Mathematics Clinic

Final Report for
South Gate CEHAT

Community Air Quality Monitoring and Analysis

May 6, 2021

Team Members
Adam Grobman
Carson Herness (Project Manager, Fall)
Amber Kampen
Lotenna Nwobbi
Hillary Rodriguez (Project Manager, Spring)

Advisor
Julie Medero

Liaisons
Victor Ferrer
Samir Patel
Liz Ruiz

Abstract

We collaborated with the Community Environmental Health Action Team (CEHAT) of South Gate, CA to create an air quality website and analyze air quality data. The website is accessible to non-technical audiences in both English and Spanish. Our work can be used to inform the CEHAT, regulators, and South Gate residents about local air quality.

Contents

List of Figures

List of Tables

Acknowledgments

In addition to our wonderful Advisor and Liaisons, we are especially thankful to the following individuals:

- Romie Duarte, Community Involvement Coordinator at the United States Environmental Protection Agency (EPA), for assistance in finding official EPA resources in Spanish and for helping test the functionality of the air quality dashboard
- Lelia Hawkins, Associate Professor of Chemistry at Harvey Mudd College, for insights on analysis techniques relevant to air pollution, her area of research
- Rossmery Zayas, SELA (South East Los Angeles) Youth Organizer at Communities for a Better Environment (CBE), for help with community outreach in South Gate and for finding local students to test our website

Their contributions helped make this project possible.

Chapter 1

Executive Summary

The City of South Gate is a community of about 100,000 residents in Southeast Los Angeles County. In addition to environmental conditions that impact all of Southern California, contaminated land, manufacturing plants, oil and gas wells, and transportation corridors worsen the environmental health of South Gate. Out of these concerns, the South Gate Community Environmental Health Action Team (CEHAT) formed to improve the environmental health and safety of South Gate. In recent years, the CEHAT has been focusing on air quality, with a special focus on fine particulate matter ($PM_{2.5}$).

Our work with the CEHAT this year has been focused on using their PurpleAir sensors to provide information and insight into the air quality in South Gate. We have done so along two major axes: an air quality dashboard website for residents of the city and a data analysis application for the CEHAT to use to gain further insight into South Gate's air quality. The air quality dashboard clearly and cleanly presents current and recent (past 24 hours) air quality data in both English and Spanish. The data analysis application allows the CEHAT to compare and contrast air quality in different parts of South Gate and across Southern California, which they can use to better inform legislators and regulators.

In the chapters that follow, we provide a brief overview of South Gate and air quality, describe the functionality and technology of both the dashboard and the data analysis application, and provide suggestions for future work. We also provide user guides to both the administrative features of the dashboard and the data analysis application so that the CEHAT can easily use these programs in the future.

Chapter 2

Introduction

In this section, we introduce the community of South Gate and the CEHAT's relationship to the community. From there, we establish the basics of air quality and the sensors we used. We also outline the goals of our project.

2.1 The City of South Gate

With just under 7.5 square miles in total area and close to 100,000 residents, South Gate is a densely populated city in Los Angeles County (City of South Gate, n.d.; Los Angeles County Department of Public Health, 2018). The historically industrial economy contributes to environmental health and justice concerns in this predominantly Latine community. Ninety-five percent of the population is Latine, and many residents are monolingual Spanish speakers. There are three Environmental Protection Agency (EPA) designated Superfund sites in the City of South Gate: Cooper Drum, Jervis Webb, and Southern Avenue Industrial Area (United States Environmental Protection Agency, n.d.c). These sites—by virtue of their designation by the EPA as Superfund sites—are among "the nation's most contaminated land," posing risks to human health and the environment.

In addition to these sites, there are many other manufacturing plants still operating in or near the city that impact the community's health. The city is also home to two oil and gas wells, potentially exposing residents to the toxic chemicals contained within the products and byproducts of these facilities (Los Angeles County Department of Public Health, 2018).

South Gate's location along high-traffic transportation corridors also affects the environmental health of the community. Both Interstates 105

(Glenn Anderson Freeway) and 710 (Long Beach Freeway) run alongside and through the City. The 710 alone carries an average of over 20,000 trucks and other diesel vehicles through the City of South Gate daily (California State Transportation Agency, 2016). In addition to high truck traffic, the Alameda Corridor—a freight rail network connecting the Ports of Long Beach and Los Angeles to Downtown LA—runs through South Gate, sending an average of 30 to 40 trains through the city each day in recent years (excluding 2020 and 2021 due to the COVID-19 pandemic) (Alameda Corridor Transportation Authority, 2021). These transportation corridors make South Gate particularly vulnerable to primary pollutants such as nitrogen oxides (NO_x) which contribute heavily to the aerosolization of particulate matter of diameter less than 2.5 microns, commonly referred to as $PM_{2.5}$ (Clean Air Technology Center, 1999).

2.2 The CEHAT

Concerns about the conditions mentioned in Section 2.1, among others, led to the formation of our sponsor, the South Gate Community Environmental Health Action Team (CEHAT), a community organization comprised of "city residents, business owners, community activists, members of civic groups, and state, local and federal government" dedicated to "[making] a South Gate that is beautiful, safe, educated, represented, healthy, sustainable, prosperous, clean, and united" (Centers for Disease Control and Prevention, 2016). Over the years, the CEHAT has worked to prioritize the community's environmental health concerns and implement an action plan to address these concerns.

As will be relevant in Section 3.5.2, the CEHAT is not a legally recognized organization and therefore does not have tax-exempt, non-profit status. While this reflects the CEHAT's grassroots nature, it can pose difficulties in funding and renders them ineligible for non-profit discounts.

The CEHAT was the first group in California to utilize the Protocol for Assessing Community Excellence in Environmental Health assessment to evaluate environmental health and justice (Centers for Disease Control and Prevention, 2016). In more recent years, the group's focus has been on air quality. They have tackled air quality through endeavors such as educational workshops, collaborating with schools on an air quality flag program, and working on Assembly Bill 617 (AB617) measures to reduce emissions in and around South Gate. This Clinic project serves as an extension of these efforts.

2.3 Air Quality

The CEHAT has been focusing on air quality in South Gate because air pollution can seriously impact health. Under the Clean Air Act, the EPA sets air quality standards for pollutants such as carbon monoxide, ground-level ozone, nitrogen dioxide, particulate matter, and sulfur dioxide (United States Environmental Protection Agency, n.d.a). To clearly communicate air quality information for each of these pollutants on a common scale, the EPA developed the Air Quality Index (AQI). Running from 0 to 500, the AQI indicates how good or bad the air is, with higher numbers meaning higher levels of pollution and greater health risk (AirNow, n.d.a). An AQI of 100 represents the national standard for air quality. Values below 100 are considered safe, whereas values above 100 mean the air is unhealthy, at first only for sensitive groups but as the AQI goes higher, for everyone. In rare cases, such as during wildfires, the air may be of such poor quality that the AQI exceeds 500. The EPA considers this "beyond the AQI."

The EPA's AQI formula for particulate matter (such as $PM_{2.5}$) is based on a 24-hour average (AirNow, n.d.b). However, this 24-hour average means that it is primarily useful for historical understanding of air quality rather than as an in the moment tool for people to use to adjust their habits. For this reason, the EPA has developed and published the NowCast formula AirNow (2020). The NowCast formula uses smaller periods of time to produce more up to date AQI information that the public can use in the moment to make decisions about their activities. During periods of high air quality volatility, the NowCast formula uses only more recent data, whereas when the air quality is more consistent, it uses up to 12 hours of particulate matter data.

The CEHAT is particularly concerned with $PM_{2.5}$ because these fine particles can enter the lungs and bloodstream, causing heart and lung problems (United States Environmental Protection Agency, n.d.b). This is especially dangerous for sensitive populations such as youth and elderly people. As we saw in Section 2.1, the transportation corridors in and around South Gate make the community especially vulnerable to $PM_{2.5}$ pollution. Wildfires also increase South Gate's exposure.

2.4 PurpleAir Sensors

Despite the local concerns of South Gate, there has historically been a lack of data for the CEHAT and South Gate residents to use. The South Coast Air

Quality Management District (AQMD) is the regulatory agency tasked with monitoring the air quality throughout the South Coast Air Basin (which contains South Gate), serving about half the population of California with 40 air quality sensors (South Coast Air Quality Management District, n.d.a). While it is valuable to know the air quality in the region, the air quality can change substantially depending on factors such as proximity to large traffic ways and industrial buildings. The closest AQMD sensor is nearly 10 miles away from South Gate and monitors the area of south Los Angeles, which South Gate borders, but is not part of. Due to the high variability of the Southern California air shed, the AQMD sensor in the region is unable to capture a more detailed picture of air quality in South Gate, particularly regarding $PM_{2.5}$ (South Coast Air Quality Management District, n.d.b).

To combat this deficiency in air quality data, the UCLA Kaiser Permanente Center for Health Equity and the UCLA Jonsson Comprehensive Cancer Center selected the CEHAT as a 2019 Community Seed Grantee (UCLA Kaiser Permanente Center for Health Equity, 2019). This money has allowed the CEHAT to purchase and place low-cost PurpleAir sensors throughout the City of South Gate. These low-cost sensors measure $PM_{2.5}$ concentrations in real-time and connect via Wi-Fi to transmit data to PurpleAir, where it is displayed on an interactive map which anyone can access at their website (PurpleAir, 2021b). PurpleAir sensors are a widely used low-cost option for air quality monitoring, so popular that they have drawn the attention of the EPA (PurpleAir, 2021a). Recently, the EPA has started integrating PurpleAir sensors into their own Fire and Smoke Map, which further substantiates the use of PurpleAir sensors in South Gate (Moffitt, 2020). The EPA also publishes a correction algorithm to account for differences between the federal monitors and the PurpleAir sensors (Barkjohn et al., 2020). Using this correction algorithm in tandem with the NowCast (or standard) AQI formula allows PurpleAir data to be used to understand current (and historical) air quality.

While the PurpleAir sensors upload data directly for display to the PurpleAir website, the website can confuse technical and non-technical audiences alike. In addition, the map can be difficult to navigate in regions where there are many PurpleAir sensors. Since many communities in the Southern California area have these sensors, locating South Gate on the larger map is nontrivial and a barrier to using the site. The graphs on the PurpleAir website are hard to read, especially for non-technical users, and the website provides no explanation of AQI for users who are just learning about air quality. Additionally, while PurpleAir's partner ThingSpeak stores historical data, it does so only for a period of fourteen days, meaning any in-depth analysis of air quality would require a separate database of historical data.

2.5 Goals

Our work this year has aimed to help the CEHAT in their efforts to positively impact the residents of South Gate, CA by providing information, analysis, and education on air quality in their city. To provide information and education, we created a website for residents of South Gate to see the air quality in their area quickly and easily, allowing them to make decisions based on this information. As part of this website, we also collect historical data from the PurpleAir sensors deployed throughout South Gate. We then use this data to work towards the other major goal of our project: comprehensive data analysis of air quality in South Gate. Chapters 3 and 4 describe our work this year towards these goals.

Chapter 3

Air Quality Dashboard

The air quality dashboard can be accessed at https://sg-cehat-air-quality.web.app. The source code is available on GitHub and is released under the Hippocratic License 2.1.

3.1 Major Features

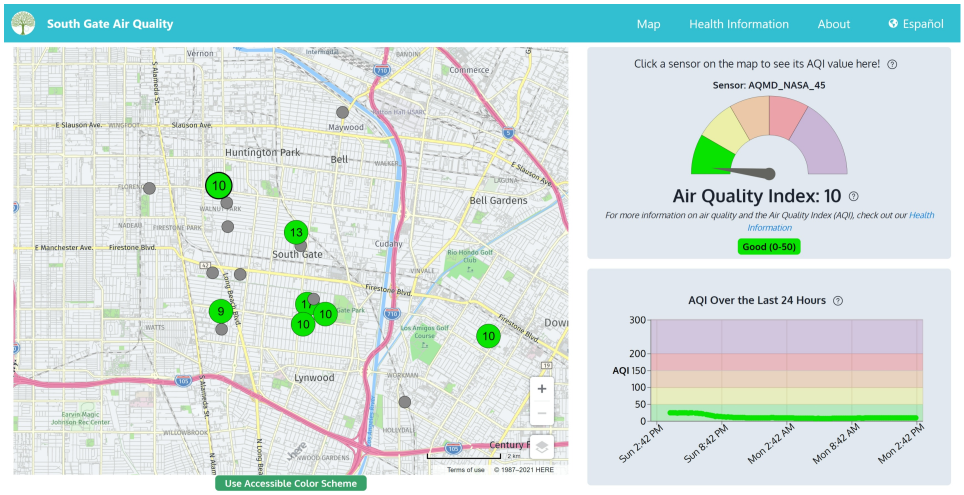
The air quality dashboard is feature rich. In this section, we describe some of the major functionality that users and administrators will experience.

3.1.1 Sensor Data

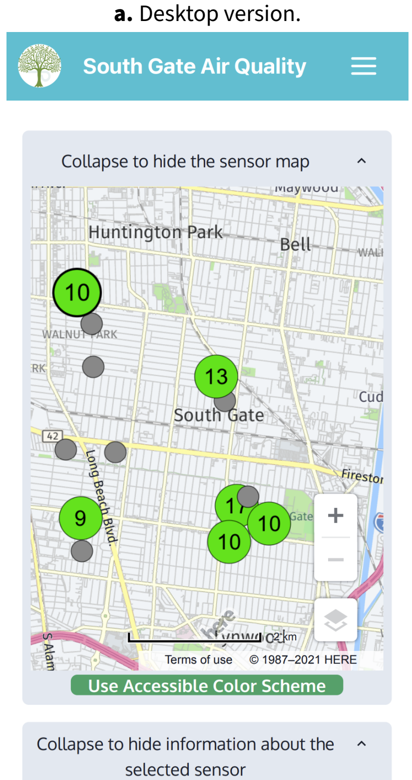
Since the major goal of the website is to allow residents of South Gate to understand the quality of the air around them quickly and easily, visitors to the website are greeted with a home page that provides information collected by the sensors. The home page as rendered on both desktop and mobile browsers can be seen in Figure 3.1. In the sections that follow, we discuss the ways the home page displays data to the user.

Map

The most prominent feature on the home screen is a map that superimposes sensor data onto a map of South Gate at the locations of the PurpleAir sensors. When data from a given sensor is sufficient to compute a NowCast AQI, the sensor is displayed as a circle in the color of the AQI's air quality category with the AQI inside the circle. When a NowCast AQI cannot be computed, the sensor is instead displayed as a smaller grey circle to indicate



a. Desktop version.



b. Mobile version.

Figure 3.1 Home page of the air quality dashboard.

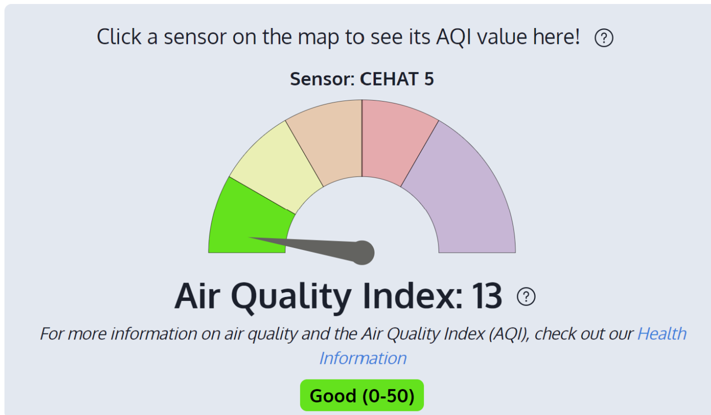


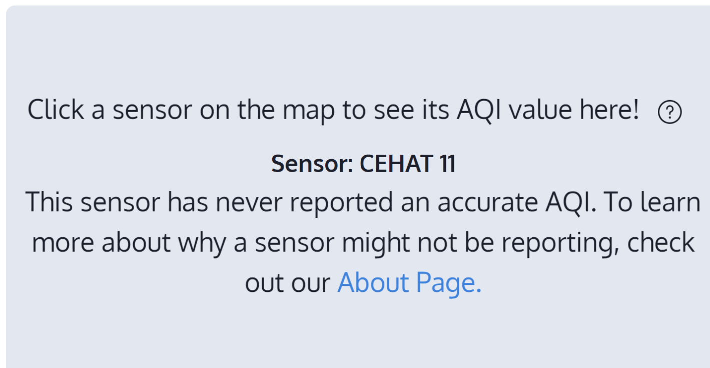
Figure 3.2 AQI gauge showing good air quality at the CEHAT 5 sensor.

that a sensor is physically present at the location but does not currently provide enough data to calculate a NowCast AQI. Whether a sensor is or is not reporting a NowCast AQI, the circular markers representing the sensor are clickable. When clicked, the other two components on the home page update to provide more detail on the selected sensor. We explore these features in the subsequent sections.

