1, w2 = 0.5, 0.5

total_population_data['Concern Metric'] = (

   w1 * total_population_data['ESTIMATE'] +

   w2 * total_population_data['Growth Rate']

)

ranked_data = total_population_data.sort_values(by=['YEAR', 'Concern Metric'],
ascending=[True, False]).reset_index(drop=True)

result = ranked_data[['PANEL', 'YEAR', 'ESTIMATE', 'Growth Rate', 'Concern Metric']]

result.to_csv('Government_Concern_Metric.csv', index=False)

pd.read_csv('Government_Concern_Metric.csv')

grouped_data = result.groupby(['YEAR', 'PANEL']).mean().reset_index()

pivot_df = grouped_data.pivot(index='YEAR', columns='PANEL', values='Concern Metric')

plt.figure(figsize=(20, 12))

pivot_df.plot(kind='area', stacked=True, alpha=0.7, figsize=(20, 12))

plt.title('Government Concern Metric by Drug Overdose Type and Whole Year',
fontsize=24)

plt.xlabel('Year', fontsize=18)

plt.ylabel('Concern Metric', fontsize=18)

plt.legend(title='Drug Overdose Type', fontsize=12, title_fontsize=16)

plt.show()
```



To rank and label the concern the government should have for specific drug overdose types, I developed a metric called the "Concern Metric." This metric combines two key factors: the absolute number of overdose deaths (ESTIMATE) and the growth rate of these deaths over time. The formula used is Concern

Metric = 0.5 * ESTIMATE + 0.5 * Growth Rate

, where both the number of deaths and the growth rate are given equal weights (w1 and w2 are both 0.5).

The process starts by loading the relevant data from the dataset and filtering it to focus on the total population data. Then, I calculate the growth rate for each drug type by determining the percentage change in the number of deaths from the previous year. Missing values in the growth rate are filled with the mean growth rate to ensure that no data is excluded due to missing values.

Next, I transform the growth rates into absolute values to account for both increases and decreases in deaths. The Concern Metric is then calculated by averaging the absolute number of deaths and the growth rate, which provides a balanced view of both the current severity and the trend of the problem.

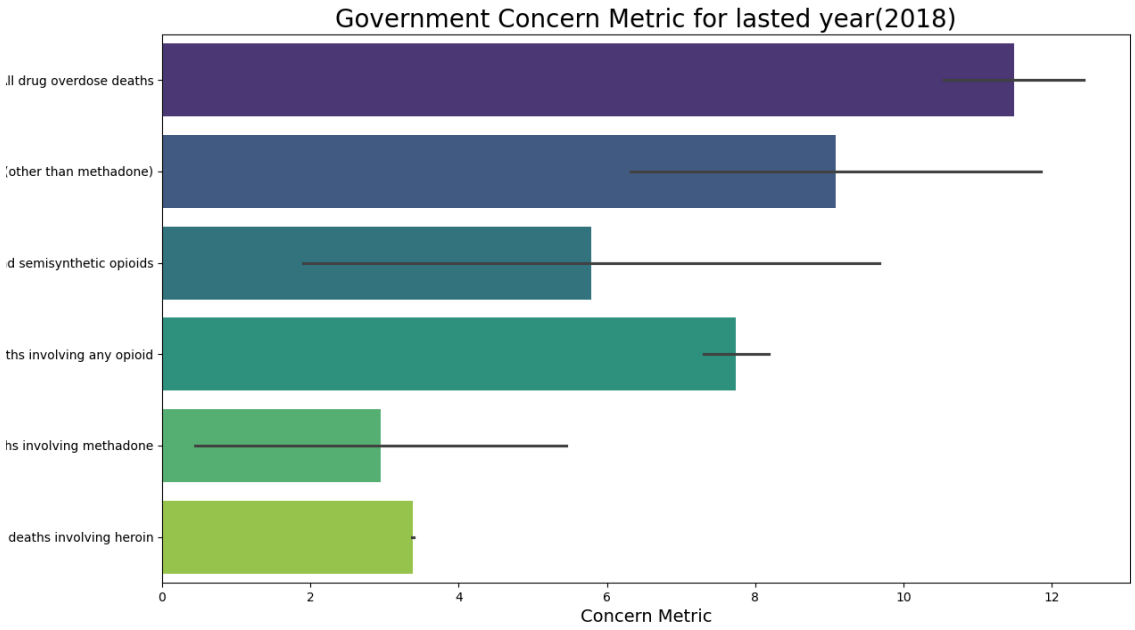
After computing the Concern Metric, I sort the data by year and concern metric to identify the most critical drug overdose types each year. To effectively communicate the results, I create a stacked area chart that displays the Concern Metric for each drug type over the years. This visualization allows for a clear understanding of long-term trends and helps identify which drug types have been the most concerning over time.

Visualization Analysis: The stacked area chart provides a comprehensive view of the Government Concern Metric for various drug overdose types from 1999 to 2018. Each colored area represents a different drug overdose type, and the total height of the stacked areas at any point in time represents the overall concern metric.

Key Observations: There is a significant peak in the concern metric around the year 2000, followed by a sharp decline. This indicates the early 2000s saw a peak in overdose death concerns, followed by a period of stability. From 2000 to 2010, the overall concern metric remains relatively stable, with minor fluctuations. During this period, the contributions from different drug types are more evenly distributed. Starting around 2012, there is a noticeable increase in the concern metric, peaking around 2016. This suggests a rising concern for drug overdose deaths in recent years. The brown area, representing deaths involving other synthetic opioids (excluding methadone), becomes increasingly significant, especially post-2015. This indicates a growing concern for synthetic opioids. Natural and Semisynthetic Opioids in the purple area also show a steady increase, indicating a consistent concern over time. This analysis provides valuable insights for policymakers, indicating where resources and attention should be focused to address the most pressing issues related to drug overdoses. The rising trend in synthetic opioid-related deaths, in particular, calls for targeted interventions to mitigate this growing threat.

Concern Metric Across All Years for Drug Overdose Types

