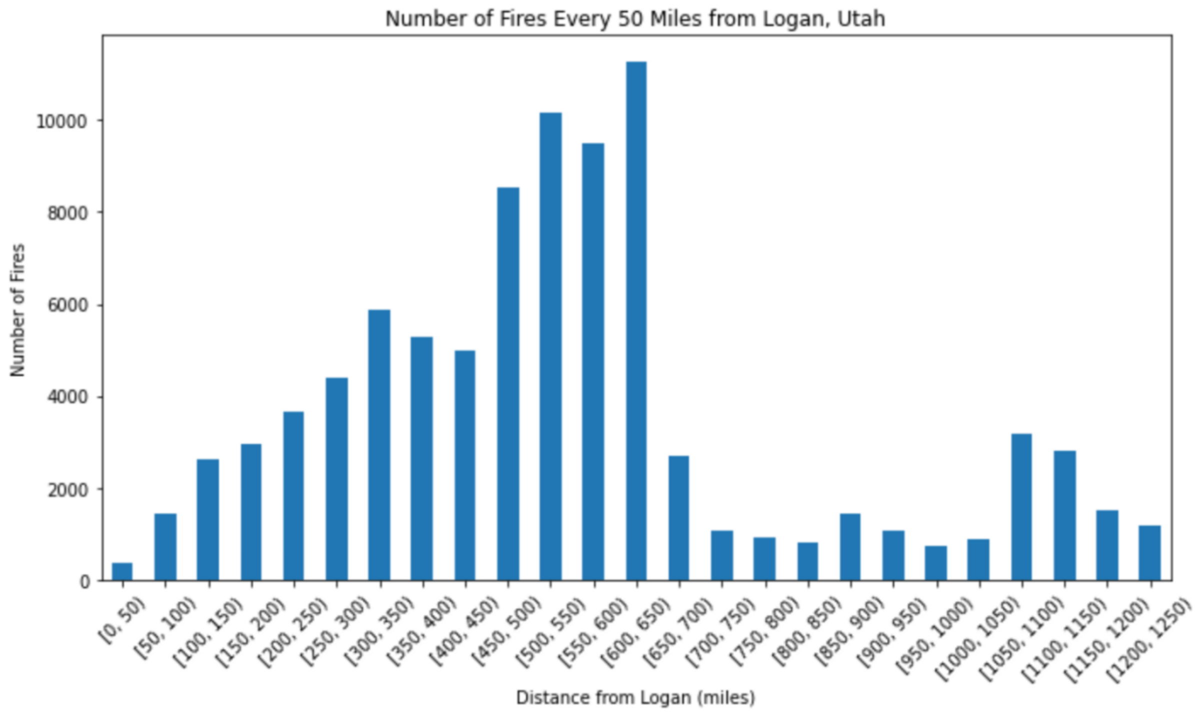
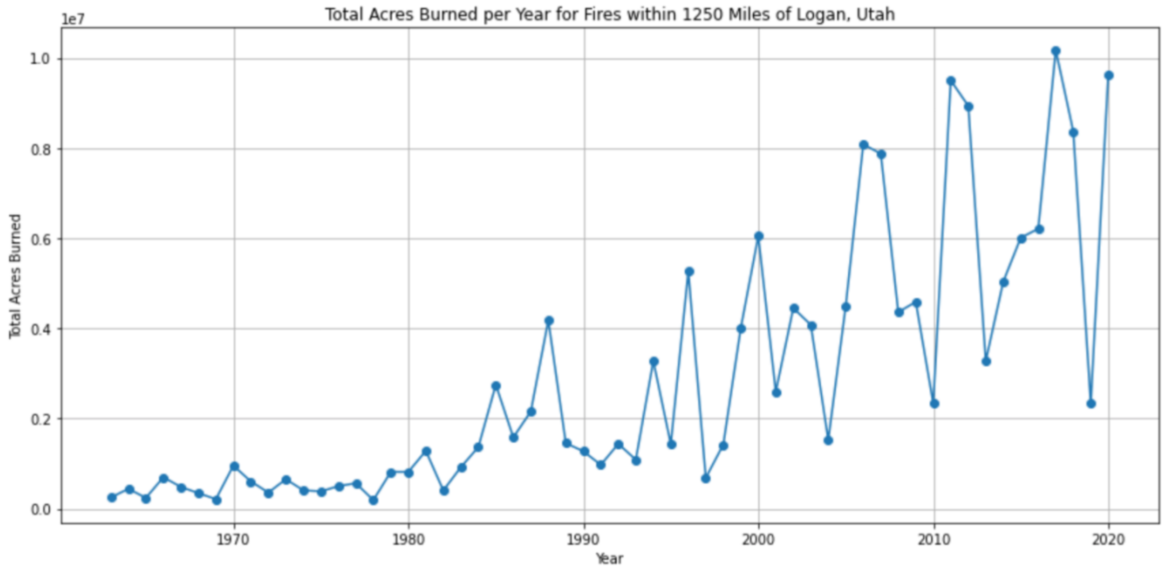
Visualization 1: Produce a histogram showing the number of fires occurring every 50 mile distance from your assigned city up to the max specified distance.



