mapStats: An R Package for Geographic Visualization of Survey Statistics
Samuel Ackerman (Temple University Statistics), Dr. Richard Heiberger (Advisor)

Introduction
- Many surveys or datasets contain a geographic variable (e.g., state). Often, the distribution of a variable (such as income) depends on geography, and visualizing this distribution is a basic step of analysis.
- In R, visualization of variables on a map is difficult because the same code generally has to be manually changed for each map, familiarity with graphics is needed, and calculating statistics is hard.
- My package mapStats (available on CRAN (R >= 3.0.2)) provides tools for calculating statistics and generates maps with one call.
- Example: Environmental Protection Agency (EPA) data used to calculate the mean daily ozone reading in April of 2012 and 2013 of each site.
- See package demo for more detailed examples.

Calculating Statistics of Variables
- The functions calcStats and calcQuantiles calculate means, sums, and quantiles for numeric variables, weighted or unweighted, at levels of one or two class variables.
- For mapping, one class variable is the geography, the other is optional.
- Functions adapt survey package output for means and totals, and rq package output for quantiles, and emulate SAS PROC TABULATE.
- Only one variable but multiple statistics are allowed.

Initial Visualization
- The variable is the mean April ozone measurement (parts per million) by monitoring site in the years 2012 and 2013.
- Calculate the mean and maximum of the measurements by state.
- For printing multiple plots per page, the default output is confusing because the same colors represent different values on each plot.
- Set parameter separate to FALSE so that color breaks are the same across all statistics so panels can be compared.
- Function call:
  ```r
  mapStats(d=epa.means, var="ozone.mean", stat=c("mean","quantile"), quantiles=1, d.geo.var="state", map.file=us, map.geo.var="NAME", map.label.names="STUSPS", wt.label=FALSE, var.pretty="April ozone measurements (ppm)", palette=c("Reds"), num.row=2, separate=FALSE)
  ```

By Variables and Color Palettes
- Calculate the mean and maximum for the variable by year (parameter by.var); the same color scale will be used for all levels of the by variable.
- Use 3 colors for the mean and 5 for the maximum (parameter ngroups) and different colors (parameter palette).
- Sequential HCL palettes can be used instead of the default RColorBrewer palettes by using the col parameter.
- Suppress state labels (parameter map.label) to improve visibility.

Adding Spatial Features
- Superimpose shapefiles, points, or text (sp.layout.pars).
- Include locations of electric power plants in the US on the map by defining a sp.points object.
- Shade regions in green where ozone emissions consistently exceed national standards by defining a sp.polygons object.

Layout of Panels
- Use num.col and num.row parameters to control arrangement of panels by page, and lattice layout argument for layout of plots within each panel.
- Default titles display across plots for each statistic. When there are two or more columns, a newline is placed after the variable name in the titles to prevent overlap. In the right panel, titles would still overlap, so use the titles parameter to place newline markers in titles.
- References