Saisriram Gurajala
Common Analysis
10/30/2024
DATA512

For this assignment, I focused on analyzing wildfire trends, generating a smoke impact metric, and forecasting this metric. Given the data-intensive nature of the task, collaboration with classmates allowed me to approach problems from different perspectives and refine both data filtering and model optimization techniques. A significant part of this process was optimizing API calls to efficiently pull in air quality index (AQI) data from the EPA's AQI API. Daniel Vogler and I worked together on debugging and refining these calls. Daniel's approach to organizing API calls sped up data retrieval considerably, and the streamlined API process ensured that we could gather and analyze air quality information without lag, improving the accuracy and timeliness of our results. I was able to help him untangle the logic of parsing multiple results from multiple pollutants per day to consolidate into a single daily AQI data point. Another important aspect of this project was optimizing the data filtering process. Alex and I worked together and decided to pre-filter the GeoJSON data using the "Fire_Year" field in the attributes. By setting up this initial filter, I reduced the number of records requiring distance computation. This change significantly reduced the amount of time it took to process the GeoJSON File, as the distance computation was the rate-limiting-factor in processing the wildfire dataset.

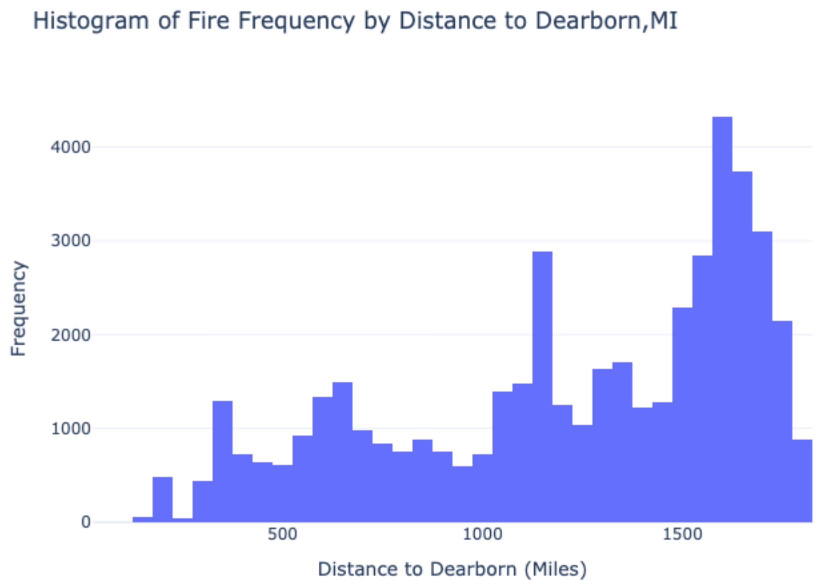
Daniel Vogler and Ed Szeryozhenkov helped me understand and apply the inverse-square law in constructing a smoke impact metric. The inverse-square law models the dispersion of smoke as a function of distance, enabling me to approximate the impact of wildfire smoke at varying proximities. Daniel and Ed's input was essential for calculating a metric that combines the area burned and distance from the wildfire source. This metric became the foundation for estimating smoke intensity and its potential effect on air quality in Dearborn. For the time series forecasting of the metric itself, Alex Netzley and Jake Flynn guided me in incorporating Holt-Winters Exponential Smoothing models. They recommended this approach to account for the seasonal patterns and volatility I observed in the smoke metric we had constructed. Working with Alex and Jake gave me hands-on exposure to implementing seasonal and trend adjustments within our model. They additionally gave me a code snippet to get me started with using the method in python, which I adapted heavily to better model my own data. This guidance highlighted the importance of using methods that account for seasonality when working with metrics derived from environmental data. These metrics naturally fluctuate over time resulting in volatile and noisy data that needs to be accounted for in modeling.

Saisriram Gurajala
Common Analysis
10/30/2024
DATA512

Through this collaboration, I learned not only the technical details of model implementation and data filtering but also how different aspects of the workflow are interdependent. For example, streamlining the API calls enabled me to access data more quickly, which allowed me to test the forecasting models in real time without delays. The pre-filtering approach to GeoJSON data made it possible to use more sophisticated forecasting methods without being bottlenecked at data loading and filtering. While navigating these different approaches sometimes required flexibility and adjustment, each contribution added value to the overall solution. Collaboration with classmates was instrumental in shaping my understanding of both forecasting and data processing. Their input expanded my thinking about data preprocessing, model selection, and the value of using techniques tailored to the data's unique patterns and seasonality. Working together not only led to a more effective forecasting solution but also broadened my perspective on how collaborative problem-solving can lead to more innovative and efficient approaches in data analysis.

