SESUG Paper 214-2018

Five Crazy Good Visualizations and How to Plot Them

David Mintz, U.S. Environmental Protection Agency

ABSTRACT

We all have our favorite visualizations. The best ones deliver a clear message to the intended audience. Over the years, there are a few that have won my affection. I would like to share my top five with you, along with the code and a few anecdotes about why they make the list. Some of these examples are static; others are interactive. This paper will cover SAS/GRAPH® and ODS Graphics procedures. It will also touch on a few basic elements of good graphical design.

INTRODUCTION

This paper describes five plots, why I love them, and how to plot them. My hope is that you will use them, improve upon them, and customize them to fit your need. I will provide and explain the relevant segments of the code. Feel free to contact me for a copy of the complete code which, in most cases, includes code for annotate data sets, macro variables, and additional options.

TILE PLOT

The tile plot arranges data in chronological order, revealing any temporal patterns that might exist. It is called a tile plot because one square, or tile, is plotted for each unit of time. In the Figure 1 example, the tiles represent one day. The color of each tile represents the Air Quality Index (AQI) category that corresponds to the measured ozone concentration that day. The orange, red, and purple tiles represent days when ozone concentrations were in unhealthy ranges. You can see that most of the unhealthy days occurred in generally warmer months, with the highest category being predominant in the peak of summer. This plot reveals a typical seasonal pattern associated with ozone which is more readily formed in the presence of abundant heat and sunlight (in combination with the presence of certain pollutants).

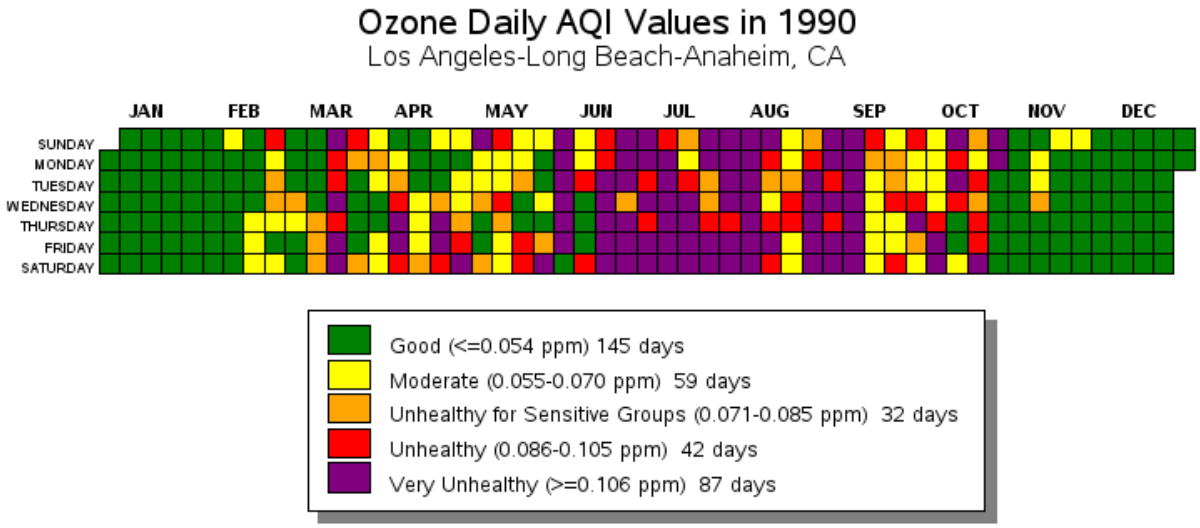

Figure 1. Daily Air Quality Index (AQI) values for ozone in Los Angeles, 1990.

I traced the origins of this plot back to George C. Tiao, a statistician and University of Chicago Professor Emeritus. According to my source, Dr. Tiao made his first plots by hand probably in the 1980s. Tom Curran brought the tile plot concept to EPA and developed a prototype in SAS circa 1990. Terence FitzSimons sharpened that code and developed the version that EPA first published in its 1991 annual air quality trends report.[1] Since its initial publication, this plot has also been featured in SESUG 95 and SUGI

22 conference proceedings and in a book entitled *Visualization of Time-Oriented Data*.[2,3,4] The plot is also featured as a "Bright Idea" on Michael Friendly's DataVis website.[5] EPA currently provides this plot as a data visualization tool on the AirData website.[6]

I love this plot because it arranges data in a chronological order that promotes pattern detection. It also plots a lot of data in a small space.

