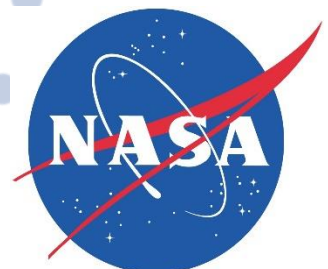


# Tutorial #2

## Plotting the Ratio of TROPOMI Formaldehyde and Nitrogen Dioxide Satellite Data: An Indicator of Ozone Sensitivity

Jennifer McGinnis  
Tracey Holloway  
Jenny Bratburd

*Nelson Institute Center for sustainability and the Global Environment  
University of Wisconsin-Madison  
Madison, WI 53726*



## • • • • • **Acknowledgements**

Work for this report was funded by the NASA Health and Air Quality Applied Sciences Team (HAQAST).

We thank all the groups that make their data publicly available for air quality and health analysis, especially those we used in this report: TROPOMI and Earth Engine Data Catalog. In addition, we would like to thank the broader NASA HAQAST team for helpful comments at various stages of the project.

# CONTENTS

<b>Acknowledgements</b> .....	2
<b>Executive Summary</b> .....	4
<b>Tutorial</b> .....	4
<b>Conclusion</b> .....	13
<b>Code</b> .....	14
<b>References</b> .....	18

## Executive Summary

In this tutorial, we will employ Google Earth Engine to compute and compare the Formaldehyde to Nitrogen dioxide Ratio (FNR) on higher-ozone days and lower-ozone days. The classification of these days as higher or lower ozone levels is determined utilizing data provided by the Environmental Protection Agency (EPA). We strongly recommend completing Tutorial 1 available on <https://hollowaygroup.org/tutorials> before attempting this tutorial. Tutorial 1 goes over how to create an account, identify the datasets, calculate averages, and divide. This tutorial focuses on New York City to replicate imagery shown in the Air and Waste Management EM Magazine December 2023 issue. The code used in this tutorial is provided at the bottom of this tutorial where you can copy and paste it directly in Google Earth Engine, click "Run" to immediately see the output.

Key Aspects:

- Identifying higher-ozone and lower-ozone days based on EPA data
- Plotting the average FNR for higher-ozone and lower-ozone days
- Downloading the images created

## Tutorial

*Last updated and tested on 8/1/2023*

### 1) Determine the ozone season and location of interest.

- a. We will look at the 2022 ozone season over the New York City Metropolitan Statistical Area (MSA).
- b. As defined by the EPA, the ozone season is from May 1<sup>st</sup> - September 30<sup>th</sup>.

### 2) Categorize higher-ozone and lower-ozone days.

- a. Go to the EPA website and locate the Air Data – Tile Plots
  - i. <https://www.epa.gov/outdoor-air-quality-data/air-data-tile-plot>
- b. Input ozone as the pollutant, the year of interest, geographic area of interest, and include all monitor sites. Click "Plot Data."

## Air Data - Tile Plot

Plot daily AQI values for a specific location and time period. Each square or "tile" represents one day of the year and is color-coded based on the highest daily AQI value at the selected monitor - or among all monitors in the geographic area if "All Sites (Highest Daily AQI)" is selected. The legend tallies the number of days in each AQI category.

1. Pollutant

2. Year

3. Geographic Area

-- or --

Select a County ...

All Sites (Highest Daily AQI)

340030006

340130003

340170006

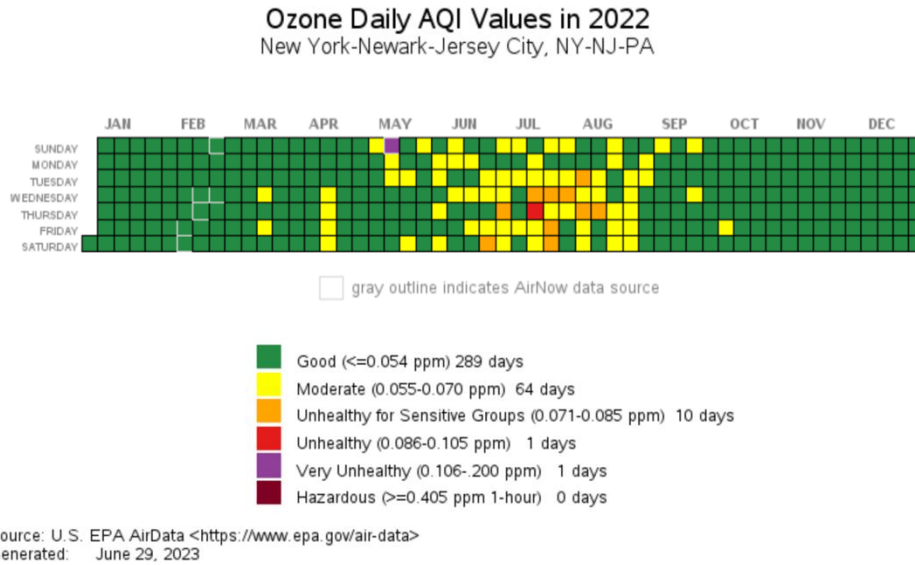
340190001

340230011

4. Monitor Site

- c. This output (example shown below) will be used to categorize the days within the ozone season (May 1<sup>st</sup> - September 30<sup>th</sup>) into higher-ozone days and lower-ozone days. For this analysis, green/Good and yellow/Moderate are considered lower-ozone days; orange/Unhealthy for Sensitive Groups or higher (0.071 ppm or higher) are considered higher-ozone days.

