Purple Air Data

Documentation

Project Overview

Yellow = need to update

Context

- Code for Chicago wants to drive better outcomes for our city

Hypothesis

- We hypothesize that local (neighborhood-level) air data differ meaningfully from the EPA (AirNow) data, which are only present in x (?) locations in the Chicago area
- This implies:
    - Policy is designed based on incomplete information
    - We may be leaving out key populations from appropriate air quality warnings
    - We may need to invest in more sensors

Methodology

- We use local air sensors (PurpleAir) to show they differ significantly from the air quality index (AQI) levels provided by the EPA

Data

- Purple Air data - thingspeak API
- EPA Data - Airnow API

Outcomes

- We aim to partner with a local organization and advocate for investment in air quality sensors
- We hope to generate fact-based analysis for the need of these sensors

The analysis is a three step process

We hypothesize that local air conditions differ from the EPA (AirNow) data, so we need more sensors

1. Download Purple Air Data
2. Download AirNow data (if needed)
3. Conduct correlation analysis of the data

1. Downloading Purple Air Data

PurpleAir Parameters overview

Parameter to place in pa_sensors.json highlighted in blue

STEP	DESCRIPTION	NOTES
Identify	● Choose the neighborhood to run analysis on	● A neighborhood may not have any PurpleAir sensors
Browse	● Visit purple air’s website and use the interactive map to identify any sensors in the target neighborhood ● A sensor appears as a bubble - yellow, green or orange	
Inspect	● Open your developer tools on Chrome by pressing “Ctrl + Shift + C” or “Command + Shift + C” on Windows and Mac respectively ● Click on “Network” which will show you all the HTTP requests sent by your browser	● This is how your browser is communicating with servers under the hood
Click	● Click on a sensor you want details for	●
Gather	● In the browser URL at the top, you will see a segment like “?select=” which is followed by your **“SensorID”** ● You will also observe two few requests in the Networks tab that start with “.csv?start=”... ○ You may have to refresh a few times and look at the timing of those requests right after you click on the bubble ○ This is the browser using the Thingspeak API to retrieve data - and you must recreate this URL in the notebook to receive the data ○ You need to note a few parameters here: “**Thingspeak ID**” which is the code after “channels/” and “**Thingspeak Key**”, which is the alphanumeric code after “api_key=” ○ Please note that there are two API calls for Sensor A and Sensor B, with different parameters ○ The request that appears first is Sensor A and the one that appears slightly later is Sensor B	● Example URL for Evanston: https://map.purpleair.com/1/mAQI/a10/p604800/cC0?select=**119791**#10.57/41.9601/-87.654 ● Example API for 119791: https://api.thingspeak.com/channels/**1472157**/fields/8.csv?start=2021-12-08%2020:24:43&average=10&round=2&days=4&api_key=**5YLLDB7EWIG3FSMH**

PurpleAir Parameters overview

STEP	DESCRIPTION	NOTES
Analyze	• Create a copy of the "1.1-purpleair-data-extraction.ipynb" and use that in conjunction with the pa_sensors.json to make necessary edits and retrieve the data	• Be sure to create a copy!!