The GMAP procedure works by matching location information from the "data" data set to a "map" data set. The map data set does not have to represent geography. It is simply a grid on which you plot your data based on a common id variable. Practically, you specify the two input data sets in the PROC GMAP statement, like this:

```
proc gmap map=grid data=values;
```

The two input data sets must contain a common id variable. This is how you construct the id variable in the "map" data set:

```
data grid;
do ix=0 to 53; *53 weeks a year;
 do iy=0 to 6; *7 days a week;
   id=100*iy+ix;
   x=ix;y=iy;output;
   x=ix+1;y=iy;output;
   x=ix+1;y=iy+1;output;
   x=ix;y=iy+1;output;
 end;
end;
drop ix iy;
output;
run;
```

In concept, the resulting grid looks like this, where I have labeled each cell to illustrate the associated id.

600	601	602	603	604	...
500	501	502	503	504	...
400	401	402	403	404	...
300	301	302	303	304	...
200	201	202	203	204	...
100	101	102	103	104	...
0	1	2	3	4	...

Then you must create an id associated with each day in your "data" data set:

```
data values;
     keep sasdate id val;
     set sourcedata;
     year=year(sasdate);
     frstday=mdy(1,1,year);
     yplot=7-weekday(sasdate);
     xplot=intck('week',frstday,sasdate);
     id=100*yplot+xplot;
run;
```

When you run PROC GMAP with these two data sets and matching id variable, it will overlay the data onto the grid.

```
goptions colors=(green yellow orange red purple);
pattern value=msolid;

proc gmap map=grid data=values;
   id id;
   choro val/discrete
   annotate=labels
   cempty=gray;
run;
quit;
```

MULTIYEAR TILE PLOT

While the original tile plot in Figure 1 shows a single year of data (unless you stack multiples), the multiyear tile plot, shown in Figure 2, strings out the data for each year into a single row and stacks multiple years of data. This arrangement enables you to detect long-term trends in the data. Plus, you can see if there are any changes in seasonal patterns – if they are shifting or lengthening or shortening, for example. You can see if there have been any multiday episodes or anomalous days.

Ben Wells, an EPA colleague, designed this plot. Though there is a similar version on *Robert Allison's SAS/Graph Examples* website, Ben developed this design independently.[7] EPA currently provides this plot as a data visualization tool on the AirData website.[8]