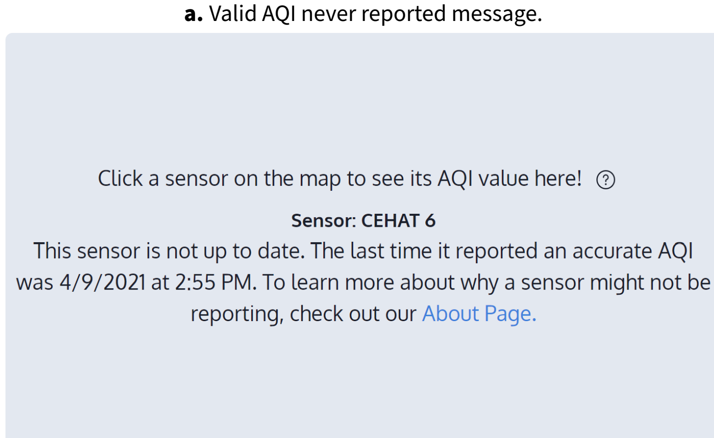
Current AQI Gauge

To clearly communicate the current air quality for a selected sensor, the website features a user-friendly gauge component to put the NowCast AQI into perspective (Figure 3.2). A tag appears below the gauge to show the EPA air quality category in which the selected sensor's current NowCast AQI falls, along with a link to the "Health Information" page (see 3.1.2) to help clarify the AQI scale. When the air quality enters the "hazardous" category (defined by the EPA as an AQI greater than 300), the gauge displays the needle fully to the right to make clear that while the AQI scale goes to 500 (or beyond), the air quality is so poor that it is incredibly dangerous, and all precautions should be taken.

If a sensor does not produce sufficient data to compute a NowCast AQI, the gauge component instead displays a message indicating this and noting when (if ever) a valid AQI was last reported. This can be seen in Figure 3.3.



a. Valid AQI never reported message.



b. Valid AQI reported at some time in the past message.

Figure 3.3 Messages displayed in place of AQI gauge when AQI cannot be computed.

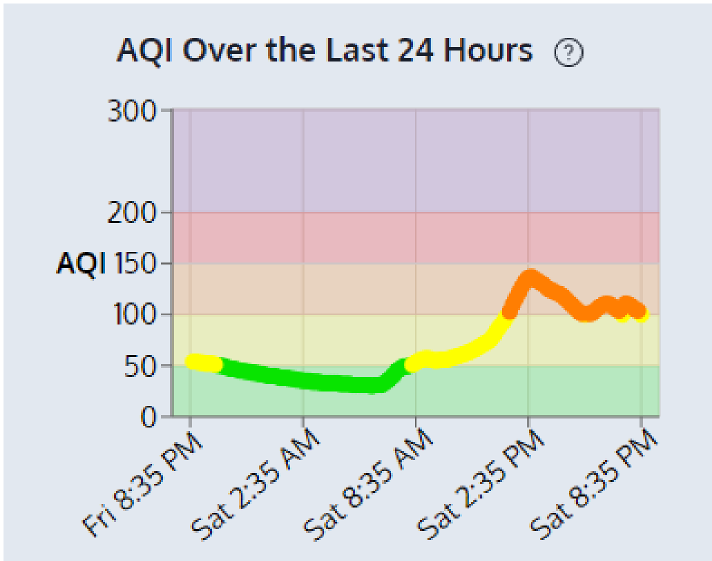


Figure 3.4 Graph of the NowCast AQI at the CEHAT 7 sensor over a 24-hour period.

24 Hour AQI Graph

To provide context of the air quality in South Gate, we display a graph of the NowCast AQI for each sensor over the past 24 hours. This graph is color coded to correspond to the EPA air quality colors to provide at-a-glance information. For times when an AQI was unavailable, no point is displayed on the graph. This means that even a sensor lacking a current AQI can provide a meaningful graph if it is undergoing a temporary lapse in data. An example graph can be seen in Figure 3.4.

3.1.2 Static Content

As a part of the CEHAT's efforts to educate and inform the community, it is important that we contextualize the air quality data. The website provides both health information and information about the project and CEHAT. Similar to the way Chapter 2 serves as context to this report, these pages provide context to residents of South Gate using the website. Importantly, these sections provide both health recommendations on how to avoid exposure to poor air quality as well as ways to get involved with the CEHAT (including becoming a sensor host). We hope this helps to create change in the community and engage more people with the CEHAT.

3.1.3 Administrative Tools

Our goal is to ensure that the website is useful for years to come once the project is over. Since the CEHAT does not have computer scientists or information technology specialists heavily involved with the organization, we wanted to make sure that the CEHAT can administer the website with ease. Therefore, we created a comprehensive set of administrative tools that allow the CEHAT to add and remove sensors from the website, investigate sensor failures, and download or delete sensor data without needing to use technical systems. We provide a full guide to the administrative tools in Appendix A.

3.2 Accessibility

As a community engagement project, we wanted to ensure that *all* members of the community can benefit from the website. Therefore, we have focused on making sure the website is accessible to a wide range of audiences, taking as broad a definition of accessibility as possible.

3.2.1 Assistive Technology Support

Low and no vision users can navigate the web using screen readers, making it important that our website supports these technologies. We have worked to incorporate alternative text for graphics and to support keyboard navigation throughout the website. In some ways, we have succeeded with this goal. In our experiments with the Microsoft Narrator screen reader, a user was successfully able to navigate between pages and have the static content read aloud to them. Navigating the home page was slightly more difficult due to accessibility issues with the mapping software. While we have made significant strides in this area, it continues to be an area for additional development.

3.2.2 Alternative Color Scheme

The EPA has set a standard for communicating air quality categories by color. However, the color scheme they use is not accessible for those with color vision difficulties. To be consistent with other resources members of

AQI Range	EPA Color Scheme	Accessible Color Scheme
0-50	Good	Good
51-100	Moderate	Moderate
101-150	Unhealthy for Sensitive Groups	Unhealthy for Sensitive Groups
151-200	Unhealthy	Unhealthy
201-300	Very Unhealthy	Very Unhealthy
301+	Hazardous	Hazardous

Table 3.1 Comparison of the EPA and accessible color schemes.

the South Gate community may use, we default to the EPA color scheme to communicate air quality information. However, we also allow users to switch to an alternative color scheme that is more accessible to users for whom the EPA color scheme is inaccessible. A comparison between these color schemes is available in Table 3.1.

3.2.3 Spanish Translation

Knowing that South Gate has a large monolingual Spanish speaking community (see 2.1), we wanted to make sure that the website was accessible to this community. For this reason, the website is available in both English and Spanish. The website automatically detects which language a user has set in their browser, but the website also includes a button for users to switch between languages. This allows Spanish speaking users to interact with the website with ease.

We create the Spanish content through a multi-stage process. All five members of the team speak Spanish, with one team member being a native bilingual speaker. For each addition to the website, team members wrote out translations and then had the translations reviewed by other team members, including the native speaker. In addition, the Liaisons provided feedback on the translations, allowing the team to iterate and improve the user experience for Spanish speakers.

3.3 Technologies

Creating the dashboard required many different technologies. In this section, we describe those we used throughout the project.

3.3.1 Front End

The website is a React Progressive Web Application, which allows users to install it as if it were an app on their computers and phones in addition to visiting it like a standard website. We used the HERE Maps tools to create the map interface as it provided a variable cost feature set that integrated well with the React software stack and is free to use at the expected usage level of the website. Attempts to use the completely free and open source OpenLayers library were unsuccessful. The Chakra UI user component library helped us quickly create an easy-to-use and accessible user interface, while the `react-i18next` library facilitated our bilingualization efforts.

3.3.2 Back End

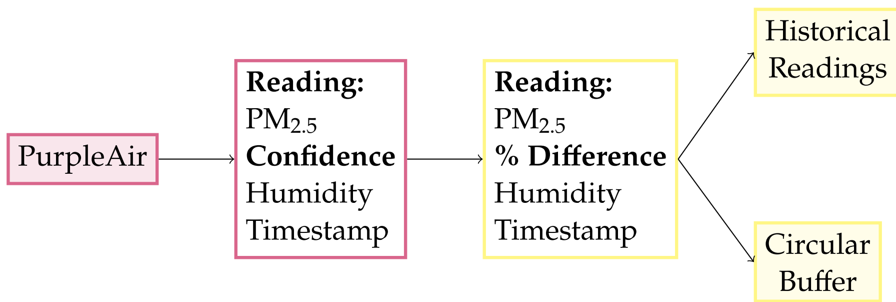
The website uses the serverless architecture of Google's Firebase. We opted for a serverless architecture because it facilitated quicker development time and it requires minimal maintenance. We chose Google's Firebase based on a combination of familiarity, cost, and environmental considerations among competing offerings. The website is hosted through Firebase Hosting. We use the NoSQL Cloud Firestore database to store air quality data, which we obtain from the PurpleAir API (Application Programming Interface). Firestore also stores other website-maintenance data. This data is processed using Cloud Functions for Firebase. Firebase Authentication provides authentication services to the administrative tools described in Section 3.1.3.

3.3.3 Development Tools

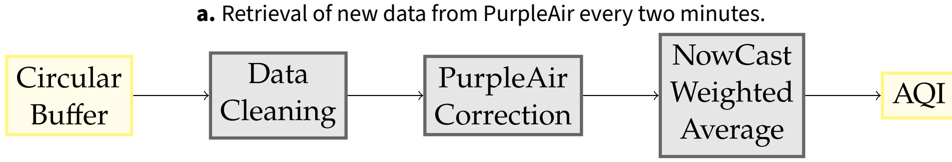
We wrote all code for the website using the TypeScript programming language as it allowed us to write front end React code and back end code in the same language while also helping us to write better code through its typing system. To ensure the highest code quality, we enforced the use of a common code style through ESLint and Prettier (as combined by `gts`) and reviewed each other's code thoroughly using GitHub Pull Requests. We deployed accepted code automatically using GitHub Actions.

3.4 Data Pipelines

Our website heavily relies on up-to-date air quality data. In this section, we describe how our website gets and processes air quality data from PurpleAir.



a. Retrieval of new data from PurpleAir every two minutes.



b. Computation of NowCast AQI from $PM_{2.5}$ data.

Figure 3.5 Pipelines to get and use air quality data.

These processes are visualized in Figure 3.5.

PurpleAir sensors use "PMSX003 laser counters to measure particulate matter in real time, with each laser counter alternating 5-second readings averaged over 120 seconds" PurpleAir (2021a). Every two minutes, we query the PurpleAir API for any new data for the CEHAT's sensors. In particular, we ask for the $PM_{2.5}$ reading, the confidence value of that reading, the humidity reading, and the time at which the reading was generated. We also ask for any error flags for each sensor, which can indicate partial or complete sensor malfunction. We then convert PurpleAir's confidence value to a mean percent difference value between the two laser counter readings in each sensor; this value is used to potentially ignore readings in the data cleaning process. The PurpleAir confidence value is documented in Figure 3.6, and the conversion process from the confidence value to mean percent difference is documented in Figure 3.7.

The data for each of the CEHAT's sensors is then stored in two places in Cloud Firestore: a collection of historical readings for future data analysis and a circular buffer of the last twelve hours of readings for easier use during the AQI calculation process.

As a balance between accuracy and cost, we calculate the NowCast AQIs for each sensor every ten minutes. For this calculation, we obtain the last twelve hours of $PM_{2.5}$ data from the sensor's circular buffer, which is stored in a single document in Firestore. We remove invalid readings in the data

Result: Confidence value for a sensor
Input: Pseudo-average values of channel A and B (a, b)

```
d ← |a − b|                          // difference
avg ← (a + b)/2                      // average
m ← (d/avg) * 100                    // mean percent difference
p ← max((m/1.6) − 25, 0)             // percent confidence
return max(100 − p, 0)               // confidence score
```

Figure 3.6 PurpleAir confidence calculation

Result: Mean percent difference between channel A and B of a sensor
Input: confidence

```
switch confidence do
    case 0 do return 2
    case 100 do return 0
    otherwise do return 0.01 · (100 − confidence + 25) · 1.6
```

Figure 3.7 Mean percent difference calculation cleaning process before applying the PurpleAir correction factor described by Barkjohn et al. (2020) to ensure comparability to EPA data. While Barkjohn et al. (2020) requires that both channels of the PurpleAir sensor agree within a relative and absolute tolerance, the PurpleAir API does not give sufficient data to calculate the absolute difference between channels. As a result, we modify the protocol to use only the relative difference (the mean percent difference value from earlier). If not enough data is present or the data varies by more than the allowed amount, no AQI is reported for that sensor. After the data cleaning phase, we apply the NowCast algorithm for $PM_{2.5}$ (AirNow, 2020). As with before, if there is not sufficient data to apply the NowCast algorithm, we report no AQI for that sensor. Lastly, we convert the NowCast $PM_{2.5}$ to an AQI (Office of Air Quality Planning and Standards, 2018). These AQIs are also stored in a circular buffer for 24 hours, used for the graphs on the home page.

3.5 Technical Challenges and Solutions

The data pipeline described in Section 3.4 was not the original design. Throughout the project, we modified the pipeline to overcome new challenges. In this section, we describe those challenges and how we overcame them.

3.5.1 API Limitations and Deprecation

Originally, the data pipeline could not ask PurpleAir directly for all the new data related to the CEHAT's sensors in one request since this feature was not available. PurpleAir publicizes two (now deprecated) APIs on their website: a rudimentary JSON (JavaScript Object Notation) API that allows users to get some current data for a single sensor and another through their partner ThingSpeak that allows users to get additional information for a single sensor. The ThingSpeak API requires keys obtained from the JSON API. Both APIs only provide data for a single sensor. Therefore, originally, every two minutes we would, for each sensor, ask PurpleAir for some information (including their ThingSpeak keys) and ThingSpeak for additional information. This required two external API calls for each sensor.

To protect the availability of their services, PurpleAir and ThingSpeak limit the number of requests that can be made by a single application. Therefore, we had to artificially delay each API call to comply with this limit, which limited the number of sensors CEHAT could deploy. While researching ways to overcome this barrier, we discovered that PurpleAir intended to deprecate the JSON API, replacing it with a more full-featured API.

This new API has a feature for group queries, where applications can create a group of sensors (in our case, all the CEHAT sensors), and in one request, obtain relevant data on all sensors in the group. When making queries to their API for multiple sensors more frequently than every 10 minutes, they ask that users use the group queries. There is functionally no limit on the number of sensors in a group. However, PurpleAir does not make all of the data for each sensor available through this API. Notably, they pre-average the two channels of data, reporting a confidence value representing the variance between what they call the "pseudo-average" between the channels as opposed to the values of the two channels themselves. The raw difference between the channels is not available despite the correction factor of Barkjohn et al. (2020) needing the absolute difference. While we asked PurpleAir if they would be willing to provide each channel's reading in the group API call, they were unable to make changes within the timeframe of our project. Since PurpleAir is deprecating the original APIs, we had to transition to this new format, despite the loss of information.

3.5.2 Database Costs

The duplication of data to both a historical readings collection and the circular buffer described in Section 3.4 and visualized in Figure 3.5a was not part of the original data pipeline. Initially, we obtained data for the AQI calculation by requesting the last 360 readings (each of which has a document in the Cloud Firestore NoSQL database) for each sensor from the historical readings database. However, Google charges per document read for Cloud Firestore, meaning that the cost of calculating the AQI was beyond the CEHAT's very small budget. By implementing the circular buffer, we were able to significantly cut the number of database reads each day with only a slight increase in the number of database writes (also charged by Google). This change keeps the website below Google's free allotment of database operations. The dramatic reduction in the number of reads can be seen in Figure 3.8, which shows the database usage on the day we started using the circular buffer.

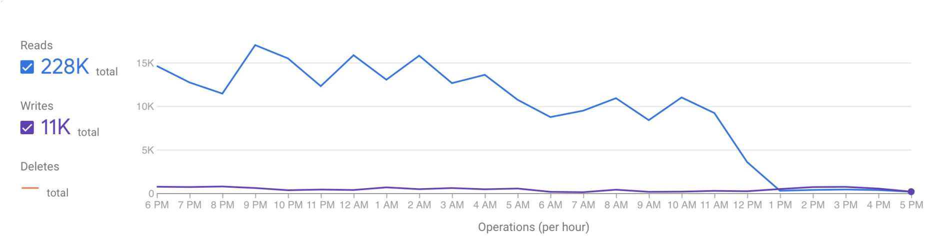


Figure 3.8 Reduction in database operations after transition to usage of circular buffer.

3.6 User Testing

We wanted to make sure that the website was easy to use for members of the community. Therefore, we engaged in user testing to adapt the website to best meet user's needs. We gathered feedback from South Gate residents across many age groups through partnerships with a local youth group and posts in online community forums. In addition, we reached out to non-South Gate residents to get insight from more people on the ease-of-use of the website. This testing guided us towards providing additional tooltips on the website to provide instructions and in bolstering our explanation of the AQI.

Chapter 4

Data Analysis

The source code for the interactive math report is available on GitHub and is released under the Hippocratic License 2.1. We have provided a guide on how to install R and RStudio, get the app to run, and interpret all the visualizations in the app, which can all be found in Appendix B. Users also have the option to run the app online.

4.1 Cleaning the Raw Data

The first step of processing data downloaded from the website is formatting it for data analysis. We refer to this process as "cleaning" data. As mentioned in Section 3.5.1, the format of the raw sensor data changed after PurpleAir deprecated its API, and the website started using the new API on March 30, 2021. We use two different data cleaning processes, one for the old data (data from before March 30, 2021) and one for the new data (all data from after March 30, 2021).

The function that cleans the old data assumes that the comma-separated values (CSV) file has two $PM_{2.5}$ reading channels, as well as timestamp, humidity, latitude, and longitude columns. We correct the timestamp from UTC to the America/Los Angeles time zone) and get rid of any values that exceed the standard EPA thresholds. Then, we use the EPA correction factor (the same correction factor used by the dashboard) to adjust the data so that it meets EPA standards.

The function that cleans the new data assumes the CSV file has a single $PM_{2.5}$ column, as well as columns for timestamp, sensor names, latitude, longitude, humidity, and mean percent difference. To clean the new data, we adjust the timestamp from UTC to the America/Los Angeles time zone, and make sure to remove any negative $PM_{2.5}$ readings in the data.

