Olina (Qiuyu) Zhu | qz258
BIS634 Final Project Report | Fall 2021

Accidental Drug Related Deaths 2012-2020 (Connecticut)

Please note that the web interface has been modified since the presentation. The homepage and 'Missing values' pages were edited and additional information was added into the new analysis option 'Dataset View'.

Overview

Illegal drug use and drug abuse has plagued the United States since early 19th century, but this issue has likely been present prior to such systematic research and efforts to control drug abuse. The prevalence of drug use in the United States as of 2020 is approximately 19%, with over 31.9 million illegal drug users aged 12 or older, and if including the drugs tobacco and alcohol – which are not illegal, but regardless very harmful to human health – this figure rises to over 53 million (NCDAS, 2021). During the time between the beginning of 2020 and present day, there has been over 700,000 recorded drug overdoses, and in the same year the federal government issued a $35 billion budget for drug control, but the results were not very significant (NCDAS, 2021). The rise of drug-involved overdose deaths increased relatively consistently from 1999 to 2019 (Fig. 1), and there were significantly more incidences occurring among men compared to among women (National Institute on Drug Abuse, 2021).

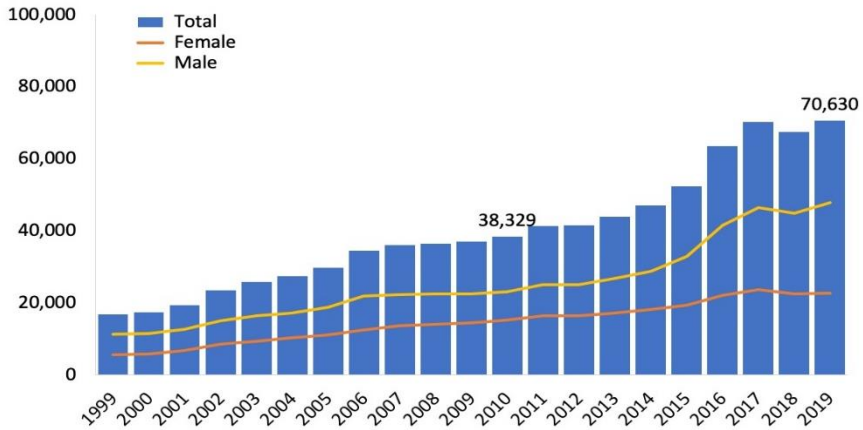

Figure 1. National Drug-Involved Overdose Deaths. Number among all ages, by gender, 1999-2019. Graph adapted from NCDAS found at: https://www.drugabuse.gov/drug-topics/trends-statistics/overdose-death-rates.

More specifically, 22% of American men and 17% of American women have reported use of illegal drugs, and the most common age range in which such incidences occur is 18-25 years of age (NCDAS, 2021). A more detailed review of age distribution of illegal drug use in the United States is highly alarming: approximately 5% of 8th-graders, 20% of 10th-gradesr, 20% of 12th-graders, and 47% of all students who have not yet completed 12th grade have reported using illegal drugs at least once. Moreover, the exposure of children younger than five years of age to marijuana increased by 1.5-fold over a seven-year period (NCDAS, 2021).

Due to the significant impact and prevalence of drug abuse incidences in the United States, I have decided to conduct research on drug-related deaths in the state that I am currently residing in, Connecticut.

Dataset

Source

The dataset "Accidental Drug Related Deaths 2012-2020" was acquired from Data.ct.gov, which can be accessed through clicking this link. This dataset did not specify any licensing information but was classified as a public dataset and is intended for public access and use. Moreover, the dataset could be downloaded as CSV, RDF, JSON, and XML formats, and the metadata file is available for download on the same webpage as a JSON file. This dataset was published on November 12th of 2020, and the latest update occurred on November 29th of 2021.

The preprocessed dataset included 7,679 entries and 41 columns (variables), including:

- Incidence ID
- Incidence recorded date
- Date type
- Age
- Sex
- Race
- Residence city, county, and state
- Death city, county, and state
- Location if 'other'
- Description of injury
- Injury place, city, county, and state
- Cause of death
- Manner of Death
- Geos of death city, residence city, and injury
- Drug types:
  - Heroin
  - Cocaine
  - Fentanyl
  - Fentanyl analogue
  - Oxycodone
  - Oxymorphone
  - Ethanol
  - Hydrocodone
  - Benzodiazepine
  - Methadone
  - Hydromorphone
  - Xylazine
  - Amphet
  - Tramad
  - Morphine (not heroin)
  - Other
  - Opiate NOS
  - Any Opioid

FAIRness (GO FAIR, 2021)

FAIR Components	Description	Satisfies Component?
Findable	The dataset was easily findable on Data.ct.gov, which can also be accessed with Data.gov. The data is downloadable in various human- and machine-readable formats, such as JSON, RDF, and XML, and metadata is in JSON format.	Yes
Accessible	The dataset was accessible as it does not require any authorization process for access or download, was classified as a public dataset, and the metadata was equally accessible.	Yes
Interoperable	The dataset and metadata are downloadable as formal and broadly applicable languages (CSV, JSON, RDF, and XML).	Yes
Reusable	The dataset and metadata were described with accurate and relevant attributes (variables) but did not specify license.	Moderately

Processing

After data cleaning, the dataset contained 7679 entries and 28 variables. The cleaning process is described below in the subsections: ***Redundant Variables***, ***Data Type***, and ***Missing Values***.

