

# Investigation of Atmospheric Arctic Wildfire Tracers at Summit, Greenland

Jashan Chopra<sup>1</sup>

**Abstract**—Ethane/methane and acetylene/methane ratios have no meaningful usage as atmospheric tracers of wildfires when measured from Summit, Greenland. This result stems from the statistical analysis of hybrid single-particle Lagrangian integrated back trajectories for each compound measurement value as they intersect NASA satellite fire counts. Sensitivity analysis between hourly time matching, scale of location matching, and seasonal differences were used to verify this conclusion. High uncertainty in identifying sources of outliers from harmonically fitted residuals with these ratios at high-latitudes still persists, and cannot be solely attributed to fire counts as shown here.

## INTRODUCTION

Climate change has been most pronounced in the Arctic. Increases in temperature and overall dryer conditions have led to more frequent and intense wildfires counts and burned areas. Resulting emissions, composed of particulates/soot and trace gases, have a positive feedback on the Arctic climate, exacerbating the already large warming seen at high-latitudes. Large uncertainties in quantitatively assessing the impact of wildfires at high latitudes are due to the scarcity of monitoring of wildfire atmospheric emissions and chemical impacts in the region. Atmospheric monitoring of wildfire emissions, including methane, and non-methane hydrocarbons (NMHC) is currently ongoing at the Greenland Environmental Observatory at Summit (GEOSummit). The Summit monitoring generates continuous data of atmospheric concentrations with hourly time resolution.

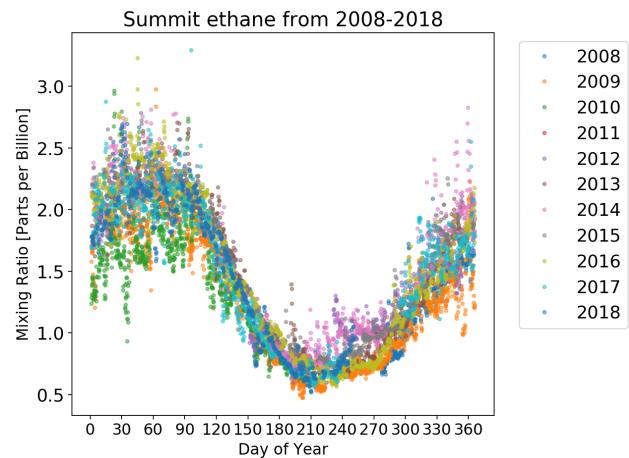
This project serves as an investigation to identify if these data can be used as a tracer of wildfires in the Arctic region. In particular, methane and the NMHCs ethane and acetylene are the central compounds explored in this research. As an undergraduate research project through the University of Colorado Boulder, the goal of this project was centrally to learn. Thus, throughout this report, skills I've developed alongside the research will be highlighted.

The main technical focus of the project was to develop Python code to analyze and interpret the data. Following initial analyses into the data was the cleaning and trimming of poor data. This included removing potentially polluted data from meteorological results, developing harmonic fits and identifying high residuals, and performing classical statistical analysis. The last step was creating the back

trajectories of Summit measurements, and finding overlap with NASA fire counts to draw a conclusion. Atmospheric chemistry, Python experience, and back trajectory analysis skills were all acquired on the job, with no prior experience.

## INITIAL DATA ANALYSIS AND HARMONIC FITTING

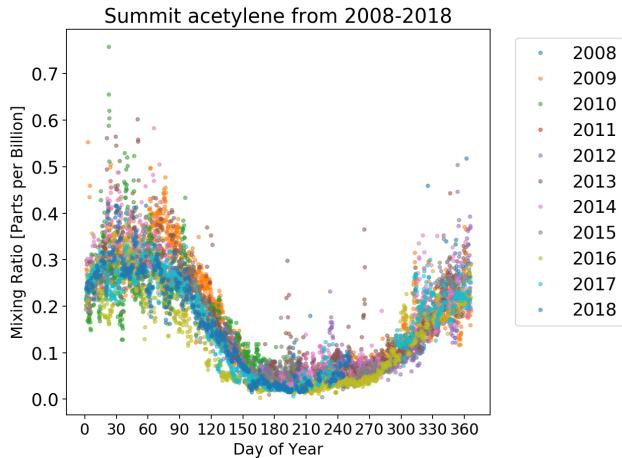
**Early Time Series and Ratio Creation:** Across all levels of Python experience and data analysis, the first step is creating a time series of the data. I developed a simple script to import past data of NMHC mixing ratios across time, convert the dates to the julian day format, and plot it.



**Fig. 1:** Ethane Mixing Ratio [ppb] from Summit, Greenland, measured from 2008-2018

After doing this for a single compound, I was able to convert the code into a function to run for each NMHC compound. As a side note, NMHC data is recorded at Summit, Greenland by a gas chromatography flame ionization detector (GC-FID). Outputted chromatograms are analyzed by peakSimple, which assigns an area and retention time to each peak to identify the compound. Peak areas below 0.10 volt-seconds are below the GC-FID limit, and register as zero for our analysis. In this case, they have been removed from the data. This data is displayed for ethane and acetylene in fig. 1 and fig. 2 respectively. These time series helped me understand many of the important features of the raw data. The first is the noticeable seasonal cycle, seeing higher mixing ratios in the winter, and lower mixing ratios in the summer.<sup>[3]</sup> We typically define the summer as Julian days 120 to 305, with the winter days being the values outside of that range. Also visible is a

<sup>1</sup>Institute of Arctic and Alpine Research (INSTAAR), University of Colorado, Boulder, USA



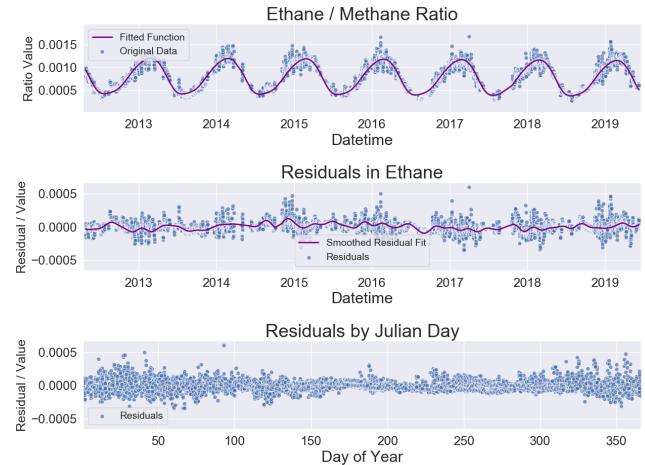
**Fig. 2:** Acetylene Mixing Ratio [ppb] from Summit, Greenland, measured from 2008-2018

greater number of potential outliers in the acetylene data. The analytical uncertainty is higher closer to the GC-FID detection limit, which helps explain this increase of outliers. Finally, these graphs helped explore differences between each year, and I started to see small peaks that could be a pollution plume, potentially from a wildfire.