Saisriram Gurajala
Common Analysis
10/30/2024
DATA512

Visualization 1: Histogram of Fire Occurrence within 1800 miles of Dearborn, Michigan

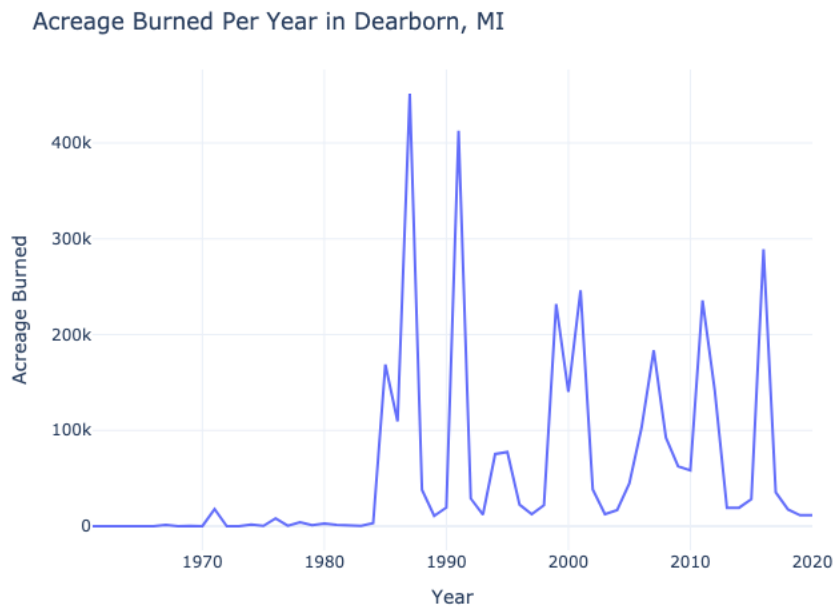


Description:

This graphic represents the occurrence of wildfires within 1800 miles of Dearborn, Michigan. The original data is from a GeoJSON file of wildfires from 1800s-Present. This data was processed through the following steps: the fires were restricted from the years 1961 to 2021. Additionally, distance from Dearborn, Michigan was assessed by finding the closest point on the polygon to Deaborn's latitudinal and longitudinal coordinates. The distance calculations themselves were carried out via the pyproj library in python and represent the geodesic distance between two selected points. Fires were additionally filtered to be "Wildfires" and not "Prescribed burns". The x-axis of this plot is the shortest distance from the fire to selected coordinates of Dearborn. Distances are bucketed into bins of width 50 miles. The y-axis is the frequency at which fires occur at a given distance. Looking into the graphic itself, the viewer can see the frequency of fires increases with distance. The highest concentration of fires occurs at greater than 1500 miles from Dearborn, and there's also a peak at around 1250 miles.

Saisriram Gurajala
Common Analysis
10/30/2024
DATA512

Visualization 2: Time Series Plot of Total Acreage Burned within 650 Miles of Dearborn, MI

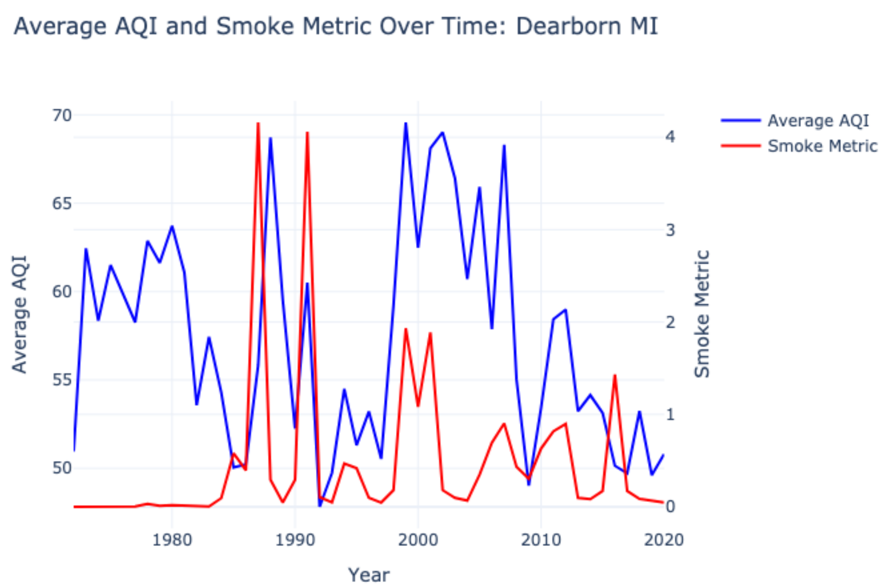


Description:

The preprocessing of data for this graphic is similar to the preprocessing described above. Briefly, the original data is from a GeoJSON file of wildfires from 1800s-Present. This data was processed through the following steps: the fires were restricted from the years 1961 to 2021. Additionally, the fires themselves were restricted to those within 650 miles of Dearborn, MI. Distance from Dearborn, Michigan was assessed by finding the closest point to Deaborn's latitudinal and longitudinal coordinates. Prescribed fire burns were discarded from this calculation, as these are intentional and not wildland fires. Distance calculation utilized the pyproj library in python and measured the geodesic distance between two selected points. The x-axis is the year between 1961 and 2020 (2021 had no data for the distance range). The y axis is the total acreage burned per year, summed over the filtered fires. Looking into the graphic itself, the viewer can see there tends to be specific spikes in the acreage burned. We can see before 1985 there were very few fires, although this may have occurred because of poor record keeping. The largest spike in acreage burned occurred around 1987, with other peaks around 2000 and 2016.