```
# 2018
latest_year = total_population_data['YEAR'].max()
latest_data = total_population_data[total_population_data['YEAR'] == latest_year]
ranked_data_2018 = latest_data.sort_values(by='Concern Metric',
ascending=False).reset_index(drop=True)
result_2018 = ranked_data_2018[['PANEL', 'YEAR', 'ESTIMATE', 'Growth Rate', 'Concern
Metric']]
result.to_csv('2018_Government_Concern_Metric.csv', index=False)
pd.read_csv('2018_Government_Concern_Metric.csv')
plt.figure(figsize=(14, 8))
sns.barplot(data=ranked_data_2018, x='Concern Metric', y='PANEL', palette='viridis')
plt.title('Government Concern Metric for lasted year(2018)', fontsize=20)
plt.xlabel('Concern Metric', fontsize=14)
plt.ylabel('Demographic Group', fontsize=14)
```



For this Concern Metric, focusing on the most recent year (2018) to get a snapshot of the current state. Sort the data for 2018 by the concern metric to identify the most critical drug overdose types. Then, Create a bar chart to display the Concern Metric for each drug type in 2018. Compared with the whole year, focusing on the latest year allows us to capture the most current trends and emerging issues, making it easier to allocate resources effectively to the most pressing problems.

Visualization Analysis: The bar chart provides a clear view of the Government Concern Metric for various drug overdose types in the year 2018. Each bar represents a different drug overdose type, and the length of the bar indicates the concern metric value.

Key Observations: The highest concern metric is observed for drug overdose deaths involving other synthetic opioids (excluding methadone). This indicates a significant concern for synthetic opioids in 2018. Drug overdose deaths involving heroin also show a high concern metric, highlighting the critical impact of heroin on public health in 2018. However, Drug overdose deaths involving methadone, natural and semisynthetic opioids, and all drug overdose deaths have lower concern metrics compared to synthetic opioids and heroin.

Comparison with All Years Analysis

The 2018-specific analysis provides a snapshot of the most urgent issues, whereas the all-years analysis offers a long-term view of trends. The latest year analysis is crucial for immediate action and resource

allocation, highlighting which drug types require the most attention right now. Analyzing all years is also important for understanding overall trends and identifying consistent problems, but it may dilute the focus on the most recent and pressing issues. By combining both approaches, policymakers can make informed decisions that balance immediate needs with long-term strategies.

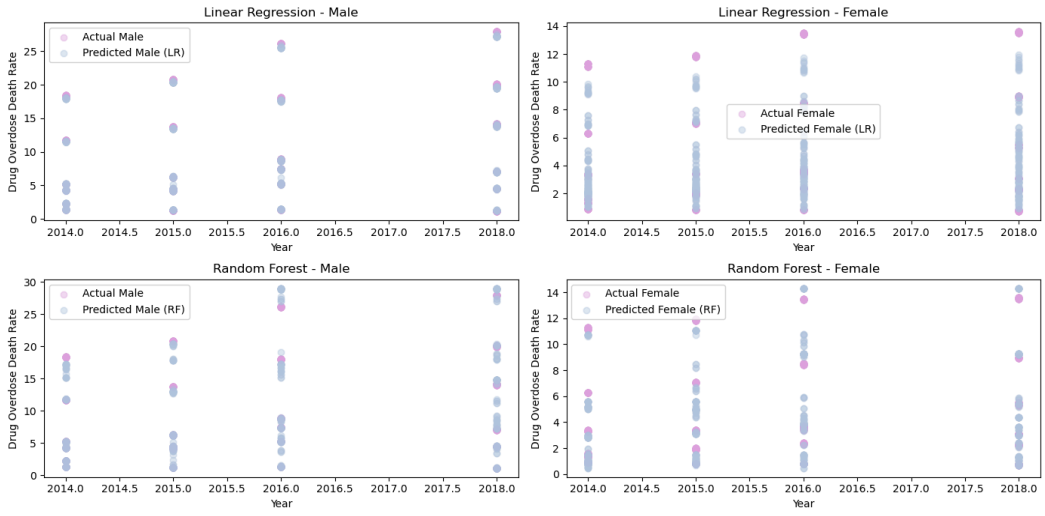
Part 3- Develop a predictive model to forecast future drug overdose death rates for each demographic group.

Sex:

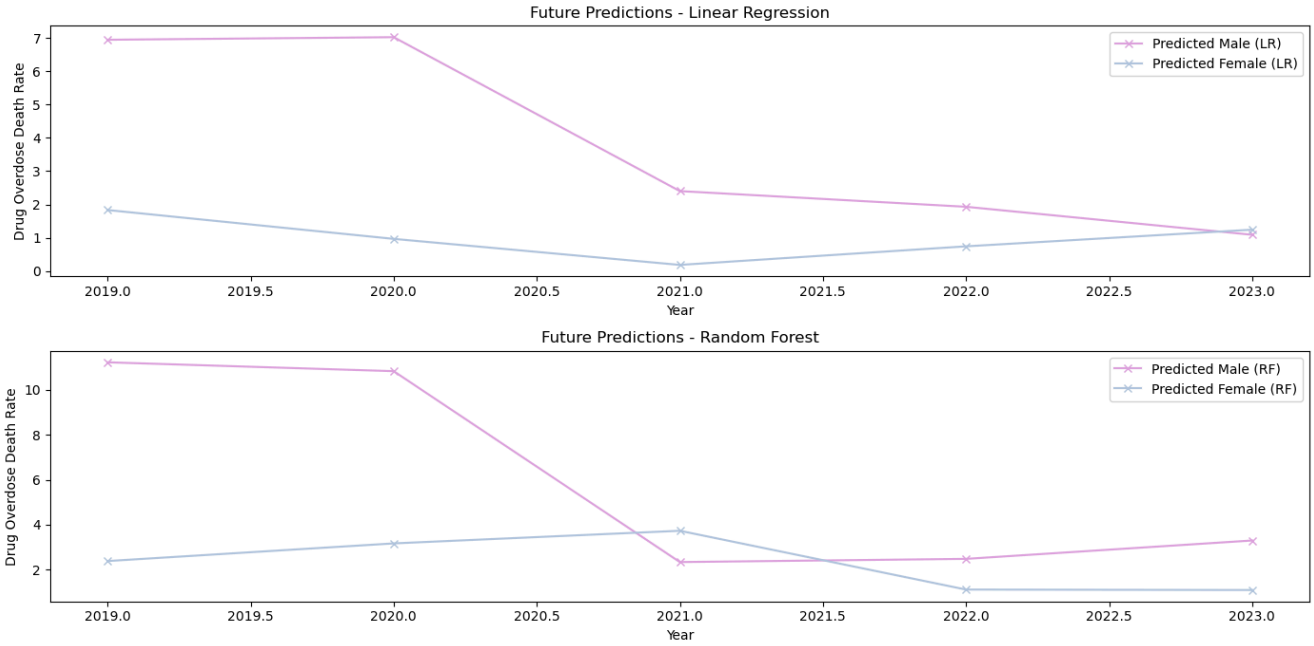
```
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
lr_female = LinearRegression()
lr_female.fit(X_train, y_train_female)
lr_male = LinearRegression()
lr_male.fit(X_train, y_train_male)
# Model Training with Random Forest
rf_female = RandomForestRegressor(n_estimators=100, random_state=42)
rf_female.fit(X_train, y_train_female)
rf_male = RandomForestRegressor(n_estimators=100, random_state=42)
rf_male.fit(X_train, y_train_male)
from sklearn.metrics import mean_squared_error
# Model Evaluation
y_pred_male_lr = lr_male.predict(X_test)
y_pred_female_lr = lr_female.predict(X_test)
y_pred_male_rf = rf_male.predict(X_test)
y_pred_female_rf = rf_female.predict(X_test)
rmse_male_lr = np.sqrt(mean_squared_error(y_test_male, y_pred_male_lr))
rmse_female_lr = np.sqrt(mean_squared_error(y_test_female, y_pred_female_lr))
rmse_male_rf = np.sqrt(mean_squared_error(y_test_male, y_pred_male_rf))
rmse_female_rf = np.sqrt(mean_squared_error(y_test_female, y_pred_female_rf))
last_row = sex_reshaped[['Male_Lag1', 'Male_Lag2', 'Female_Lag1',
'Female_Lag2']].tail(1).values.flatten()
future_lr_male = []
future_lr_female = []
for _ in range(5):
  next_value_lr_male = lr_male.predict([last_row])
  next_value_lr_female = lr_female.predict([last_row])
```