To help normalize the differences between acetylene and ethane, I was informed to explore the ratio of acetylene/methane and ethane/methane. At Summit, Methane was previously sampled every three hours, eight times a day. The NMHC's were sampled less frequently, every eight hours, three times a day. These sampling rates changed following the data recording for this analysis. Because of this sampling difference, an average mixing ratio of methane in a  $\pm 3$  hour bracket of each NMHC date was used to create these ratios. Reference to any of these compounds past this point will refer to these ratio values.

**Harmonic Fitting:** To get a better sense of the data and perform more statistically advanced analyses, a fit of the data was created. This proved to be impossible with my experience, since the data is essentially a combination of three signals. It includes the long term trend, the nonsinusoidal yearly cycle, and short term variations.<sup>[8]</sup> I didn't know the complex fast Fourier transforms and fitting methods required for this. Thankfully, Kirk Thoning from the National Oceanic and Atmospheric Administration Earth System Research Laboratory's Global Monitoring Division (NOAA ESRL GMD) developed Python code to apply his curve fitting methods to imported datasets.<sup>[8]</sup>

One of the more difficult coding tasks I faced early on was dealing with the multitude of datetime formats and conversions for Kirk Thoning's CCGCRV tool, as it required an input text file with a very specific date format. Additionally challenging was optimizing my solutions to work with massive datasets without taking weeks to



**Fig. 3:** Fig A: Ethane/Methane ratio time series with fitted function. Fig B: Residuals of data from fitted function time series with smoothed residual fit. Fig C: Residuals time series by Julian day of year

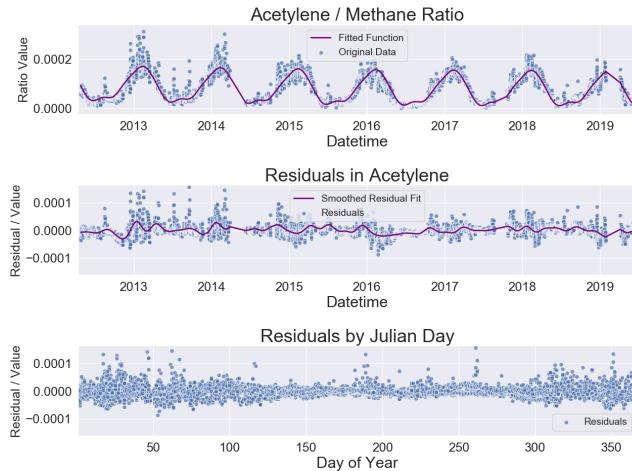
create dates. Eventually, I found a Python package called Numba. It takes basic NumPy array indexing calculations and converts them to extremely efficient C code. Using the power of this tool, I developed a series of functions to convert datetimes for large data sets between decimal year, Julian day, separated format, and datetime index format.

Throughout this whole process, I was manually integrating new data from Summit, quality checking it, and adding it to yearly spreadsheets. I incorporated this new data in the updated analyses, did the cleanup and feature engineering required, and then used the CCGCRV tool to create these datasets with the harmonic fitted function. More information on the specific coding processes outlined here, along with the sources of data are presented in Appendix A and B respectively. This data is displayed for ethane and acetylene in fig. 3 and fig. 4 respectively.

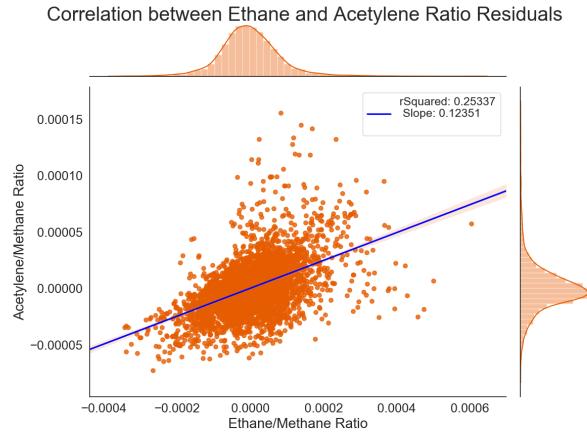
The final analysis on the two ratios alone was to create a linear regression plot comparing their residuals, as seen in fig. 5. The correlation coefficient between the residuals of the two ratios was only 0.2537. There are plenty of times we see a spike in the ethane values but have little change in acetylene, and vice versa. Generally this means their sources and sinks differ heavily, which is a first sign that these two compounds may not be ideal when used in conjunction to trace a fire. This can also be affected by topics not addressed in this paper such as boundary layer dynamics during mixing and transport of the atmospheric compounds.

## IDENTIFYING AND REMOVING CAMP POLLUTION FROM METEOROLOGICAL DATA

**Meteorological Data Analysis:** The next step after the initial importation and exploration of the ethane and



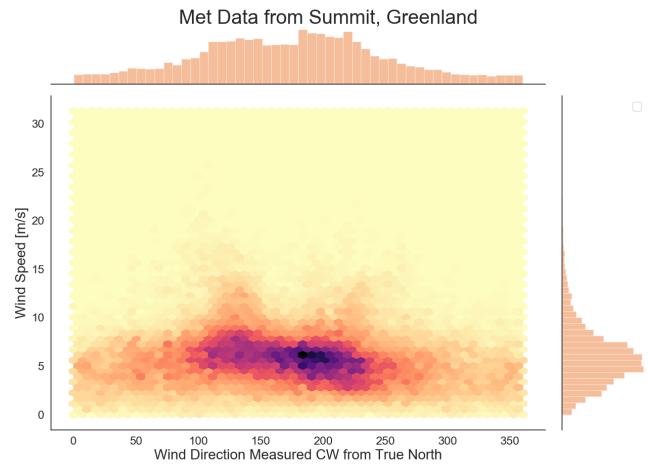
**Fig. 4:** Fig A: Acetylene/Methane ratio time series with fitted function. Fig B: Residuals of data from fitted function time series with smoothed residual fit. Fig C: Residuals time series by Julian day of year



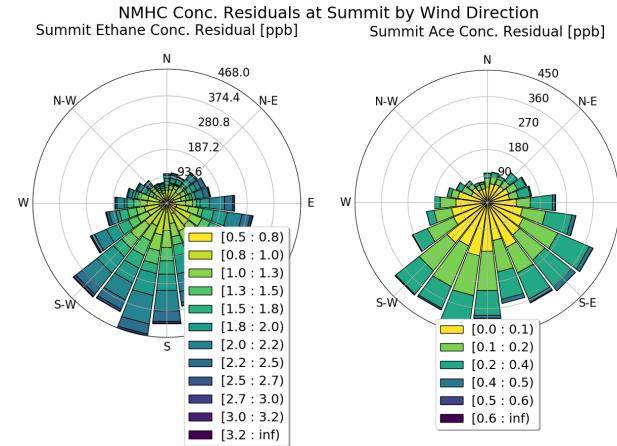
**Fig. 5:** A linear regression plot of the ethane/methane ratio residuals to the acetylene/methane ratio residuals.

acetylene ratios was to trim and clean the data. Since I was interested in the outliers, I had to make sure that any outliers were a spike in concentration from a potential pollution event, and not local pollution close to the Temporary Atmospheric Watch Observatory (TAWO) measurement collection facility. To do this, I imported NOAA meteorological data from Summit, Greenland and used it to identify which atmospheric data I should trim from the original data set.<sup>[4]</sup> The source of this data can be found in Appendix B.