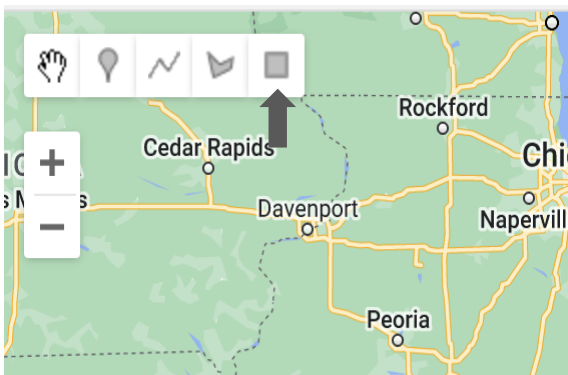
- d. We find that at least 10 days need to be in each category (34+ preferred) to reduce the “noise” in the satellite data (Vigouroux et al., 2020).



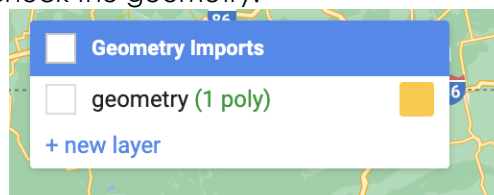
The following data link is active for the next 10 minutes, after which you must resubmit your query.  
[Download CSV \(spreadsheet\)](#)

### 3) Define the geographic region in Google Earth Engine.

- a. Create a new file in Google Earth Engine.
- b. Draw a polygon over the area of interest.
  - i. To do this, click the square shown below.
  - ii. Click and drag to create a box over the area of interest.



- iii. To hide the polygon from displaying, hover over the “Geometry Imports” and uncheck the geometry.



#### 4) Create two variables to store the formaldehyde and nitrogen dioxide data.

- a. We will use the offline datasets, the tropospheric columns for formaldehyde (HCHO) and nitrogen dioxide (NO<sub>2</sub>), and the ozone season of 2022.
- b. Any names can be used for the variables, as long as they are kept consistent.

#### 5) Create four new variables to store the HCHO and NO<sub>2</sub> data on higher-ozone and lower-

```
FNR Tutorial 2 Get Link Save Run Rese
Imports (1 entry)
  var geometry: Polygon, 4 vertices
1
2 var formaldehyde = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_HCHO')
3   .select('tropospheric_HCHO_column_number_density')
4   .filterDate('2022-05-01', '2022-10-01');
5
6 var no2 = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_NO2')
7   .select('tropospheric_NO2_column_number_density')
8   .filterDate('2022-05-01', '2022-10-01');
```

#### ozone days.

- a. In the screenshots and text below:
  - i. HCHO on higher-ozone days is "formaldehyde\_high"
  - ii. HCHO on lower-ozone days is "formaldehyde\_low"
  - iii. NO<sub>2</sub> on higher-ozone days is "no2\_high"
  - iv. NO<sub>2</sub> on lower-ozone days is "no2\_low"
- b. After the formaldehyde\_high variable is created, set it equal to the original formaldehyde variable.
  - i. Use the .filter() and ee.Filter.or() functions to prepare the variable for the list of higher-ozone days.

```
FNR Tutorial 2 * Get Link Save Run Reset
Imports (1 entry)
  var geometry: Polygon, 4 vertices
1
2 var formaldehyde = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_HCHO')
3   .select('tropospheric_HCHO_column_number_density')
4   .filterDate('2022-05-01', '2022-10-01');
5
6 var no2 = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_NO2')
7   .select('tropospheric_NO2_column_number_density')
8   .filterDate('2022-05-01', '2022-10-01');
9
10 var formaldehyde_high = formaldehyde.filter(ee.Filter.or(
11 ));
```