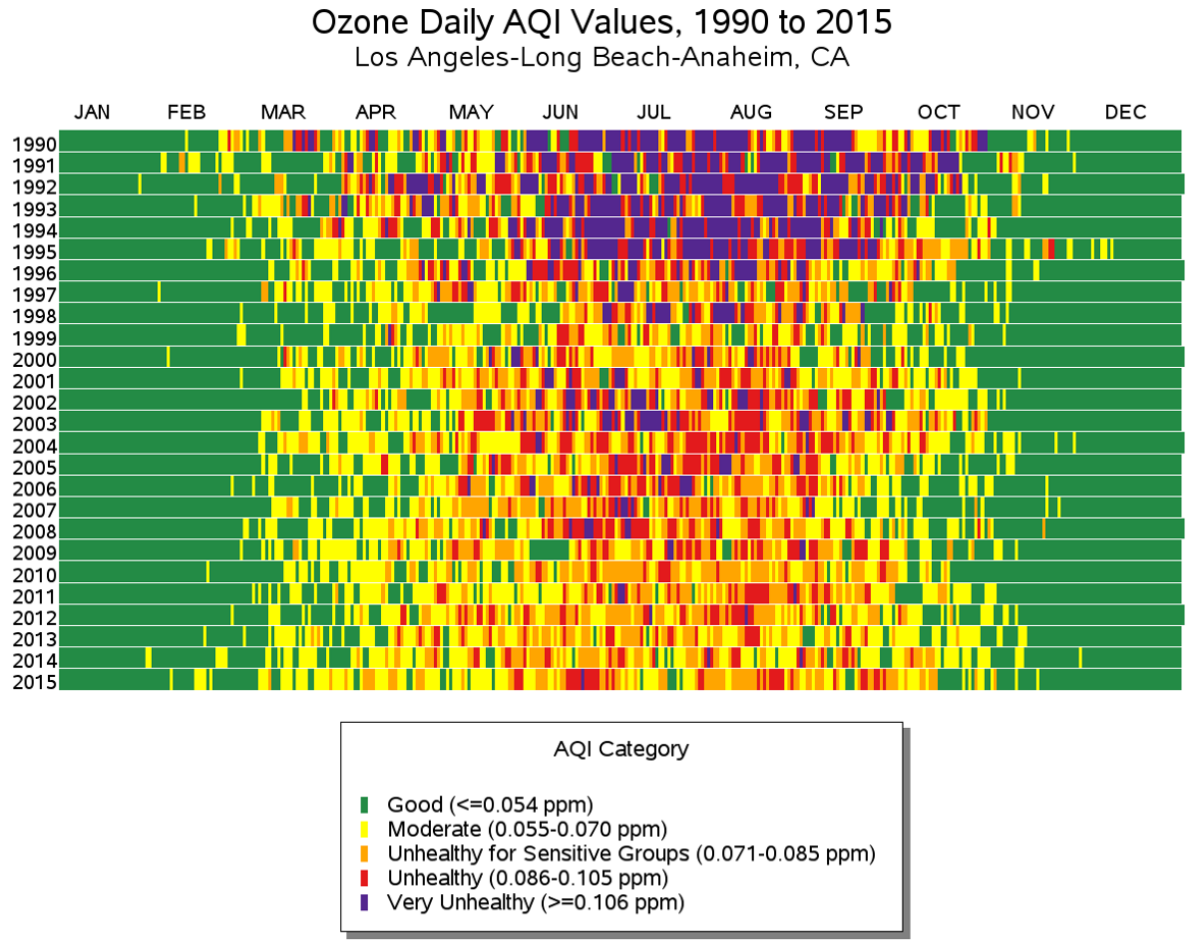

Figure 2. Daily Air Quality Index (AQI) values for ozone in Los Angeles, 1990 - 2015

I love this plot because it displays a lot of data in a small space. You can potentially fit as many as 50 years on a single page – that's about 18,000 data points! It also makes use of small multiples. Small multiples are thumbnails of the same graphic design repeated in several frames on the same page. "The design remains constant through all the frames, so that attention is devoted entirely to shifts in the data."[9] This is useful because once readers understand the design of one plot, they can quickly interpret and compare all the others. And, like the single year tile plot, it also arranges data in chronological order and promotes pattern detection.

The basic code structure is the same as it is for the single year tile plot - there are two input data sets with a common id variable that PROC GMAP uses to match them. The primary difference is how you draw the grid and assign ids to each cell.

This code creates the "map" data set, or grid, for the years 1990 to 2015. (The full code accepts different start and end years with the use of macro variables.)

```
data grid;
do ix=0 to 366; *366 days per year;
 do iy=0 to 25; *number of years;
   id=1000*iy+ix;
   x=ix;y=iy*7;output;
   x=ix+1;y=iy*7;output;
   x=ix+1;y=y+7;output;
   x=ix;output;
 end;
end;
drop ix iy;
output;
run;
```

The following code creates an id associated with each day in the "data" data set:

```
data values;
     keep sasdate id val;
     set sourcedata;
     year=year(sasdate);
     frstday=mdy(1,1,year);
     yplot=2015-year;
     xplot=intck('day',frstday,sasdate);
     id=1000*yplot+xplot;
run;
```

The GMAP procedure is the same as it is for the single year tile plot, except for the *coutline* option. The *coutline=same* assignment makes the outline of each cell the same color as the cell. Different colored cell outlines would detract from the visual. And to maintain a reasonable overall image width, you cannot afford extra horizontal pixel space to outline each cell.

```
goptions colors=(green yellow orange red purple);
pattern value=msolid;

proc gmap map=grid data=values;
   id id;
   choro val/discrete
   annotate=labels
   cempty=gray
   coutline=same;
run;
quit;
```

STACKED HISTOGRAM

This stacked histogram is a great example of a “once and done” plot, designed to answer a single question. It illustrates John W. Tukey’s assertion that “an approximate answer to the right question is worth a great deal more than a precise answer to the wrong question." Let me give some background and then pose the question this plot answers.

In recent years, there has been an increase in the production and use of air quality sensors that can measure and report pollutant concentrations at 1-minute intervals. This is great but also challenging because there are no available studies that indicate the health impact of a 1-minute reading. So, it’s difficult to know what level of a 1-minute reading should be of concern. EPA’s national air quality standard for ozone is based on data averaged over an 8-hour period, and there are ample health studies based on 8-hour exposures. So, how do we relate 1-minute values for which we have no health information to 8-hour averages for which we do have health information? This is tricky because 1-minute concentrations can be quite variable over an 8-hour period. The precise question is, “If I see a 1-minute value of x, what’s the likelihood it is part of an 8-hour average that is above a level of concern (like the level of the standard)?”