First, I created a heat map to give me a general sense of typical wind directions and wind speeds in this meteorological data. We see from fig. 6 that a majority of data comes from the south, between 170 and 210 degrees, measured clockwise from true north. We also see that most data is between 5 to 7 meters per second. As per the Summit clean air management plan, air at the TAWO building is



**Fig. 6:** A distribution plot of wind direction v. wind speed from meteorological data at Summit, Greenland

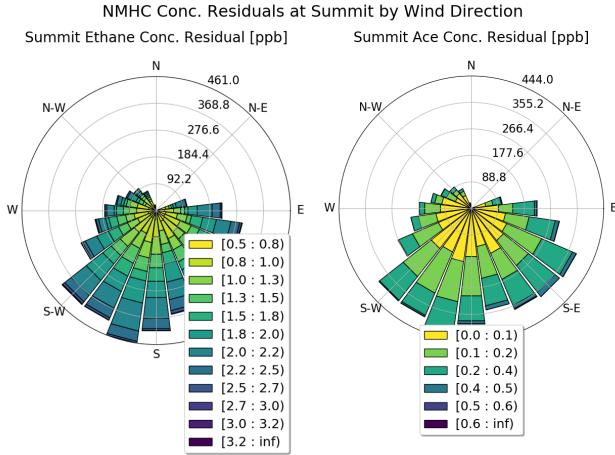


**Fig. 7:** Windrose of ratio residuals before trimming values

considered to be impacted by local sources when the wind direction is between 342 and 72 degrees (i.e., from the NNE) or when the wind speed is less than 1.03 meters per second.

**Windrose Plotting:** To check how much of my original data could be impacted by pollution at the collection site, I created a windrose plot of the atmospheric concentrations using the same meteorological data. The windrose plot displays which direction the wind travels from. The size of the bar displays the number of values in each slice, similar to a histogram. The color of the bar displays the actual ratio's residual value. Documentation on the windrose plotting can be found in Appendix A.

Lastly, I wrote a function that combines the atmospheric data with the meteorological data and drops values flagged as impacted by potential local pollution. The before and after windrose plots can be seen in fig. 7 and fig. 8 respectively. These windrose plot shows that a vast majority of data is from the south, and will not be impacted. However, there is



**Fig. 8:** Windrose of ratio residuals after trimming values

some data in the impacted quadrants described by the clean air management plan.

**Removal of Winter Cycle:** To explore the usage of these tracers with wildfires, I decided to remove data from winter months for the central analysis. Specifically, data was removed during the Julian days of 120 to 305. Wildfires happen less frequently in the winter<sup>[1]</sup>, and these trajectories simply take space on the final graph. The differences between summer and winter data is still addressed in the discussion. All of these modifications to the original dataset are performed by a single script, combining multiple functions that I've written throughout the course of my Python development. This script also handles minor data trimming details such as further removal of zero or NaN values, assigning appropriate headers, and resetting the index for further analysis.

#### CREATING BACK TRAJECTORIES AND OVERLAYING WITH FIRE COUNTS

**Back Trajectories:** Following all the initial analysis, cleaning, and prepping, we now have the final data product. It consists of the ethane/methane and acetylene/methane ratio values, the residuals, the function fit, and properly formatted datetimes. The next step was to create back trajectories originating from the TAWO building at Summit, Greenland for each data point.

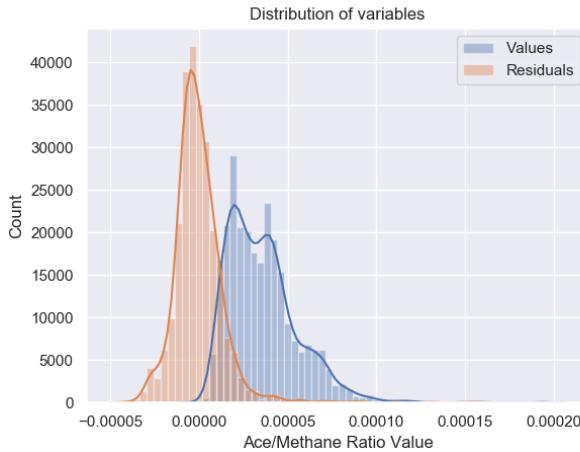
To create the back trajectories I used NOAA's Hybrid Single-Particle Lagrangian Integrated Trajectory model (HYSPLIT).<sup>[7]</sup> This model identifies air parcel trajectories, essentially uniform and small volumes of air, and follows them back in time using meteorological data. To use the model, I employed the help of PySPLIT, a python environment for working with HYSPLIT, developed by Mellissa S.C. Warner.<sup>[9]</sup> Further modifications to PySPLIT were made here at INSTAAR by Brendan Blanchard in order to run long periods of back trajectories in chunks. Finally, I made modifications to the software to use NOAA meteorological data from the Global Data Assimilation

System (GDAS) at the 0.5° resolution<sup>[5]</sup>, and adjusted for my specific datetimes. I chose to run back trajectories 72 hours in the past to cover a large region of potential fire sources, anything longer would incorporate too much uncertainty into the analysis. Additionally, the starting altitude was estimated at 50 meters.

**Fire Counts:** After creating the back trajectories, the next step was to see if they linked up with wildfires. The fire products used were from the NASA Visible Infrared Imaging Radiometer Suite (VIIRS).<sup>[6]</sup> Only reported fires with high confidence were used, and from past verified data. Similarly, near real time data was neglected in order to remain positive that only wildfires or large fire sources being detected. In order to link the fires up with the back trajectory, I cross checked the latitudes and longitudes to a one degree accuracy ( 11 kilometers). Other results are reported for a one-tenth degree accuracy ( 1.2 kilometers) in the discussion. If a longitude and latitude match existed, I then verified if the time of the fire product and the back trajectory were within a period of 24 hours. Thus, a match is completely defined as a back trajectory crossing over a fire within the same day and within one degree latitude and longitude difference. These matching conditions were extremely generous to compensate for any potential error in the times and back tracing of tracers. Even with these loose boundary conditions, we only see a few matches. More information on all of these models and data can be found in Appendix B.