Saisriram Gurajala
Common Analysis
10/30/2024
DATA512

Visualization 3: AQI and Smoke Estimate Metrics from 1961-2021 in Dearborn, MI



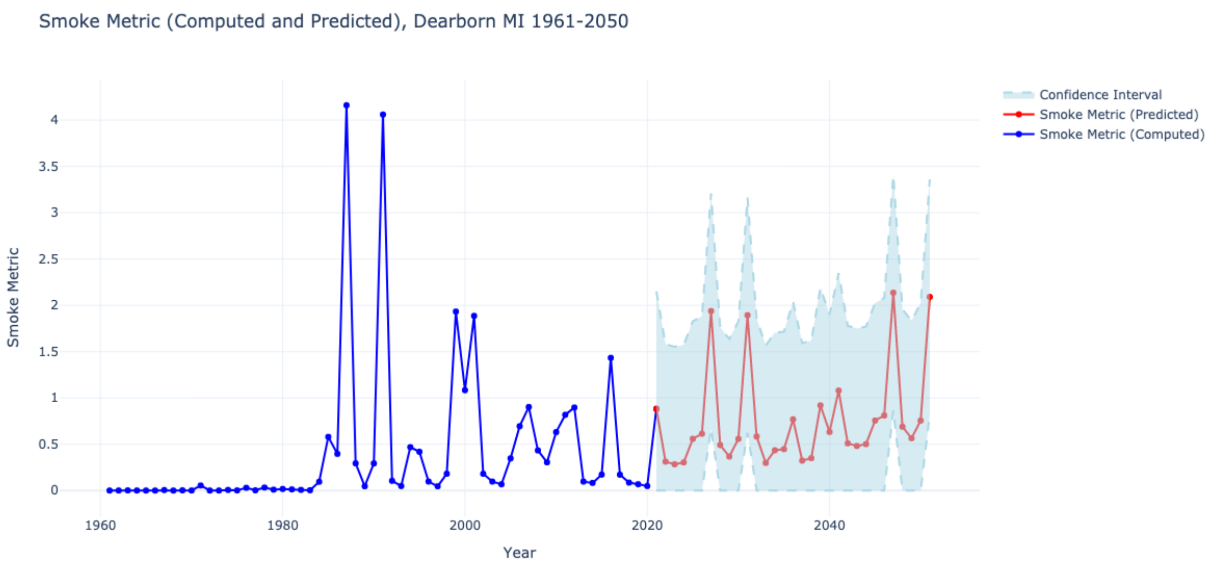
Description:

This graph contains Air Quality Index (AQI) and Smoke Metrics from the years 1961-2020 (No data available for fires in 2021). The filtering on fires was conducted as described in the above two visualizations. AQI was collected by calling the EPA's AQI API. Results from this API were filtered to be within the years 1961 and 2021 and filtered to be between May first and October thirty-first of each year to accurately estimate AQI during fire season. The AQI API provides a daily summary of air quality index corresponding to a given contaminant. Multiple stations report results for the same contaminant per day, so in these cases we took the average of the pollutant AQI per day. After that, to consolidate the AQI values from multiple pollutants per day into a single value, we took the maximum of the average AQI per pollutant per day in accordance the EPA's recommendations. The smoke metric is a function of two aspects of a given fire. I based the metric off the inverse-square law, modeling the intensity of light or gas across a given distance as a function of the inverse of the squared distance. The smoke metric itself is the ratio of the acres burned over the squared distance. To generate a smoke metric per year we summed the smoke impact over fires for a given year. The x axis is the year the metric has been computed for. There are two y axis: the left is the average AQI, and the right is the smoke metric for a given year. Before 1985, we don't see the smoke metric change much, likely due poor record keeping. However, after 1985, we see similar shifts in the smoke metric.

Saisriram Gurajala
Common Analysis
10/30/2024
DATA512

Although the magnitude in AQI change doesn't match the magnitude of change in the smoke metric, the general trend seems to hold. This observation is especially visible before and after 1990, and the two peaks around 2000.

Visualization 4: Computed and Forecasted Smoke Metric Values



Description:

This graphic is not strictly required within the specifications of the project, but it is a nice visualization of the forecasting work. The smoke metric's formulation and computation were defined above in visualization three's description. To forecast this metric, I employed an exponential smoothing time series modeling algorithm from Holtmann-Winters series of models. After conversations with Jake Flynn, I decided to use this algorithm due to its smoothing of noisy data for volatile datasets, matching the Smoke Metric is. The smoke metric was forecasted in python, using the statsmodel.api package. Looking at the trend prior, we can see that peaks and valleys occur regularly, about three to four years apart. Therefore, I assume some seasonality in the model. The x-axis is the year of the computed or predicted smoke metric, and the y-axis is the smoke metric value. The graph is divided into two subsets: the computed smoke metric, in blue, and the predicted smoke metric, in red. Around the predicted smoke metric is a blue dashed region, corresponding to a 95% confidence interval for the predictions. We can see looking at the graphic that the trend of smoke quality spikes and low smoke quality years continues until 2051.