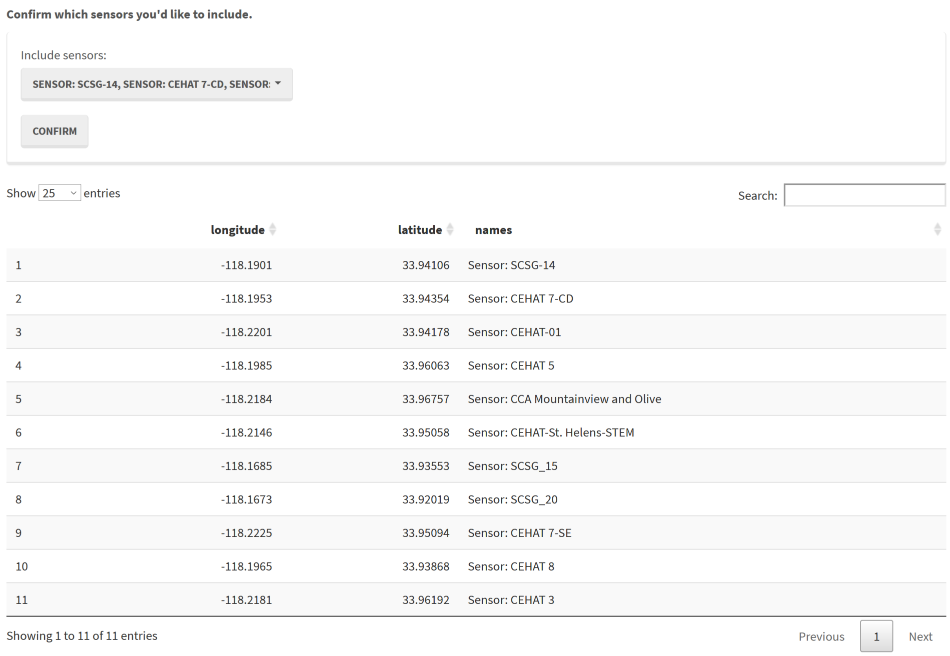
4.2 `Shiny`

We primarily used R for encoding our statistical computations and visualizations of the PurpleAir data. Specifically, we employed a popular R package, called `Shiny`, so that we could create an interactive web application, capable of running calculations on data downloaded from the air quality dashboard (see Sections 3.4 and A.4.6). This is especially useful for our project because it allows the CEHAT members and regulators (as well as any other invested parties) to conduct the same calculations, cleaning, and analyses on future datasets. One of our main goals for the `Shiny` app was making it both digestible and accessible so that the CEHAT members can get a full understanding of the South Gate air quality during a particular time period. In the next two sections, we will discuss the components and visualizations found in our app.

4.3 App Features and Visualizations

The following sections discuss each component of the interactive `Shiny` Application, including their function, the corresponding calculations and visuals, the intended benefits for the CEHAT from each page of the application, as well as our process behind the creation of each page. We wanted to ensure that members of the CEHAT would be able to produce the most basic summary calculations for air quality data, including means, medians, maximums, minimums, and diurnal patterns, as well as be able to produce results that are directly comparable to other AB617 communities with active AQMD $PM_{2.5}$ sensors. We also intended for the CEHAT members to be able to monitor the functionality of their sensor network through both summative information about particular sensors and interpolation calculations that reveal outliers in the sensor network.

To use any of the app features, users must first upload a comma-separated-values (CSV) file from their computer; this file is created by the download data function on the website, as explained in Section A.4.6. After users confirm the format of the data, they can include (or exclude) desired sensors in the app's calculations. The result of this is shown in Figure 4.1.



	longitude	latitude	names
1	-118.1901	33.94106	Sensor: SCSG-14
2	-118.1953	33.94354	Sensor: CEHAT 7-CD
3	-118.2201	33.94178	Sensor: CEHAT-01
4	-118.1985	33.96063	Sensor: CEHAT 5
5	-118.2184	33.96757	Sensor: CCA Mountainview and Olive
6	-118.2146	33.95058	Sensor: CEHAT-St. Helens-STEM
7	-118.1685	33.93553	Sensor: SCSG_15
8	-118.1673	33.92019	Sensor: SCSG_20
9	-118.2225	33.95094	Sensor: CEHAT 7-SE
10	-118.1965	33.93868	Sensor: CEHAT 8
11	-118.2181	33.96192	Sensor: CEHAT 3

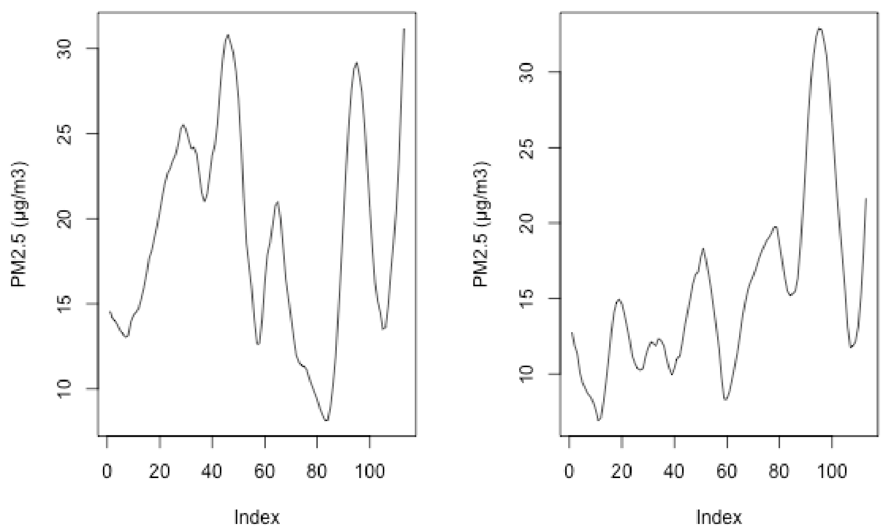
Figure 4.1 Sensor selection on the *Overview* page of the `Shiny` app.

Users are then able to explore the rest of the app's features. Once they adjust the plots to their desired formats, they can return to the *Overview* page of the app to generate a report of the analysis, which contains most of the same visualizations and descriptions, in a portable document format (PDF) or as a Microsoft Word document.

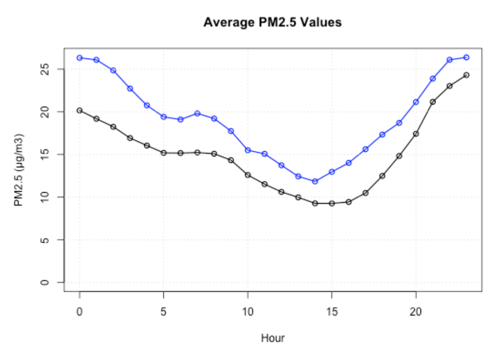
4.3.1 Summary Statistics

The *Summary Statistics* page is the first section of our data analysis. For each calculation, we refer to the *general* hourly $PM_{2.5}$ for South Gate PurpleAir sensors. The general hourly readings are the aggregated $PM_{2.5}$ values by hour, which is an average of each sensor's hourly $PM_{2.5}$ readings. This functionality demonstrates the importance of our app's sensor-exclusion feature: our app allows users to exclude a particular sensor (or multiple sensors) from the data (refreshing each tab with the updated dataset) so that faulty sensors do not skew or invalidate any reported results.

The *Summary Statistics* page includes charts that depict a user-specified 72 hours of the general *hourly* air quality in South Gate (Figure 4.4b), as well



a. Maximum averages for distinct 8-hour periods



b. Diurnal patterns of average $PM_{2.5}$ values for distinct time periods.

Figure 4.2 Line graphs displaying 8-hour averages and the diurnal patterns.

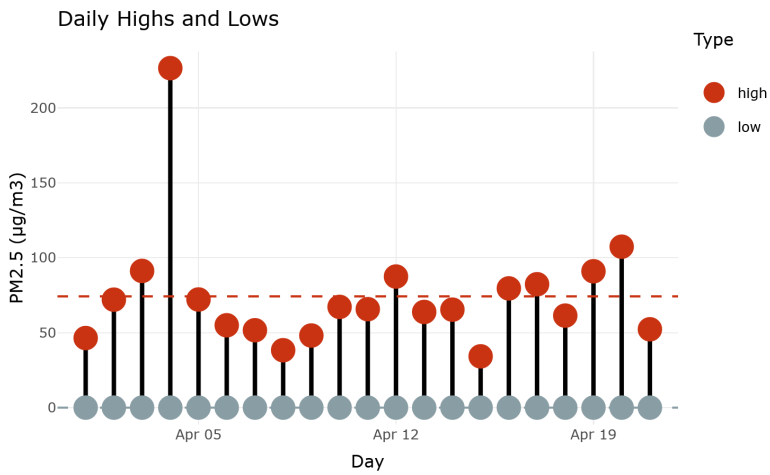


Figure 4.3 High and low $PM_{2.5}$ values in South Gate for April 2021.

as a user-specified 30 days of the general *daily* air quality (Figure 4.4a). In addition, the page documents the spread of peak values for $PM_{2.5}$ over 24 hours, as shown in Figure 4.4c. The *Summary Statistics* page also directly reports the percentiles of the data, the hour when most peak values are recorded, and the overall second maximum for recorded $PM_{2.5}$. Moreover, this page allows users to compare 8-hour averages and diurnal patterns for two different time periods, which is shown in Figure 4.2a and Figure 4.2b, respectively. This functionality allows the CEHAT group to focus specifically on periods of increased $PM_{2.5}$ values, like during times of wildfires or national holidays. For example, on New Year's Eve, there are spikes in $PM_{2.5}$ levels because of the increased use of fireworks. The goal is for the CEHAT to both understand how air quality typically varies in South Gate and to generate results that are clear and comparable to other air quality data so that the CEHAT can assess how South Gate fares in relation to other locations with federally regulated sensors.

When it came to visualizing the summary statistics for South Gate, we made sure to use intuitive graphs, including lollipop charts, bar charts, and line charts. Lollipop charts are useful for comparing categorical data. We used them to compare the highs and lows of $PM_{2.5}$ values by sensor and for the entire city of South Gate, as seen in Figure 4.3.

Bar charts are effective for comparing data by groups. We used them to look at the trends in the hourly and daily $PM_{2.5}$ values, as well as for visualizing the peak $PM_{2.5}$ values during a selected period. Some of the bar charts found on the *Summary Statistics* page are shown in Figure 4.4.

Line graphs are typically used to understand how patterns evolve over time. We used them to compare $PM_{2.5}$ values between two different times, and to observe the diurnal patterns over a 24-hour period, as shown in Figure 4.2.

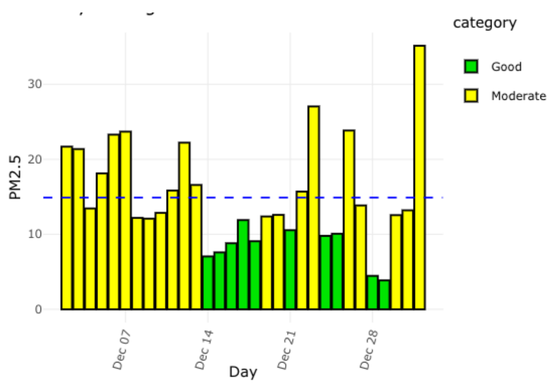
4.3.2 Sensor Summaries

Although PurpleAir sensors are low-cost and easy to install, they sometimes malfunction and begin reporting improbable values for $PM_{2.5}$, or not reporting any values at all. As we have previously mentioned, we address this via the sensor exclusion feature. However, to even be able to scrutinize the performance of particular sensors, users need to inspect how each sensor has performed in the uploaded dataset.

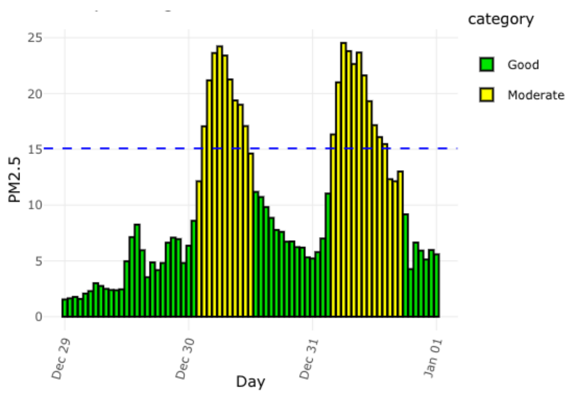
The *Sensor Summaries* page allows the CEHAT to inspect the consistency of each sensor, identify which sensors typically report higher or lower values of $PM_{2.5}$ (depicted in Figures 4.5a and 4.5b), and identify which sensors typically report outliers. This page is particularly helpful in identifying any geographically specific air quality patterns that may affect a sensor's typical functionality, and in identifying where air quality in South Gate is better (or worse). The goal is for the CEHAT to be able to use this page of the `Shiny` app to maintain their sensor network: repairing those that produce the most implausible readings and investigating those that tend to go down often.

On the *Sensor Summaries* page, users can find the breakdown of each sensor's collected readings by EPA AQI category (found in Figure 4.6b), the AQI category of readings by percentage (found in Figure 4.6a), the number of days spent not reporting data, the historical percent difference data (only available for data of the latest data format, discussed further in Section 4.4.2), and the breakdown of readings that both exceed and fall below the general median calculation by hour, in both geographical (map-included) and non-geographical (bar chart) formats.

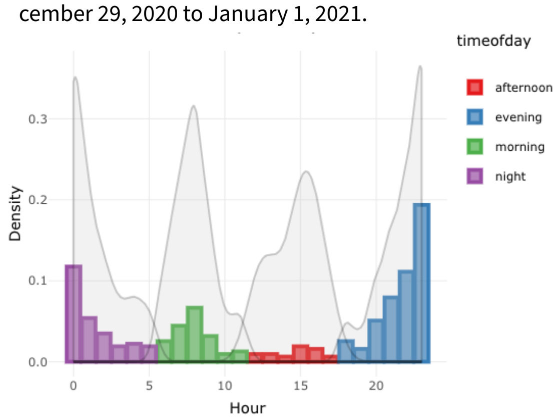
Similar to the graphics in Section 4.3.1, we used bar charts to visualize the sensor summary data. Additionally, we used spatial charts so that the user can get a better idea of how sensors behave in certain areas of South Gate. These visualizations can be found in Figure 4.5.



a. Daily averages of $PM_{2.5}$ values in South Gate for December 2020.

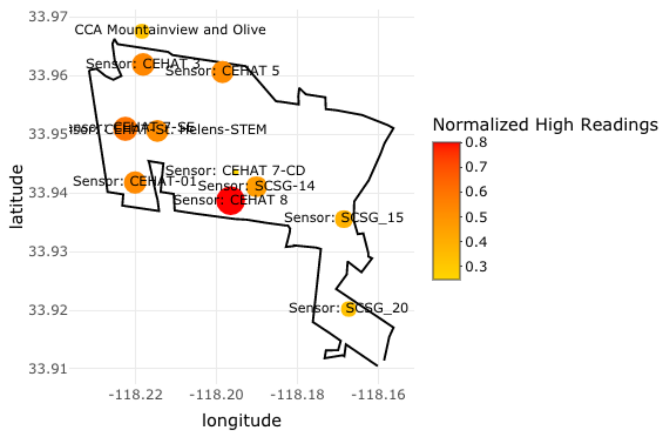


b. Hourly averages of $PM_{2.5}$ values in South Gate from December 29, 2020 to January 1, 2021.

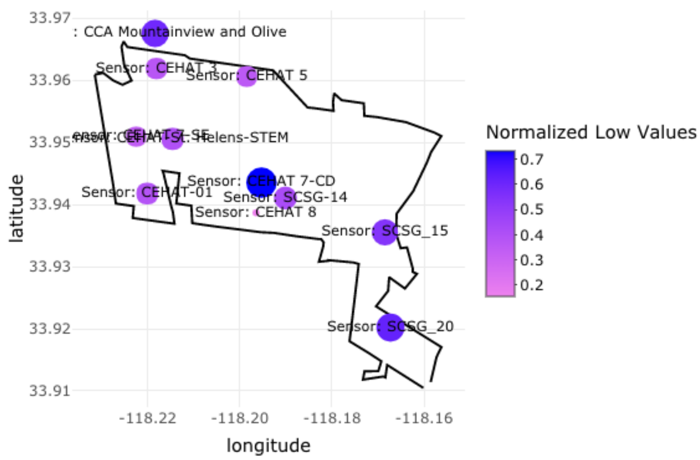


c. Peak $PM_{2.5}$ values by density over a 24-hour period.

Figure 4.4 Bar charts displaying hourly and daily data.

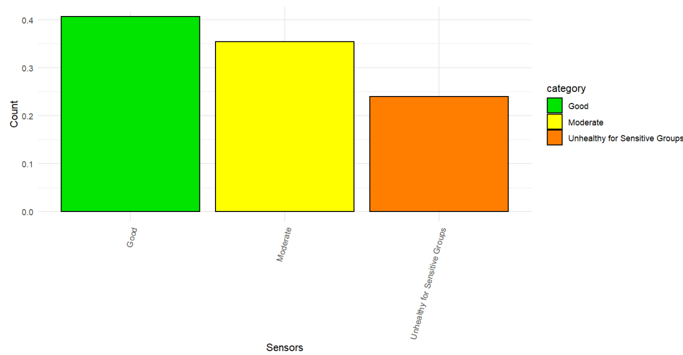


a. Readings over the median per sensor in April 2021.