**Statistical Inferences & Plotting:** An initial hypothesis I had was that the wildfires would be linked to peaks in the values of either of the explored ratios. To check this condition I used the z-score, a statistical parameter that determines number of standard deviations away from the mean. In particular, I was more interested in the acetylene/methane data as acetylene is a primary emission from forest fires.<sup>[1]</sup> However, when looking at the distribution of the raw ratio values we see a somewhat bimodal distribution, which can be concerning to use normal standard deviations with. The distribution of residuals from the fitted function follow a better normal distribution though, so the z-scores were calculated from this parameter. These distributions can be seen in fig. 9

For a display of all these features together, I used the Cartopy Python library, specifically designed for geospatial data processing.<sup>[2]</sup> I centered the graph on the longitude, latitude coordinates of Summit, Greenland, (-38.4592, 72.5796). The globe background is a nearside perspective projection, with any points being a geodetic projection, and any lines being Plate Carree projections to preserve real distances. Back trajectories are colored with a diverging red and blue colorbar with relation to the magnitude of the z-score for the tracer measurement that each particular back trajectory originates from. Each fire product is plotted with a small orange dot, and identified fire



**Fig. 9:** Distribution of values and residuals from harmonic fitted function of acetylene/methane ratio

matches are plotted with a larger purple star. Throughout this process statistical data was calculated on the average z-score of all the trajectories and the average z-score of the matched trajectories, reported in the discussion.

## DISCUSSION & CONCLUSIONS

At first glance of fig. 10 we notice that matches between a fire product and a back trajectory are limited, especially considering the immense count of both of the products over the six plus year timespan. The number of matches and their percentage as a whole of both fire counts and trajectories is found in table. I. The graph displayed was the one that presented the most number of matches, at just over 3% of all trajectories. The conditions for this graph, as noted, were extremely broad. The time frame where a trajectory will pick up fire particulates is uncertain, but 24 hours is still a huge range. Also seen from the table is that the number of matches between acetylene/methane and ethane/methane is similar. It is known that acetylene is a more prominent fire tracer<sup>[1]</sup>, which is why we see the slight lead in matches for that compound under the same conditions. The globe plot for ethane/methane is not displayed, but statistics for it are reported in tables. I and II.

Beyond the low number of matches for the given conditions, we also see that there is little correlation between the matches and high z-score values. Presumably, if the z-score of the matches was significantly higher, we would see that peaks in the data may indeed come when the trajectory passes over a wildfire. This quantitative data can be found in table II. For the acetylene/methane ratio, we do see the matches have a slightly higher average z-score, but this is reversed for the ethane/methane ratio. When the latitude and longitude location matches are tightened to within a tenth of a degree, we see no matches in each scenario, with the exception of acetylene/methane in the summer. In this single case, we can see that the average

z-score of the matches is less than the overall average. This shows that even for the few matches found, they don't correlate with spikes in the data. The last varying parameter was seasonal data. The primary graphs use only summer data because of the nature of wildfires in these regions. When testing winter data, the number of fires drops significantly, and we see no matches between trajectories and fires even for broad conditions.

From this preliminary investigation we can draw the conclusion that the acetylene/methane and ethane/methane ratios analyzed here are not strong tracers of wildfires for the Summit, Greenland location. This is backed up by the low number of matches over a long period of time. No positive correlations between the ratio residual's z-scores and actual wildfires can be asserted. There is no strong increase in average match z-score between data modification variations.

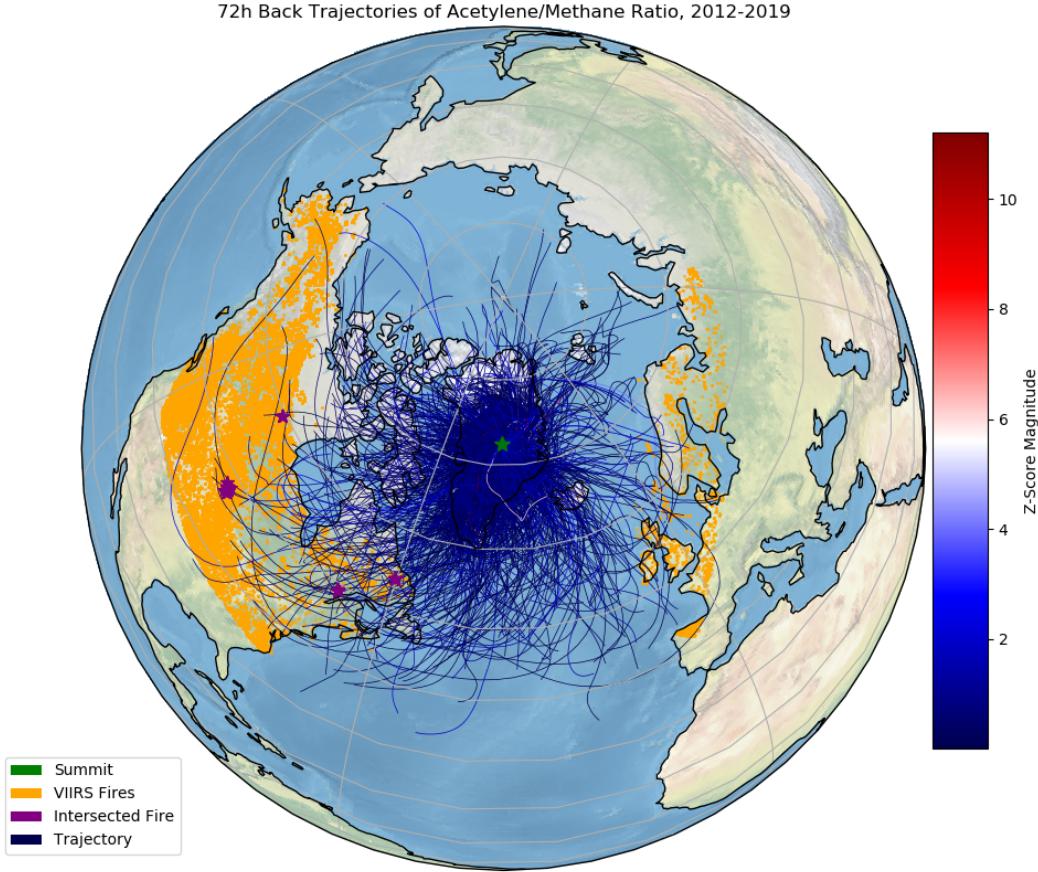
By no means is this a fully conclusive result about the usage of acetylene/methane and ethane/methane ratios as atmospheric tracers of wildfires at other locations. There are a number of uncertainties and further questions to be asked. There is a chance that the Summit, Greenland site is not the best measuring site to trace wildfires. Fires happen infrequently in Greenland. To track fires in more common locations like North America, the Lagrangian integrated trajectories needed to go back in time up to 72 hours. It is uncertain if particles from fires would follow these back trajectories perfectly to Summit because of complicated boundary layer dynamics and various mixing/transporting effects. Although a sensitivity analysis was performed on the time comparison variable, it is still unknown how close in time a fire must be to the trajectory to have contributed any emissions.