- ii. Use ee.Filter.Date(\*Start Date Inclusive\*, \*End Date Exclusive\*), to specify the days with higher-ozone in the ozone season. See below for the correct format.
- iii. Note: If the days are consecutive, you can group them together. For example, July 13<sup>th</sup> and July 14<sup>th</sup> are higher-ozone days. So as I am filtering the data to only include higher-ozone days I use ee.Filter.Date('2022-07-13', '2022-07-15'). This way July 13<sup>th</sup> is included (because the start date is

inclusive), July 14<sup>th</sup> is included, and July 15<sup>th</sup> is not included (because the end date is exclusive).

```
FNR Tutorial 2 Get Link Save Run
Imports (1 entry)
  var geometry: Polygon, 4 vertices
1
2 var formaldehyde = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_HCHO')
3   .select('tropospheric_HCHO_column_number_density')
4   .filterDate('2022-05-01', '2022-10-01');
5
6 var no2 = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_NO2')
7   .select('tropospheric_NO2_column_number_density')
8   .filterDate('2022-05-01', '2022-10-01');
9
10 var formaldehyde_high = formaldehyde.filter(ee.Filter.or(
11   ee.Filter.date('2022-05-08', '2022-05-09'),
12   ee.Filter.date('2022-06-25', '2022-06-26'),
13   ee.Filter.date('2022-06-30', '2022-07-01'),
14   ee.Filter.date('2022-07-13', '2022-07-15'),
15   ee.Filter.date('2022-07-20', '2022-07-21'),
16   ee.Filter.date('2022-07-22', '2022-07-24'),
17   ee.Filter.date('2022-07-27', '2022-07-28'),
18   ee.Filter.date('2022-08-02', '2022-08-03'),
19   ee.Filter.date('2022-08-04', '2022-08-05'),
20   ee.Filter.date('2022-08-11', '2022-08-12')
21 ));
22
```

- c. Repeat step 5b for the formaldehyde\_low variable using the days categorized as lower-ozone days within the ozone season (example on next page).
  - i. For this variable, the trick of grouping days together is very useful.
    1. Notice the dates in the formaldehyde\_high and formaldehyde\_low variables fit together like a puzzle.
    2. Apart from May 1<sup>st</sup> and October 1<sup>st</sup>, every day that is listed as a start date in the formaldehyde\_high variable will be found as an end date somewhere in the formaldehyde\_low variable, and vice versa. Checking that this is true will help ensure every day is included in the analysis.
- d. Repeat steps 4b-c for NO<sub>2</sub> using the original no2 variable (example on next page). The dates in no2\_high should be the same as the dates in formaldehyde\_high and the dates in no2\_low should be the same as the dates in formaldehyde\_low.

```

FNR Tutorial 2 Get Link Save Run
Imports (1 entry)
  var geometry: Polygon, 4 vertices
1
2 var formaldehyde = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_HCHO')
3   .select('tropospheric_HCHO_column_number_density')
4   .filterDate('2022-05-01', '2022-10-01');
5
6 var no2 = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_NO2')
7   .select('tropospheric_NO2_column_number_density')
8   .filterDate('2022-05-01', '2022-10-01');
9
10 var formaldehyde_high = formaldehyde.filter(ee.Filter.or(
11   ee.Filter.date('2022-05-08', '2022-05-09'),
12   ee.Filter.date('2022-06-25', '2022-06-26'),
13   ee.Filter.date('2022-06-30', '2022-07-01'),
14   ee.Filter.date('2022-07-13', '2022-07-15'),
15   ee.Filter.date('2022-07-20', '2022-07-21'),
16   ee.Filter.date('2022-07-22', '2022-07-24'),
17   ee.Filter.date('2022-07-27', '2022-07-28'),
18   ee.Filter.date('2022-08-02', '2022-08-03'),
19   ee.Filter.date('2022-08-04', '2022-08-05'),
20   ee.Filter.date('2022-08-11', '2022-08-12')
21 ));
22
23 var formaldehyde_low = formaldehyde.filter(ee.Filter.or(
24   ee.Filter.date('2022-05-01', '2022-05-08'),
25   ee.Filter.date('2022-05-09', '2022-06-25'),
26   ee.Filter.date('2022-06-26', '2022-06-30'),
27   ee.Filter.date('2022-07-01', '2022-07-13'),
28   ee.Filter.date('2022-07-15', '2022-07-20'),
29   ee.Filter.date('2022-07-21', '2022-07-22'),
30   ee.Filter.date('2022-07-24', '2022-07-27'),
31   ee.Filter.date('2022-07-28', '2022-08-02'),
32   ee.Filter.date('2022-08-03', '2022-08-04'),
33   ee.Filter.date('2022-08-05', '2022-08-11'),
34   ee.Filter.date('2022-08-12', '2022-10-01')
35 ));