The x-axis of the bar chart represents the distance from Logan in miles, ranging from 50 to 1250 in increments of 50 miles. The y-axis represents the number of wildfires that occurred within each distance range. The best way to read this figure is to look at the x axis to get an idea of the ranges and then look at the y axis to see how many fires fit into each of those buckets. Each row in my dataset represents a fire and each fire has a specific distance from Logan, Utah. This distance was calculated between the coordinates of the city and the coordinate point within the fire polygon which is closest to the city. Only the fires with a distance less than or equal to 1250 miles were included in the final dataset. It is also important to note that before the 1990's, there was significantly less data available on wildfires and diligent documentation was not as prevalent. There could be some bias in the data if the nature of the landscape was different before 1990. This bias would skew the distribution of fires we see in the histogram.

We can see clearly that the bulk of wildfires that are within 1250 miles are within 650 miles of the city. There is also a range of wildfires that occur between 650 and 950 miles from Logan, with a slight decrease in frequency beyond 950 miles. It picks up again past 1000 miles and drops after 1100. This suggests that the proximity to forests, grasslands, and other fire-prone areas plays a significant role in the occurrence of wildfires near Logan.

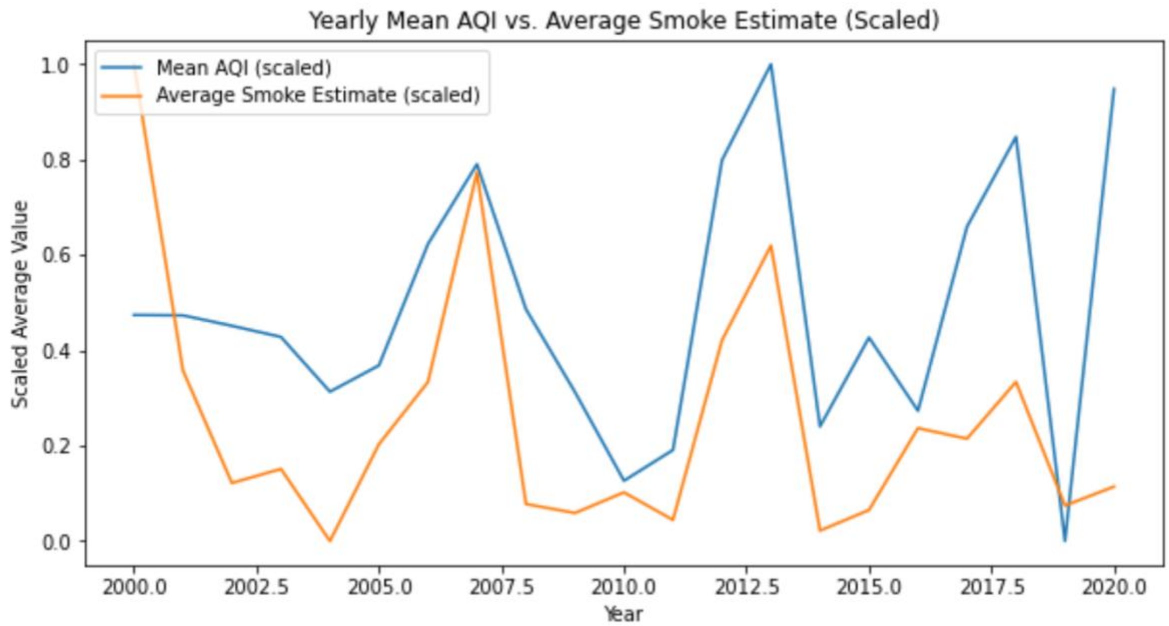
Visualization 2: Produce a time series graph of total acres burned per year for the fires occurring in the specified distance from your city.