```
  future_lr_male.append(next_value_lr_male[0])
  future_lr_female.append(next_value_lr_female[0])
  last_row = np.roll(last_row, -1)
  last_row[-2] = next_value_lr_female
  last_row[-1] = next_value_lr_male
last_row_rf = sex_reshaped[['Male_Lag1', 'Male_Lag2', 'Female_Lag1',
'Female_Lag2']].tail(1).values.flatten()
future_rf_male = []
future_rf_female = []
for _ in range(5):  next_value_rf_male = rf_male.predict([last_row_rf])[0]
  next_value_rf_female = rf_female.predict([last_row_rf])[0]
  future_rf_male.append(next_value_rf_male)
  future_rf_female.append(next_value_rf_female)
  last_row_rf = np.roll(last_row_rf, -1)
  last_row_rf[-2] = next_value_rf_female
  last_row_rf[-1] = next_value_rf_male
plt.figure(figsize=(14, 7))
years_test = sex_reshaped['YEAR'].iloc[-len(y_test_male):]
plt.subplot(2, 2, 1)
plt.scatter(years_test, y_test_male, label='Actual Male', color='plum', alpha=0.4)
plt.scatter(years_test, y_pred_male_lr, label='Predicted Male (LR)',
color='lightsteelblue', alpha=0.4)
plt.title('Linear Regression - Male')
plt.xlabel('Year')
plt.ylabel('Drug Overdose Death Rate')
plt.subplot(2, 2, 2)
plt.scatter(years_test, y_test_female, label='Actual Female', color='plum', alpha=0.4)
plt.scatter(years_test, y_pred_female_lr, label='Predicted Female (LR)',
color='lightsteelblue', alpha=0.4)
plt.title('Linear Regression - Female')
plt.xlabel('Year')
plt.ylabel('Drug Overdose Death Rate')
plt.legend()
plt.subplot(2, 2, 3)
plt.scatter(years_test, y_test_male, label='Actual Male', color='plum', alpha=0.4)
plt.scatter(years_test, y_pred_male_rf, label='Predicted Male (RF)',
color='lightsteelblue', alpha=0.4)
plt.title('Random Forest - Male')
plt.xlabel('Year')
plt.ylabel('Drug Overdose Death Rate')
plt.legend()
```

```
plt.subplot(2, 2, 4)
plt.scatter(years_test, y_test_female, label='Actual Female', color='plum', alpha=0.4)
plt.scatter(years_test, y_pred_female_rf, label='Predicted Female (RF)',
color='lightsteelblue', alpha=0.4)
plt.title('Random Forest - Female')
plt.xlabel('Year')
plt.ylabel('Drug Overdose Death Rate')
years_future = range(sex_reshaped['YEAR'].max() + 1, sex_reshaped['YEAR'].max() + 6)
plt.figure(figsize=(14, 7))
plt.subplot(2, 1, 1)
plt.plot(years_future, future_lr_male, label='Predicted Male (LR)', color='plum',
marker = 'x')
plt.plot(years_future, future_lr_female, label='Predicted Female (LR)',
color='lightsteelblue',marker = 'x')
plt.title('Future Predictions - Linear Regression')
plt.xlabel('Year')
plt.ylabel('Drug Overdose Death Rate')
plt.legend()
plt.subplot(2, 1, 2)
plt.plot(years_future, future_rf_male, label='Predicted Male (RF)', color='plum',
marker = 'x')
plt.plot(years_future, future_rf_female, label='Predicted Female (RF)',
color='lightsteelblue', marker = 'x')
plt.title('Future Predictions - Random Forest')
plt.xlabel('Year')
plt.ylabel('Drug Overdose Death Rate')
plt.legend()
plt.tight_layout()
plt.show()
```



```
from scipy.stats import t
residuals_male_lr = y_test_male - y_pred_male_lr
residuals_female_lr = y_test_female - y_pred_female_lr
std_residuals_male_lr = np.std(residuals_male_lr)
std_residuals_female_lr = np.std(residuals_female_lr)
se_predictions_male_lr = std_residuals_male_lr / np.sqrt(len(y_test_male))
se_predictions_female_lr = std_residuals_female_lr / np.sqrt(len(y_test_female))
t_value = t.ppf(0.975, df=len(y_test_male)-1)
margin_of_error_male_lr = t_value * se_predictions_male_lr
margin_of_error_female_lr = t_value * se_predictions_female_lr
print('Margin of Error for Linear Regression - Male:', margin_of_error_male_lr)
print('Margin of Error for Linear Regression - Female:', margin_of_error_female_lr)
predictions_male_rf = np.array([tree.predict(X_test) for tree in rf_male.estimators_])
predictions_female_rf = np.array([tree.predict(X_test) for tree in
rf_female.estimators_])
std_predictions_male_rf = np.std(predictions_male_rf)
std_predictions_female_rf = np.std(predictions_female_rf)
se_predictions_male_rf = std_predictions_male_rf / np.sqrt(len(y_test_male))
se_predictions_female_rf = std_predictions_female_rf / np.sqrt(len(y_test_female))
t_value = t.ppf(0.975, df=len(y_test_male)-1)
margin_of_error_male_rf = t_value * se_predictions_male_rf
margin_of_error_female_rf = t_value * se_predictions_female_rf
```