```

```

FNR Tutorial 2 Get Link Save Run
36
37 var no2_high = no2.filter(ee.Filter.or(
38   ee.Filter.date('2022-05-08', '2022-05-09'),
39   ee.Filter.date('2022-06-25', '2022-06-26'),
40   ee.Filter.date('2022-06-30', '2022-07-01'),
41   ee.Filter.date('2022-07-13', '2022-07-15'),
42   ee.Filter.date('2022-07-20', '2022-07-21'),
43   ee.Filter.date('2022-07-22', '2022-07-24'),
44   ee.Filter.date('2022-07-27', '2022-07-28'),
45   ee.Filter.date('2022-08-02', '2022-08-03'),
46   ee.Filter.date('2022-08-04', '2022-08-05'),
47   ee.Filter.date('2022-08-11', '2022-08-12')
48 ));
49
50 var no2_low = no2.filter(ee.Filter.or(
51   ee.Filter.date('2022-05-01', '2022-05-08'),
52   ee.Filter.date('2022-05-09', '2022-06-25'),
53   ee.Filter.date('2022-06-26', '2022-06-30'),
54   ee.Filter.date('2022-07-01', '2022-07-13'),
55   ee.Filter.date('2022-07-15', '2022-07-20'),
56   ee.Filter.date('2022-07-21', '2022-07-22'),
57   ee.Filter.date('2022-07-24', '2022-07-27'),
58   ee.Filter.date('2022-07-28', '2022-08-02'),
59   ee.Filter.date('2022-08-03', '2022-08-04'),
60   ee.Filter.date('2022-08-05', '2022-08-11'),
61   ee.Filter.date('2022-08-12', '2022-10-01')
62 ));
63

```



## 6) Calculate the mean values for the geometry of interest.

- a. This will be done for:
  - i. HCHO and NO<sub>2</sub> for the entire ozone season
  - ii. HCHO and NO<sub>2</sub> for lower-ozone days
  - iii. HCHO and NO<sub>2</sub> for higher-ozone days

```
FNR Tutorial 2 Get Link Save Run Reset  
63  
64 // Calculate the mean values for the geometry of interest  
65 var formaldehyde_avg = formaldehyde.mean().clip(geometry);  
66 var no2_avg = no2.mean().clip(geometry);  
67  
68 var formaldehyde_low_avg = formaldehyde_low.mean().clip(geometry);  
69 var no2_low_avg = no2_low.mean().clip(geometry);  
70  
71 var formaldehyde_high_avg = formaldehyde_high.mean().clip(geometry);  
72 var no2_high_avg = no2_high.mean().clip(geometry);  
73  
74
```

## 7) Calculate the number of pixels used to create the mean values.

- a. Create a layer storing the number of pixels averaged to create the means in Step 6.
  - i. This layer of data is going to be used to create a “mask” to filter out mean values that were created with less than 10 pixels.
  - ii. Although NO<sub>2</sub> data can be displayed as a single day, to maintain consistency and ensure outliers don’t greatly impact the FNR, NO<sub>2</sub> data will be masked as well as HCHO data.

```
FNR Tutorial 2 Get Link Save Run Reset  
75 // Calculate the number of pixels used to create the averaged pixel  
76 var formaldehyde_avg_count = formaldehyde.count().clip(geometry);  
77 var no2_avg_count = no2.count().clip(geometry);  
78  
79 var formaldehyde_low_count = formaldehyde_low.count().clip(geometry);  
80 var no2_low_count = no2_low.count().clip(geometry);  
81  
82 var formaldehyde_high_count = formaldehyde_high.count().clip(geometry);  
83 var no2_high_count = no2_high.count().clip(geometry);  
84
```

## 8) Create the masks.

- a. Using the respective HCHO and NO<sub>2</sub> count variables above, create a mask that ensures at least 10 HCHO pixels and 10 NO<sub>2</sub> pixels were used to create the average values for the total ozone season, the lower-ozone days, and the higher-ozone days.

```
FNR Tutorial 2 Get Link Save Run Reset  
86 // Create mask for each raito to only keep pixels that were created by  
87 // averaging at least 10 pixels  
88 var mask = formaldehyde_avg_count.gte(10).and(no2_avg_count.gte(10));  
89 var mask_low = formaldehyde_low_count.gte(10).and(no2_low_count.gte(10));  
90 var mask_high = formaldehyde_high_count.gte(10).and(no2_high_count.gte(10));
```

## 9) Calculate the ratio and apply the corresponding masks.

- a. Calculate the ratio and rename the variable to describe more accurately what is now stored in the variable.
- b. Using the .updateMask() function, apply the corresponding mask created in Step 8 to the FNR layers.

```

FNR Tutorial 2 Get Link Save
93 // Apply the Masks
94 var fnr_avg = formaldehyde_avg.divide(no2_avg)
95   .rename('Total Ratio')
96   .updateMask(mask);
97
98 var fnr_low = formaldehyde_low_avg.divide(no2_low_avg)
99   .rename('Low Ratio')
100  .updateMask(mask_low);
101
102 var fnr_high = formaldehyde_high_avg.divide(no2_high_avg)
103   .rename('High Ratio')
104   .updateMask(mask_high);

```

10) Add the FNR values to the map.