To answer this question, I binned the distribution of 8-hour averages at 1-minute intervals from 0.01 to 0.13 parts per million (ppm). The plot in Figure 3 shows all the distributions stacked vertically. The vertical reference line indicates the level of the standard (which at the time was 0.075 ppm and was revised to 0.070 ppm in 2015). From this plot you can see that the higher the 1-minute value, the more likely the associated 8-hour average is to be above the level of the standard. For example, when you get to a 1-minute value of 0.11 ppm, 70 percent of the associated 8-hour averages are above the level of the standard (from calculated z-scores). When you get a 1-minute value of 0.13 ppm, 89 percent of 8-hour averages are above the level of the standard. This graphic is a simple illustration of a complex relationship between 1-minute readings and 8-hour averages. EPA is using information from a similar analysis to develop public messaging for 1-minute readings to help people interpret sensor data and take appropriate cautionary action if necessary.

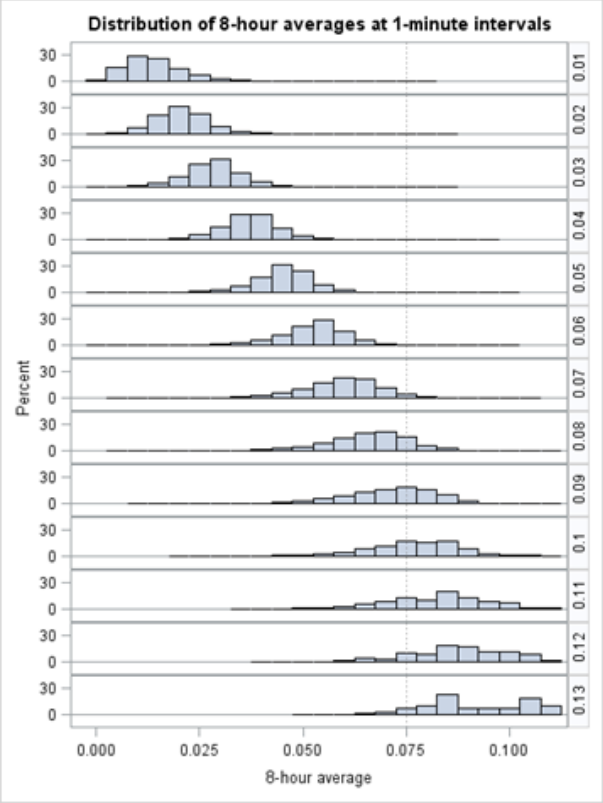

Figure 3. Distribution of ozone 8-hour averages at 1-minute intervals

I love the stacked histogram because it simplifies a relatively complex problem. It also makes use of small multiples.

This plot uses ODS Graphics, specifically the SGPANEL procedure. It takes only eight lines of code, including the RUN statement.

```
proc sgpanel data=dat.intervals_all_sorted noautolegend;
     where interval ge .01;
     panelby interval / layout=rowlattice uniscale=row spacing=0 onepanel
proportional novarname;
     histogram conc_8hr_average / binwidth=.005;
     refline .075 / lineattrs=(pattern=dot color=gray) axis=x;
     rowaxis discreteorder=data;
     title "Distribution of 8-hour averages at 1-minute intervals";
run;
```

INTERACTIVE SVG MAP

The inspiration for this interactive map came from the New York Times Upshot map in Figure 4, published June 26, 2014 by Alan Flippen. At the time of this writing, the map and related article were still available on the web.[10] It is a simple county choropleth map. What makes it great is that it is interactive and data-rich. You can get information for all 3,000 or so counties simply by moving your mouse from county to county.