Redundant Variables

The dataset variables included multiple violations of PHI according to HIPAA standards, such as GeoSpacing to below the state level for residency, death, and injury, as well as longitude and latitude information provided for these variables. Such violations to PHI were removed from the dataset. Moreover, due to the fact that the dataset is focused on the Connecticut region, there is very little reason to conduct analysis by state or region. Instead, analysis focused on age, gender, missing values, and descriptive statistics of each of the 17 drug types. Removed columns involved the following:

- 'residence city'
- 'residence county'
- 'residence state'
- 'death city'
- 'death county'
- 'location'
- 'location if other'
- 'injury city'
- 'injury county'
- 'injury state'
- 'manner of death'
- 'deathcitygeo'
- 'residencecitygeo'
- 'injurycitygeo'
- ‘other significant conditions’

Data Type

The data type of the columns ‘date’ and ‘age’ were changed to dtype = ‘datetime’ and dtype = ‘float’ respectively. The column names were changed to lower case, such that analysis would be more efficient.

Missing Values

The drug type columns (Specified in **Dataset → Source** section) originally only marked detected drug used with a ‘Y’ in the corresponding row, whereas the drug uses not detected or not present were simply left blank. However, this may not be an accurate representation of the data, as in missing value analysis, such blank cells would be interpreted as missing data and not ‘drug not used’. Therefore, all blank cells in the drug type column were filled with ‘N’.

Furthermore, the column ‘other significant conditions’ was removed because the column contained > 90% missing values (‘NaN’). This column did not have its blank cells filled in with ‘N’ as did the drug type columns because it is hard to intuitively determine whether blank cells represented the absence of other significant conditions or the lack of information collected, therefore, to avoid confusion, this column was removed.

Summary Statistics

The summary statistics analysis and some general data explorations were included within the web interface as the analysis options ‘Age’, ‘Age by sex’, and ‘Dataset View’, and they are reproduced below:

74.37% of cases were of men.

25.47% of cases were of women.

The other cases are of gender "Unknown".

The average age in dataset is 42.48 years old.

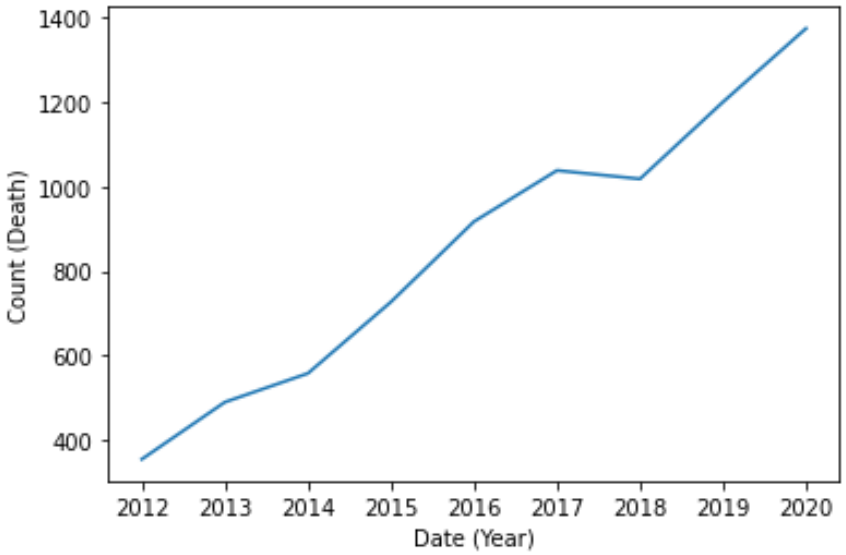

Figure 2. Time series of the number of accidental drug-related deaths in Connecticut from 2012-2020.