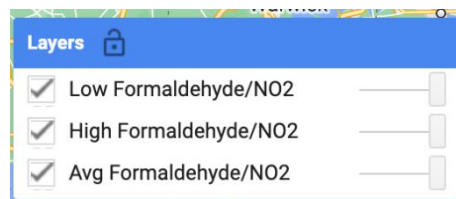
- Define the color palette. The colors, minimum value, and maximum value can be changed to better display your data.
- Add the three FNR layers.
- Click “Run”. The data may take a couple of minutes to appear.

```

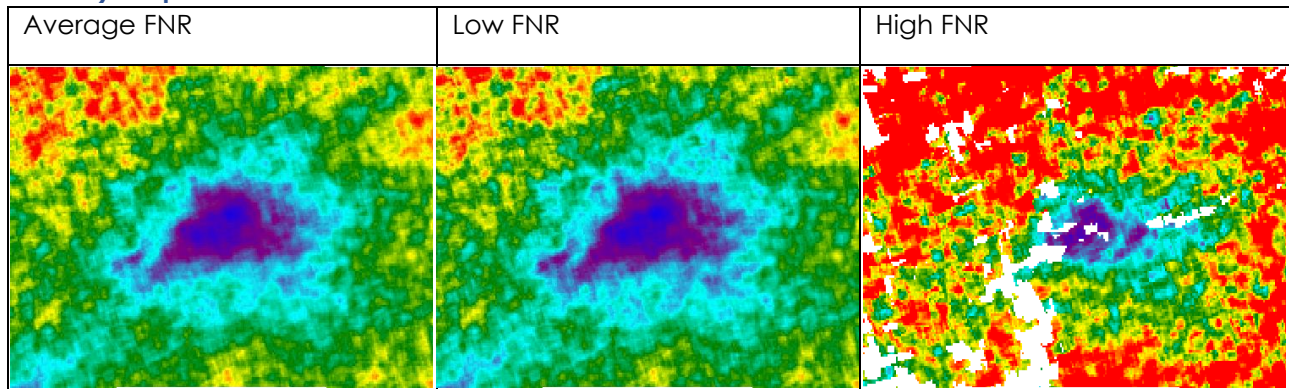
FNR Tutorial 2 Get Link Save Run
107
108 // FNR Color Palette
109 var palette = ['black', 'blue', 'purple', 'cyan', 'green', 'yellow', 'red'];
110
111 var band_viz = {
112   min: 0,
113   max: 6,
114   palette: palette
115 };
116
117 Map.addLayer(fnr_avg, band_viz, 'Avg Formaldehyde/NO2');
118 Map.addLayer(fnr_high, band_viz, 'High Formaldehyde/NO2');
119 Map.addLayer(fnr_low, band_viz, 'Low Formaldehyde/NO2');
120

```

- Hover over the “Layers” icon towards the right of the screen to toggle on and off which layers you would like displayed.



## 11) Output



## 12) Add a legend.

- Copy and paste the following code and click "Run" for the legend to appear:

```
//Creating a legend
function makeColorBarParams(palette) {
  return {
    bbox: [0, 0, 1, 0.01],
    dimensions: '10x100',
    format: 'png',
    min: 0,
    max: 1,
    palette: palette,
  };
}

var colorBar = ui.Thumbnail({
  image: ee.Image.pixelLonLat().select(0),
  params: makeColorBarParams(band_viz.palette),
  style: { stretch: 'horizontal', margin: '0px 0px', minWidth: '200px', maxHeight: '24px'},
});

var legendLabels = ui.Panel({
  widgets: [
    ui.Label(band_viz.min, { margin: '4px 0px'}),
    ui.Label(
      ((band_viz.max - band_viz.min) / (2 + band_viz.min)),
      { margin: '4px 0px', textAlign: 'center', stretch: 'horizontal'}
    ),
    ui.Label(band_viz.max, { margin: '4px 0px' }),
  ],
  layout: ui.Panel.Layout.flow('horizontal')
});

var legendTitle = ui.Label({
  value: 'Map Legend: FNR',
  style: {fontWeight: 'bold',
  padding: '5px 0px'
  }
});
```