## SOURCE CODE

All of the original Python source code for this project can be located at <https://github.com/JashanChopra/SummitWildfireTracers>.

## REFERENCES

- [1] Andreae, M. O., and Merlet, P., Emission of trace gases and aerosols from biomass burning, *Global Biogeochem. Cycles*, 15, 955–966, 2001.
- [2] Cartopy. v0.1.2. 22-Jul-2019. Met Office. UK. <https://github.com/SciTools/cartopy/archive/v0.11.2.tar.gz>
- [3] Detlev Helmig et al. (2016) Reversal of global atmospheric ethane and propane trends largely due to US oil and natural gas production. *Nature Geoscience*. DOI: 10.1038/NGEO2721
- [4] Mefford, T.K., M. Bieniulis, B. Halter, and J. Peterson, Meteorological Measurements, in *CMDL Summary Report*



**Fig. 10:** Nearside perspective projection globe centered at Summit, Greenland displaying z-score color coded back trajectories overlayed on NASA VIIRS fire products. Whole degree latitude/longitude comparison within 24 hours.

	Total Number		Number of Matches		Percent Matches / Total	
	Ace/Methane	Ethane/Methane	Ace/Methane	Ethane/Methane	Ace/Methane	Ethane/Methane
Fire Counts	255110	255110	59	45	0.02%	0.02%
Trajectories	1887	1632	59	45	3.12%	2.76%

**TABLE I:** General statistics on the matches between fire counts and trajectories between acetylene/methane and ethane/methane ratios.

	Winter (1°)		Summer (1°)		Winter (0.1°)		Summer (0.1°)	
	Avg Z (all)	Avg Z (match)	Avg Z (all)	Avg Z (match)	Avg Z (all)	Avg Z (match)	Avg Z (all)	Avg Z (match)
Ace/Methane	n/a	n/a	0.68	0.79	n/a	n/a	0.68	0.63
Ethane/Methane	n/a	n/a	0.76	0.66	n/a	n/a	n/a	n/a

**TABLE II:** Average z-score between acetylene/methane and ethane/methane ratios across seasonal variation and match size variation, using a six hour time difference.

1994 - 1995, No. 23, 1996, pg. 17.

[5] National Centers for Environmental Prediction/National Weather Service/NOAA/U.S. Department of Commerce. 2015, updated daily. NCEP GDAS/FNL 0.25 Degree Global Tropospheric Analyses and Forecast Grids. Research Data Archive at the National Center for Atmospheric Research, Computational and Information Systems Laboratory. <https://doi.org/10.5065/D65Q4T4Z>. Accessed 07-23-2019.

[6] Schroeder, W., Oliva, P., Giglio, L., & Csiszar, I. A. (2014). The New VIIRS 375 m active fire

detection data product: algorithm description and initial assessment. *Remote Sensing of Environment*, 143, 85-96. doi:10.1016/j.rse.2013.12.008

[7] Stein, A. et al. NOAAs HYSPLIT atmospheric transport and dispersion modeling system. *Bulletin of the American Meteorological Society* 96, 20592077 (2015).

[8] Thoning, Kirk & P. Tans, Pieter & D. Komhyr, Walter. (1989). Atmospheric carbon dioxide at Mauna Loa Observatory: 2. Analysis of the NOAA GMCC data, 19741985. *Journal of Geophysical Research*. 94. 8549-8565. 10.1029/JD094iD06p08549.

[9] Warner, M. S., 2018: Introduction to PySPLIT: A Python Toolkit for NOAA ARLs HYSPLIT Model. *Comput. Sci. Eng.*, 20, 4762, <https://doi.org/10.1109/MCSE.2017.3301549>.

#### ACKNOWLEDGEMENTS

Thanks to Brendan Blanchard for providing the tools and support I needed to rapidly develop my Python skills. Thanks to Helene Angot for providing crucial insight behind the science, statistics, and models required for this research. Thanks to Detlev Helmig for providing me with this amazing opportunity and guiding me through common hurdles in research. This work was funded by the University of Colorado Boulder's Undergraduate Research Opportunities Program (UROP) and by INSTAAR's Atmospheric Research Lab (ARL).

#### APPENDIX A: DATA VISUALIZATION REPRODUCTIONS

In the paper I skipped over some of the heavier coding descriptions to neatly describe the results. In this appendix section I will provide a step by step instruction on how to reproduce the final plot shown in fig. 10. Every other visualization in this paper was an interim step towards the final graph, but they are not necessary to reproduce this graph. Nevertheless, the original code for each and every one of these graphs can be found in the GitHub repository at <https://github.com/JashanChopra/SummitWildfireTracers>. Documentation on creating the windrose plots can also be found at this repository. Procedures for getting or creating data will be highlighted in Appendix B.

**Central Data Script:** Before jumping into the explanation on visualization, a flowchart of the central data cleaning process is provided in fig. 12. Creating the graph is defined by this central script, which has a number of conditions explored more in the docstring of the code. Following initial data reading and some trimming, the code works on creating datetimes. First it calls the `decToDatetime()` function to create datetime indexes from the original decimal year format. This function simply takes the integer year from the decimal format and assigns this as the starting date. It then

calculates a `timedelta` of days from the remainder decimal, incorporating a leap year check, and adds that to get a final datetime. Next, the main script gives each datetime a Julian day equivalent, which it uses to trim either summer or winter days depending on the input condition. Pollution from certain zones is also removed during this step by the `metRemoval()` function, which can be seen in flowchart form in fig. 13. The `sciPy` statistics module then calls the `zscore()` function to identify z-scores of the residual value column of the original dataset. The absolute value of these scores are taken and data is removed for z-scores below the inputted threshold. Finally the `trajPlot()` function is called to display the final plot, which is explained in flowchart form in fig. 14. Fire data is imported inside the plotting function, and is described in flowchart form in fig. 11.

**metRemoval() Function:** This function is a simple and classic data trimming script. First we import the data with a separate function specialized for this type of meteorological data import. We reassign column names, replace -999 and similar values with `NaN` and drop them. Finally, we create datetimes in the exact way as before, and drop useless columns. Back into the main combination script, we do nearly the same thing, except we first merge the atmospheric data with the returned met data. This uses Pandas `merge_asof()` built in function, which grafts the meteorological data onto each atmospheric data point by the nearest datetime in a set tolerance. The default tolerance is a single hour.