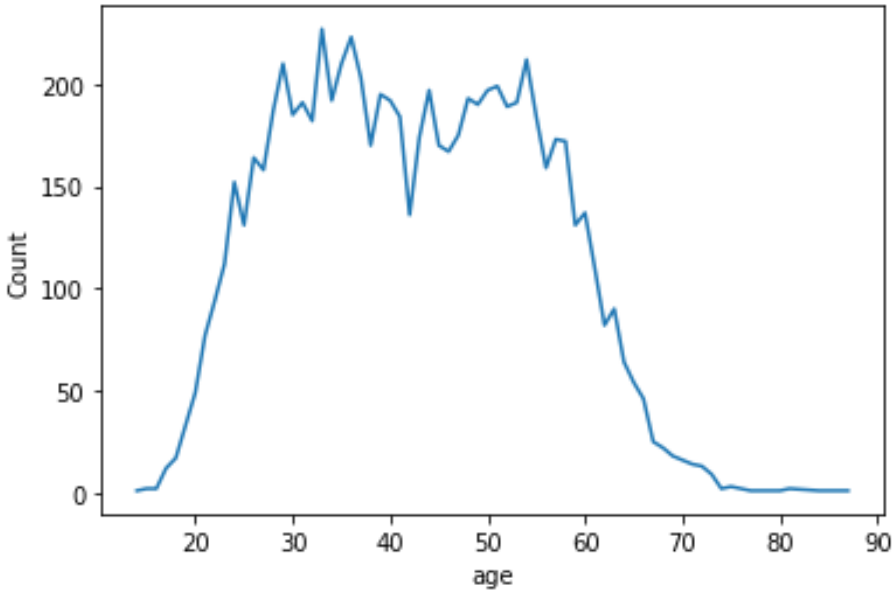

Figure 3. Number of accidental drug-related deaths by age in Connecticut from 2012-2020.

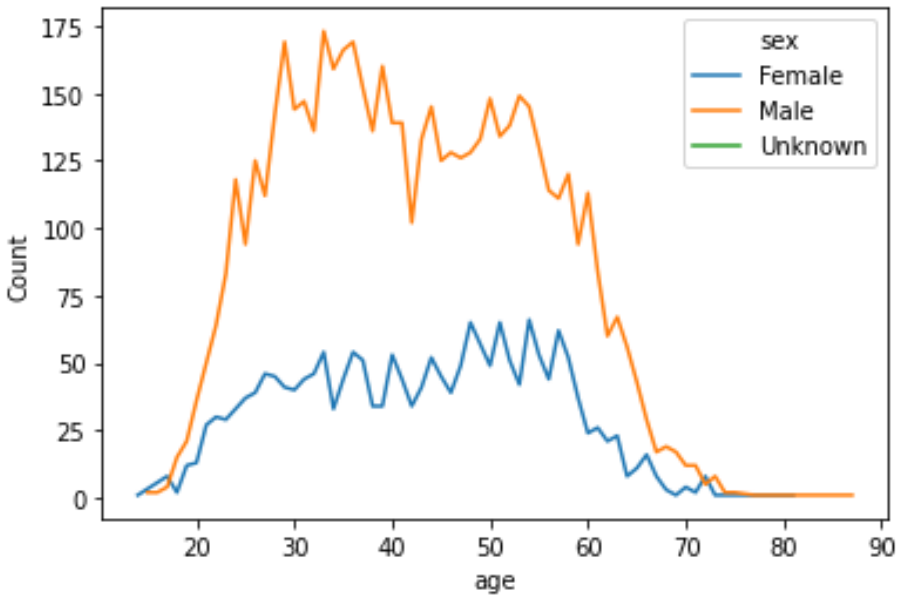

Figure 4. Number of accidental drug-related deaths by gender and age in Connecticut from 2012-2020. Blue line indicates female incidences, orange line indicates male incidences, and green (barely visible, towards older ages) indicates incidences of unknown gender.

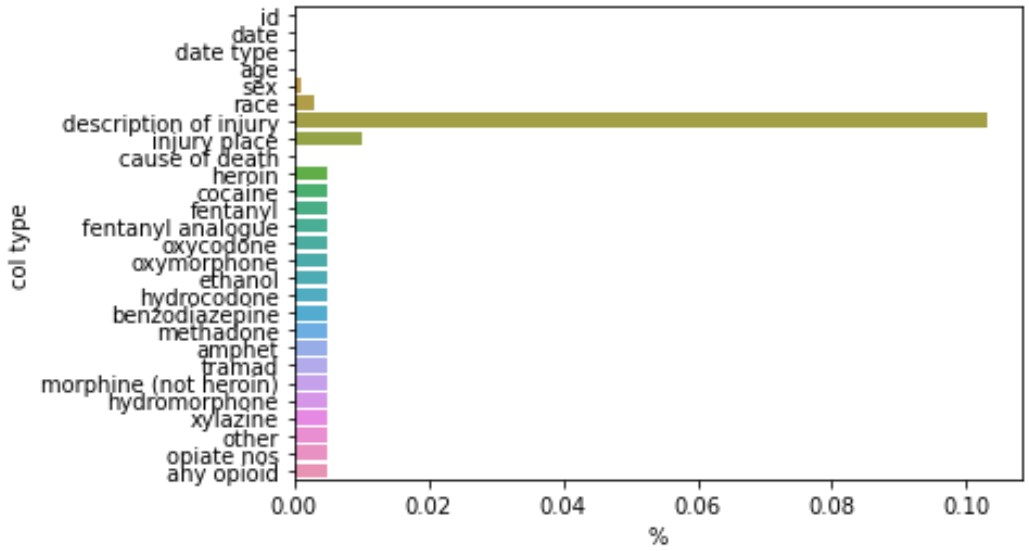

Figure 5. Percent of each column that are missing values. The column type is on the y-axis labeled col-type and the percentage of specific column that is missing values in on the x-axis.