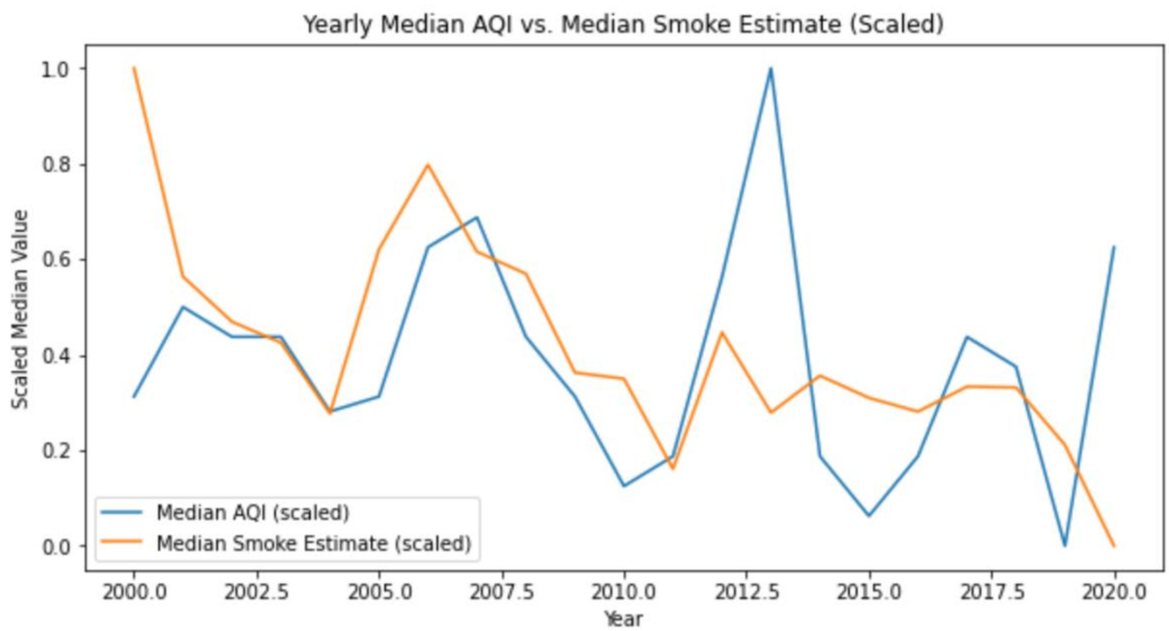


My graph depicts the total acres burned due to wildfires within 1250 miles of Logan, Utah, from 1970 to 2020. The visualization employs a line graph to showcase the trend of total acres burned over time. The x-axis represents the year, ranging from 1963 to 2020, while the y-axis represents the total acres burned in each year, with a maximum value of approximately 10,000 acres. The calculations behind this visualization were straightforward, our given dataset (the USGS Combined wildland fire dataset) contained a feature called GIS Acres that provided us the acreage of the fire; we only included fires that were within 1250 miles of Logan, Utah. This distance was calculated between the coordinates of the city and the coordinate point within the fire polygon which is closest to the city.

The line graph clearly demonstrates a significant increase in the total acres burned due to wildfires over the years. In 1970, only around 2000 acres were burned. However, the trend steadily increased throughout the 1980s and 1990s, reaching a peak of nearly 10,000,000 acres in 2020. This suggests that the frequency and intensity of wildfires in the region surrounding Logan have been on the rise.

Visualization 3: Plot 3: Produce a time series graph containing your fire smoke estimate for your city and the AQI estimate for your city.





My graphs showcase the relationship between the average AQI (Air Quality Index) and the average smoke estimate for the years 2000 to 2020. The visualization employs a line graph to display the trends for both metrics over time. The reason that is starts at 2000 is because that is when the estimates became available from the EPA for my specific city.