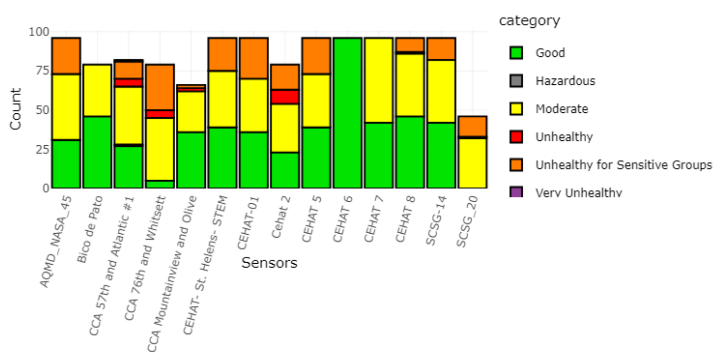


b. Readings under the median per sensor in April 2021.

Figure 4.5 Bar charts and spatial charts that summarize South Gate readings.



a. EPA categories for the $PM_{2.5}$ readings of sensor CEHAT 5 for April 2021.



b. EPA categories for the $PM_{2.5}$ readings of all South Gate sensors for April 2021.

Figure 4.6 EPA categories of $PM_{2.5}$ readings in South Gate for all sensors and for sensor CEHAT 5.

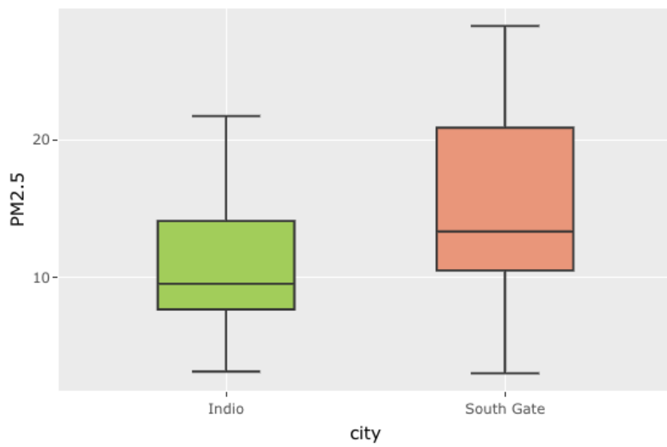
4.3.3 Comparisons between Cities

One of our goals is to make sure that the CEHAT understands how South Gate's air quality compares to the air quality in other parts of Southern California. We provide users with the ability to compare the South Gate $PM_{2.5}$ values to $PM_{2.5}$ values in a CSV file from the AB 617 Community Air Monitoring Website using t-tests and bar plots. Based on the results of the t-tests, the app provides a p-value which can inform the user if South Gate or the provided region has worse air quality. The visualizations include box plots for both locations along with a side-by-side bar chart of the daily $PM_{2.5}$ values for each city. The plots are made interactive through the `plotly` package in R so that the user can easily find specific statistical measurements for both datasets by hovering over the plots. These visuals can be seen in Figure 4.7

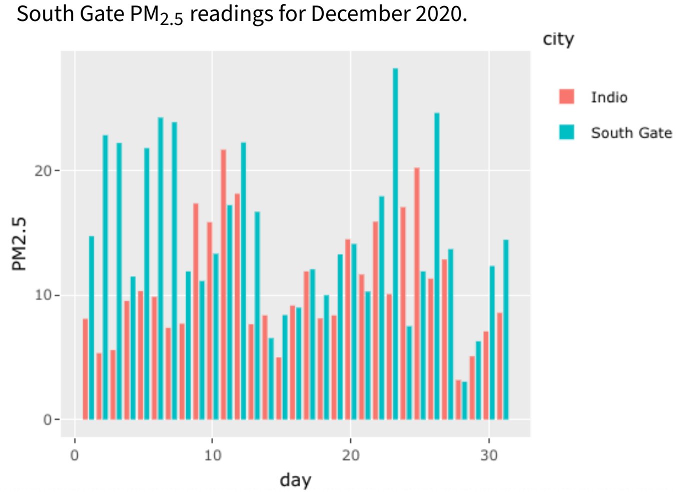
4.3.4 Interpolation and Sensor Placement

The *Interpolation and Sensor Placement* page conducts interpolation calculations for the uploaded PurpleAir dataset at a specified timestamp. The method of interpolation that that our app runs is known as ordinary kriging. This method of spatial modeling invokes a semivariogram, which models the relationship between sensor locations to indicate the variability of the measure in relation to the physical distance of separation. The app automatically optimizes the parameters of the semivariogram and then calculates the ordinary kriging interpolation output for the selected timestamp. Each set of ordinary kriging calculations provides three pieces of information: predicted $PM_{2.5}$ values across South Gate, the variance of those values, and the corresponding standard deviations. The visualizations for these calculations are found in Figure 4.8.

We determined that ordinary kriging was the most suitable method of interpolation through an iterated process of cross-validation for several methods, in which we sampled 14 timestamps in two different datasets (October, then December) and ran cross-validation on each timestamp. Using the root mean-squared error as a metric, we determined that ordinary and simple kriging tended to produce estimations with the least amount of error amongst the following methods: inverse-distance weighting, ordinary kriging, simple kriging, universal kriging, thin plate spline interpolation, local polynomial interpolation, and completely regularized spline interpolation. See Table 4.1 for the error values. The decision to use ordinary kriging as

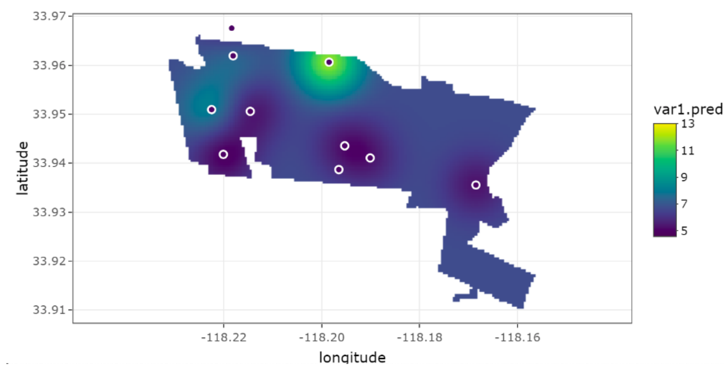


a. Box plot comparison of the Indio $PM_{2.5}$ readings (left) vs. Box plot of the South Gate $PM_{2.5}$ readings for December 2020.

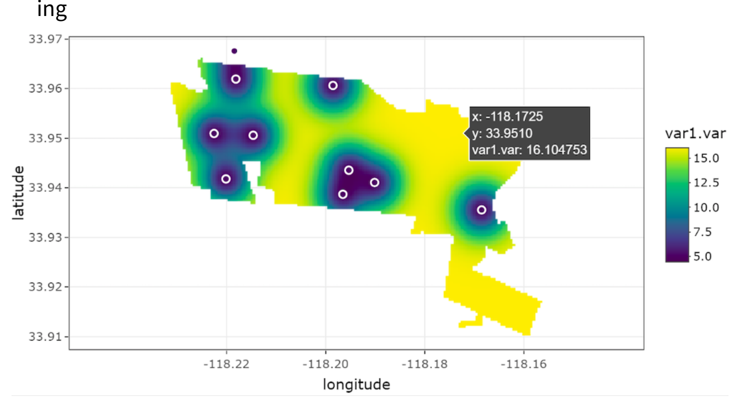


b. Side-by-side bar chart of $PM_{2.5}$ readings in Indio and South Gate for December 2020.

Figure 4.7 Comparisons between South Gate $PM_{2.5}$ values and $PM_{2.5}$ values in other regions in Southern California.



a. Estimated $PM_{2.5}$ values for South Gate calculated through ordinary kriging



b. Variance of estimated $PM_{2.5}$ values for South Gate calculated through ordinary kriging

Figure 4.8 The results of the ordinary kriging computed in the `Shiny` app. These calculations show first the estimated values, and then their spread.

Interpolation Method	Average Root Mean-Squared Error
Thin-Plate Spline	2.6915
Local Polynomial Interpolation	2.3735
Completely Regularized Spline	1.6008
Ordinary Kriging	1.548
Simple Kriging	1.3016
Universal Kriging	1.5903
Inverse Distance Weighting	2.5787

Table 4.1 Root mean-squared errors for various feasible interpolation methods.

opposed to simple kriging was determined by the feasibility of automating the kriging process in R, which was made possible for ordinary kriging through the R package `automap`.

The goal of the *Interpolation and Sensor Placement* page is for the CEHAT group to be able to get additional air quality estimates throughout different locations in South Gate without needing to augment their sensor network. In addition, the CEHAT gets another tool for managing their sensor network, as the variance calculations provide a measure of where they should deploy more sensors, since better network coverage directly reduces spatial variance.

4.4 Technical Challenges and Solutions

We have faced two significant problems while working on the data analysis report. In this section, we will describe those challenges and the solutions we produced.

4.4.1 Switching to an Interactive User Interface

As we neared the end of the Spring semester, we began to develop a clear picture for how to adequately represent our work. Instead of keeping our data analysis in an `R Markdown` file, which is a framework that is well-suited for in-line code documents, we switched to a `Shiny` application format. The app better accommodated the distinct facets of our work. Also, given that our `Shiny` app is wholly interactive, this change ensured that the CEHAT members would not have to manually change the code, which they otherwise would have needed to do had we kept the `R Markdown` file. We switched to a user interface (UI) `Shiny` app close to the end of the project. This proved to be very time-consuming as we had to integrate our analyses into a server driven UI, with which we had very little experience. Despite the difficulties that came with acquainting ourselves better with `Shiny`'s coding structure and UI formatting, we were able to successfully produce an intuitive and user-friendly report.

4.4.2 Change in Data Format

As mentioned in Section 3.5.1, PurpleAir deprecated their API which resulted in a change of data format. Instead of having two $PM_{2.5}$ readings for each sensor, PurpleAir provides one reading along with a confidence value for it. In correspondence with this change, we had to make a function that cleans the new raw sensor data. We also had to modify the functions that depended on the format of the old data so that the functions can handle the new data as well. In our app, we ask users if their data is new or old ("old data" meaning that the data is from before PurpleAir switched their API) so that we can properly handle the data that they are uploading.

4.5 Testing

Testing of the app was primarily small-scale, as our main concerns with usage of the app was ensuring that it functions without difficulty or error on other computers. Therefore, we composed a complete guide to downloading R and RStudio and installing the necessary packages, as well as a tutorial for how to run the `Shiny` application. We distributed this guide to all members of the CEHAT Clinic team, including the Liaisons. The testing of the `Shiny` application informed us about improvements that could be made to the guide to streamline the user experience during setup. App testing also helped in locating any bugs that remained in the app, allowing us to address and solve them prior to the completion of the project. Once the app was deployed to the web, we employed the same small-scale testing to make sure it would work on other devices.

Chapter 5

Conclusion

Throughout the year, we have worked diligently to accomplish major goals. In doing so, we also investigated other directions for the project and planned future phases of work. We conclude by presenting our work on a mobile sensor unit and future work for the CEHAT.

5.1 Mobile Sensor

We investigated using an incredibly low-cost PlanTower PMS5003 air quality sensor attached to an Arduino as a form of mobile sensor. This could be used to gather data in areas where a fixed sensor cannot be located, comparing that data to interpolated values. In this way, the mobile sensor could help the CEHAT better determine where new sensors could be placed. Additionally, the mobile sensor could be used by school groups as an educational experience.

We constructed one of these sensors (Figure 5.1) and sent it to Claremont for our Advisor to test in Southern California (due to the COVID-19 pandemic, team members were located throughout the country during this project). Unfortunately, the sensor performed poorly in even slightly windy conditions, making it infeasible for its desired use. However, the CEHAT may wish to consider alternative mobile sensor options in the future. Additionally, the mobile sensor could be a useful tool in educational settings for teaching about air quality and air quality sensors.

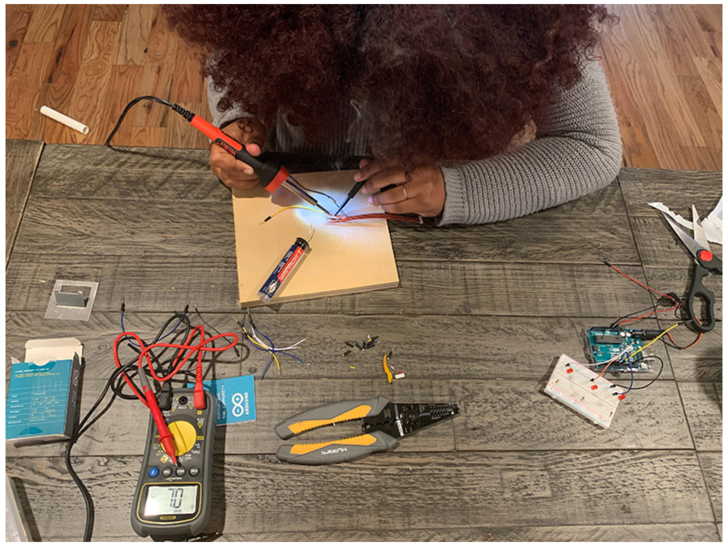


Figure 5.1 Team member Hillary Rodriguez building a mobile sensor.

5.2 Future Directions

Throughout the development of both the air quality dashboard and the data analysis application, we collaborated with the CEHAT on a wide variety of desired features and their respective priorities. In this section, we explore how the CEHAT can build off the work completed in this project to add additional desired functionality.

5.2.1 Dashboard

Throughout the year, the CEHAT was interested in exploring the following functionalities that did not make it into the final website:

- **Interpolation**: Similar to the work done in the analysis application (Section 4.3.4), color the map with the estimated air quality throughout the city.
- **Forecasting**: The EPA does work on forecasting the AQI into the near future (AirNow, n.d.b). Additional functionality forecasting the AQI could be built into the dashboard.
- **Crowd sourcing**: The AQMD has a crowd-sourcing program to collect odor complaints. The CEHAT indicated interest in either integrating

with that system or designing their own crowd-sourcing program for the dashboard.

- **Alerting**: In times of extreme air quality, the CEHAT would like to create an alerting system, placing banners on the website and texting users.

- **App-like enhancements**: The website is a progressive web app, as mentioned in Section 3.3.1, meaning that users can download the website like an app on their phone or tablet. However, improvements could be made to the app to be more mobile friendly and feel more native on phones. For example, the static content of the website should be made available offline once a user has visited the website at least once. Additionally, while the map page functions well on mobile, it could be enhanced to feel more intuitive to use for mobile users.

- **Generalizing the application**: As mentioned in Chapter 3, the code for the website is released under the Hippocratic License for other people to use. Thus, another community could use our work from this semester to create an air quality website for their community. While we documented the process of cloning the website on GitHub, this process could be better streamlined and explained such that users with less technical experience in web development could better adapt and expand on the work we created this semester.

Since these are areas in which the CEHAT has already expressed interest, they would be the best areas for future development work. Additionally, as mentioned in Section 3.2, the HERE Maps API has accessibility issues for users of assistive technologies that the CEHAT may wish to address.

5.2.2 Data Analysis

For the data analysis, the CEHAT and members of the Clinic team collaborated on what would be most useful to include in the data analysis app. The most poignant concerns involved the using external datasets to identify correlations, but we did not determine the sources of such datasets during this project. The following features did not make it into the `Shiny` app:

- **Forecasting:** As mentioned in 5.2.1 the ability to generate forecasts into the immediate future using existing PurpleAir data was not included.

The large amount of data required to produce accurate forecasts as well as the computation involved posed major obstacles.

- **Value-based Correlations:** In line with the focus on environmental justice in this project, comparing the recorded $PM_{2.5}$ values to the rates of asthma and other respiratory diseases, and comparing concentrations of $PM_{2.5}$ to concentrations of other pollutants like ozone or sulfur oxides were of interest to the CEHAT. Fitting the app with the capacity to handle other sets of data would permit this functionality.

- **Spatial Correlations:** Some of the desired correlations focused on the spatial difference as variable. As such, there was interest in comparing the air quality in South Gate to other locations in the United States, in comparing air quality in residential areas to non-residential areas, in identifying and correlations between tree cover and $PM_{2.5}$, and in comparing the air quality between census tracts, focusing particularly on any correlation with income level. The compactness of South Gate and the high variability of air quality data hinders the feasibility of any of the internal correlations. That said, being able to handle data from various different sources as input would once again make much of this possible in the `Shiny` app.

- **$PM_{2.5}$ With Respect to Distance From Highways:** Determining how the air quality varies in regards to the distance from the two major highways that border South Gate was initially very favorable to the CEHAT, and this is a feature that could be addressed with a mobile sensor, ideally not the one mentioned in Section 5.1.

Many of these additional features would require data uploads from a variety of sources. The two endeavors that would not require a separate data upload are forecasting and determining $PM_{2.5}$ with respect to distance from highways. These features would actually require either a much larger dataset, for the former, or additional sensors, for the latter. If the various sources of data are reliable and verified with the CEHAT, then the avenues of spatial and value-based correlations are the most logical areas for expansion of the Shiny app's functionality.

Appendix A

Administrator Guide to the Air Quality Dashboard

Throughout our development we have prioritized making the website easy to maintain for its administrators (i.e., the CEHAT). This guide describes all the ways that administrators can control the site, including adding new sensors, removing existing sensors, downloading data from the website database, and deleting data from the database.

A.1 Accessing the Administrative Panel

To access the administrative panel, navigate to the *About* page of the website and scroll to the bottom of the page. Under the *Administrators* heading, there is a clickable link to the administrative pane. You can also navigate to this page directly at https://sg-cehat-air-quality.web.app/admin.

If you are not already signed in, you will be prompted to sign in either with email or Google (Figure A.1). If you have trouble signing in one way, it is possible that you set up your account to only work with *Sign in with email* or *Sign in with Google*, so you should try the other sign in method.