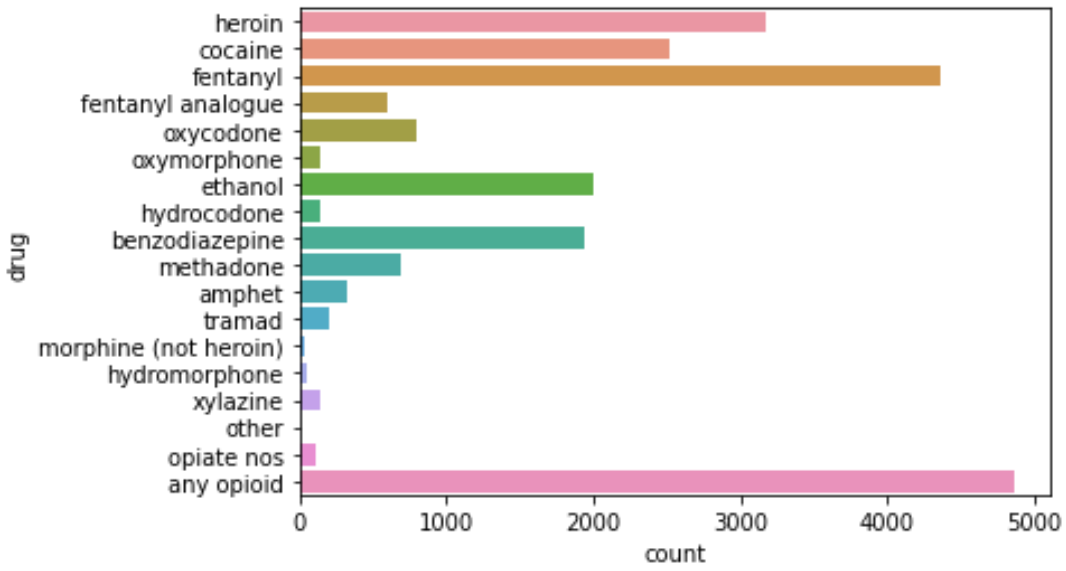

Figure 6. Number of accidental drug-related deaths by drug type in Connecticut from 2012-2020. The drug type is on the y-axis labeled col-type and the number of incidences in on the x-axis.

Web Interface

Design and Development

The goal that propelled the design and development of this web interface is to answer the following questions:

- What is the overall trend in accidental drug-related deaths in Connecticut from 2012 to 2020?
- What does the age distribution look like?
- How complete is the dataset?
- What is the distribution of missing data?
- Does age, gender, and race distribution look difference for different types of drug use?
- How many total cases for each drug?
- Which types of drug incidences are more frequent?

The web interface was designed using Flask and written in Python, HTML, CSS, and utilized Jinja2 templates. The main code could be found in the file app.py, and the html templates include: index.html (homepage), info.html (all analysis except dataset view), datasetview.html, and failure.html.

The Home page provides a screenshot (with clickable link to data source website) and a brief description of the data source with a clearly identified link to the data source website.

The webpage is designed to adapt to device width, such that even if one were to view the page on their phones, the words will not shrink to fonts so small that they are illegible.

All pages except the Home page includes a clickable link that redirects user to Home page.

Selection of invalid analysis options or options with no data available (e.g., 'Drug type: other') will redirect the user to failure.html.

All data processing and preparation for presentation were dynamically performed in the APIs in app.py, including index(), info(), by_age(), by_gender_age(), missing_vals(), show_drug_type_stats(drug), and dataset_view(analysis_type). The option to select a type of analysis from the 21 options listed below was sent from the Home page (index.html) to the View Information page (info.html) or processed by info() to be sent to datasetview.html if the selected option were Dataset View. The former then receives files sent from the corresponding APIs depending on chosen analysis type via the send_file() method, and displays the information in html format. The latter renders the datasetview.html page and displays information there instead of in the info.html page.

For more specific information on the processing of data, please refer to the code attached in the appendix section of this report.

Analysis

The analysis that are available to view on the web interface is the following:

- Home:
  - Time series plot of the number of incidences in Connecticut from 2012 to 2020
  - The number of incidences by drug type
- Age: A line plot of the number of incidences by age
- Age by sex: A line plot of the number of incidences by age and gender
- Missing values:
  - The total number of missing values in the cleaned dataset
  - A bar plot of the percent of each column that is occupied by missing values
- Dataset View:
  - The proportion of cases that were of each gender
  - The average age in the dataset
  - The complete cleaned dataset (table format)
- Drug type: {drug}
  - A labeled line plot of the age distribution of incidences for the selected drug
  - The number of total incidences in dataset related to selected drug
  - The number of male and female cases within the total incidence count
  - The average age
  - The most seen race

Results

The results of above analysis are described below:

- Home:
  - The overall trend in drug usage was a relatively consistent increase from 2012 to 2020, with a small dip in 2018 that conforms with the national trend in that year, and a significant increase in the years 2016-2017.
  - The most frequently seen drug in the dataset is opioid, which includes all analogues and derivatives of opioid drugs. The second most seen is fentanyl, and the third most seen is heroin.
- Age:
  - The age range in which most incidences occur is approximately 20-55.
  - Despite the average age in dataset being approximately 42 years old, there was a significant dip in the plot around age 42.
  - The plot is slightly skewed to the right, indicating that more young Americans in Connecticut have accidental drug-related deaths compared to older individuals.
  - There are very few incidences of people under age 20 or over age 65.
- Age by sex:
  - The age distributions for both genders are very similar to the overall age distribution and to each other.
  - There are significantly more incidences of men than of women.