```
residuals_male_lr = y_test_male - y_pred_male_lr
residuals_female_lr = y_test_female - y_pred_female_lr
std_residuals_male_lr = np.std(residuals_male_lr)
std_residuals_female_lr = np.std(residuals_female_lr)
se_predictions_male_lr = std_residuals_male_lr / np.sqrt(len(y_test_male))
se_predictions_female_lr = std_residuals_female_lr / np.sqrt(len(y_test_female))
t_value = t.ppf(0.975, df=len(y_test_male)-1)
margin_of_error_male_lr = t_value * se_predictions_male_lr
margin_of_error_female_lr = t_value * se_predictions_female_lr
r2_male_lr = r2_score(y_test_male, y_pred_male_lr)
r2_female_lr = r2_score(y_test_female, y_pred_female_lr)
rmse_male_lr = np.sqrt(mean_squared_error(y_test_male, y_pred_male_lr))
rmse_female_lr = np.sqrt(mean_squared_error(y_test_female, y_pred_female_lr))
predictions_male_rf = np.array([tree.predict(X_test) for tree in rf_male.estimators_])
predictions_female_rf = np.array([tree.predict(X_test) for tree in
rf_female.estimators_])
y_pred_male_rf = np.mean(predictions_male_rf, axis=0)
y_pred_female_rf = np.mean(predictions_female_rf, axis=0)
std_predictions_male_rf = np.std(predictions_male_rf, axis=0)
std_predictions_female_rf = np.std(predictions_female_rf, axis=0)
se_predictions_male_rf = np.mean(std_predictions_male_rf) / np.sqrt(len(y_test_male))
se_predictions_female_rf = np.mean(std_predictions_female_rf) /
np.sqrt(len(y_test_female))
margin_of_error_male_rf = t_value * se_predictions_male_rf
margin_of_error_female_rf = t_value * se_predictions_female_rf
r2_male_rf = r2_score(y_test_male, y_pred_male_rf)
r2_female_rf = r2_score(y_test_female, y_pred_female_rf)
rmse_male_rf = np.sqrt(mean_squared_error(y_test_male, y_pred_male_rf))
rmse_female_rf = np.sqrt(mean_squared_error(y_test_female, y_pred_female_rf))
data = {
  "Metric": ["R^2", "RMSE", "Margin of Error"],
  "Male_LR": [r2_male_lr, rmse_male_lr, margin_of_error_male_lr],
  "Female_LR": [r2_female_lr, rmse_female_lr, margin_of_error_female_lr],
  "Male_RF": [r2_male_rf, rmse_male_rf, margin_of_error_male_rf],
  "Female_RF": [r2_female_rf, rmse_female_rf, margin_of_error_female_rf]
}
table = pd.DataFrame(data)
```

	Metric	Male_LR	Female_LR	Male_RF	Female_RF
0	R^2	0.936455	0.327713	0.926644	0.297278
1	RMSE	2.017694	3.370785	2.167864	3.446240
2	Margin of Error	0.164767	0.265747	0.058408	0.057430

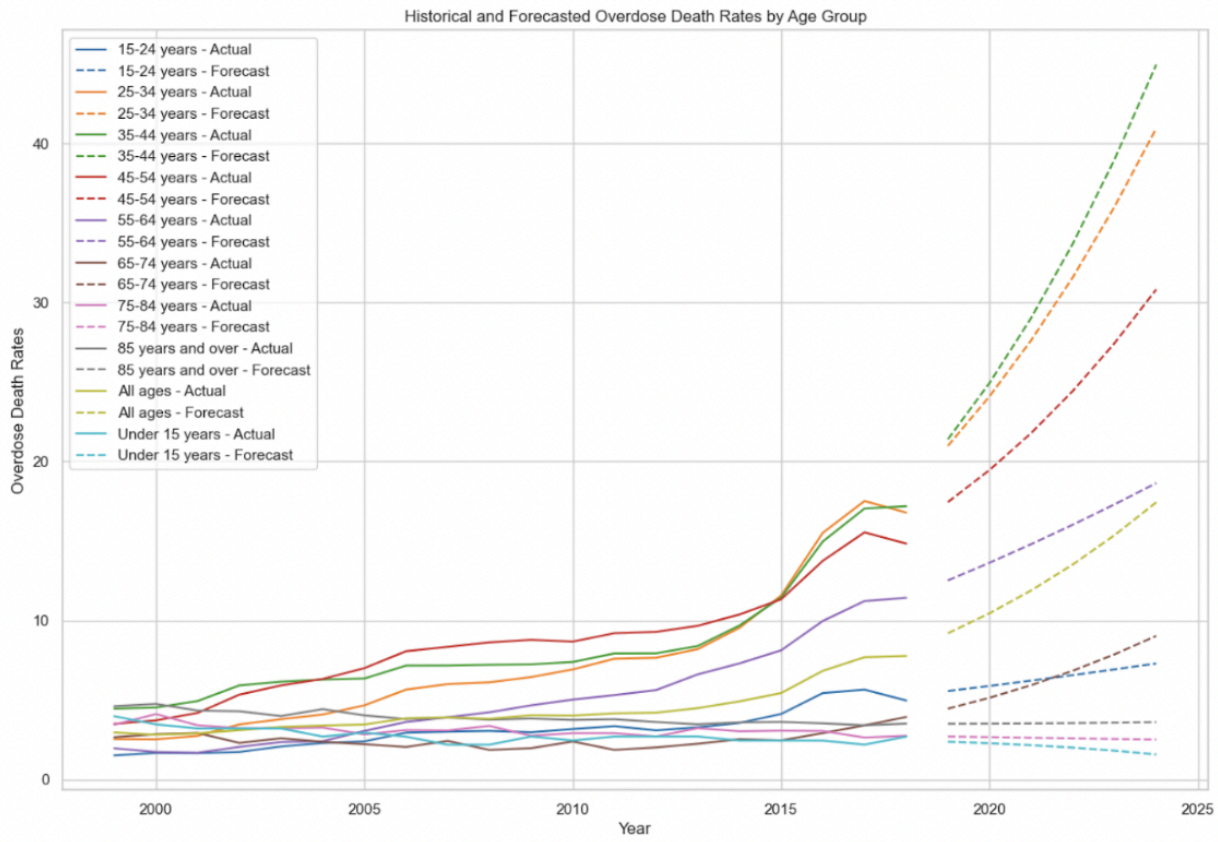
In this part, we analyze the predicted male drug overdose death rates from a linear regression model, which show a steep initial decline followed by a gradual decrease and eventual stabilization, starting at approximately 7 and dropping to around 2 by the end of the forecast period. Both linear regression and random forest models indicate a decline in drug overdose death rates for both genders over the next five years, with linear regression demonstrating a more stable, consistent decline, whereas random forest models exhibit more fluctuations, suggesting a nuanced understanding of trends. We also evaluate these two models. And in general, the Linear Regression model shows higher R^2 values for both genders, indicating better overall accuracy compared to the Random Forest model. However, the Random Forest model, with its lower RMSE and consistent margins of error, captures the nuanced fluctuations in the data more effectively. However, linear regression fails to capture the variability in the data for both genders, showing a poor fit, while random forest provides a better fit by capturing more actual trends but still misses some peaks and troughs. The analysis suggests that gender may not be a strong predictor of drug overdose death rates, indicating that other factors might be more significant in determining future rates. Future analysis should consider including additional variables such as age and drug types involved to enhance prediction accuracy and better capture the complexity of drug overdose trends.

Age

Polynomial Regression