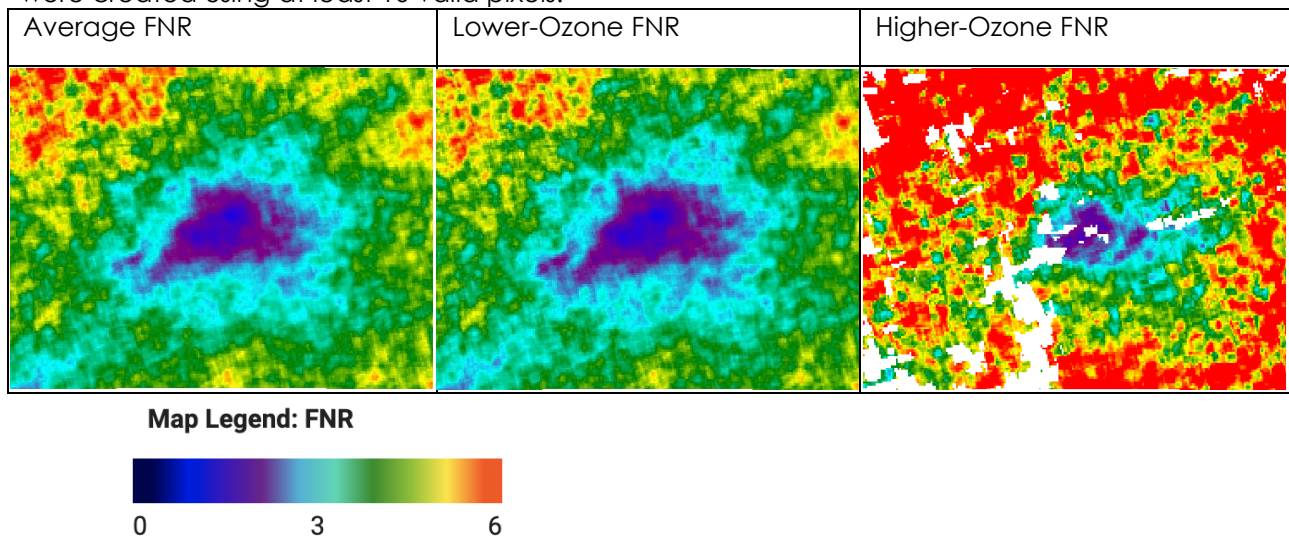
```
var legendPanel = ui.Panel([legendTitle, colorBar, legendLabels]);
Map.add(legendPanel);
```

### 13) Analyzing the images.

Jin et al., 2022 calculated the overall transitional range using an empirical observation-based analysis from seven cities, one of them being New York City. Since they analyzed New York City, we will use the city specific threshold calculated, [2.9-3.8] instead of the overall threshold mentioned in Tutorial 1, [3.2-4.1].

We can see in the figure below that on during the ozone season, the city center of New York City is predominately less than 2.9 and is therefore most likely VOC-limited. Further away from the city center begins to shift into the NO<sub>x</sub>-limited regime. This is consistent on lower-ozone days. However, when the higher-ozone days are plotted together, there is a shift towards a more NO<sub>x</sub>-limited regime and more of the area is NO<sub>x</sub>-limited than on lower-ozone days. This suggests that reductions in NO<sub>x</sub> emissions may decrease ozone production, especially on the most polluted days.

One thing to note is that on the higher-ozone days there are more pixels missing. This is because we applied the mask created in Steps 8 and 9 that only allowed pixels to be plotted if the averages were created using at least 10 valid pixels.



### 14) Download the images.

- The background map will not be downloaded with your satellite data.
- Copy and paste the following code and click "Run":
  - Note: if you changed the max value in the visualization parameters above, change the max value below as well.

```
var thumbnailFNRavg = fnr_avg.getThumbURL({
  'min': 0,
  'max': 6,
  'palette': palette,
  'dimensions': 500,
  'crs': 'EPSG:3857',
  'region': geometry,
});
```

```

print('FNR Avg', thumbnailFNRavg);
var thumbnailFNRhigh = fnr_high.getThumbURL({
  'min': 0,
  'max': 6,
  'palette': palette,
  'dimensions': 500,
  'crs': 'EPSG:3857',
  'region': geometry,
});
print('FNR High', thumbnailFNRhigh);
var thumbnailFNRlow = fnr_low.getThumbURL({
  'min': 0,
  'max': 6,
  'palette': palette,
  'dimensions': 500,
  'crs': 'EPSG:3857',
  'region': geometry,
});
print('FNR Low', thumbnailFNRlow);

```

- c. Click “Console” in the top right corner. You should see three links labeled for each FNR layer.



- d. Click on the link and it will open a new tab. It may take a couple of minutes to load but once you see the image in the web browser you can right click and save the image. We recommend screenshotting the legend to save it.

## • • • • Conclusion

This tutorial aims to provide support to air quality managers in effectively harnessing satellite data for ozone assessment in support of public health endeavors. A key aspect of this tutorial involves plotting the Formaldehyde to Nitrogen dioxide Ratio (FNR) to determine the region's sensitivity to NO<sub>x</sub> and VOCs, thereby facilitating the development of mitigation strategies, State Implementation Plans, and other applications. Furthermore, a critical component of comprehending ozone sensitivities on highly polluted days involves a comparative analysis between higher-ozone days and lower-ozone days.