    - The occurrence of incidences in the gender category ‘Unknown’ is insignificant, but surprisingly occurred in ages > 70 years.
- Missing values:
    - There is a total of 1,539 missing values in the cleaned dataset.
    - The most prominent proportion of missing values occurred within the column ‘description of injury’ at approximately 10%.
    - The drug type columns all consistently had 0.35% of their data missing.
    - The columns ‘id’, ‘date’, ‘age’, and ‘cause of death’ all had 0 missing values.
- Dataset View:
    - Almost three-quarters of the dataset are occupied by incidences of men, over one-quarter by incidences of women, and the rest are of gender "Unknown".
    - The average age in dataset is 42.48 years old.
    - The complete dataset without analysis is presented.
- Drug type: {drug}
    - The age distribution for most drug types had the mode age range of approximately 25-55.
    - Some age distributions (heroin, cocaine, fentanyl, oxycodone, ethanol, amphet, and any opioid) were slightly right skewed.
    - The number of total incidences in dataset related to valid drug type selection ranged from 40 to 4366.
    - Age data for drug types of fewer incidences tend to produce graphs that are not as smooth, and hence less informative of a clear trend. Perhaps due to this, some of them seemed left skewed (morphine (not heroin), tramad, methadone, and hydrocodone). However, if such results were accurate, then it implies that such incidences tend to occur among older individuals.
    - The number of male and female cases within the total incidence count varied by drug, but in general there were more male incidences than female incidences.
    - The average age for all drug types were slightly over 40 years old.
    - The most seen race is White for all drug type analyses, but this does not imply that White people are more likely to have such incidences compared to other races, but just that the base population is majority White, hence White people are statistically more likely to appear in this dataset.

Limitations

Some limitations and difficulties in the presentation and development of the web interface, as well as corresponding implemented or potential resolutions are listed below:

Limitation	Potential Resolution
The web interface made limited use of CSS, hence the design is not very aesthetically pleasing or fancy.	Implement more CSS code in html pages.
The failure.html page does not specify which kind of error.	Create separate html pages for different kinds of error.
There is a redundant 'date' text in the web interface's 'Dataset View' page, the source of which I cannot figure out.	Cannot figure out error: complete code earlier so there is time to consult with Professor.
The analyses were simple plotting and calculations.	Implement other potential analysis types such as KNN or PCA to predict 1) the age of incidence based on drug type or 2) other drugs used if given the use of a type of drug.
PHI data were dropped, and the resulting location information column consists mostly of value 'CT', which is not useful for spatial analysis.	Explore datasets that cover a wider geographic range.

Difficulty	Implemented Resolution
Generating and presenting graphs dynamically	Saved output graph in a folder named 'output' with a specific name for each type of analysis, such that multiple queries for the same analysis type would only result in one file for the latest query. File sent to info.html or datasetview.html via corresponding API.
Showing dataframe as table on webpage	Googling: https://stackoverflow.com/questions/52644035/how-to-show-a-pandas-dataframe-into-a-existing-flask-html-table
Presenting complete missing value analysis figure without having column names cut off	Googling and using bbox_inches = 'tight': https://stackoverflow.com/questions/45239261/matplotlib-savefig-text-chopped-off
Sending parameters to API	Add '/<parameter>' in route and include parameter in function.

Acknowledgements

I would like to thank Professor McDougal for spending an hour with me on zoom and responding quickly to a number of my emails crying for help on the final project. Thank you for your generous and extensive help, concern for our comfort with the pace and difficulty of class, and a challenging yet rewarding semester. I would also like to thank our wonderful teaching fellow Wenxin Xu for helping me on various homework assignments and a zoom meeting. I have learnt a great amount from this course and despite it being the most prominent contributor to making this an intimidating first semester at Yale Graduate School, I do not regret taking it and deeply appreciate the guidance and knowledge obtained during the course.

Finally, I will also acknowledge how stackoverflow.com and geeksforgeeks.com has helped answer a countless number of trivial problems that I have encountered throughout this final project. A few are listed below:

I. https://stackoverflow.com/questions/28207761/where-does-flask-look-for-image-files
II. https://www.google.com/url?sa=i&url=https%3A%2F%2Fgiphy.com%2Fgifs%2FYaleAlumni-transparent-yale-yalealumni-YlYX7mrtL1R4Z5huWJ&psig=AOvVaw2a6huJn2rxb6Yo8GubeCVx&ust=1640110782628000&source=images&cd=vfe&ved=0CAsQjRxqFwoTCMDxoPf-8vQCFQAAAAAdAAAAABAD
III. https://www.rawpixel.com/image/1201197/pink-watercolor-background
IV. https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.istockphoto.com%2Fvector%2Fwhite-exclamation-mark-symbol-on-red-circle-caution-icon-isolated-on-white-gm1248649489-363706105&psig=AOvVaw384yr0anRu2wlW5SnvqxiB&ust=1640113583889000&source=images&cd=vfe&ved=0CAwQjhxqFwoTCKjqna-J8_QCFQAAAAAdAAAAABAD
V. https://stackoverflow.com/questions/52644035/how-to-show-a-pandas-dataframe-into-a-existing-flask-html-table
VI. https://stackoverflow.com/questions/45239261/matplotlib-savefig-text-chopped-off
VII. https://stackoverflow.com/questions/35277075/python-pandas-counting-the-occurrences-of-a-specific-value
VIII. https://stackoverflow.com/questions/42406233/how-to-add-title-to-seaborn-boxplot
IX. https://datascienceparichay.com/article/pandas-extract-year-from-datetime-column/
X. https://python-forum.io/thread-33935.html

References

Data.ct.gov. (2020, November 12). *Accidental Drug Related Deaths 2012-2020*. Retrieved from https://catalog.data.gov/dataset/accidental-drug-related-deaths-2012-2018.

GO FAIR. (2021, October 18). *Fair principles*. GO FAIR. Retrieved December 21, 2021, from

Institute of Medicine (US) Committee on Opportunities in Drug Abuse Research. (1996, January 1). *Drug abuse research in historical perspective*. Pathways of Addiction: Opportunities in Drug Abuse Research. Retrieved December 22, 2021, from https://www.ncbi.nlm.nih.gov/books/NBK232965/

NCDAS. (2021, September 25). *Substance abuse and addiction statistics [2021]*. Drug Abuse Statistics. Retrieved December 18, 2021, from https://drugabusestatistics.org/

National Institute on Drug Abuse. (2021, February 25). *Overdose death rates*. National Institute on Drug Abuse. Retrieved December 20, 2021, from https://www.drugabuse.gov/drug-topics/trends-statistics/overdose-death-rates