```
from sklearn.preprocessing import PolynomialFeatures
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import KFold
from scipy.stats import t
drug['ESTIMATE'].fillna(drug['ESTIMATE'].mean(), inplace=True)
drug['YEAR'] = drug['YEAR'].astype(int)
drug['AGE'] = drug['AGE'].astype(str)
agg_data = drug.groupby(['YEAR', 'AGE'])['ESTIMATE'].mean().unstack()
best_poly_models = {}
best_degrees = {}
rmse_values = {}
X = agg_data.index.values.reshape(-1, 1)
degree_range = range(1, 6)
def cross_val_rmse(model, X, y, folds=5):
  kf = KFold(n_splits=folds, shuffle=True, random_state=1)
  rmses = []
  for train_index, test_index in kf.split(X):
      X_train, X_test = X[train_index], X[test_index]
      y_train, y_test = y[train_index], y[test_index]
      model.fit(X_train, y_train)
      predictions = model.predict(X_test)
      rmse = np.sqrt(mean_squared_error(y_test, predictions))
```

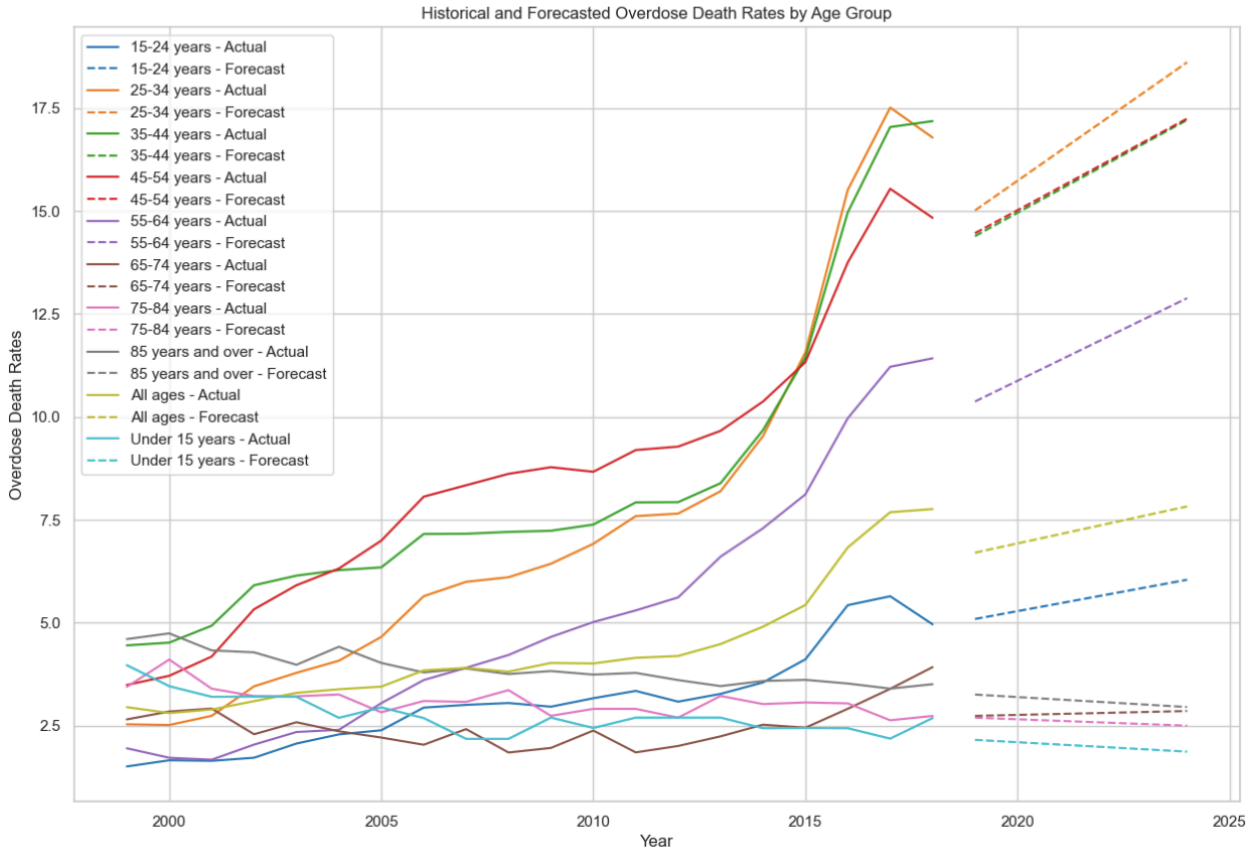
```
        rmses.append(rmse)
    return np.mean(rmses)
for age_group in agg_data.columns:
    y = agg_data[age_group].values
    best_rmse = float('inf')
    best_degree = None
    best_model = None
    for degree in degree_range:
        poly = PolynomialFeatures(degree=degree)
        X_poly = poly.fit_transform(X)
        model = LinearRegression()
        rmse = cross_val_rmse(model, X_poly, y)
        if rmse < best_rmse:
            best_rmse = rmse
            best_degree = degree
            best_model = LinearRegression().fit(X_poly, y)
    best_degrees[age_group] = best_degree
    best_poly_models[age_group] = best_model
    rmse_values[age_group] = best_rmse
print("Best Polynomial Degrees:\n", best_degrees)
print("RMSE Values:\n", rmse_values)
forecast_years = list(range(2019, 2025))
future_predictions = pd.DataFrame(index=forecast_years, columns=agg_data.columns)
for age_group, model in best_poly_models.items():
    degree = best_degrees[age_group]
    poly = PolynomialFeatures(degree=degree)
    future_X_poly = poly.fit_transform(np.array(forecast_years).reshape(-1, 1))
    future_predictions[age_group] = model.predict(future_X_poly)
plt.figure(figsize=(15, 10))
colors = plt.cm.get_cmap('tab10', len(agg_data.columns))
for i, age_group in enumerate(agg_data.columns):
    plt.plot(agg_data.index, agg_data[age_group], label=f'{age_group} - Actual',
color=colors(i))
    plt.plot(future_predictions.index, future_predictions[age_group], linestyle='--',
label=f'{age_group} - Forecast', color=colors(i))
plt.xlabel('Year')
plt.ylabel('Overdose Death Rates')
plt.title('Historical and Forecasted Overdose Death Rates by Age Group')
plt.legend()
plt.show()
residuals = {}
for age_group in agg_data.columns:
    model = best_poly_models[age_group]
    degree = best_degrees[age_group]
    poly = PolynomialFeatures(degree=degree)
    X_poly = poly.fit_transform(X)
    predicted = model.predict(X_poly)
    actual = agg_data[age_group].values
    residuals[age_group] = actual - predicted
margin_of_errors = {}
confidence_level = 0.95
degrees_freedom = len(X) - 1
t_critical = t.ppf((1 + confidence_level) / 2, df=degrees_freedom)
for age_group, res in residuals.items():
    std_error = np.std(res, ddof=1)
    margin_of_error = t_critical * std_error
    margin_of_errors[age_group] = margin_of_error
print("Margin of Errors:\n", margin_of_errors)
```



Linear regression

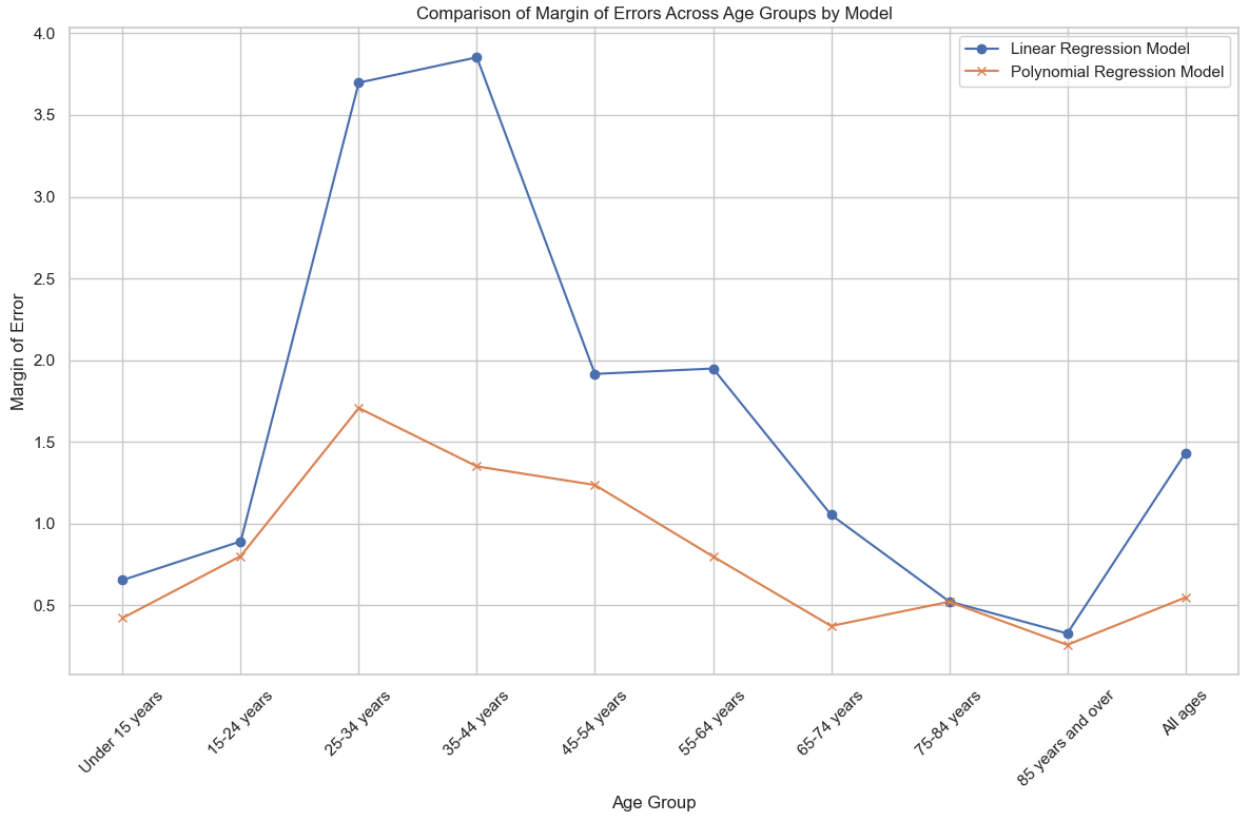
```
drug = ;
linreg_models = {}
rmse_values = {}
X = agg_data.index.values.reshape(-1, 1)
forecast_years = list(range(2019, 2025))
future_X = np.array(forecast_years).reshape(-1, 1)
future_predictions = pd.DataFrame(index=forecast_years, columns=agg_data.columns)
for age_group in agg_data.columns:
  y = agg_data[age_group].values
  model = LinearRegression().fit(X, y)
  linreg_models[age_group] = model
  future_predictions[age_group] = model.predict(future_X)
  in_sample_predictions = model.predict(X)
  rmse = np.sqrt(mean_squared_error(y, in_sample_predictions))
  rmse_values[age_group] = rmse
plt.figure(figsize=(15, 10))
colors = plt.cm.get_cmap('tab10', len(agg_data.columns))