Once you are signed in, you should see the home page of the admin panel (Figure A.2), which has the options *Manage Account*, *Manage Sensors*, *Manage Users*, and *Sign Out*. The rest of this guide details how admins can use these features.

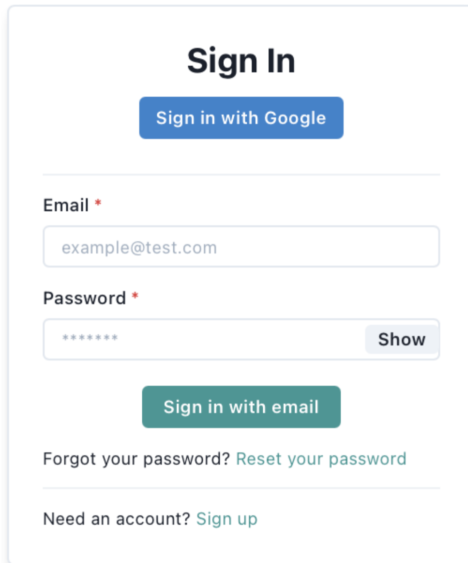


Figure A.1 Sign in page for website administrators.

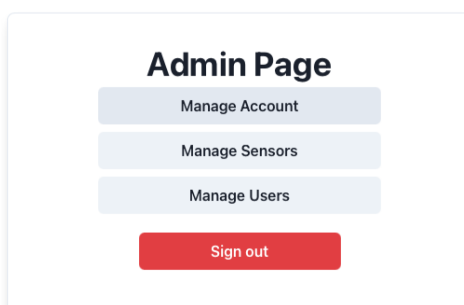


Figure A.2 Home page of the administrative pane. You can navigate to all admin features from this page.

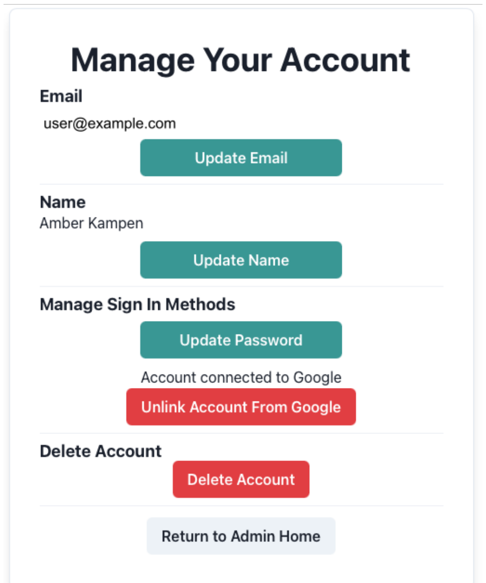


Figure A.3 *Manage Account* page of the administrative pane.

A.2 Managing Your Administrator Account

The *Manage Account* page (Figure A.3) is simple. It allows users to update their name, email, and password. If you do not have a Google account associated with your email address, you must use an email and password to access your account. If you do have a Google account associated with your email address, the *Manage Account* page will also allow you to connect your account to Google (if you signed up with email and password) or add a password to your account (if you signed up with Google). The sign in methods you use changes nothing about your access to administrative features: we simply offer both options so that signing in is as easy as possible.

Note that only non-administrative users can delete their account. Thus, if you wish to delete your account, first remove yourself as an admin user in the *Manage Users* page.

A.3 Adding and Removing Administrators

The *Manage Users* page (Figure A.4) can be used to add and remove administrators. To become an administrator, a user must first create a non-

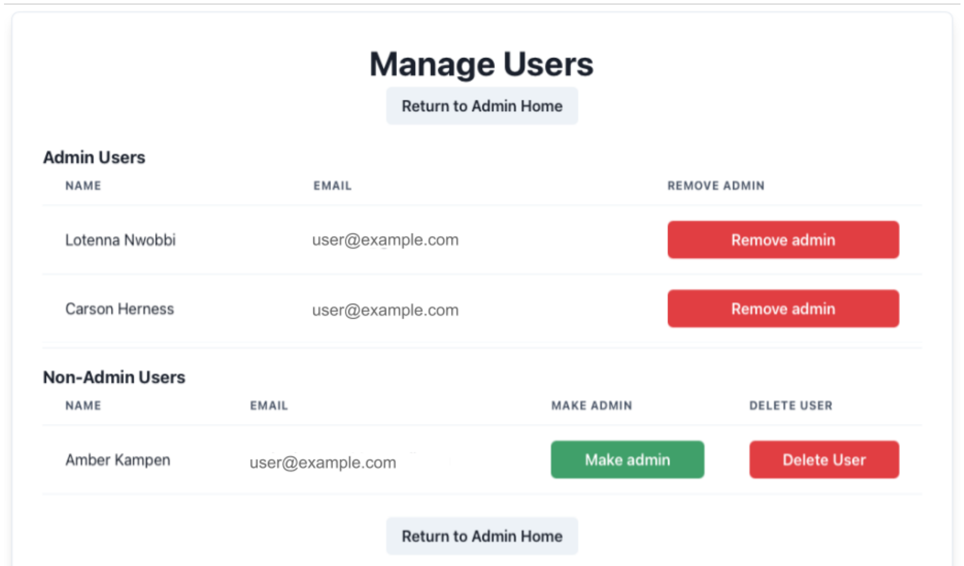


Figure A.4 *Manage Users* page which allows administrators to change the administrator status of other users. To make the non-admin user an admin, click the green button that says "Make admin". To remove Lotenna Nwobbi as an admin, click the red button next to his email that says "Remove admin".

administrator account. They can do this by navigating to the sign in page at https://sg-cehat-air-quality.web.app/admin and click the sign-up link that appears after "Need an account?". Until this account has been given administrator status by a user who is already an administrator, the only option available will be to manage their account.

Once the new account has been created, this user will appear under "Non-Admin Users" in the *Manage Users* page. By clicking the *Make admin* button and pressing *Confirm*, the selected user will have administrator status and will see the other options on the administrative pane.

Since administrators can add and remove sensors from the network as well as irreversibly delete data from the database, it is a good idea to keep the list of authorized administrators somewhat small, only granting administrative access to users who are responsible for monitoring the website. We have provided the ability to remove existing non-administrative users so that if a member of the CEHAT moves out of the area or takes on new responsibilities, you can continue to keep your administrator list small.

Removing existing administrators is a similar process as adding new

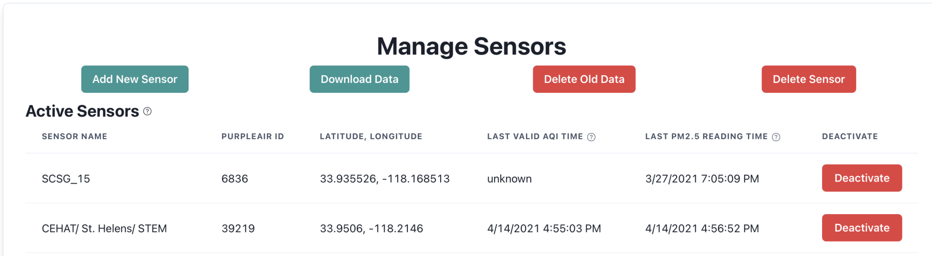


Figure A.5 *Manage Sensors* page of the administrative pane.

administrators. In the *Manage Users* page under the "Admin Users" heading, you can click the *Remove admin* button and press *Confirm* to remove the user's administrator status. This will move the user under the "Non-Admin Users" heading, where they can be added as an administrator again in the future.

Note that we do give administrators the power to remove themselves from the admin list. If this action is confirmed and you want to be an admin again, another administrator on the site needs to add you back. As a safety feature, if there is only one admin on the site, they cannot be removed until more admins are added.

A.4 Managing Sensors

The most important feature in the administrative pane is the ability to manage the sensors and their historical data. All the features discussed in this section are available on the same page (Figure A.5) which is accessed by clicking the *Manage Sensors* button from the admin home.

A.4.1 Understanding the Manage Sensors Page

Active and Inactive Sensors

This page is split into two sections, "Active Sensors" and "Inactive Sensors". Active sensors appear on the map and their historical $PM_{2.5}$ readings are recorded in the database. Inactive sensors are in our database and may have historical $PM_{2.5}$ readings from when they were once active, but they do not appear on the map and any $PM_{2.5}$ values that are recorded while the sensor is marked as inactive are not stored in our database. By default, you should assume that you want a sensor to be active. Reasons to mark a sensor as inactive include:

- The sensor has been returned to you by the host, so it should not show up on the map anymore.
- The sensor has not reported in a long time and you think there is something wrong with the hardware. For this reason, you do not want it to be on the map anymore.
- You are preparing to delete the sensor.

Sensor Information

For each sensor on the *Manage Sensors* page, you will see the following values:

- Sensor Name
- PurpleAir ID
- Latitude and Longitude
- Last Valid AQI Time
- Last $PM_{2.5}$ Reading Time
- Recent Sensor Errors
- Recent AQI Calculation Errors

The sensor's name, PurpleAir ID, latitude, and longitude come directly from PurpleAir. The PurpleAir ID, latitude, and longitude are automatically generated by PurpleAir and the sensor name is configured (via PurpleAir) when you first set up a new sensor.

The last valid AQI time, last $PM_{2.5}$ reading time, recent sensor errors, and recent AQI calculation errors are values that the Clinic team found useful and are calculated in the back end of the website. These values are described in detail in the following section A.4.2. Note that recent errors only appear for active sensors.

A.4.2 Troubleshooting Sensors that are not Reporting

The *Manage Sensors* page of the admin panel contains detailed information for why a sensor may not be reporting an AQI. Using the details provided, you can distinguish between a sensor that is down for a few hours versus a sensor that has a systematic problem with its hardware or software and must be returned to PurpleAir. The four fields that can help you troubleshoot a sensor are *Last Valid AQI Time*, *Last* $PM_{2.5}$ *Reading Time*, *Recent Sensor Errors*, and *Recent AQI Calculation Errors*.

Last $PM_{2.5}$ Reading Time

This value is the last time any $PM_{2.5}$ value was recorded by the sensor and sent to PurpleAir. For most sensors, this value should be very recent (within the last few minutes). If this is not the case, there may be an issue with the sensor itself. See the following sections for specific troubleshooting advice.

Last Valid AQI Time

This value is the last time the website reported an AQI for this sensor. This time might be different from the last $PM_{2.5}$ time for a few reasons. First, the website gets new $PM_{2.5}$ values every 2 minutes but only calculates new AQI values every 10 minutes, so the times will not match exactly. Secondly, a sensor may not have enough satisfactory data to calculate AQI even though it is getting values from PurpleAir. If this value is in the last few hours, it is likely that the sensor will start reporting again without any action from the CEHAT. If the value is more than a week old, there is a good chance there is something wrong with the sensor. See the following sections for specific troubleshooting advice.

Recent Sensor Errors

Any errors in this column are related to the PurpleAir sensor itself, not the AQI calculation process. A given sensor can have between 0 and 6 values in this column. All errors that appear in this column were generated from the most recent sensor reading, so the existence of an error does not necessarily mean that something must be fixed. We advise that these errors be troubleshooted only when the sensor does not have recent times for "Last $PM_{2.5}$ Reading Time" or "Last Valid AQI Time".

Many of these errors could potentially be fixed by resetting the sensor: unplugging it from its power source and plugging it back in. However, there is also a chance that there is something wrong with either the hardware or the software of the sensor that can only be repaired by PurpleAir. The following list explains each error and possible troubleshooting ideas.

- **Reading Not Received:** This means that the last time the website requested data from PurpleAir (which happens every 2 minutes), this sensor did not report any values. This might happen if the sensor has been unplugged or if there is a problem with the host's internet. If the sensor is plugged in and connected to strong internet, there may be something wrong with the sensor.

- **No Humidity Reading Received:** This error means that the website did receive data for this sensor, but the humidity was not reported. A PurpleAir sensor's humidity is essential in calculating the AQI via the NowCast algorithm, so it will not have updated AQI values if it does not report humidity. This could be an error with the PurpleAir sensor's software but is likely an issue with the PurpleAir sensor's humidity sensor hardware.

- **Incomplete Sensor Reading:** This error means that the website did receive data for the sensor, but some value was missing. If this error appears with the "No Humidity" error, it is likely referring to the humidity, but if the "No Humidity" error is not present, it means some other value ($PM_{2.5}$, confidence, latitude, longitude, timestamp) is not being reported.

- **Channels Diverged:** PurpleAir Sensors have two channels that measure $PM_{2.5}$. If the values reported by those two sensors are too far apart, they have "diverged" and the reading is discarded. Something is wrong with one or both of the $PM_{2.5}$ channels.

- **Channel A Downgraded:** PurpleAir "downgrades" channel A if it believes the channel is not reporting accurately. Having both channels is essential to accurate AQI calculation, so this error will prohibit a sensor from reporting AQI on the CEHAT website. Something is wrong with channel A, and if standard troubleshooting does not solve the problem (rebooting, checking Wi-Fi, cleaning), there is likely something wrong with the sensor that must be repaired by PurpleAir.

- **Channel B Downgraded:** This error is the same as the previous error but for channel B instead of channel A.

Recent AQI Calculation Errors

Any errors in this column are related to the AQI calculation. While errors in this column may signify a hardware issue with the PurpleAir sensor, it may also be related to the AQI calculation process. If the "Last Valid AQI Time" is recent, these errors do not need to be addressed as the sensor may recover soon. However, if the "Last Valid AQI Time" is more than a few days old, there is likely an issue with the sensor.

- **AQI Too High:** This error means that the calculated AQI was greater than 500. This value is larger than the EPA's AQI scale and signifies that the sensor is reporting extremely high values that may not accurately reflect the concentration of $PM_{2.5}$. If this error occurs for a short period of time, it may be that something hazardous was directly in front of the sensor and it will return to normal soon. If this error occurs for an extended period of time, there is likely something wrong with the sensor that must be addressed by PurpleAir.

- **Not Enough Readings:** The NowCast AQI calculation that is used to convert $PM_{2.5}$ to AQI uses data over the last 12 hours. For an hour to be used in the calculation, there need to be at least 23 readings with sufficient data (any of the errors discussed in the previous section would lead to a reading not having sufficient data). Of the most recent 3 hours, at least 2 hours need to have sufficient data for an AQI to be calculated. This error occurs if even **before** checking that the readings had sufficient data, there were not enough recent readings for AQI calculation.

- **Not Enough Valid Readings:** This error is slightly different from the above error. If a sensor is still emitting readings but our confidence in those readings is low due to channel divergence or a channel downgrade, we do not include those readings in our data calculation. Any of the errors mentioned under "Recent Sensor Errors" may contribute to this error.

A.4.3 Relocating a Sensor

It is likely that at some point you will need to relocate a sensor that you already have (for example, when a sensor host moves). There are two options for how to do this: (a) keep this sensor on the website, which keeps old historical readings or (b) remove the sensor and re-add it as a new sensor.

Read the following descriptions carefully to determine which option is right for you.

Keep the Sensor History

This is the easier of the two options. The sensor may be unplugged from its power source, stored for any amount of time, and plugged in again when it has a new home. If a sensor is going to be stored for an extended period, you should deactivate this sensor (which you can manage from the *Manage Sensors* page) so that it does not appear on the map on the home page of the website, but nothing bad will happen if this step is skipped. When the sensor is set up in its new home (and activated, if the sensor was deactivated in its storage time), it should just keep on collecting data as normal.

As with any new sensor, it will need to collect good data for several hours before it reports AQI values, but as long as it is getting $PM_{2.5}$ values from PurpleAir it should behave normally within 24 hours. The latitude and longitude of the sensor will automatically update both on the map on the home page of the website and in the *Manage Sensors* page of the admin pane.

An important thing to consider with this method of relocating a sensor is that the sensor will have readings in the database from both before and after it is relocated. This means that if you download this sensor's data, some rows of the CSV file will have the original latitude and longitude and some rows of the CSV file will have the new latitude and longitude. If you are using this data to perform analysis on $PM_{2.5}$ values over time, take care to note what date the sensor was moved as this could impact trends you notice. If having data from two distinct locations is not a concern for you, feel free to use this method for relocating a sensor. If you would rather keep the data separate from one another, follow the directions in Section A.4.3.

Remove and Re-Add the Sensor

The previous section explains why you might want to remove and re-add a sensor that is being relocated instead of simply plugging it in to a power source in a new location. If you would rather keep the data from before and after the move separate, follow the steps outlined here.

When the sensor is done recording at its original location, start by deactivating the sensor so that it no longer reports $PM_{2.5}$ values and disappears from the map. Then you will want to download the data for the sensor you are deactivating. This can be done with the *Download Data* button on the *Manage Sensors* page, as explained by Section A.4.6. Make sure to keep the generated CSV file somewhere safe so that it does not get lost. We recommend saving the file on your local computer as well as in a cloud storage solution such as Google Drive, Box, Microsoft OneDrive, or Dropbox.

Next, delete the sensor from the database using the *Delete Sensor* button on the *Manage Sensors* page, as outlined in Section A.4.5.

While the sensor is being moved or if it is sitting in storage, you should wait until the sensor is ready to collect data again before finishing the process. Once the sensor is ready, add it to the network using the *Add New Sensor* button, following the instructions of Section A.4.4. Then, the sensor should be ready to go and all the data in the database for the relocated sensor will be from after the sensor was moved.

A.4.4 Adding New Sensors to the Network

Adding new sensors is relevant whenever you want to add a sensor to your network. This sensor may be brand new, a sensor that you already have that has not been in use, or a sensor that you are relocating. To add a new sensor to the site, click the *Add New Sensor* button on the *Manage Sensors* page of the administrative pane. The site will prompt you to enter a PurpleAir ID for the sensor you want to add (Figure A.6a).