APPENDIX

CONTENTS

```python
# import libraries
import pandas as pd
import seaborn as sns
import numpy as np
from statistics import mode
from matplotlib import pyplot as plt
from flask import Flask, render_template, redirect, request, url_for, jsonify, send_file
import requests
import jinja2
import tempfile
from pandas.plotting import table
import uuid

# import data
# data retrieved from  https://catalog.data.gov/dataset/accidental-drug-related-deaths-2012-2018
# Public Access: This dataset is intended for public access and use
# Publisher: data.ct.gov
data = pd.read_csv("Accidental_Drug_Related_Deaths_2012-2020.csv")
data.columns= data.columns.str.lower()

# ____________________________________________ DATA PREPARATION
# ____________________________________________

# drop redundant variables
# set to lower case column names
data = data.drop(columns=['residence city', 'residence county', 'residence state', 'death city', 'death county', 'location', 'location if other', 'injury city', 'injury county', 'injury state', 'manner of death', 'deathcitygeo', 'residencecitygeo', 'injurycitygeo'])

# set type of variable of date column to Datetime
data['date'] = pd.to_datetime(data['date'])

# clean dataset - NaN values in type of drug columns are to be interpreted as 'N' for No and not missing
# only keep NaN if all entries in row for columns in type_of_drug is NaN
type_of_drug = data[["heroin", "cocaine", "fentanyl", "fentanyl analogue", "oxycodone", "oxymorphone", "ethanol", "hydrocodone",   "benzodiazepine", "methadone", "amphet", "tramad", "morphine (not heroin)", "hydromorphone", "xylazine", "other", "opiate nos",   "any opioid"]]
row = 0
col_num = len(type_of_drug.columns)
while row < len(type_of_drug):
    num_nan = type_of_drug.iloc[row].isna().sum()
    if num_nan != col_num: type_of_drug.iloc[row].fillna("N", inplace=True)
    row += 1
data[["heroin", "cocaine", "fentanyl", "fentanyl analogue", "oxycodone", "oxymorphone", "ethanol", "hydrocodone",   "benzodiazepine", "methadone", "amphet", "tramad", "morphine (not heroin)", "hydromorphone", "xylazine", "other", "opiate nos",  "any opioid"]] = type_of_drug

# generate dataframe image (done in Data Exploration file)
# ax = plt.subplot(111, frame_on=False)
# ax.xaxis.set_visible(False)
# ax.yaxis.set_visible(False)
# table(ax, data)
# plt.savefig('/static/data.png')

average_age = round(np.mean(data['age']), 2)
percent_male = round(len(data[data['sex']=='Male'])/len(data), 4)*100
percent_female = round(len(data[data['sex']=='Female'])/len(data), 4)*100

# ____________________________________________ END DATA PREPARATION
# ____________________________________________



# ____________________________________________ END API FUNCTIONS
# ____________________________________________
# ____________________________________________ FLASK FUNCTIONS
# ____________________________________________
```

```
59
60 # global variables to be used in html pages
61 # select options for generating and viewing analysis
62 drug_types = ["heroin", "cocaine", "fentanyl", "fentanyl analogue", "oxycodone", "oxymorphone", "
   ethanol", "hydrocodone",    "benzodiazepine", "methadone", "amphet", "tramad", "morphine (not heroin)"
   , "hydromorphone", "xylazine", "other", "opiate nos",   "any opioid"]
63 analysis_options = ["Drug type: " + i for i in drug_types]
64 analysis_options = ["Age", "Age by sex", "Missing values", "Dataset View"] + analysis_options
65
66 # implement a server that provides three routes using flask
67 app = Flask(__name__)
68
69 # the index/homepage written in HTML which prompts the user to select an item for analysis
70 # and provides a button, which passes this information to /info
71 @app.route("/")
72 def index():
73     return render_template("index.html", analysis_options=analysis_options, drugcount='/static/
   drug_count.png', trend='/static/trend.png', gif='/static/yalegif.gif', img='/static/datagov.PNG')
74
75 # finding relationship between age and accidental drug-caused death
76 @app.route("/age")
77 def by_age():
78     analysis_type = request.args.get("Actions")
79     print(analysis_type)
80     ages = data.groupby('age')['id'].count().reset_index()
81     plt.figure()
82     age_plt = sns.lineplot(data=ages, x='age', y='id')
83     age_plt.set(ylabel='Count')
84     path = 'output/age.png'
85     plt.savefig(path)
86     return send_file(path, mimetype='image/png')
87
88 # find relationship between age and accidental drug-caused death by gender
89 @app.route("/genderage")
90 def by_gender_age():
91     age_gender = data.groupby(['sex', 'age'])['id'].count().reset_index()
92     age_gender = age_gender[age_gender.sex != 'unknown']
93     plt.figure()
94     age_gender_plt = sns.lineplot(data=age_gender, x='age', y='id', hue='sex')
95     age_gender_plt.set(ylabel='Count')
96     path = 'output/genderage.png'
97     plt.savefig(path)
98     return send_file(path, mimetype='image/png')
99
100 # count missing values and the proportion of each column that is missing
101 @app.route("/missingvals")
102 def missing_vals():
103     nan_dict = {}
104     cols = data.columns
105     tot = 0
106     for col in cols:
107         nan_dict[col] = [data[col].isna().sum(), round(data[col].isna().sum()/len(data[col]), 3)*100]
108         tot += data[col].isna().sum()
109     title = 'Total: ' + str(tot)
110     # plot missing values onto chart
111     missing_df = pd.DataFrame.from_dict(nan_dict, orient='index').reset_index()
112     missing_df.columns = ['col type', 'crude count', '%']
113     plt.figure()
114     missing_plt = sns.barplot(data=missing_df, x='%', y='col type').set_title(title)
115     path = 'output/missingvals.png'
116     plt.savefig(path, bbox_inches='tight')
117     return send_file(path, mimetype='image/png')
118
119 # select type of drug used and show stats
120 @app.route("/drugtype/<drug>")
121 def show_drug_type_stats(drug):
122     selected_drug = data[data[drug] == "Y"]
123     # age distribution
124     age = selected_drug.groupby('age')['id'].count().reset_index()
```

```
125     plt.figure()
126     age_plt = sns.lineplot(data=age, x='age', y='id')
127     age_plt.set(ylabel='Count')
128     age_plt.set(title=drug)
129     path = 'output/drugtype.png'
130     plt.savefig(path)
131     return send_file(path, mimetype='image/png')
132 
133 # view descriptive statistics and cleaned dataset
134 @app.route("/datasetview/<analysis_type>")
135 def dataset_view(analysis_type):
136     return render_template('datasetview.html', analysis_type=analysis_type, tables=[data.to_html(
    classes='data')], titles=data.columns.values, female=percent_female, male=percent_male, aa=average_age
    )
137 
138 # a web page that takes the name of the state as a GET argument and
139 # (1) if one item is selected, displays the same information as the API 'view' in an HTML page
140 # or (2) displays an error page if none chosen
141 # includes a link back to the homepage
142 @app.route("/info", methods=["GET"])
143 def info():
144     analysis_type = request.args.get("Actions")
145 
146     # check if selected option is valid
147     if str(analysis_type) in analysis_options:
148         # initialize variables
149         isdrug = False
150         fp = "/"
151         tc = ""
152         mc = ""
153         fc = ""
154         aa = ""
155         rm = ""
156 
157         # if chosen to analyze a type of drug
158         if str(analysis_type).split(" ")[0] == "Drug":
159             isdrug = True
160             drug = str(analysis_type).split(": ")[1]
161             drug = drug.lower()
162             fp = f'/drugtype/{drug}'
163             selected_drug = data[data[drug] == "Y"]
164             total_count = len(selected_drug)
165 
166             # if no data available render failure page
167             if total_count == 0:
168                 return render_template("failure.html")
169 
170             # if drug type data is available
171             male_count = len(selected_drug[selected_drug["sex"] == 'Male'])
172             female_count = len(selected_drug[selected_drug["sex"] == 'Female'])
173             average_age = np.mean(selected_drug['age'])
174             race_mode = mode(selected_drug['race'])
175 
176             # upating strings
177             tc = "There were " + str(total_count) + " incidences related to "+str(drug) + "."
178             mc = "Of which, the number of males is " + str(male_count) + ";"
179             fc = "The number of females is " + str(female_count) + "."
180             aa = 'The average age is ' + str(average_age) + "."
181             rm = 'The most seen race is ' + str(race_mode) + "."
182         # view descriptive statistics and complete dataset (cleaned)
183         elif str(analysis_type) == "Dataset View":
184             return dataset_view(analysis_type)
185 
186         # missing values
187         elif str(analysis_type) == "Missing values":
188             fp = '/missingvals'
189 
190         # analysis by age
191         elif str(analysis_type) == "Age":
```

```
            fp = '/age'

        # by gender age is only valid option left
        else:
            fp = '/genderage'
        return render_template("info.html", analysis_type=analysis_type, fp=fp, isdrug=isdrug, tc=tc,
mc=mc, fc=fc, aa=aa, rm=rm)

    # analysis type not included in choices: failure
    return render_template("failure.html")

# ____________________________________________ END FLASK FUNCTIONS
____________________________________________

if __name__ == "__main__":
    app.run(debug=True)