for i, age_group in enumerate(agg_data.columns):
  plt.plot(agg_data.index, agg_data[age_group], label=f'{age_group} - Actual',
color=colors(i))
  plt.plot(future_predictions.index, future_predictions[age_group], linestyle='--',
label=f'{age_group} - Forecast', color=colors(i))
plt.xlabel('Year')
plt.ylabel('Overdose Death Rates')
plt.title('Historical and Forecasted Overdose Death Rates by Age Group')
```

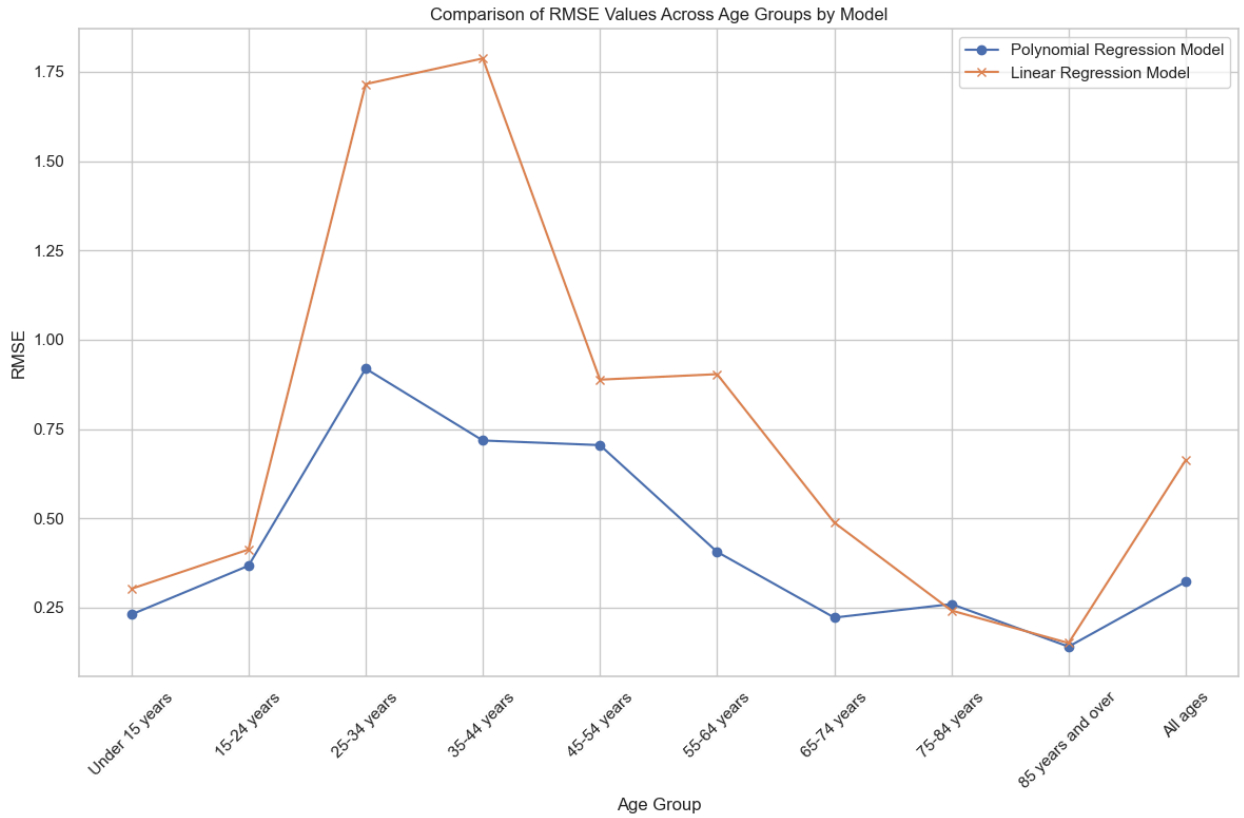
```
plt.show()
```




```
age_groups = ['Under 15 years', '15-24 years', '25-34 years', '35-44 years', '45-54
years', '55-64 years', '65-74 years', '75-84 years', '85 years and over', 'All ages']
margin_errors_model1 = [0.6531630901171845, 0.8900034198290717, 3.6983070511034635,
3.8532829829418866, 1.9150420703847961, 1.9481217739355186, 1.0524108497716191,
0.5221179301392773, 0.32581283151258716, 1.4299630826782754]
margin_errors_model2 = [0.42066839106330517, 0.7998817343181052, 1.7061553109107028,
1.3488958921659182, 1.2350658705608828, 0.7973978693235392, 0.3719254404002682,
0.520155134787402, 0.2564191025980815, 0.5462092991300304]
plt.figure(figsize=(12, 8))
plt.plot(age_groups, margin_errors_model1, label='Linear Regression Model',
marker='o')
plt.plot(age_groups, margin_errors_model2, label='Polynomial Regression Model',
marker='x')
plt.title('Comparison of Margin of Errors Across Age Groups by Model')
plt.xlabel('Age Group')
plt.ylabel('Margin of Error')
plt.xticks(rotation=45)
plt.show()
```