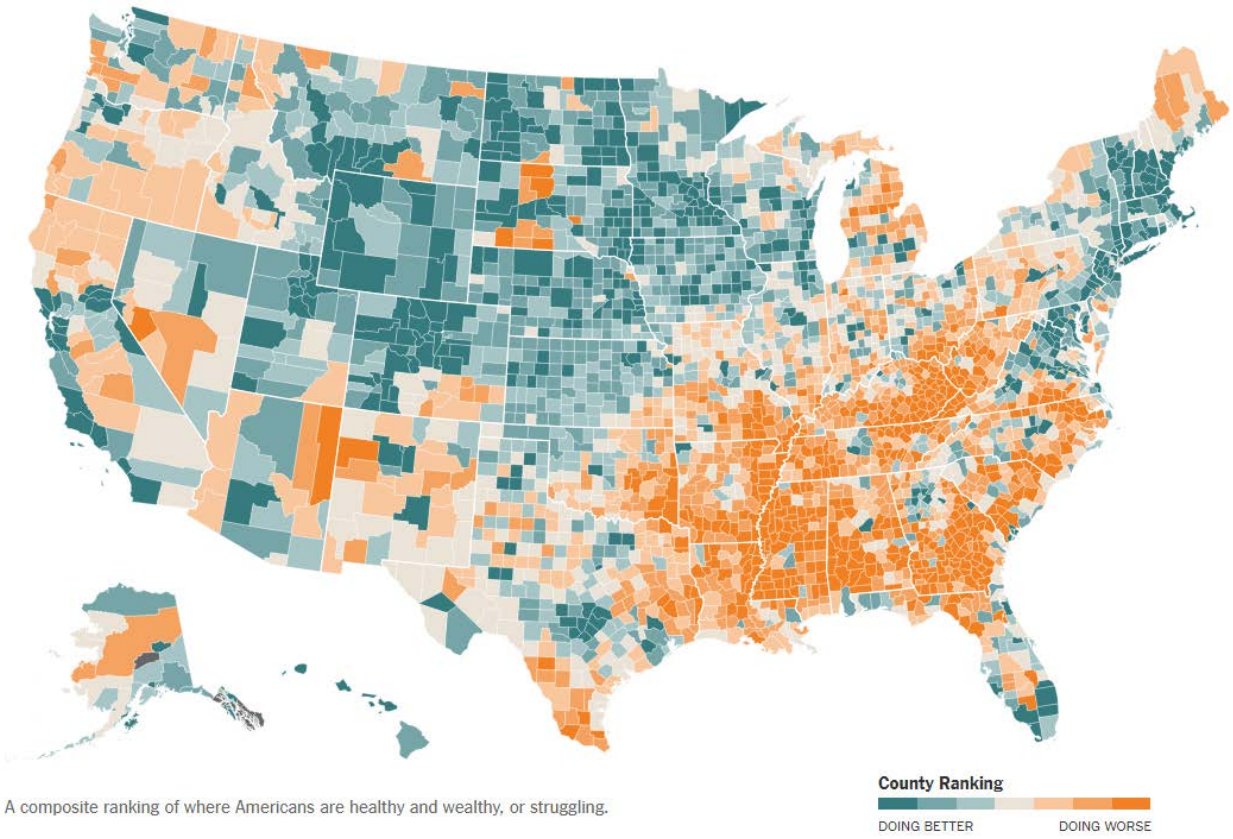

Figure 4. Interactive map from the New York Times Upshot website.

The SAS version modeled after this map is shown in Figure 5. You can play with the interactive features of this map on EPA's AirCompare website.[11] The binary color scheme simply denotes whether a county has air quality monitoring data (darker shaded counties have data). The more important information is revealed when you mouseover the map.