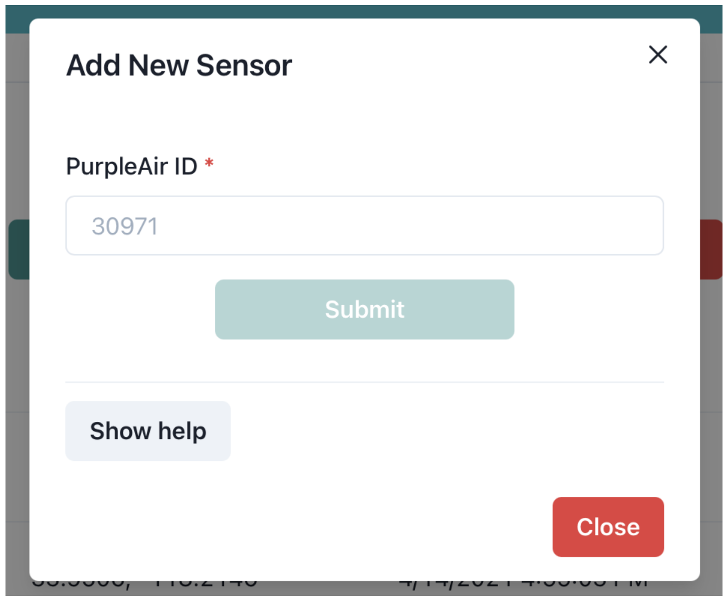
To find the PurpleAir ID, go to the map on PurpleAir's website and click on the sensor you want to add. On the pop-up for the sensor, hover over "Get This Widget", which is at the bottom of the pop-up. You should see something that looks like Figure A.7.

The box that appears when you hover over "Get This Widget" contains a white box with text. The PurpleAir ID of this sensor appears between ''PurpleAirWidget_" and "_module". In the case shown in Figure A.7, the PurpleAir ID is 39155.

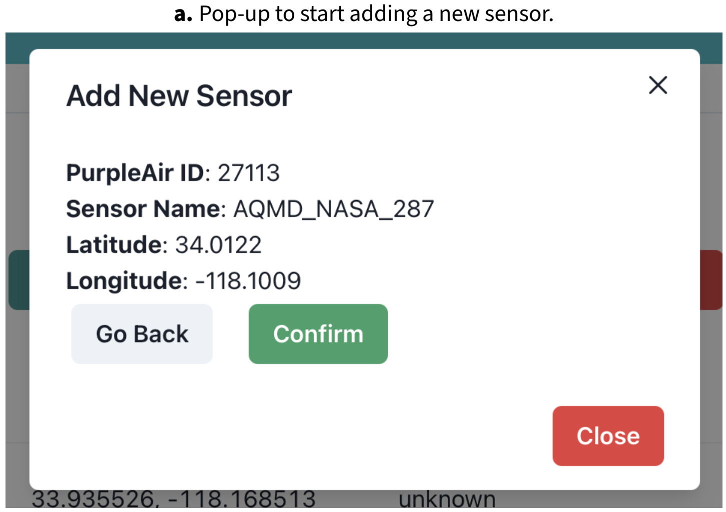
A sensor must be on PurpleAir's map to be added to the website. If you cannot find the sensor on the PurpleAir map, there is a good chance that something else is wrong with the sensor or the way it has been set up. You should troubleshoot this with PurpleAir, as it is unrelated to the CEHAT Air Quality website.

When you click submit, it will show you the sensor's name, latitude, and longitude (Figure A.6b). Confirm that these values match your expectations and press confirm to finish adding the sensor to the CEHAT's sensor network.

Because the NowCast AQI formula requires several hours of $PM_{2.5}$ data to calculate an AQI, it may take up to a day to start displaying an AQI, but



a. Pop-up to start adding a new sensor.



b. Confirmation page for adding a new sensor.

Figure A.6 Screens displayed while adding a new sensor.

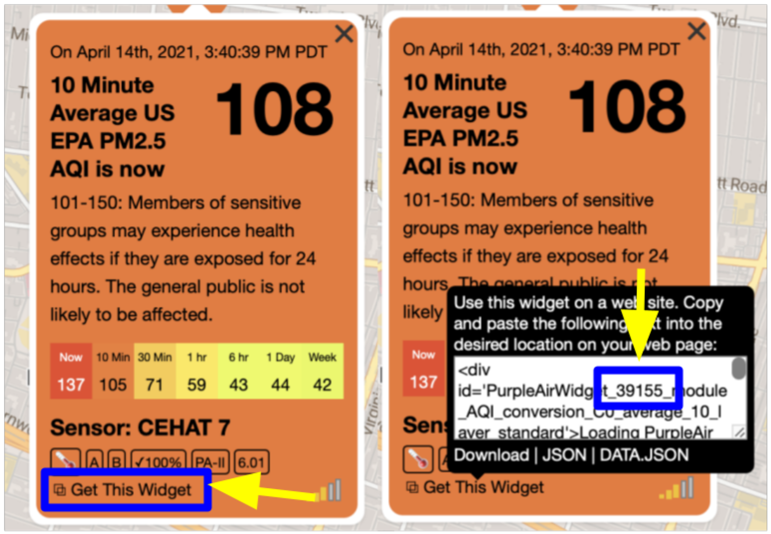


Figure A.7 Location on the PurpleAir website with a sensor's PurpleAir ID.

the sensor should be added to the map and the list of sensors on the *Manage Sensors* page within 30 minutes.

A.4.5 Removing Sensors from the Network

Removing sensors is relevant when you take down or relocate a sensor or if a sensor is broken beyond repair. If a sensor is no longer in use, it is desirable to delete it from your network so that its data does not contribute to your database storage costs and so administrators can more easily focus on the sensors that are in use. Since removing a sensor also deletes all its historical readings, it is critical that you first download the data for that sensor if you ever want to access the data again. Instructions for this are in Section A.4.6.

On the *Manage Sensors* page of the admin pane there is a red button labeled *Delete Sensor*. Make sure to read the paragraph on the pop-up (Figure A.8) to affirm that this is the action you are trying to execute. From the drop-down menu, you can select a sensor to delete. As an extra protection measure, administrators can only delete sensors that have been deactivated. If the sensor you are looking for is not in the drop-down menu, go back to the *Manage Sensors* page and deactivate it first by clicking the *Deactivate* button for the sensor. Now, the sensor will appear in the drop-down menu.

Once a sensor has been selected, its PurpleAir ID is displayed. Confirm that this is the PurpleAir ID of the sensor you are trying to delete and type

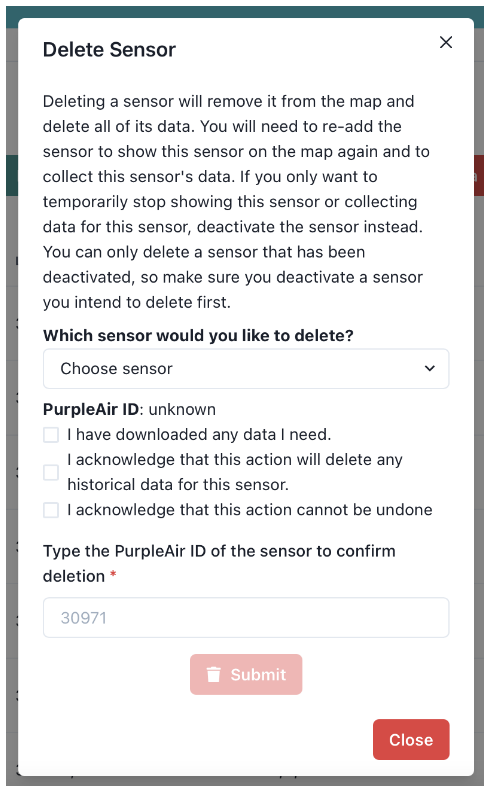


Figure A.8 Pop-up for deleting a sensor from the website and database.

the PurpleAir ID in the text box at the bottom of the pop-up. Next, read the prompts for the three check boxes and confirm that you have downloaded the data, and acknowledge that you are aware that deleting a sensor irreversibly deletes data from your database. Once all three check boxes are selected, click the *Submit* button to delete the sensor.

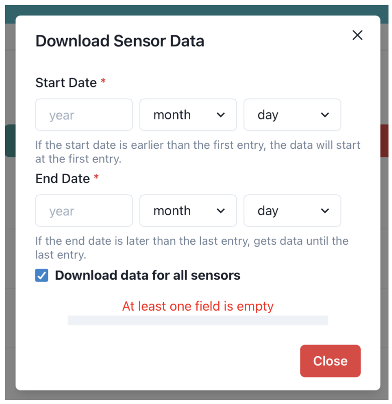
A.4.6 Downloading Sensor Data

The ability to download sensor data is an important feature of the admin pane. Use this feature by clicking the *Download Data* button on the *Manage Sensors* page. Data downloaded from the website is stored in a comma-separated values (CSV) file. These files can be viewed in spreadsheets applications such as Microsoft Excel, Google Sheets, and Apple Numbers. They can also be viewed in normal text editors where they will appear as lines of values separated by commas instead of as a spreadsheet of cells.

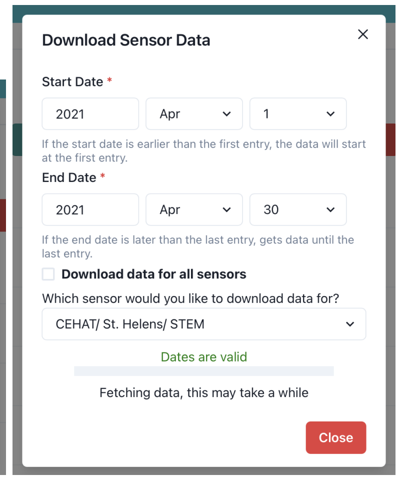
Clicking the *Download Data* button opens a pop-up that prompts you for start and end dates (Figure A.9a). The generated CSV will contain all $PM_{2.5}$ values in the database between those two dates. This means that you can download the data for a particular day, month, season, or year. The website will allow you to download data for large ranges of dates, but it is advisable to download data for at most a couple of months at a time to limit file size. Large files take longer to generate and download, and they can be more cumbersome when viewing them on your computer.

You have the option to download the data for all sensors in your database (including inactive sensors) or for a single sensor. The default selection is to download the data from all sensors. To download data for a single sensor simply uncheck the *Download data for all sensors* option. This reveals a drop-down menu from which you can select which sensor's data you want to download.

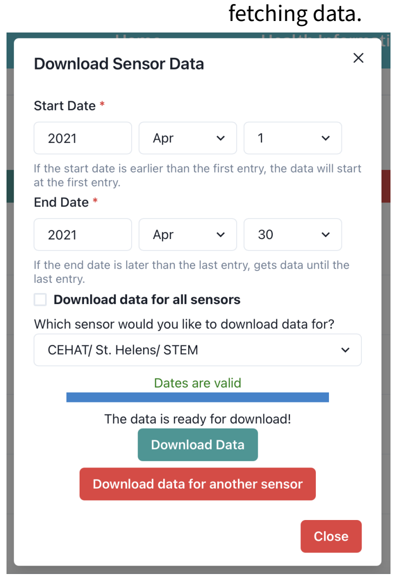
When your settings are configured, click the *Fetch Data* button. Once clicked, the *Fetch Data* button will be replaced with the message "Fetching data, this may take a while" and the progress bar will soon start to advance (Figure A.9b). Generating the CSV for a small amount of data will not take very long (for example, all sensors for few days or a single sensor for a month). Generating the CSV for a larger amount of data could take several minutes. Make sure the browser window stays open and your computer does not go to sleep in this time. For reference, on a 2019 MacBook Pro, downloading the data for all sensors for 3 months took between 5 and 10 minutes. If you find that you are waiting far too long for the data to finish



a. Initial pop-up to begin downloading sensor data.



b. Download sensor data pop-up while fetching data.



c. Download sensor data pop-up when data is ready to be downloaded.

Figure A.9 Stages of the downloading sensor data process.

pm25_2021-04-09T07_00_00.000Z_to_2021-04-12T07_00_00.000Z

Timestamp	Name	PM2.5	Percent Difference	Humidity	Latitude	Longitude
2021-04-09T07:01:27.000Z	CEHAT- St. Helens- STEM	8.4	0	54	33.9506	-118.2146
2021-04-09T07:03:27.000Z	CEHAT- St. Helens- STEM	7.6	0	54	33.9506	-118.2146
2021-04-09T07:05:27.000Z	CEHAT- St. Helens- STEM	6.8	0	54	33.9506	-118.2146
2021-04-09T07:07:27.000Z	CEHAT- St. Helens- STEM	8	0	54	33.9506	-118.2146
2021-04-09T07:09:27.000Z	CEHAT- St. Helens- STEM	8.5	0	54	33.9506	-118.2146
2021-04-09T07:11:27.000Z	CEHAT- St. Helens- STEM	7.9	0	54	33.9506	-118.2146

Figure A.10 Sample CSV file downloaded from the website viewed in a spreadsheet application.

"fetching", you can abort the process and by closing the pop-up and trying again with a much smaller date range. For this reason, we suggest one month as a reasonable date range for a single CSV file.

Once the website has finished generating the CSV, the "Fetching data..." message is replaced by "The data is ready for download!" and 2 buttons appear, *Download Data* and *Download data for another sensor* (Figure A.9c). You must click the *Download Data* button to download the data to your machine. The file will appear both in your browser (the location in your browser depends on which internet browser you use) and either in your computer's downloads folder or in the folder you chose to download the file to. The title of a downloaded file takes the form "pm25_start-timestamp_to_end-timestamp" if you download all data and "purple-air-id_pm25_start-timestamp_to_end-timestamp" if you download data for a specific sensor.

We strongly advise that you open every file you download as soon as you download it (as we do in Figure A.10) and take a quick glance to double check that it matches your expectations. Generating the data for download costs money (albeit small amounts: about $0.06 per 100,000 sensor readings), so you should avoid generating the same file multiple times.

If you want to download a second file (perhaps for another sensor or date range), you can click *Download data for another sensor* or close the pop-up and click *Download Data* again. Either of these actions will replace the download buttons with the *Fetch Data* button that allows you to generate a new CSV file.

A.4.7 Deleting Data from the Database

The website allows administrators to delete data from the database. The task of irreversibly deleting data may seem intimidating, but it is an important part of the maintenance for this site. The main reason to delete data from the database is to avoid storage costs from Google Firebase (the service that hosts the website and the database). You can learn more about when you should delete data in A.5.

The most important part about deleting data is making sure that you have downloaded all the data that is to be deleted before starting with the deletion process. These operations cannot happen simultaneously: make sure you download all the data, open the files and double check that they look right, and upload the data to a safe storage location **before** deleting the data. The steps for downloading the data are outlined in Section A.4.6.

Once the data has been downloaded and stored, delete data by clicking the *Delete Old Data* button on the *Manage Sensors* page. The pop-up (Figure A.11) contains two paragraphs of text that should be read every time data is to be deleted. Then, you must read and click all three checkboxes to confirm that you are aware of the action you are taking. Finally, you need to select a date that is no less than 7 days before the current date. We do not allow dates less than 7 days in the past to avoid deleting data that is relevant to calculating current AQI values, which ensures that there is no time where the website misbehaves because of missing data. Upon clicking the delete button, all data in the database before the specified date will be deleted. The back end of the website deletes data in batches once a day, not instantaneously, but this does not affect the front end of the website or the site maintenance as an administrator.

A.5 Maintenance Costs for the Website

In this section, we outline the main costs for the website and how the main website administrator (whoever has access to the Google Firebase Account) can check how much the website is costing. All components of the South Gate CEHAT Air Quality Dashboard are hosted via Google Firebase at https://console.firebase.google.com. Once you are logged in, you can navigate directly to the usage and billing page at https://console.firebase.google.com/u/0/project/sg-cehat-air-quality/usage.

The costs for the website fall under a few main categories: Cloud Firestore, Functions, Cloud Storage, and Hosting Storage. These costs are broken down by the Console in a helpful graphic like that of Figure A.12. Overall, you should expect the monthly bill to be less than $5.00, but this cost can go higher if a significant amount of data is being stored in the database. In the

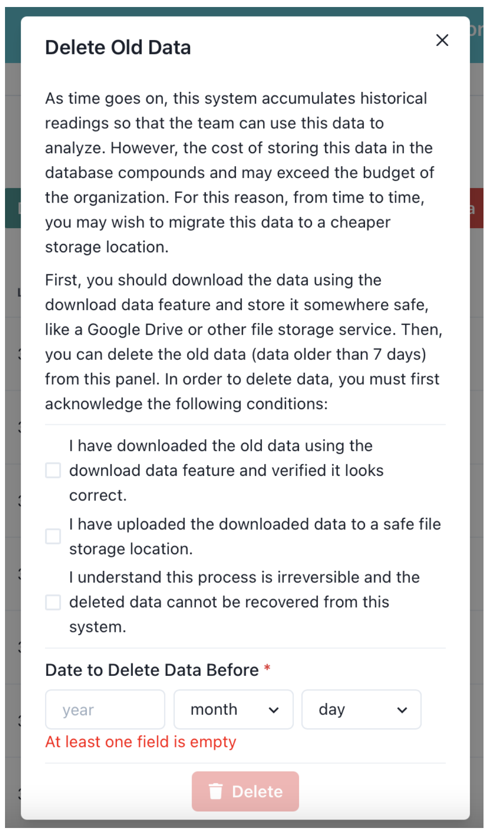


Figure A.11 Interface for deleting all data before a certain date.