```
rmse_values_model1 = [0.2310392903599993, 0.3675517622603396, 0.9193634742960677,
0.7181636378052114, 0.7052244362869698, 0.4063779435809085, 0.2224682944069633,
0.2602339025407311, 0.14114905917156972, 0.32280012855470697]
rmse_values_model2 = [0.3030215396408503, 0.4128987241361757, 1.715753253125379,
1.7876511392483212, 0.8884442575297552, 0.9037909034937891, 0.4882443004783337,
0.24222584138444292, 0.1511541410308047, 0.663401869301903]
plt.figure(figsize=(12, 8))
plt.plot(age_groups, rmse_values_model1, label='Polynomial Regression Model',
marker='o')
plt.plot(age_groups, rmse_values_model2, label='Linear Regression Model', marker='x')
plt.title('Comparison of RMSE Values Across Age Groups by Model')
plt.xlabel('Age Group')
plt.ylabel('RMSE')
plt.xticks(rotation=45)
plt.legend()
plt.show()
```


Comparison of Margin of Errors Across Age Groups by Model
Linear Regression Model
Polynomial Regression Model
Margin of Error
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
Under 15 years
15-24 years
25-34 years
35-44 years
45-54 years
55-64 years
65-74 years
75-84 years
85 years and over
All ages
Age Group


Comparison of RMSE Values Across Age Groups by Model
Polynomial Regression Model
Linear Regression Model
RMSE
1.75
1.50
1.25
1.00
0.75
0.50
0.25
Under 15 years
15-24 years
25-34 years
35-44 years
45-54 years
55-64 years
65-74 years
75-84 years
85 years and over
All ages
Age Group

This section, as the graph shows, involved a thorough investigation utilizing polynomial regression to forecast future drug overdose fatality rates across different age groups. The graph skillfully presents historical data together with anticipated rates through 2025, highlighting real historical rates and expected trends for various age groups. The much smaller margin of error seen with polynomial models supported the choice to choose them over simpler linear models. The sophisticated, non-linear patterns shown in the overdose data are skillfully captured by these models; patterns that linear models frequently find difficult to portray because of their inherent simplicity.

Carefully preparing the data—including managing missing values—and creating polynomial characteristics based on the past patterns of each age group were part of the analysis. Through the application of cross-validation, we reduced the margin of error by optimizing the polynomial degree for every demographic group, so improving the accuracy and predictive capacity. This meticulous process of selection made sure that the models were robust in their capacity to generalize across various data sets and tailored to capture the particular dynamics of each age group. The graph acts as an illustration of the accuracy of these models by showing a notable difference between the predicted and real data points, so emphasizing how well the polynomial method captures the complex dynamics of drug overdose trends.

The overdose death rates by age group, wherein each demographic follows a different trajectory, are graphically shown in the graph that is provided. The projections for age groups like 25–34 and 35–44 years highlight a serious worry about the growing drug abuse in these generations. The comparatively steady lines for the very young (those under 15) and very old (those 85 years of age and above) on the other hand, point to less variation in these categories. The graph skillfully shows the possible future problems if current trends go unchecked by using dashed lines to distinguish forecasts from actual historical data.

Planning and drafting of public health policies depend on an examination of these trends. Realizing that middle-aged persons are expected to have the biggest increases in overdose death rates makes focused programs meant to prevent and treat these populations necessary. This knowledge can guide the distribution of resources, such money for educational initiatives and addiction treatment centers, which are geared especially at the most vulnerable groups. In order to slow down this concerning trend and enhance public health outcomes, legislators and medical experts should proactively address the growing issues of drug overdose by predicting these patterns.

Race:

```
from sklearn.preprocessing import PolynomialFeatures
from sklearn.model_selection import train_test_split, cross_val_score
```

```
from sklearn.metrics import mean_squared_error

from scipy.stats import norm

file_path =
'Drug_overdose_death_rates__by_drug_type__sex__age__race__and_Hispanic_origin__United_
States_20240518.csv'

data = pd.read_csv(file_path)

race_keywords = ['White', 'Black or African American', 'American Indian or Alaska
Native', 'Asian or Pacific Islander']

filtered_race_data_labels =
data[data['STUB_LABEL'].str.contains('|'.join(race_keywords), case=False)]

filtered_race_data_labels['Race'] =
filtered_race_data_labels['STUB_LABEL'].apply(lambda x: 'White' if 'White' in x else
                                                                    'Black
or African American' if 'Black or African American' in x else


'American Indian or Alaska Native' if 'American Indian or Alaska Native' in x else
                                                                    'Asian
or Pacific Islander' if 'Asian or Pacific Islander' in x else None)

race_avg_data = filtered_race_data_labels.groupby(['YEAR', 'Race']).agg({'ESTIMATE':
'mean'}).reset_index()

race_avg_data_pivot = race_avg_data.pivot(index='YEAR', columns='Race',
values='ESTIMATE')

future_years = 10

future_years_range = np.arange(race_avg_data_pivot.index[-1] + 1,
race_avg_data_pivot.index[-1] + 1 + future_years)

future_X = future_years_range.reshape(-1, 1)

confidence_level = 0.95

z_value = norm.ppf((1 + confidence_level) / 2)

forecasts = {}

results = []

def find_best_degree(X, y, max_degree=5):

  best_degree = 1

  best_score = float('inf')

   for degree in range(1, max_degree + 1):

      poly = PolynomialFeatures(degree)

      X_poly = poly.fit_transform(X)


      model = LinearRegression()

      scores = cross_val_score(model, X_poly, y, scoring='neg_mean_squared_error',
cv=5)

      rmse = np.sqrt(-scores.mean())


      if rmse < best_score:

          best_score = rmse
```

```
            best_degree = degree

    return best_degree, best_score
for race in race_keywords:
    if race in race_avg_data_pivot.columns:
        y = race_avg_data_pivot[race].values
        X = race_avg_data_pivot.index.values.reshape(-1, 1)
        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
        linear_model = LinearRegression()
        linear_model.fit(X_train, y_train)
        y_pred = linear_model.predict(X_test)
        linear_rmse = np.sqrt(mean_squared_error(y_test, y_pred))
        linear_std_error = np.std(y_test - y_pred) / np.sqrt(len(y_test))
        linear_margin_of_error = z_value * linear_std_error
        future_linear_pred = linear_model.predict(future_X)
        forecasts[race] = future_linear_pred
        results.append({
            'Race': race,
            'Model': 'Linear',
            'RMSE': linear_rmse,
            'Margin of Error': linear_margin_of_error
        })
        best_degree, poly_rmse = find_best_degree(X_train, y_train)
        poly = PolynomialFeatures(degree=best_degree)
        X_poly_train = poly.fit_transform(X_train)
        X_poly_test = poly.transform(X_test)
        poly_model = LinearRegression()
        poly_model.fit(X_poly_train, y_train)
        y_poly_pred = poly_model.predict(X_poly_test)
        poly_std_error = np.std(y_test - y_poly_pred) / np.sqrt(len(y_test))
        poly_margin_of_error = z_value * poly_std_error
        X_poly = poly.fit_transform(X)
        poly_model.fit(X_poly, y)
            future_X_poly = poly.transform(future_X)
        future_poly_pred = poly_model.predict(future_X_poly)
        forecasts[race] = future_poly_pred
        results.append({
            'Race': race,
            'Model': 'Polynomial (Degree {})'.format(best_degree),
```