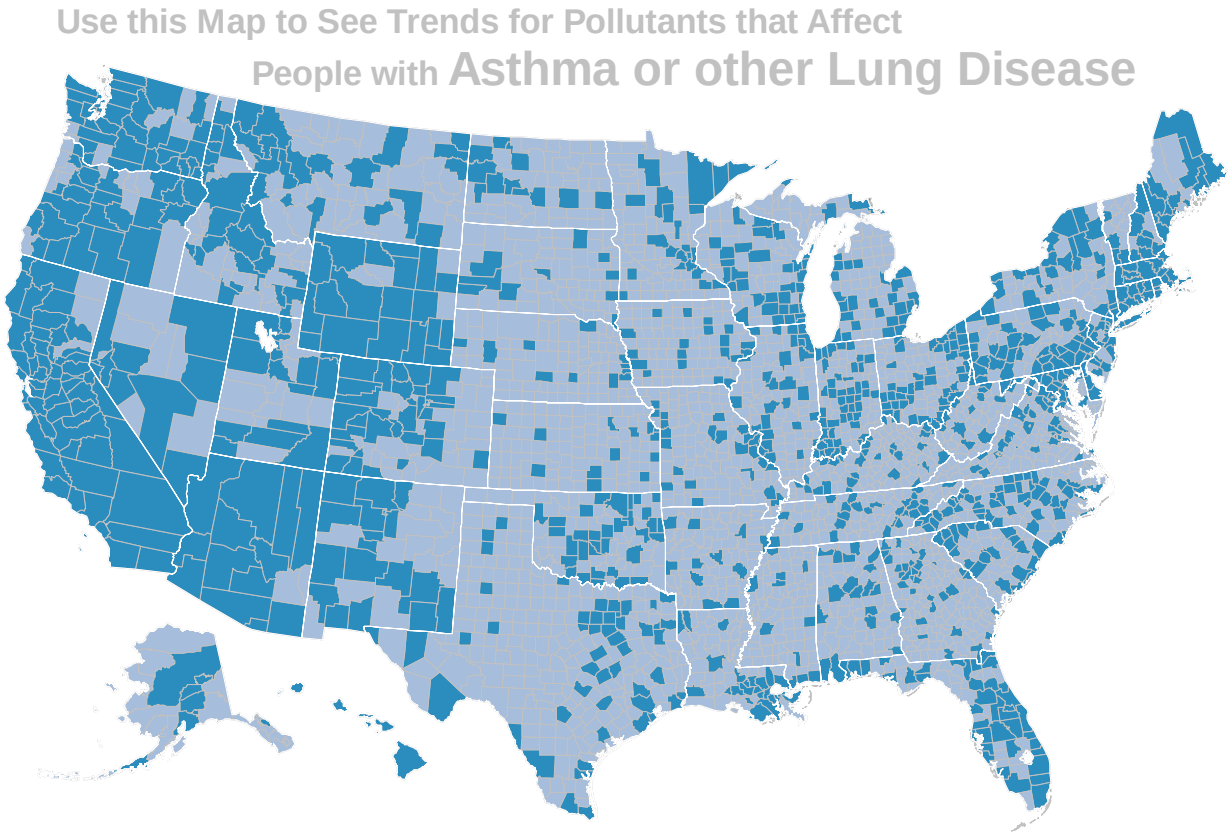

Figure 5. Interactive SVG map from AirCompare website

I love this interactive map because you can get so much information, including plots, quickly and easily just by hovering over the map.

I chose to create this map as a scalar vector graphic (SVG) image because, beginning with SAS 9.3, the SVG graphics device has special attribute called ONMOUSEOVER=HTML.[12] This attribute allows you to link to images or text that will appear when you mouseover any part of the SVG. The following example shows how to link to both conditionally. If a county has data, then it links to a pre-generated plot of that data. If a county has no data, then it links to text that says, "NO DATA." This is all done in the map data set that is fed into PROC GMAP. Here is how to specify the ONMOUSEOVER=HTML attribute in the map data set:

```
data newmapdata;
     drop density drillurl imgurl;
     length htmlvar $600.;
     set mapdata;
     *make plot appear over county;
     if xanchor gt .073 then do;
          xpix=round(1200/0.726398*xanchor+615.40643,1) - 335; *reposition
graph to left of eastern counties (all counties east of st.louis);
     end;
     else do;
```

```
          xpix=round(1200/0.726398*xanchor+615.40643,1) + 10; *for
everything west of st.louis;
     end;
     ypix=round(320-(727/0.45422*yanchor),1) + 100; *adjust so it does not
display off screen at top;
     *point to plots;
     drillurl='./aqiplot.html#rpt';
     imgurl='./areaname.svg';
     *only want counties with data to be clickable;
     if mondata=1 then do;
     htmlvar="href="||quote(trim(imgurl))||
      " ONMOUSEOVER=ShowImage("||quote(trim(imgurl))||
      ","||compress(xpix)||","||compress(ypix)||",300,300);
changeOpacity(.5)";
     end;
     else do;
     altvar='title="'||trim(left(areaname))||' - NO DATA"';
     htmlvar=trim(left(altvar));
     end;
run;
```

There is nothing special about the “data” data set. It simply contains the common id variables – state and county – and one other variable that indicates whether the county has data or not.

PROC GMAP uses the map data set and the “data” data set to generate the map as follows:

```
goptions reset=all dev=svg;

ods html body="yourmap.html" path=gout style=listing
    parameters=("drilldownmode"="html");
filename gout "C:\yourdirectory";

pattern1 value=msolid color=cx2b8cbe; *dark blue;
pattern2 value=msolid color=cxa6bddb; *light blue;

proc gmap map=newmapdata data=values all anno=anno_outline;
id state county;
choro mondata / cempty=gray html=htmlvar
                coutline=ltgray nolegend;
run;
quit;
ods html close;
```

CUMULATIVE FREQUENCY PLOT

The cumulative frequency plot has been around for almost 200 years (no kidding).[13] Yet, it is still as sexy as ever. ESPN's mobile app uses this plot to show "game flow" which is the real-time cumulative score for each team. Figure 6 shows the score progression of Game 4 of the 2018 NBA Eastern Conference finals. Here you can see that Boston fell behind early in the first period and played from behind the rest of the game. (Aside: Cleveland went on to win the series, but lost in the finals to Golden State.)