## Code

```
var geometry = ee.Geometry.Polygon(
  [[[-75.21227074821776, 41.428499909904694],
    [-75.21227074821776, 40.02149602878241],
    [-72.75682641228026, 40.02149602878241],
    [-72.75682641228026, 41.428499909904694]]]);

var formaldehyde = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_HCHO')
  .select('tropospheric_HCHO_column_number_density')
  .filterDate('2022-05-01', '2022-10-01');

var no2 = ee.ImageCollection('COPERNICUS/S5P/OFFL/L3_NO2')
  .select('tropospheric_NO2_column_number_density')
  .filterDate('2022-05-01', '2022-10-01');

var formaldehyde_high = formaldehyde.filter(ee.Filter.or(
  ee.Filter.date('2022-05-08', '2022-05-09'),
  ee.Filter.date('2022-06-25', '2022-06-26'),
  ee.Filter.date('2022-06-30', '2022-07-01'),
  ee.Filter.date('2022-07-13', '2022-07-15'),
  ee.Filter.date('2022-07-20', '2022-07-21'),
  ee.Filter.date('2022-07-22', '2022-07-24'),
  ee.Filter.date('2022-07-27', '2022-07-28'),
  ee.Filter.date('2022-08-02', '2022-08-03'),
  ee.Filter.date('2022-08-04', '2022-08-05'),
  ee.Filter.date('2022-08-11', '2022-08-12')
));

var formaldehyde_low = formaldehyde.filter(ee.Filter.or(
  ee.Filter.date('2022-05-01', '2022-05-08'),
  ee.Filter.date('2022-05-09', '2022-06-25'),
  ee.Filter.date('2022-06-26', '2022-06-30'),
  ee.Filter.date('2022-07-01', '2022-07-13'),
  ee.Filter.date('2022-07-15', '2022-07-20'),
  ee.Filter.date('2022-07-21', '2022-07-22'),
  ee.Filter.date('2022-07-24', '2022-07-27'),
  ee.Filter.date('2022-07-28', '2022-08-02'),
  ee.Filter.date('2022-08-03', '2022-08-04'),
  ee.Filter.date('2022-08-05', '2022-08-11'),
  ee.Filter.date('2022-08-12', '2022-10-01')
));

var no2_high = no2.filter(ee.Filter.or(
  ee.Filter.date('2022-05-08', '2022-05-09'),
  ee.Filter.date('2022-06-25', '2022-06-26'),
  ee.Filter.date('2022-06-30', '2022-07-01'),
  ee.Filter.date('2022-07-13', '2022-07-15'),
  ee.Filter.date('2022-07-20', '2022-07-21'),
  ee.Filter.date('2022-07-22', '2022-07-24'),
  ee.Filter.date('2022-07-27', '2022-07-28'),
  ee.Filter.date('2022-08-02', '2022-08-03'),
  ee.Filter.date('2022-08-04', '2022-08-05'),
```

```

ee.Filter.date('2022-08-11', '2022-08-12')
));

var no2_low = no2.filter(ee.Filter.or(
  ee.Filter.date('2022-05-01', '2022-05-08'),
  ee.Filter.date('2022-05-09', '2022-06-25'),
  ee.Filter.date('2022-06-26', '2022-06-30'),
  ee.Filter.date('2022-07-01', '2022-07-13'),
  ee.Filter.date('2022-07-15', '2022-07-20'),
  ee.Filter.date('2022-07-21', '2022-07-22'),
  ee.Filter.date('2022-07-24', '2022-07-27'),
  ee.Filter.date('2022-07-28', '2022-08-02'),
  ee.Filter.date('2022-08-03', '2022-08-04'),
  ee.Filter.date('2022-08-05', '2022-08-11'),
  ee.Filter.date('2022-08-12', '2022-10-01')
));

// Calculate the mean values for the geometry of interest
var formaldehyde_avg = formaldehyde.mean().clip(geometry);
var no2_avg = no2.mean().clip(geometry);

var formaldehyde_low_avg = formaldehyde_low.mean().clip(geometry);
var no2_low_avg = no2_low.mean().clip(geometry);

var formaldehyde_high_avg = formaldehyde_high.mean().clip(geometry);
var no2_high_avg = no2_high.mean().clip(geometry);

// Calculate the number of pixels used to create the averaged pixel
var formaldehyde_avg_count = formaldehyde.count().clip(geometry);
var no2_avg_count = no2.count().clip(geometry);

var formaldehyde_low_count = formaldehyde_low.count().clip(geometry);
var no2_low_count = no2_low.count().clip(geometry);

var formaldehyde_high_count = formaldehyde_high.count().clip(geometry);
var no2_high_count = no2_high.count().clip(geometry);

// Create mask for each raito to only keep pixels that were created by
// averaging at least 10 pixels
var mask = formaldehyde_avg_count.gte(10).and(no2_avg_count.gte(10));
var mask_low = formaldehyde_low_count.gte(10).and(no2_low_count.gte(10));
var mask_high = formaldehyde_high_count.gte(10).and(no2_high_count.gte(10));

// Apply the Masks
var fnr_avg = formaldehyde_avg.divide(no2_avg)
  .rename('Total Ratio')
  .updateMask(mask);

var fnr_low = formaldehyde_low_avg.divide(no2_low_avg)

