Email: derek@cassadyweb.com
Name: Derek Cassady
Program: Data Analysis
Class: Indiana Cohort

Census Bureau Data

Statement:

With a background and degree in Occupational Safety, Health, and Environmental, I recognize the need and lack of data analysis and science devoted to these departments in many organizations. Organizationally speaking, these are cost centers and often not seen as value added to the bottom lines of quality and productivity.

While I would tend to disagree, many professionals (including myself) in the field do not have the data analytics background to prove their need for full inclusion into the decision making processes found in most businesses. While it was my professional goal to be instrumental to my business partners in all departments by understanding their operational needs, methods, and workflow, I lacked the full toolkit.

The goal of this project, which will be a continuous work for the short-term future, is to build the skill set needed to produce quality and engaging dashboards and visualizations. The need by safety professionals to manipulate data during incident investigations and business planning is vital to maintenance of a safe working environment and strong relationships with business partners.

Project Description:

Retrieve data from the US Census Bureau through API connection and display the information in a choropleth map through an interactive web based dashboard.

This base of data manipulation and mapping for age and race will allow for expansion of the project to include many other data groups like: age, education, income, health outcomes, etc. Along with these expansions, the need for main.py with calls to modules and expansion of the interface and visualizations for the dashboard will be needed.

However, we walk before we run.

Data sources:

Census Bureau piblic API root: https://api.census.gov/data
API key requests: https://api.census.gov/data/key_signup.html
GEOJSON: https://raw.githubusercontent.com/plotly/datasets/master/geojson-counties-fips.json

Libraries of Note:

Requests & Urllib:
For the import of data from the web, Requests and Urllib are used. Requests was used in the last project with Bureau of Labor Statistics and Urllib is used here to pull a directly from a url .json for country FIPS and location information.

Kaleido:
Since Plottly Express and Dash are used for this, Kaleido is the appropriate choice for any static image production. Kaleido is quicker and shows more overall compatibility than orca.

Plottly Express and Dash:
Having a great interest in making dashboards that can be quickly updated for operational environments, the Plottly Express and Dash combination seemed to fit the goal. Visualizations created by these tools have a intractability for the end user that is much greater than other Python visualization tools found.

Openpyxl:
Being able to offer the data back into a format that most people can manipulate is important, so expanding my use and understanding of what can be done with Openpyxl seemed to be a good choice for this project.

Features Included from Requirements list:

Feature:

Read data in.
- Read in data from a local csv, excel file, json, or any other file type. There are many ways to do this, but using Pandas read_ functions is pretty easy.

- Use an API to pull in data. The most common library for this is "requests", while pulling the data itself is usually pretty easy, sometimes going through the specific API documentation can be kind of complicated.

Manipulate and clean your data.
- Use built-in pandas or numpy functions to do things like remove 0's and null values where they don't belong in your dataset.

Analyze your data!
- Use at least 5 different built-in Python functions to find out something about your data. If you had a list for example, finding the length of that list with len(<list>) does tell us a little bit about the data.

- Do 5 basic calculations with Pandas, like finding the sum(), median(), mean(), or mode() of a column. You could divide two columns by each other. You could multiple a column by a random integer. You could use string operations and find the most common letter in a given entry.

Visualize your data.
- Make 2 basic plots with matplotlib, seaborn, or any other kind of visualization library that you think looks interesting.

- Make a website and visualize your data with a framework like django or flask. This is definitely not necessary but some students in the past have done this and made really amazing projects, if you're looking for a challenge.

Interpret your data and graphical output.
- Write markdown cells in Jupyter explaining your thought process and code. If you make a few plots with matplotlib, explain what the reader is seeing and why you chose to plot things that way.

Visualization 1:

This is a static .png image created using Kaleido as the engine with the Plottly visualization. While Plottly is interactive, alas, a static image is not. An interactive version of this map is found within the project containing data from one source for the population and a second for the hover data.

Mapping data itself is provided through a GEOJSON file that feeds the visualization creation

Population by County