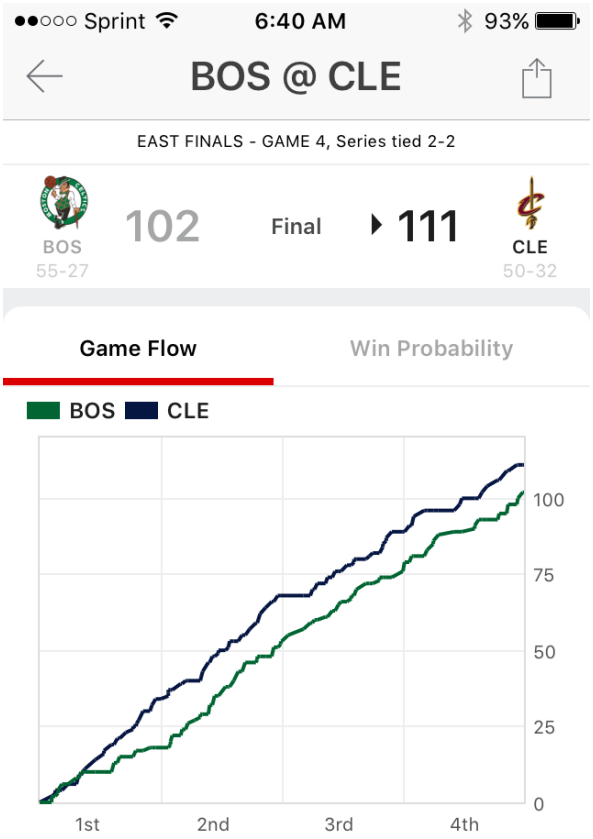

Figure 6. ESPN's "Game Flow" chart

A former mentor and EPA colleague, Terence FitzSimons, used this plot to keep a running total of how many days the ozone standard had been exceeded in different U.S. cities. His plot was published in EPA's 2003 annual air quality trends report.[14] Currently, this plot is available as a tool on EPA's AirData website.[15] There, it is connected directly to incoming ozone data, so you can track how many exceedances have occurred this year compared to last year or any previous year. Figure 7 shows an example comparing 2017 and 2018 in Los Angeles. In this example, you can see that the first exceedance of 2018 occurred later than it did in 2017. In May, it reached the same number as the previous year. Then the 2018 rate fell behind and slowed. And midway through August, there have been about 20 fewer exceedances than at the same time in 2017.

I love this plot because it helps you track the progress of one variable over time with respect to another.