```
            'RMSE': poly_rmse,
            'Margin of Error': poly_margin_of_error
        })
results_df = pd.DataFrame(results)
print(results_df)
plt.figure(figsize=(12,8))
colors = plt.cm.tab10(np.linspace(0, 1, len(race_keywords)))
for i, race in enumerate(race_keywords):
  if race in forecasts:
      plt.plot(race_avg_data_pivot.index, race_avg_data_pivot[race], label=f'{race} Historical', color=colors[i])
      plt.plot(future_X, forecasts[race], label=f'{race} Forecast', linestyle='--', color=colors[i])
plt.xlabel('Year')
plt.ylabel('Death Rate (per 100,000)')
plt.title('Drug Overdose Death Rates by Specific Races Over Time')
plt.show()
```

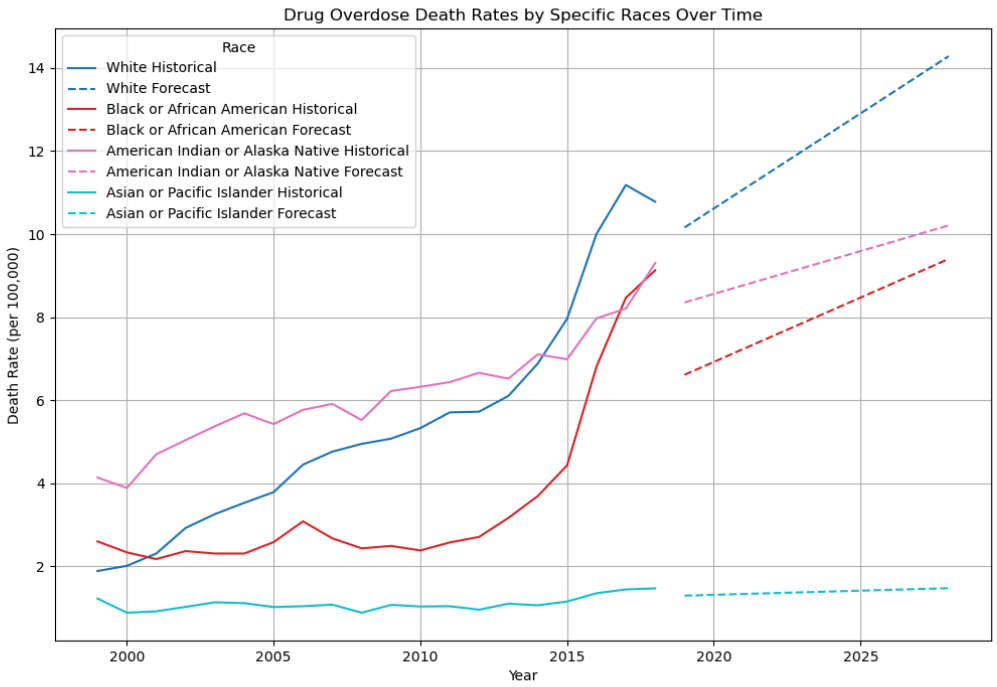
In this part, we use two different machine learning models, Linear Regression and Polynomial Regression, to predict the future drug overdose death rates from 2019 to 2028 based on the feature of races in demographic.

Linear Regression Model: For forecasting future drug overdose death rates by race, we firstly implemented a linear regression model. Initially, we defined a range of future years to forecast, extending 10 years from the last available year in the dataset. For each race, we split the historical data into training and test sets, ensuring that 20% of the data was reserved for testing. A linear regression model was then trained on the training set to learn the relationship between the years and the drug overdose death rates. Predictions were made on the test set to evaluate the model's performance, and the standard error of the predictions was calculated. Using a 95% confidence interval, we computed the margin of error for the predictions to provide an indication of their accuracy. The trained model was subsequently used to predict drug overdose death rates for the defined future years, and the predicted future rates along with their margins of error were stored for each race.

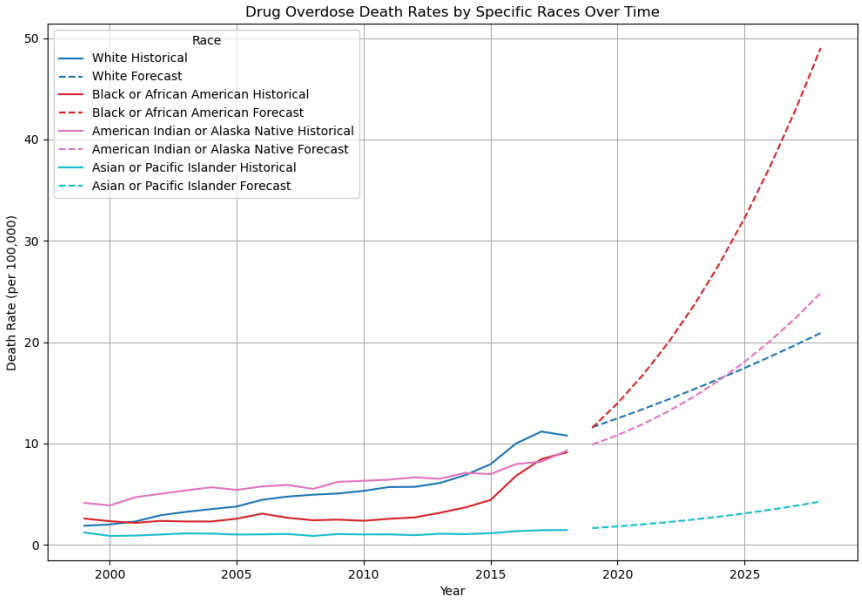
Polynomial Regression Model: To capture more complex relationships in the data and potentially improve the accuracy of our forecasts, we secondly implemented a polynomial regression model. This approach involves finding the best polynomial degree that minimizes the Root Mean Squared Error (RMSE). We defined a function to perform cross-validation and evaluate the model's performance for polynomial degrees ranging from 1 to n. For each race, the historical data was split into training and test sets, and the optimal polynomial degree was determined using the defined function. A polynomial regression model with the best degree was then trained on the training set. The model's performance was assessed by making predictions on the test set and calculating the standard error and margin of error. The trained polynomial

regression model was used to forecast future drug overdose death rates for the next 10 years. The predicted rates and their margins of error were stored and compared with the linear regression results to evaluate the improvement in forecast accuracy.

Linear Regression Model



Polynomial Regression Model



The two models generated two plots, which included historical data lines (solid lines) and future prediction lines (dashed lines), and simultaneously calculated the mortality rates of different races for each of the next ten years. Overall, both models predicted the same trend, with the predicted future drug abuse death rate consistently increasing for different races. However, Polynomial Regression models predict much higher values, for example, for Black or African Americans, The Linear Regression model predicted the highest mortality rate of 9.40 (per 100,000), while the Polynomial Regression model predicted the highest mortality rate of 49.02. The huge difference between the two models is reflected in almost all races, which may be due to the basic property of the Polynomial Regression model, that is, rapid exponential growth. Secondly, in the analysis of the Polynomial Regression model, we use the algorithm to find the best degree, so as to find the best model (minimum RMSE) in the prediction of each race. Our predictions agree with the calculation that the Polynomial Regression model produces an almost smaller Margin of Error for all races, This also means that the performance of this model is more accurate than that of the Linear Regression model.

Conclusion

We investigate drug overdose fatality rates in the United States in this extensive study along a number of factors, including drug type, gender, age, and race. This multifaceted approach highlights important patterns such as the concerning increase in opioid-related deaths, especially from synthetic opioids, and the different effects on men and women, therefore revealing the complexity and scale of the epidemic. Age-specific interventions are therefore required since, as

the age group study shows, young adults and middle-aged people are disproportionately impacted. Racial differences also surface, with larger rises in death rates in some groups, underscoring the need for culturally appropriate public health initiatives. Analyzing these patterns helps us to have a more complex picture of the pandemic, which makes it possible to create more focused and efficient measures to meet the particular requirements of every demographic group. The need for a thorough, data-driven strategy in addressing the current drug overdose problem and, in the end, saving lives is highlighted by this study.