**Steps for interfacing with PurpleAir API for retrieving station information, downloading data, and data engineering and analysis using PurpleAir API tool, Python and R**

**All current station data and referenced files can be found at the links below.**

**Git-hub repo:** [**https://github.com/Theodros-Woldeyohannes/INBRE**](https://github.com/Theodros-Woldeyohannes/INBRE)

**Google drive:** [**https://drive.google.com/drive/folders/1r5j9XYP0kJngjeCqoSqXr5drvV8b9IJg?usp=sharing**](https://drive.google.com/drive/folders/1r5j9XYP0kJngjeCqoSqXr5drvV8b9IJg?usp=sharing)

**PurpleAir API Key:** 43418ADC-5C99-11EE-A77F-42010A800009

**Links to Montana Mesonet MET stations**

**Wyola:** [**https://mesonet.climate.umt.edu/dash/unmwyola/**](https://mesonet.climate.umt.edu/dash/unmwyola/)

**Pryor:** [**https://mesonet.climate.umt.edu/dash/unmpryor/**](https://mesonet.climate.umt.edu/dash/unmpryor/)

**Read-me on repo file contents:**

* PurpleAir Download 12-3-2024 (folder containing PurpleAir data, last download 12/3):
  + Contains separate .csv files of raw PurpleAir data, one for each sensor
* Scripts (folder containing example script files using in data engineering, analysis, and plotting):
  + Data\_analysis\_V3.ipynb – Python notebook with examples for PurpleAir data summary, analysis, and plotting function
  + NewChangesForAllPlots.py – Python script for plotting PM time-series data
  + PA\_Data\_combiner.qmd – R script for formatting and combining sensor data into single table
  + Time\_agg.qmd – R script with examples for time-weight averaging PM data
  + cal\_data\_format.qmd – R script with examples for pairwise and timeseries comparison of data from co-located sensors for calibration purposes
  + get\_station\_location.py – Python script for pulling sensor location information from PurpleAir API
  + retrieve\_all\_data.py – Python script for downloading PurpleAir data by directly interfacing with API
* Stations (folder with metadata on stations)
  + StationList\_2024.xlsx – Station location and metadata
* all\_data\_12324.csv – Formatted station data (last download 12/3)

**EPA PM2.5 Fact Sheet:** [**https://www.epa.gov/system/files/documents/2024-02/pm-naaqs-monitoring-fact-sheet.pdf**](https://www.epa.gov/system/files/documents/2024-02/pm-naaqs-monitoring-fact-sheet.pdf)

**Retrieve station coordinates**

I have written a python script to retrieve station coordinates. Please refer to script file *get\_station\_location.py*

PurpleAir sensors are geolocated in WGS84

Station IDs, info, and location can be found in the spreadsheet *StationList\_2024.xlsx*

**Download station data**

Station data can be downloaded using the PurpleAir Data Download Tool. The tool can be installed here: <https://community.purpleair.com/t/purpleair-data-download-tool/3787>

1. Open program. Go to API Keys on the left side bar. Input our API key under *API Read Key*.

A screen shot of a computer

Description automatically generated

1. Go to *Download*. Enter for the following fields:

*Sensor Indexes:* 176247, 176209, 185075, 176215

*Timezone:* America/Denver

*Desired start and end time* (Our sensors were installed on 7/22/24)

*Desired Average (I usually do raw data at 0 seconds, but if you would like it averaged there are many options available)*

*Fields:* choose desired fields. I usually do the following:

* Humidity
* VOC
* pm1.0\_atm, pm2.5\_atm, pm10.0\_atm

A screenshot of a computer

Description automatically generated

1. Select directory to save data and click *Get Data* to start download. You will see a progress window. Depending on internet connection it will take a few minutes to download all data.

A screenshot of a computer

Description automatically generated

1. Data will be in .csv format, with a separate table for each sensor.

A black background with white numbers

Description automatically generated

I have also written a python script for interfacing with the PurpleAir API directly through HTTPS requests. Please refer to script file *retrieve\_all\_data.py*

**Format and combine station data**

I have written an R script for formatting and combining downloaded data into a single .csv. Please refer to script PA\_Data\_combiner.qmd

**Time series plots**

I have written a python script for creating time-series plots of PM2.5 against the EPA PM2.5 24hr standard. Please refer to script file *NewChangesForAllPlots.py*

A graph of a graph

Description automatically generated with medium confidence

See *Scripts* folder for additional scripts with plotting and other analysis functions.