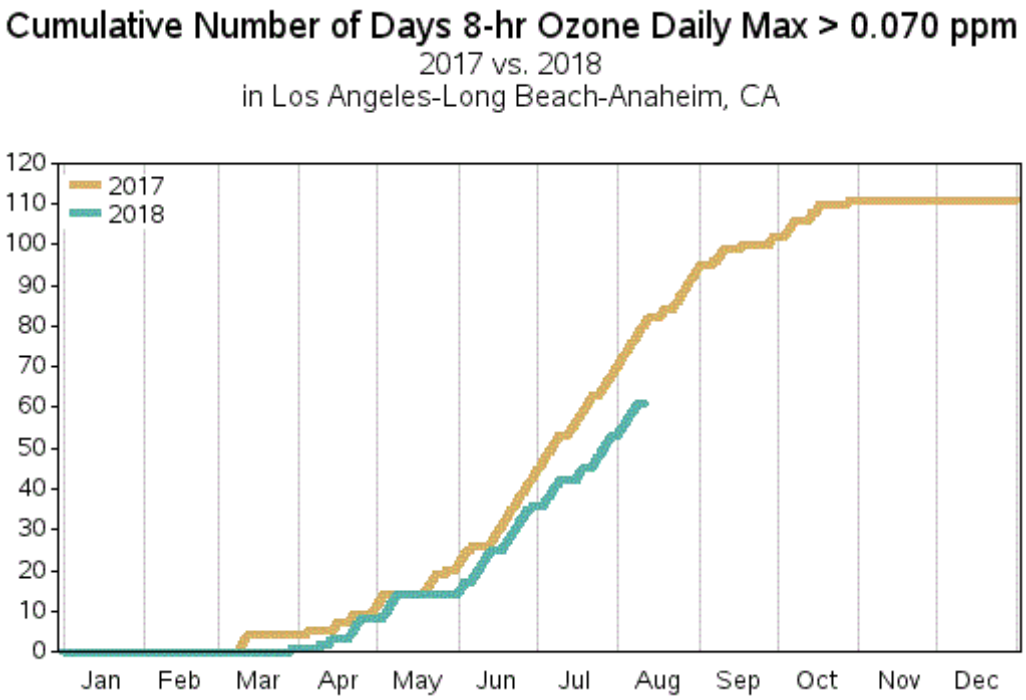

Figure 7. Cumulative number of days 8-hr ozone daily max > 0.070 ppm

This plot uses the GPLOT procedure with an overlay option to plot two lines that happen to represent cumulative frequencies. The horizontal axis represents one year from January 1st to December 31st, with reference lines that delineate months. The key part of the code is in the symbol statement. The *i=stepjr* option tells SAS how to draw the lines. Specifically, "step" tells SAS to plot the data as a step function. The "j" produces steps joined with a vertical line. And, the "r" displays the data point on the right of the step. Here is the code:

```
legend1 shape=bar(3,1) position=(top left inside) across=1 label=none
value=(j=l f=arial "2017" "2018");

pattern1 value=solid c=cxd8b365;
pattern2 value=solid c=cx5ab4ac;

axis1 offset=(0,) color=black label=none;
axis2 label=none value=none order=(1 to 365 by 1) major=none minor=none;

proc gplot data=values anno=annodata;
plot cum_count_base*jdate cum_count_comp*jdate /
          overlay
          haxis= 1 365
          href = 1 32 60 91 121 152 182 213 244 274 305 335 365
          chref=CXCCCCCC
          vminor=0
          vaxis=axis1
          haxis=axis2
          caxis=black
          legend=legend1;
symbol1 c=cxd8b365 v=none i=stepjr w=3;
symbol2 c=cx5ab4ac v=none i=stepjr w=3;
run;
quit;
```

CONCLUSION

When constructing pictures of data, careful and thoughtful design is essential. SAS/GRAPH and ODS Graphics offer myriad options for developing extremely customized static and interactive visualizations. I offer the examples in this paper as a resource for the data visualization community to use and improve upon.

REFERENCES

1. *National Air Quality and Emissions Trends Report, 1991*, EPA-450-R-92-001, U.S. Environmental Protection Agency, Office of Air Quality Planning and Standards, Research Triangle Park, NC, October 1992.
2. Mintz, Fitz-Simons, Wayland. September 1995. "Tracking Air Quality Trends with SAS/GRAPH® Software." Proceedings from the Third Annual Southeast SAS Users Group Conference (SESUG 95). Available at https://www.lexjansen.com/cgi-bin/xsl_transform.php?x=sesug1995
3. Mintz, Fitz-Simons, Wayland. March 1997. "Tracking Air Quality Trends with SAS/GRAPH® Software." Proceedings of the Twenty-Second Annual SAS Users Group International Conference (SUGI22). Available at https://www.lexjansen.com/cgi-bin/xsl_transform.php?x=sugi22
4. Aigner, Miksch, Schumann, Tominski. February 2011. *Visualization of Time-Oriented Data*. p.178. Available at https://www.springer.com/us/book/9780857290786
5. DataVis.ca website, Michael Friendly (York University). Available at http://www.datavis.ca/gallery/bright-ideas.php
6. EPA's AirData website, Single Year Tile Plot. Available at https://www.epa.gov/outdoor-air-quality-data/air-data-tile-plot
7. Robert Allison's SAS/Graph Examples! "Delays, by Carrier over Time." Available at http://robslink.com/SAS/democd40/aaaindex.htm.
8. EPA's AirData website, Multiyear Tile Plot. Available at https://www.epa.gov/outdoor-air-quality-data/air-data-multiyear-tile-plot
9. Tufte, Edward R. 1983. *The Visual Display of Quantitative Information*. Cheshire, CT.
10. New York Times Upshot website, June 26, 2014 by Alan Flippen, Where Are the Hardest Places to Live in the U.S.?, Available at https://www.nytimes.com/2014/06/26/upshot/where-are-the-hardest-places-to-live-in-the-us.html?abt=0002&abg=1&_r=0
11. EPA's AirCompare website. Available at https://www3.epa.gov/aircompare
12. Enhancing Drill-Down Behavior in SVG Presentations Using HTML Attributes, SAS/GRAPH® 9.3: Reference, Third Edition, http://support.sas.com/documentation/cdl/en/graphref/65389/HTML/default/viewer.htm#n1w35hqi9fjk5dn1w5rv8gsn7j5h.htm
13. DataVis.ca website, Michael Friendly (York University). Available at http://www.datavis.ca/milestones/index.php?group=1800%2B&mid=ms96
14. *National Air Quality and Emissions Trends Report, 2003*, EPA-454-R-03-005, U.S. Environmental Protection Agency, Office of Air Quality Planning and Standards, Research Triangle Park, NC, September 2003.
15. EPA's AirData website, AirData – Ozone Exceedances, Available at https://www.epa.gov/outdoor-air-quality-data/air-data-ozone-exceedances

ACKNOWLEDGMENTS

I would like to thank Michelle Wayland, James Hemby, and Jackie Ashley for reviewing this paper. Additionally, I would like to thank Dave Dickey who introduced me to SAS many years ago, Terence FitzSimons for being a longtime SAS mentor and friend, and Tom Curran for teaching me the intricacies of the original tile plot code.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Please contact me at:

David Mintz
U.S. Environmental Protection Agency
919-541-5224
mintz.david@epa.gov