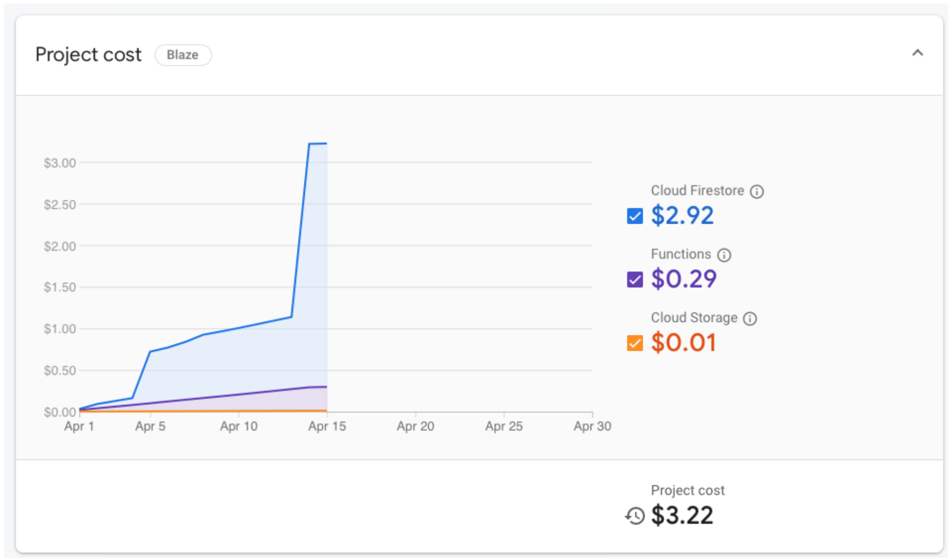


Figure A.12 Firebase Console graph outlining the total costs for the month of April 2021. This type of graph shows what parts of the website are costing the most.

subsections that follow, we describe the different costs.

A.5.1 Cloud Firestore

Cloud Firestore is the database that holds all the information you can access on the website. The costs for Firestore include:

- **Reading from and writing to the database**: these operations occur when users click around the website, when updating the database with new $PM_{2.5}$ values from PurpleAir, and when the website is updated with new AQI values.
- **Deleting from the database**: these operations occur when admins remove sensors or delete old data as outlined in Sections A.4.5 and A.4.7. Some months there will be no deletes (if no data was deleted by admins), but other months may have as many as hundreds of thousands delete operations (for example, if 6 months of old data is deleted).

Administrators cannot do anything to keep down the costs of reads and writes but can minimize delete costs by downloading and deleting old data more regularly so that it is done in smaller batches. For every 100,000 sensor readings that you download (which is about 1 month's worth of readings for 5 functional sensors), you will pay an additional $0.06 above your normal bill. For every 100,000 sensor readings that you delete, you will pay an additional $0.08 above your normal bill. Overall, since this has only a small impact on the total cost of the website, so we recommend deleting old data in whatever way works best for the admins, not to optimize cost.

However, note that we do recommend deleting old sensor data at least once per year in order to keep storage costs lower; this storage cost can turn into your **largest cost contributor**. Google gives you 1 GB of storage for free, but once the database exceeds that point, it will start to charge $0.18 per GB per month (cost accurate as of April 2021). This may not sound like much, but over time will compound and end up being the main cost of running the website. This is the main reason administrators should use the "Delete Old Data" functionality outlined in Section A.4.7. After several months (in the ballpark of 4 to 8 depending on how many sensors you have collecting data), the website will exceed the 1 GB limit. It is not essential that the data is immediately deleted, since the cost for storing extra data is only $0.18 per GB. However, it is important to keep an eye on what percentage of your free storage is being used so that you know when it might be a good time to download and save your old data and give the database a fresh start.

A.5.2 Functions

Functions refers to the Cloud Functions that are run every day to keep the database up to date. The two functions are the one that gets $PM_{2.5}$ values from PurpleAir every 2 minutes and the one that calculates updated AQI values every 10 minutes. These costs are constant, and administrators cannot mitigate them. Based on the cost in April 2021, we expect this cost to be less than a dollar each month.

A.5.3 Cloud Storage

This aspect of pricing is negligible (should not exceed a couple of cents per month) and can be ignored.

A.5.4 Hosting Storage

At the bottom of the usage page in the Firebase console under the heading "Hosting" there is a category called "Storage" pictured in Figure A.13. This

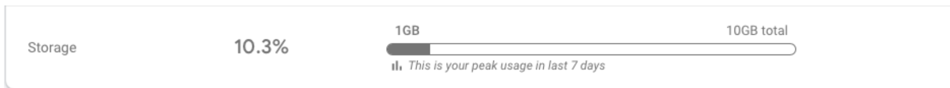


Figure A.13 Firebase Console graphic referring to the amount of storage being used by the database.

is the cost of hosting the website files on Firebase. The cost should be free or close to free, and it can be ignored.

Appendix B

Guide to Using the `Shiny` Application

To access the features in the `Shiny` application, visit https://southgatecehat.shinyapps.io/sgcehat-app. To access said features locally, it is necessary to install RStudio as well as know how to run the interactive app. In the following sections we discuss how to do this.

B.1 Software Installation

In order to run R and RStudio on your system, you need to follow the following three steps in order. Note that if you already have R installed, but it is a version that precedes 4.0.5, you will need to update R (to check your version, type `version` into the command line and hit "Enter"). To update R, visit this tutorial first before moving onto the steps in Section B.1.3 to download the necessary packages.

B.1.1 Installing R

- Based on the operating system you are using, download the binary setup file in R for Windows or the appropriate version of the .pkg file in R for Mac.
- Open the downloaded file, and install R.

B.1.2 Installing RStudio

Next, download RStudio desktop (the free, open source option), download the appropriate installer file for your operating system (Windows, macOS, Linux, etc.), and run it to install RStudio.

B.1.3 Installing Packages

You can follow the instructions below to install packages.

1. Run RStudio.

2. Click on the Packages tab in the bottom-right section and then click on install. The dialog box shown in Figure B.1 will appear.

3. Install the packages by copying and pasting the following into the "packages" line:
   `automap devtools dplyr DT futile.logger gridExtra lubridate plotly shiny shinythemes shinyWidgets testthat tidyverse tryCatchLog zoo`

4. To install the PurpleAirCEHAT package, copy the code from Figure B.2, paste it into the command line in the "Console" pane, and hit "Enter." If you are prompted to update 40 or so packages, press "Enter" again to skip this and continue. This will not affect the app's functionality, as you have already downloaded the necessary packages.

This completes the installation procedure for R Studio.

B.2 Running the Interactive Data Analysis App

1. Clone the app repository from GitHub, following these instructions from GitHub.

2. Open "app.R" in RStudio. To do this, either double-click on app.R in its specified folder location on your computer (which will automatically open RStudio), or open RStudio and do the following:

   - File → Open File. . . → Locate app.R in its specified folder and click on it once → Open

3. At this step, there may be some additional libraries that need to be installed. RStudio will inform you of this at the top of the document. Simply click on "Install" for each package that it requires.

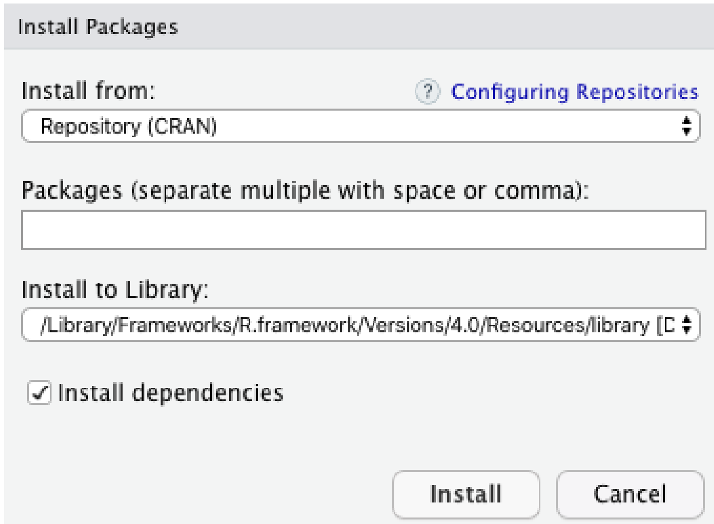


Figure B.1 Package installation dialog box in RStudio.

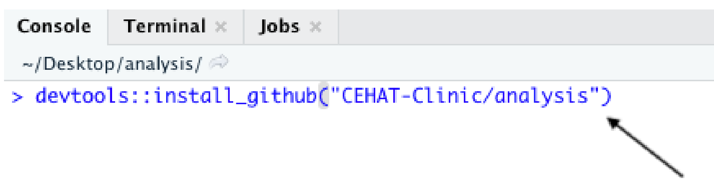


Figure B.2 How to install the package from GitHub.

4. Click "Run App" in the top right corner of the pane to run the application. Compare your screen to Figure B.3 for reference.

5. Upload data into the app to begin!

This completes the setup procedure for the Interactive Data Analysis App.

B.3 Interpreting the Graphs

As seen in Section 4.3, the visualizations on the `Shiny` app are categorized into one of the following sections: South Gate statistics, sensor summaries,

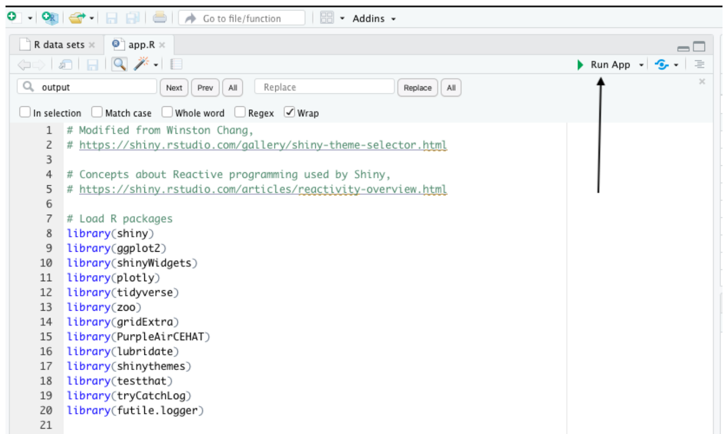


Figure B.3 How to run the app.

comparison, or interpolation and sensor placements. In this section, we will give a detailed interpretation of each plot available on the `Shiny` app.

B.3.1 South Gate Statistics

Summary

On this page, users can find a snapshot of South Gate's $PM_{2.5}$ values, with charts that represent general South Gate air quality data.

The *Summary Statistics* page includes charts that depict the general *hourly* air quality in South Gate for the 72 hours leading up to a date input specified by the user (shown in Figure 4.4b), as well as the general *daily* air quality for the 30 days leading up to a separate user-specified date (shown in Figure 4.4a).

The *Summary Statistics* page directly reports the percentiles of the data at 25, 33, 50, 66, 75, and 95 percent, the hour when most peak values are recorded, and the overall second maximum value for the uploaded dataset. The page also directly calculates the percentage breakdown of readings by their EPA AQI category.

In addition, the page documents the spread of peak values for $PM_{2.5}$ over 24 hours, as shown in Figure 4.4c

On the *Summary Statistics* page, users can also find the daily high and low $PM_{2.5}$ values in the form of a lollipop chart, which is shown in Figure 4.3.

The EPA sets the 24-hour standard for $PM_{2.5}$ to be 35 $\mu g\, m^{-3}$ per cubic meter, considering any levels above that threshold to be unsafe. The bar

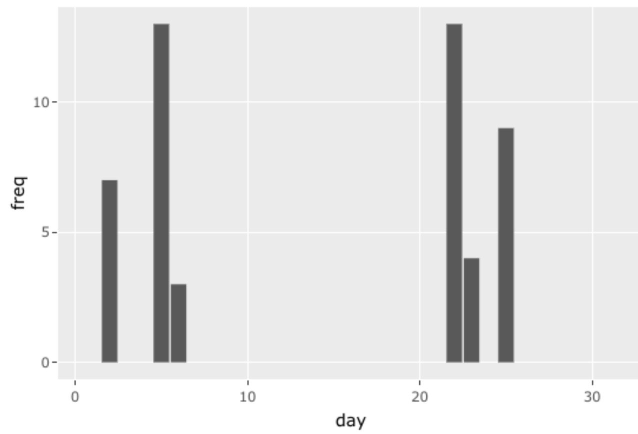


Figure B.4 Days over EPA threshold.

chart in Figure B.4 displays the number of $PM_{2.5}$ readings in South Gate that surpass this 24-hour standard for each day in the time period of the data. The datasets that we have been using to test the `Shiny` app do not contain readings that surpass the EPA threshold because they were recorded during the winter, and air quality is typically at its worse during the summer. Therefore, in order to populate the chart, we set the threshold to be 25 $\mu g\, m^{-3}$ per cubic meter instead of 35 $\mu g\, m^{-3}$ per cubic meter. The graph is interactive so hovering over a bar will display the specified day and the number of readings that exceeded the EPA threshold that day.

The *Summary Statistics* page also encourages users to compare the 8-hour rolling averages for two different time periods. The comparison in Figure 4.2a shows the 8-hour averages from December 1, 2020 to December 5, 2020 on the left, and the 8-hour averages from December 8, 2020 to December 12, 2020 on the right.

Diurnal Patterns

In this section of the app, users can explore the diurnal (day-night cycle) patterns of the data. For each graph, the diurnal pattern for a user-specified time period is shown in blue and is overlain onto the diurnal patterns for the observed time frame in the uploaded dataset. There are three diurnal graphs, each with near-identical format: mean, range, and peak/maximum (shown in Figure B.5). These are calculated by aggregating the general South Gate $PM_{2.5}$ data by hour for each of the aforementioned calculations.

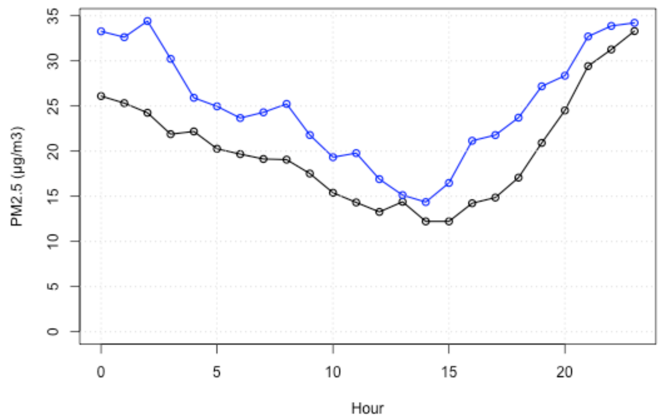


Figure B.5 The diurnal pattern for the peak $PM_{2.5}$ values in South Gate.

B.3.2 Sensor Summaries

On the *Sensor Summaries* page, users can find several charts that showcase different aspects of sensor functionality.

Network-based Observations

The plots in this section provide a holistic inspection of the sensor network, showing where sensors report higher (or lower) values, which sensors typically report outliers, and finally, how often each sensor is reported as "down" (i.e., not reporting sensor readings).

The chart in Figure 4.6b is an interactive plot that breaks down the $PM_{2.5}$ readings of each sensor by their corresponding EPA categories. In this chart, you can find the direct number of readings in each category for a particular sensor. This chart is magnified in the following section, **Single-sensor Observations**, which offers the same breakdown, but by percentage, for an individual sensor.

The chart shown in Figure B.6 reports the number of days in which a sensor did not report any $PM_{2.5}$ readings during the observed time frame of the given dataset. Note that for any sensors that are activated, or deactivated, during the observed time frame, the chart will report these changes. Hence, external information about the sensor network is required before accepting these results at their face-value.