**trajPlot() Function:** Now that all the miscellaneous data work has been done, we can develop the final plot displayed in fig. 10. First we set up the figure in the typical `Matplotlib` fashion, but using `Cartopy`'s `NearsidePerspective` projection. After adjusting the subplot spacing, we add coastlines, a stock image of the Earth, and gridlines. The first object plotted is the Summit location, as an orange star. Following this we want to create a color chart for color coding the trajectories by z-score. To do this we use `numpy.arange` to create an equally spaced distribution from a `cmap`, and then make an equivalent length array of arbitrary z-score values. We then iterate through each file in the `os.listdir` of the directory where trajectories are located. After reading the file, dropping `NaN` values, and resetting the index, we create datetime values just as before. We then merge these results with z-score values exactly as with the met data. Using a zip of the trajectory's longitude and latitude values we can create a Shapely `sgeom.LineString` feature. Now we identify the z-score of the trajectory, identify the nearest match in our preallocated array from earlier, and back index the color chart. Finally, we add the geometry with a `PlateCarree` projection to preserve true distance and color it with our according index. Before moving on, we simply append each piece of trajectory info into a list for later use.

The next step involves importing fire data, which can be

observed in fig. 11. The only thing of added interest in this fire function is the confidence level of each fire product. As stated in the research portion, only high confidence values were used. We iterate through each fire product, first identifying similar latitudes and longitudes by cross checking the fire coordinates with the full list of trajectory coordinates. We Boolean *and* both of these arrays together to find all crossing coordinate pairs. If a cross is found, we see if our timedelta of 24 hours is greater than the difference between the fire and the start of the back trajectory. If it is, we have found a match and it is appended to a separate list. Otherwise, it is appended to a list of null matches. Both of these are then plotted as scatterplots with a geodetic projection and in different colors and sizes.

Now that the fires and trajectories have been plotted, only a few design factors remain. First we create a custom located colorbar, and normalize the values with our min and max z-values from before. The colorbar base will be the same colormap as our ranged array, with a vertical orientation. We then create legend elements in the form of *mpatches.Rectangles*, followed by legend labels. We then create a legend in the lower left corner with fancybox enabled, and print a few statistics reports for general information. All culminating with a simple *plt.show* command to display the plot.

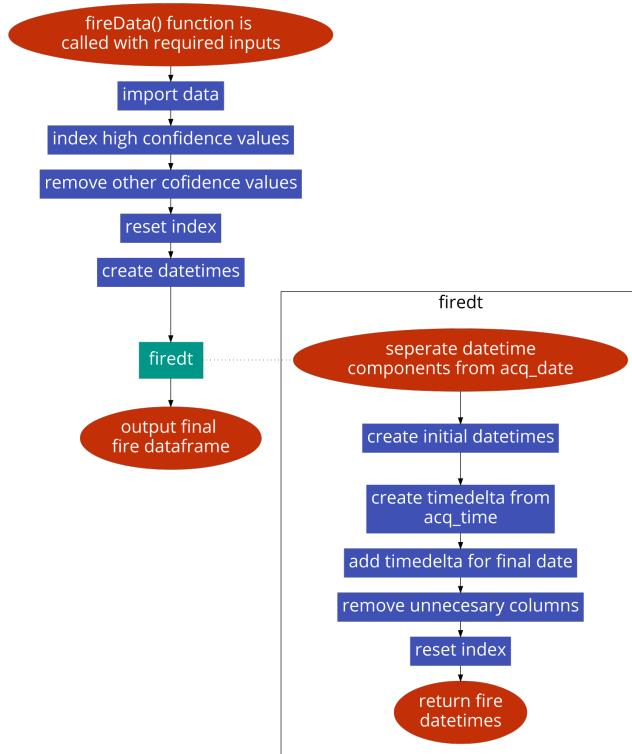


Fig. 11: Coding flowchart for importing and trimming fire data

## APPENDIX B: ACQUIRING DATA

The original sources for the data used in this project are given in this appendix, along with the required methodology to acquire them. The in house acetylene, methane, and ethane mixing ratios can be viewed in near real time (NRT) form at our [INSTAAR website](#). View the contact page if you are interested in archival data. The acetylene/methane and ethane/methane ratios are created from a combination of these data. For a coding flowchart of this process, see fig. 15.

The harmonic fit was done using the NOAA ESRL GMD CCGCRV tool developed by Kirk Thoning. The methodology, as well as Python files of the code base can be found at [this NOAA webpage](#). A tool exists via the NOAA FTP server to access the program online, but a lengthy authentication process is first required. Follow the contact page on this page for more information on using NOAA's harmonic fitting tools.

All the in-situ meteorological data to create the windrose plots and remove pollution from the dataset was acquired from NOAA's ESRL GMD. Their FTP directory with this data, specifically for Summit, can be found [at this link](#).

NOAA's HYSPLIT tool can be found [here](#). Its Python companion Pysplit can be located [at this GitHub Repo](#). Brendan Blanchard's modifications can be located [at this GitHub Repo](#). Each of these sources has its own robust documentation for how to use the tools. For my specific analysis, I used the NOAA gdas0p5 data located [on this FTP server](#).

NASA fire products can be located at the [FIRMS website](#). For archival downloads a request must be made on their website. This concludes the sources of data used in this paper. For further questions or information on data sources and methodology, contact the author at [jashan.chopra@colorado.edu](mailto:jashan.chopra@colorado.edu).