The x-axis represents the year, ranging from 2000 to 2020. The y-axes represent the average AQI in either median or mean. Both axis have been scaled because their original scales were vastly different and it would be difficult to make a comparison on the same plot. The effect of this scaling is to translate each feature individually such that it is in the given range on the training set, which is typically between zero and one. This has the benefit of bringing different features into a common scale while maintaining relationships in the data.

Let us discuss how we calculated the values. Starting with smoke estimates. The smoke impact score that I calculated is based on the size of the fire and its distance from a specific location, following the inverse square law of dispersion. This law suggests that the intensity of an effect such as smoke diminishes with the square of the distance from the source.

Here is the formula for the smoke impact score:

$$\text{Smoke Impact Score} = \frac{\text{Size of Fire}}{(\text{Distance from the City})^2}$$

In this formula:

- "Size of Fire" could be represented by the number of acres burned.
- "Distance from the City" is the distance between the fire's location and the city in question (in this case, Logan, Utah).

By applying this formula to each fire event, you can estimate the smoke impact on the city. The smoke impact score is higher for larger fires and those closer to the city, and it decreases as the distance increases. This score helps in understanding and estimating the potential smoke exposure to the city from various fires.

Every fire had one of these scores and I just simply took the mean and median of all of the fires for a given year.

Now let’s discuss the AQI score. When calling the API, this is the logic I used when deciding on which sensors to focus on:

The code is designed to retrieve daily summary data for particulate matter with a diameter of 2.5 micrometers or smaller, commonly referred to as PM2.5. The 'param' parameter in the API call is set to '88101', which is the code for PM2.5 in the EPA AQS database. PM2.5 is a significant pollutant often used as an indicator of air quality because these fine particles can penetrate deep into the lungs and even into the bloodstream, causing health issues. The concentration of PM2.5 is particularly relevant when estimating smoke impacts from wildfires, as these particles are a major component of wildfire smoke.

When looking at the two graphs, we can see how they follow each other pretty well until they reach 2012 and then it diverges more. This suggests that my smoke estimate and the AQI score are moderately similar in measuring the smoke impacts from wildfire.

Part 2 Reflection:

Working on this project felt like navigating a solo journey within a team framework. The assignment underscored the value of collaboration, yet it turned out that I spent a good chunk of time working through it on my own. This isn't to say teamwork wasn't important, but rather that the independent route I took had its own set of perks. I ended up really diving into the data and understanding it very well. Also I believe that being a part time student makes a bit more difficult to work with my peers and establish relationships. I have done so and made progress, but there is an inherit difference between being part time and being full time.

Delving into the complex data surrounding wildfires, I found that flying solo helped me really get to grips with the intricacies involved. I had the chance to test out a variety of ways to present the data, which is pretty crucial when you're trying to make sense of something this technical.

That said, going it alone didn't mean shunning all forms of help. I had my fair share of hurdles, especially when it came to coding. Here's where the internet came in handy—Stack Overflow became my go-to for untangling specific coding conundrums. There are many answers there to common problems that people have. And then there was GPT-4. This AI is very useful for getting snippets and searching through documentation. It is also useful for more verbose code.

It's worth noting that while I had all these resources, I was pretty determined not to just copy-paste my way through the assignment. It's easy to take someone else's code and call it a day, but I wanted to make sure I understood the mechanics of what I was writing. I was after the 'why' and the 'how', not just the 'what'. By building my code from the ground up, I got a clearer picture of what each line was doing and why it mattered for my project.

Using these external resources turned out to be a pretty strategic move. Stack Overflow had the collective wisdom of countless programmers who'd been in the same boat, and GPT-4 brought a level of coding insight that felt pretty futuristic. Plus, diving into the code examples that were handed out was like a masterclass in clean, efficient programming.

To wrap this up, taking the independent path through this project didn't mean shying away from help—it meant picking and choosing the right kind of help to boost my coding game. Stack Overflow, GPT-4, and those code examples didn't just make my work better; they gave me a crash course in independent problem-solving. This whole experience has been a solid reminder that the right resources can seriously level up your learning and help you churn out work you can be proud of, all on your own terms.