The bar charts and spatial plots in Figures B.7a, B.7b, 4.5a, and 4.5b

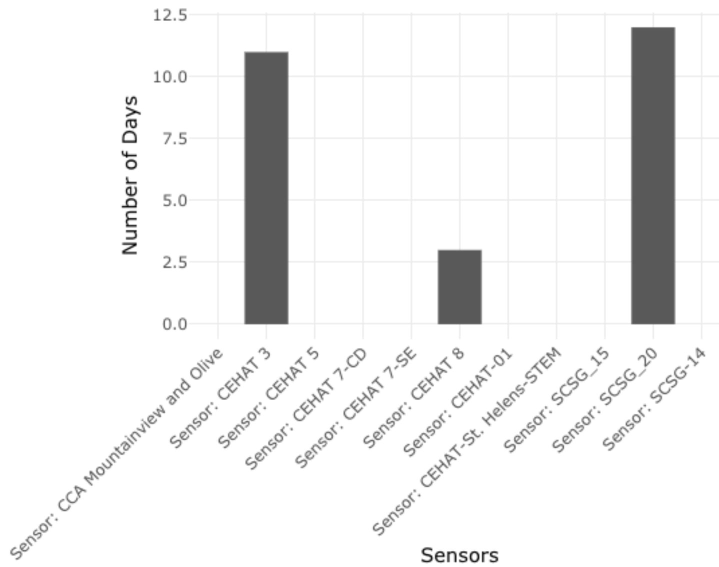
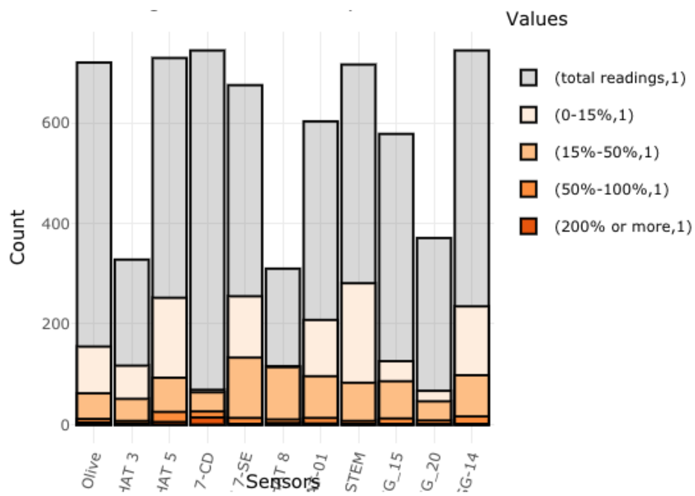


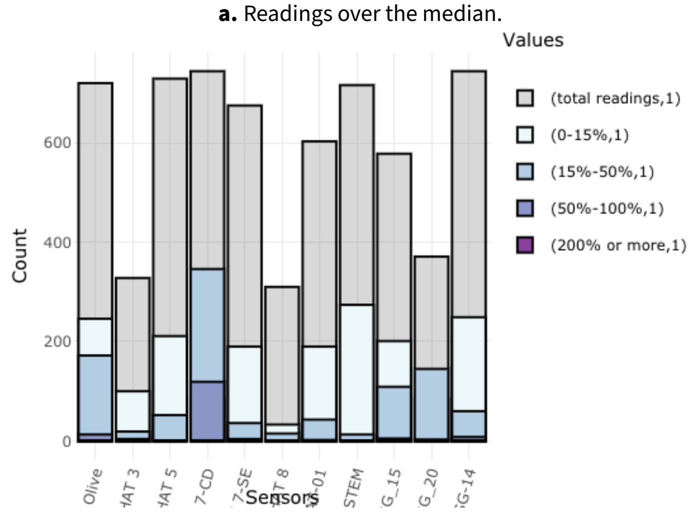
Figure B.6 Down days for all sensors in South Gate.

illustrate which sensors typically report higher or lower values compared to other sensors in the sensor network. Figures B.7a and B.7b report on the number of readings over/under the median that each sensor records, and the magnitude of this over/under in terms of percent difference. The median here is calculated at each hour for all the sensors in the network, and the readings are all taken to exclusively be over/under the median. The count of readings over the median for each sensor are further striated into categories of incremental percent difference that inform about ***how much*** greater, or lesser, than the median the observed readings are. This chart is useful for both identifying which sensors are typically reporting higher values, and of course, which sensors are consistently reporting outliers.

Figures B.7a and B.7b are paired with maps showing the corresponding spatial visualization for readings above/below the median; these map visualizations use the total count of readings over the median from the previous plot and normalizes them by the total readings collected. The map visualizations are shown in Figures 4.5a and 4.5b, included in Section 4.3.2.



a. Readings over the median.



b. Readings under the median.

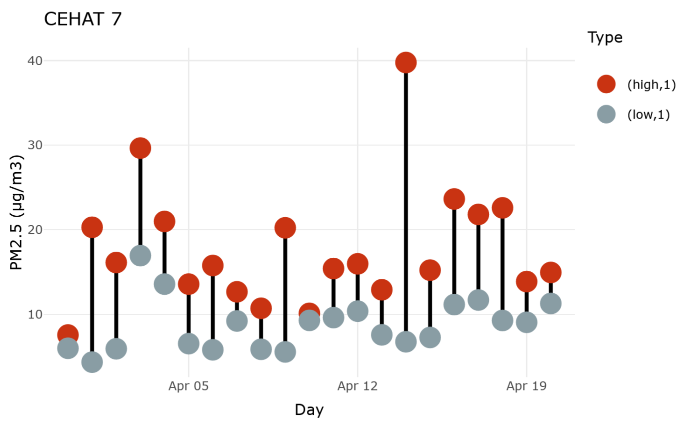


Figure B.8 Daily highs and lows for sensor CEHAT 7.

Single-sensor Observations

This section displays plots and observations for an individual sensor, which can be selected from the drop-down menu at the top of the sidebar on the *Sensor Summaries* page. In this section, you will find visualizations for high and low values, EPA categories by percentage, and historical percent difference data (for data downloaded after March 30, 2021).

The graph shown in Figure 4.6a displays the breakdown (by percentage) of a single sensor's readings by the corresponding EPA AQI category. This allows the user to compare the distribution of a sensor's readings to that of the general South Gate data, which is found on the *Summary Statistics* page.

The chart shown in Figure B.9 is a direct visualization of the mean percent difference field for a particular sensor. The method through which these values are calculated is discussed in Section 3.4.

The lollipop chart shown in Figure B.8 is a sensor-specific version of the high and low chart initially shown in Section B.3.1. This is accompanied by the histogram of high values of $PM_{2.5}$ shown in Figure B.10. This histogram is overlaid with a distribution curve to inspect normality (or near-normality) in the $PM_{2.5}$ data. This distribution helps inform how well the sensor is reporting data, and more immediately, if there may be significant environmental factors (or malfunctioning) that interfere with a sensor's reading.

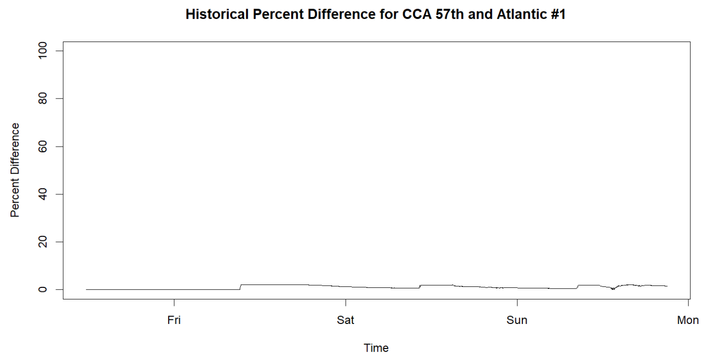


Figure B.9 Mean percent difference values for sensor CCA 57th and Atlantic #1.

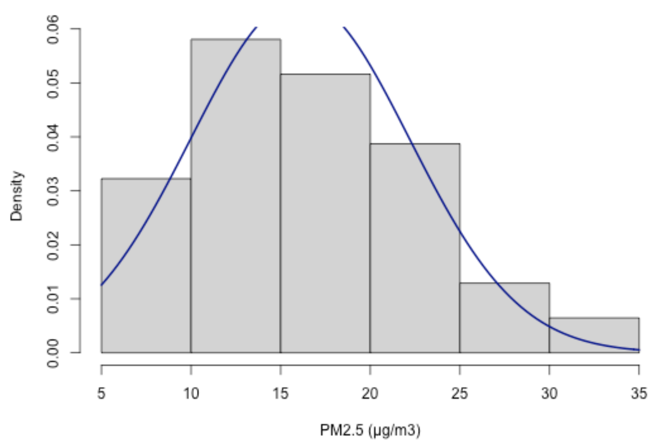


Figure B.10 Histogram of High $PM_{2.5}$ values for sensor CEHAT 7.

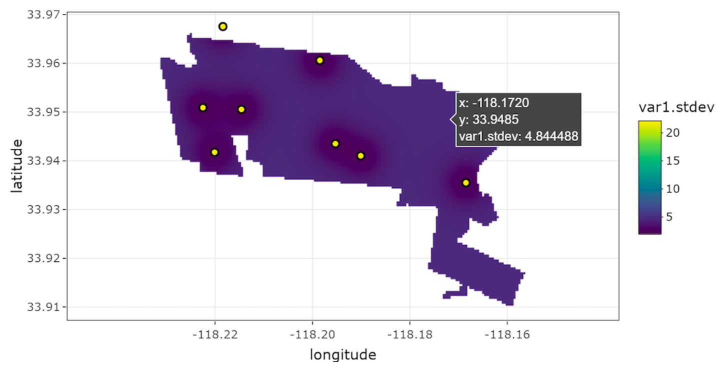


Figure B.11 Standard deviation of the ordinary kriging output computed in the `Shiny` app.

B.3.3 Comparisons

The box plots in Figure 4.7a compare the Indio $PM_{2.5}$ values (left) and the South Gate $PM_{2.5}$ values (right). The box-and-whisker graphics in the plot are separated into five sections. From bottom to top, they are the minimum, lower quartile, median, the upper quartile, the maximum. Users can see the specific values for each city's box plot by hovering over the plots.

The side-by-side bar charts in Figure 4.7b show the daily averages for $PM_{2.5}$ values in both Indio and South Gate with a legend specifying the corresponding colors. The interactive component of the `Shiny` app allows users to see both the day and the daily $PM_{2.5}$ average for every bar in the chart.

B.3.4 Interpolation and Sensor Placements

Ordinary Kriging

On the *Interpolation and Sensor Placements* page, users can find the three outputs of the ordinary kriging results calculated by the `Shiny` app. This includes the predictions along with the variance and corresponding standard deviation of those predictions at each mapped coordinate. The results of the prediction and variance calculations are shown in Figures 4.8a and 4.8b, respectively, while the standard deviation result is depicted in Figure B.11.

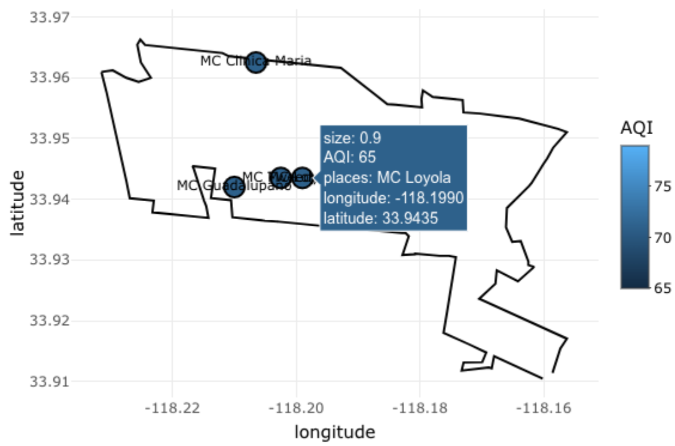


Figure B.12 Predicted AQI values at medical centers.

Sensitive Locations

On the *Interpolation and Sensor Placements* page, users can also find interpolation calculations regarding sensitive locations. We consider a place in South Gate to be a sensitive location if it is populated by people that are at greater risk for health problems, such as children, older adults, and people with pre-existing health conditions.

We provide the CEHAT with the ability to investigate the air quality near high-risk areas so that they can take measures to protect those sensitive groups, if necessary. The `Shiny` app predicts the AQI (which is defined in Section 2.3) values at 5 medical centers, 2 parks, 2 shopping centers, 3 senior centers, and 8 schools at a specific date and time using ordinary kriging.

Figure B.12, Figure B.14, and Figure B.15 display the positions of the sensitive locations within their respective category (with the exception of Figure B.13 which combines parks and shopping centers) on December 4, 2020 at 2 AM. These graphs are interactive, so hovering over a location reveals the predicted AQI value, the name of that location, and its longitude and latitude.

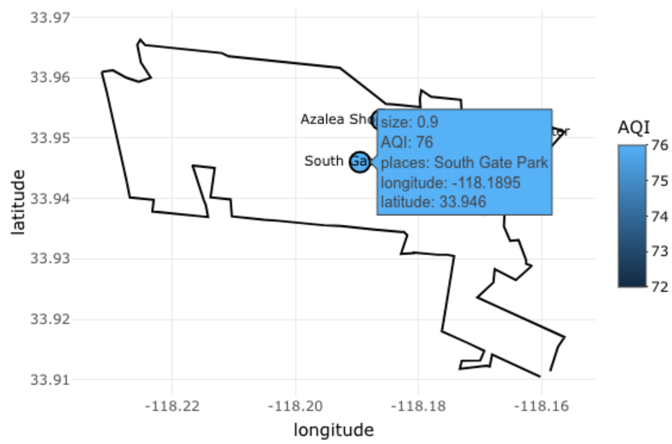


Figure B.13 Predicted AQI values at parks and shopping centers.

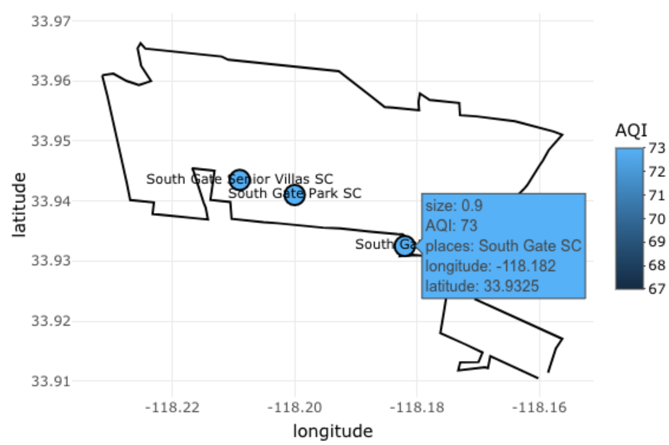


Figure B.14 Predicted AQI values at senior centers.

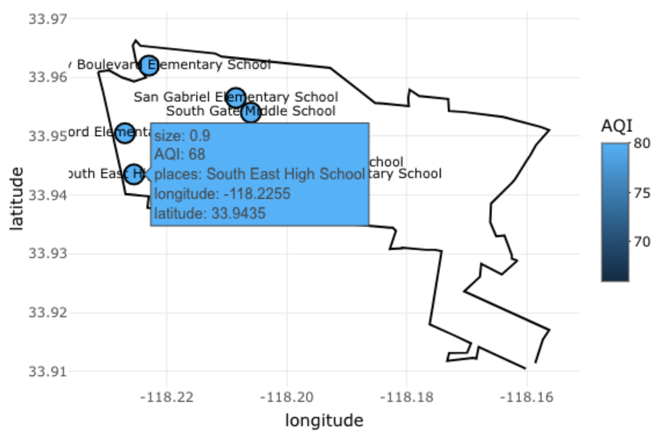


Figure B.15 Predicted AQI values at schools.

Bibliography

AirNow. 2020. How is the NowCast algorithm used to report current air quality? URL https://usepa.servicenowservices.com/airnow?id=kb_article_view&sysparm_article=KB0011856.

———. n.d.a. Air Quality Index (AQI) basics. URL https://www.airnow.gov/aqi/aqi-basics/.

———. n.d.b. Using the Air Quality Index. URL https://www.airnow.gov/aqi/aqi-basics/using-air-quality-index/.

Alameda Corridor Transportation Authority. 2021. Number of trains running on the Alameda Corridor. URL https://1popqd1sgf8034z1s33q7dj6-wpengine.netdna-ssl.com/wp-content/uploads/2021/01/CorridorTrainCounts.pdf.

Barkjohn, Karoline K., Amara L. Holder, Samuel Frederick, and Andrea L. Clements. 2020. PurpleAir PM2.5 U.S. correction and performance during smoke events. URL https://cfpub.epa.gov/si/si_public_record_report.cfm?dirEntryId=349513&Lab=CEMM&simplesearch=0&showcriteria=2&sortby=pubDate&timstype=&datebeginpublishedpresented=08/25/2018.

California State Transportation Agency. 2016. Annual average daily truck traffic on the California State Highway System. URL https://dot.ca.gov/-/media/dot-media/programs/traffic-operations/documents/census/f0017681-2016-aadt-truck-a11y.pdf.

Centers for Disease Control and Prevention. 2016. Community environmental health activism in South Gate, Los Angeles County, CA. URL https://blogs.cdc.gov/yourhealthyourenvironment/2016/02/11/community-environmental-health-activism-in-south-gate-los-angeles-county-ca/.

City of South Gate. n.d. City profile. URL http://www.cityofsouthgate.org/DocumentCenter/View/6474/City-Profile-Fact-Sheet.

Clean Air Technology Center. 1999. Nitrogen oxides (NO_x), why and how they are controlled. Tech. Rep. EPA-456/F-99-006R, United States Environmental Protection Agency. URL https://www3.epa.gov/ttncatc1/cica/files/fnoxdoc.pdf.

Los Angeles County Department of Public Health. 2018. City and community health profiles: South Gate. URL http://www.publichealth.lacounty.gov/ohae/docs/cchp/pdf/2018/SouthGate.pdf.

Moffitt, Mike. 2020. Feds create hybrid air quality site combining purpleair, airnow readings. URL https://www.sfgate.com/california-wildfires/article/PurpleAir-AirNow-Federal-hybrid-air-quality-site-15572827.php.

Office of Air Quality Planning and Standards. 2018. Technical assistance document for the reporting of daily air quality – the Air Quality Index (AQI). Tech. Rep. EPA 454/B-18-007, United States Environmental Protection Agency. URL https://www.airnow.gov/publications/air-quality-index/technical-assistance-document-for-reporting-the-daily-aqi/.

PurpleAir. 2021a. FAQ. URL https://www2.purpleair.com/community/faq.

———. 2021b. PurpleAir PA-II. URL https://www2.purpleair.com/products/purpleair-pa-ii.

South Coast Air Quality Management District. n.d.a. About. URL http://www.aqmd.gov/nav/about.

———. n.d.b. Current readings at air monitoring subregions. URL http://www.aqmd.gov/home/air-quality/current-air-quality-data/current-readings-at-air-monitoring-subregions.

UCLA Kaiser Permanente Center for Health Equity. 2019. Announcing the 2019 community seed grantees. URL https://healthequity.ucla.edu/news/announcing-2019-community-seed-grantees.

United States Environmental Protection Agency. n.d.a. Criteria air pollutants. URL https://www.epa.gov/criteria-air-pollutants.

———. n.d.b. Health and environmental effects of particulate matter (PM). URL https://www.epa.gov/pm-pollution/health-and-environmental-effects-particulate-matter-pm.

———. n.d.c. Superfund. URL https://www.epa.gov/superfund.