```

```

.rename('Low Ratio')
.updateMask(mask_low);

var fnr_high = formaldehyde_high_avg.divide(no2_high_avg)
.rename('High Ratio')
.updateMask(mask_high);

// FNR Color Palette
var palette = ['black', 'blue', 'purple', 'cyan', 'green', 'yellow', 'red'];

var band_viz = {
  min: 0,
  max: 6,
  palette: palette
};

Map.addLayer(fnr_avg, band_viz, 'Avg Formaldehyde/NO2');
Map.addLayer(fnr_high, band_viz, 'High Formaldehyde/NO2');
Map.addLayer(fnr_low, band_viz, 'Low Formaldehyde/NO2');

//Creating a legend
function makeColorBarParams(palette) {
  return {
    bbox: [0, 0, 1, 0.01],
    dimensions: '10x100',
    format: 'png',
    min: 0,
    max: 1,
    palette: palette,
  };
}

var colorBar = ui.Thumbnail({
  image: ee.Image.pixelLonLat().select(0),
  params: makeColorBarParams(band_viz.palette),
  style: { stretch: 'horizontal', margin: '0px 0px', minWidth: '200px', maxHeight: '24px'},
});

var legendLabels = ui.Panel({
  widgets: [
    ui.Label(band_viz.min, { margin: '4px 0px'}),
    ui.Label(
      ((band_viz.max - band_viz.min) / (2 + band_viz.min)),
      { margin: '4px 0px', textAlign: 'center', stretch: 'horizontal'}
    ),
    ui.Label(band_viz.max, { margin: '4px 0px' }),
  ],
  layout: ui.Panel.Layout.flow('horizontal')
});

var legendTitle = ui.Label({
  value: 'Map Legend: FNR',

```



```

    style: {fontWeight: 'bold',
      padding: '5px 0px'
    }
  });

var legendPanel = ui.Panel([legendTitle, colorBar, legendLabels]);
Map.add(legendPanel);

var thumbnailFNRAvg = fnr_avg.getThumbURL({
  'min': 0,
  'max': 6,
  'palette': palette,
  'dimensions': 500,
  'crs': 'EPSG:3857',
  'region': geometry,
});
print('FNR Avg', thumbnailFNRAvg);
var thumbnailFNRhigh = fnr_high.getThumbURL({
  'min': 0,
  'max': 6,
  'palette': palette,
  'dimensions': 500,
  'crs': 'EPSG:3857',
  'region': geometry,
});
print('FNR High', thumbnailFNRhigh);
var thumbnailFNRLow = fnr_low.getThumbURL({
  'min': 0,
  'max': 6,
  'palette': palette,
  'dimensions': 500,
  'crs': 'EPSG:3857',
  'region': geometry,
});
print('FNR low', thumbnailFNRLow);

```

## References

Eskes, H., Geffen, J. van, Boersma, F., Eichmann, K.-U., Apituley, A., Pedergrana, M., Sneep, M., Veefkind, J. P., & Loyola. (2019). *Sentinel-5 precursor/TROPOMI Level 2 Product User Manual Nitrogen dioxide*. 3.

Jin, X., Fiore, A., Boersma, K. F., Smedt, I. D., & Valin, L. (2020). Inferring Changes in Summertime Surface Ozone–NO<sub>x</sub>–VOC Chemistry over U.S. Urban Areas from Two Decades of Satellite and Ground-Based Observations. *Environmental Science & Technology*, 54(11), 6518–6529. <https://doi.org/10.1021/acs.est.9b07785>

Romahn, F., Pedergrana, M., Loyola, D., Apituley, A., Sneep, J., & Veefkind, J. P. (2022). Sentinel-5 precursor/TROPOMI Level 2 Product User Manual Formaldehyde HCHO. *Remote Sensing of Environment*, 120, 70–83. <https://doi.org/10.1016/j.rse.2011.09.027>

Vigouroux, C., Langerock, B., Bauer Aquino, C. A., Blumenstock, T., Cheng, Z., De Mazière, M., De Smedt, I., Grutter, M., Hannigan, J. W., Jones, N., Kivi, R., Loyola, D., Lutsch, E., Mahieu, E., Makarova, M., Metzger, J.-M., Morino, I., Murata, I., Nagahama, T., Notholt, J., Ortega, I., Palm, M., Pinardi, G., Röhling, A., Smale, D., Stremme, W., Strong, K., Sussmann, R., Té, Y., van Roozendaal, M., Wang, P., and Winkler, H. (2020). TROPOMI–Sentinel-5 Precursor formaldehyde validation using an extensive network of ground-based Fourier-transform infrared stations. *Atmospheric Measurement Techniques*, 13(7), 3751–3767. <https://doi.org/10.5194/amt-13-3751-2020>

Witman, S. & Holloway, Tracey & Reddy, Patrick. (2014). Integrating satellite data into air quality management: Experience from Colorado. *EM: Air and Waste Management Association's Magazine for Environmental Managers*. 34-38.