```

```
<!DOCTYPE html>

<html lang="en">
    <head>
        <meta charset="UTF-8" name="viewport" content="initial-scale=1, width=device-width">
        <title>Final Project BIS634 Fall 2021 | Olina Zhu HOMEPAGE</title>
    </head>
    <body background="/static/background.webp">

    <p style="font-size:50px">
        <h2><center>Welcome to the Homepage for Final Project BIS634 Fall 2021 | Olina Zhu <img src={{
gif}} height="80"> </center></h2>
    </p>
        <b>This webpage utilizes the dataset "accidental-drug-related-deaths-2012-2020" found <a href="
https://catalog.data.gov/dataset/accidental-drug-related-deaths-2012-2018">here.</a> </b>
        <br>
    <br> <b>The publisher is data.ct.gov and it is a public access dataset - intended for public access
 and use.</b><br>
        <br>
        <br>
        <a href="https://catalog.data.gov/dataset/accidental-drug-related-deaths-2012-2018">
        <center><img src={{img}} height="500" width="475" style="border:5px solid black"></center>
        </a>
        <br>
        <br>
        <br>
       <center><b>Please choose from below items to generate and view analysis.</b></center>
        <br>
        <br>
        <form action="/info" method="get">
            <select name="Actions">
                <option value="" disabled selected>Select From Below</option>
                {% for i in analysis_options %}
                    <option>{{ i }}</option>
                {% endfor %}
            </select>
            <input type="submit" value="Generate and View">
        </form>
        <p>
   <center><b>Plots of accidental drug-related deaths time series and drug use by type in 2012-2020 in
Connecticut is shown below.</b></center>
   <br>
   <center><img src={{trend}} height="400">
   <br>
    <br>
       <img src={{drugcount}} height="400"><br></center>
        </p>
    </body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <style>
      h2{
        background-color: bisque;
      }
    </style>
    <meta charset="UTF-8" name="viewport" content="initial-scale=1, width=device-width">
    <title>View Information</title>
  </head>
  <body background="/static/background.webp">
    <h2><center><b>You selected the analysis type: {{ analysis_type }}.</b></center></h2>
    <br>
    <br>
    <b>The results for analyzing {{ analysis_type }} is as follows</b>
      <p>
        <center><img src="{{fp}}" width="device-width" style="border:2px beige"></center>
      </p>
      {% if isdrug == True %}
        <center><b>
        {{ tc }}<br>
        {{ mc }}<br>
        {{ fc }}<br>
        {{ aa }}<br>
        {{ rm }}<br>
        </b></center>
      {% endif %}
      <br>
      <br>
      <b><a href="/">Click Here to Return Home</a></b>
  </body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <style>
        h2{
            background-color: bisque;
        }
    </style>
    <meta charset="UTF-8" name="viewport" content="initial-scale=1, width=device-width">
    <title>View Information</title>
</head>
<body background="/static/background.webp">
<h2><center><b>You selected the analysis type: {{ analysis_type }}.</b></center></h2>
<br>
<br>
<b><a href="/">Click Here to Return Home</a></b>
<br>
<br>
<br>
<b>The results for analyzing {{ analysis_type }} is as follows</b>
<p>
<center>
    <br>
    <b>{{male}}% of cases were of men.</b><br>
    <b>{{female}}% of cases were of women.</b><br>
    <b>The other cases are of gender "Unknown".</b><br>
    <b>The average age in dataset is {{aa}} years old.</b></center>
<br>
<br>
<br>
<b>The Complete Dataset Is Shown Below</b>
    {% for table in tables %}
    {{titles[loop.index]}}
    {{ table|safe }}
    {% endfor %}
</center>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8" name="viewport" content="initial-scale=1, width=device-width">
        <title>Failure</title>
    </head>
    <body background="/static/background.webp">
    <h3>You have failed to access requested information or there are no information available for your selection.</h3>
        <br>
    <b>Please return to the <a href="/">Home Page</a> and try again with a different analysis option.</b>
    </body>
</html>
```