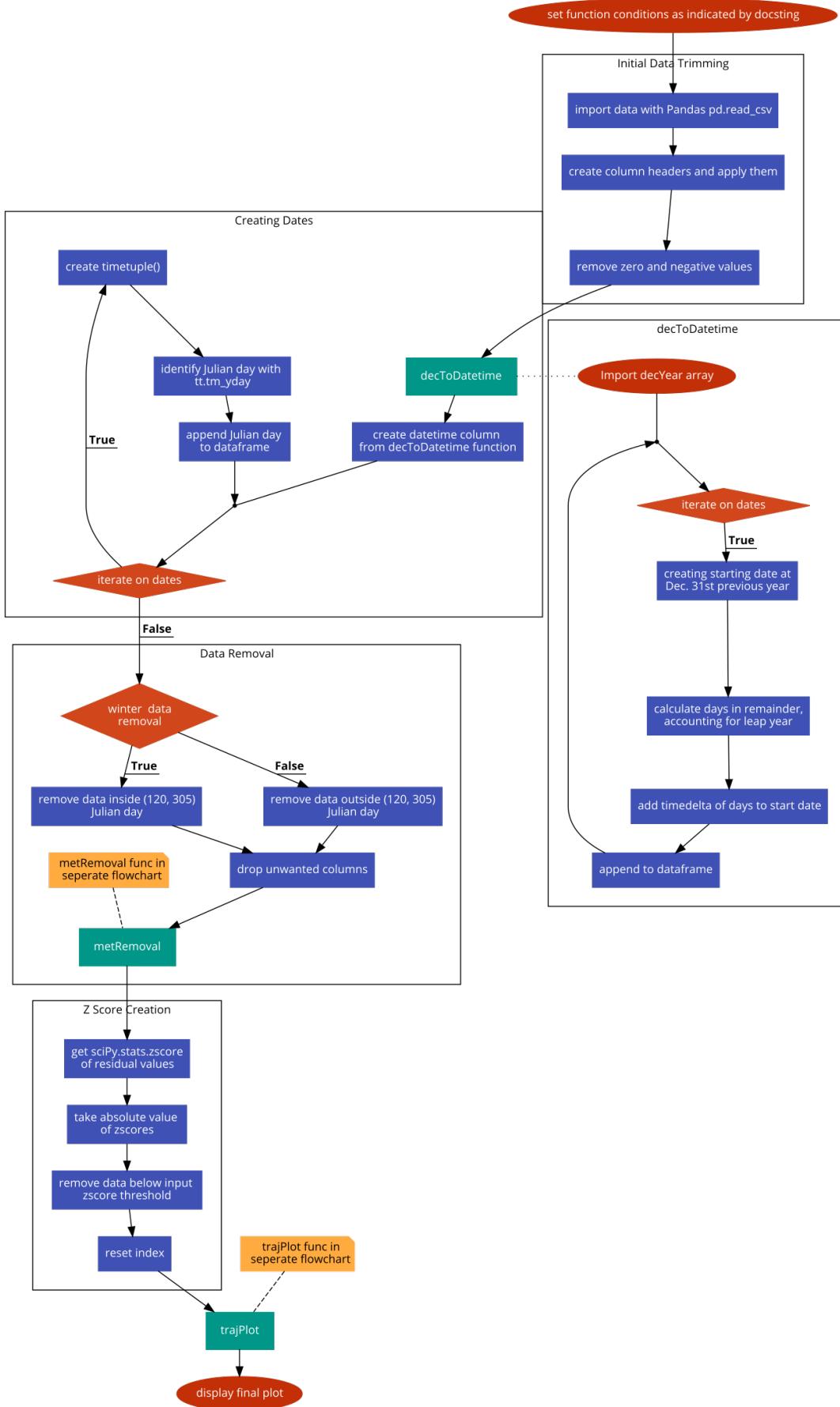
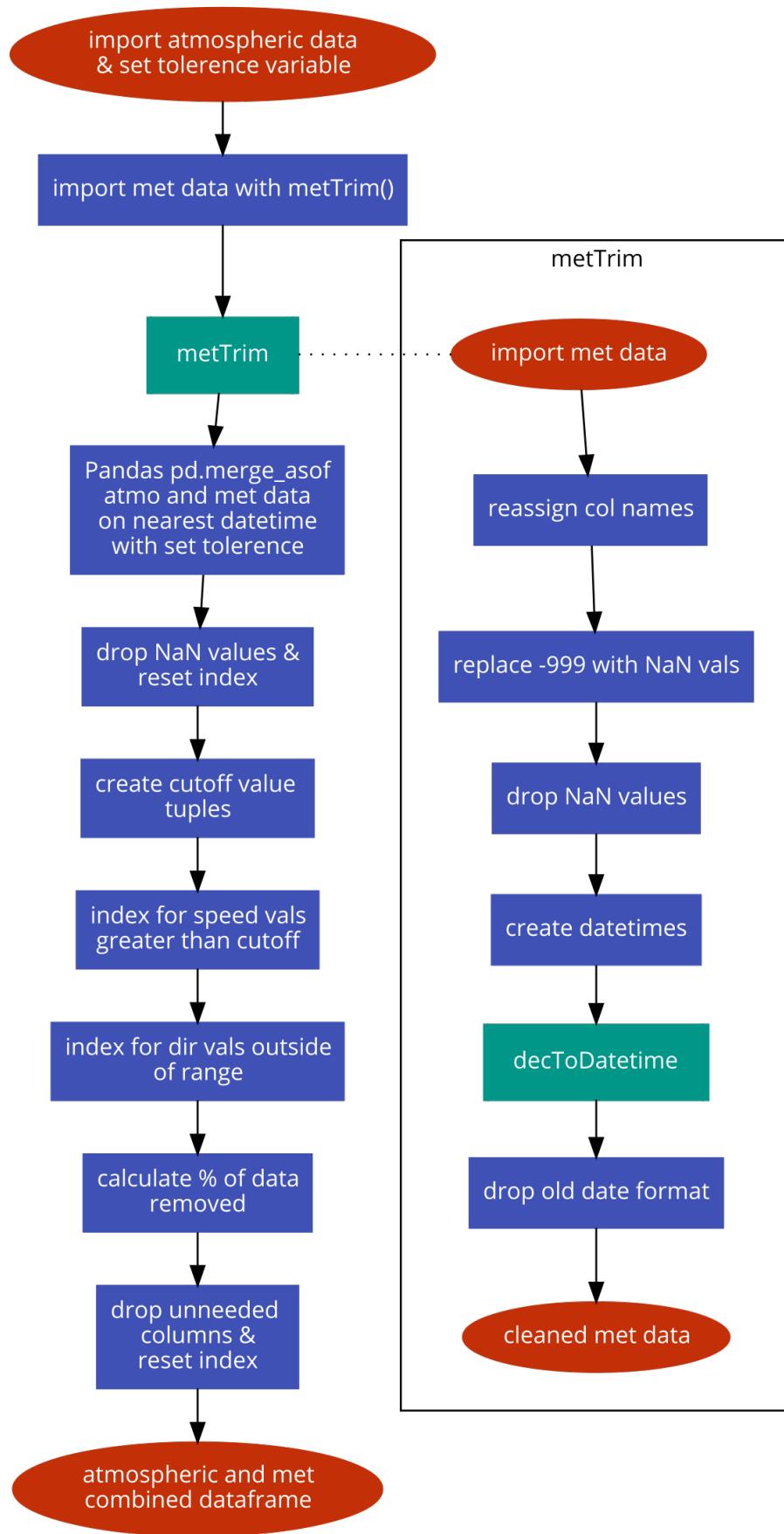
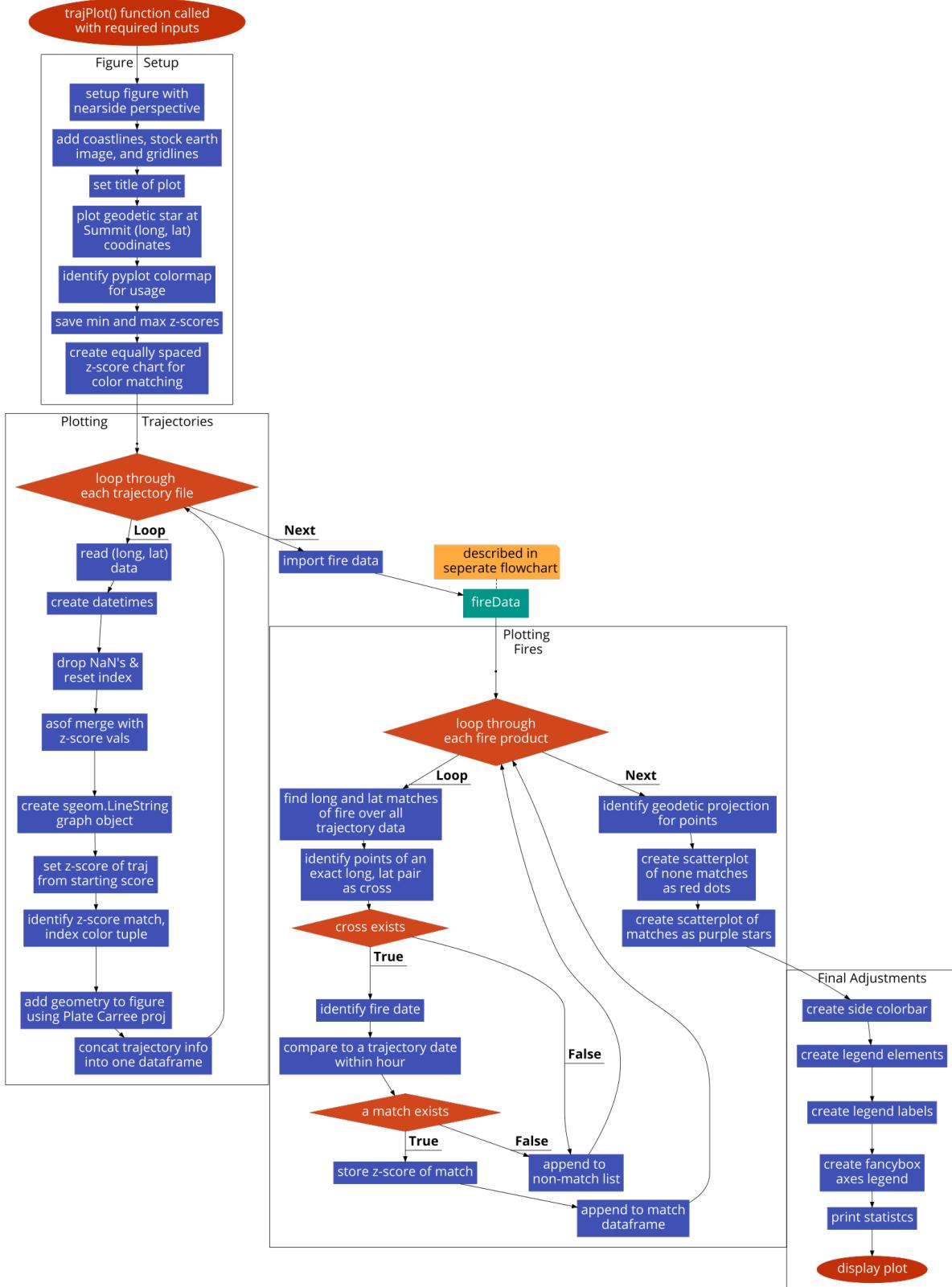


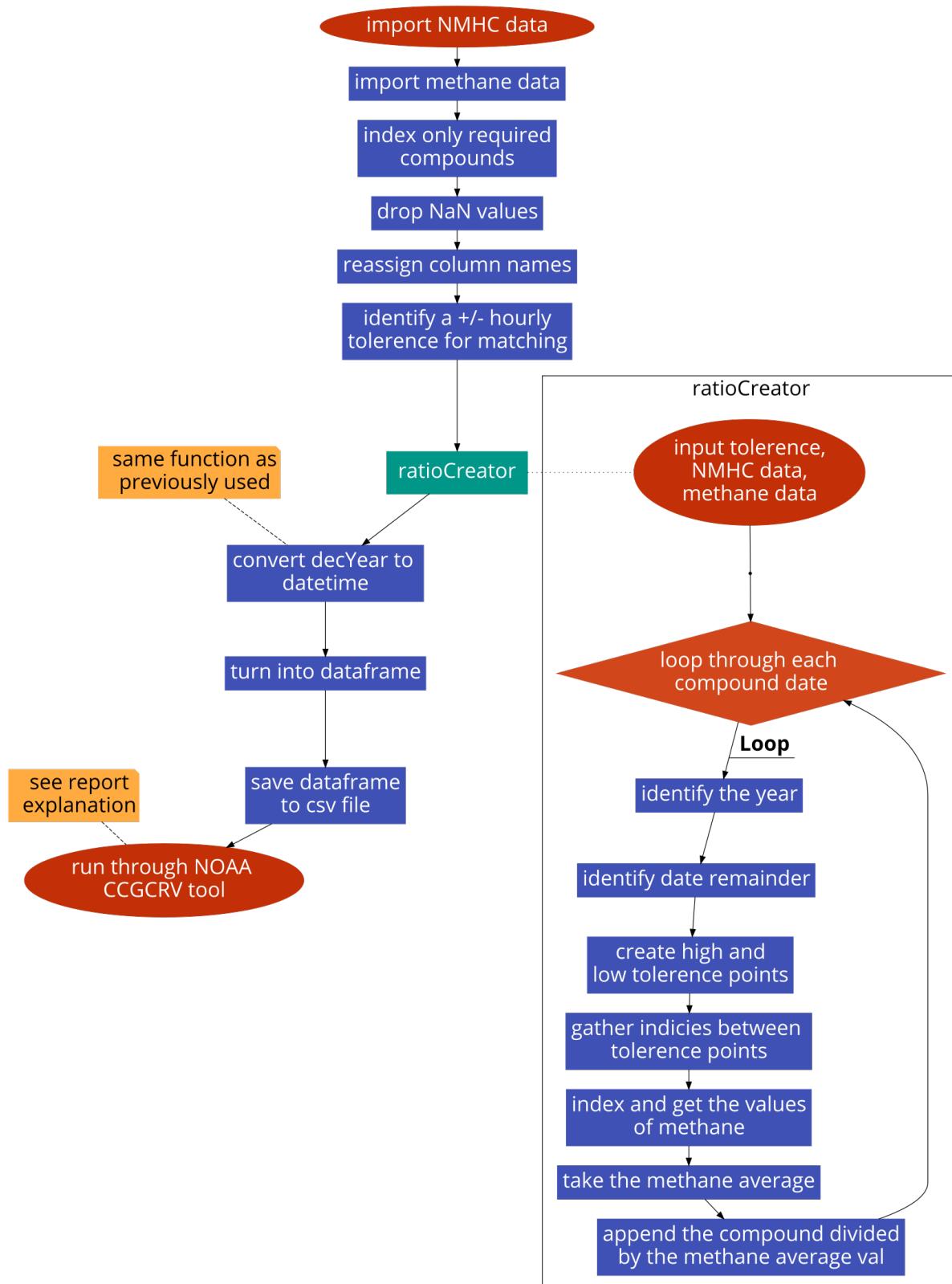
Fig. 12: Coding flowchart for preparation of data pre-plotting



**Fig. 13:** Coding flowchart for removal of pollution zones from meteorological data



**Fig. 14:** Coding flowchart for creating globe visualization of trajectories overlaid on fire data



**Fig. 15:** Coding flowchart for creation of acetylene/methane and ethane/methane ratios