Sliver

Multivariate Data Visualization Software

v1.5

http://www.sliversoftware.com

USER'S MANUAL

Table of Contents

Sliver Version Change Log

Version 1.5 (released March 18, 2017)

- Adds support for 2D scatterplots with point sizes dependent on another variable (bubble charts). With the x variable, the y variable and the color mapping, the variable point size indicates yet another dimension in a scatterplot.
- Adds zooming capability to the PC Plot Matrix window. This may be slow for large datasets given the number of lines involved, but the zoom level is remembered. The existing zoom method for all windows (Ctrl plus mouse scroll), will now also work by just pressing the + and – keys. This particularly helps when touchpads on laptops do not have the same behavior as a PC mouse when scrolling. In all cases, including the Alt key zooms horizontally while the Shift key zooms vertically.
- Adds new features for Google Earth plots:
    - A number of options specific to a plot type have been moved from the main Google Earth settings under the Options menu into the window that appears when the plot is selected under the Plots menu. This is more convenient and allows individual settings for the different plot types. Overall settings are still populated under the Options menu.
    - Adds an option to add timestamps to any Google Earth plot. The timestamps at the points can be chosen to be the row number as seconds since midnight (simply to provide a timespan), or the values of any variable at the points as seconds. If a variable time-formatted in hh:mm:ss is selected, it is converted to seconds so the player will then show the associated time. When the KML file is opened, Google Earth presents a small player control panel that also includes a second cursor to set an interval of time. This can be used to animate the plot, only showing plotted features as the slider is manually positioned or automatically played. The automatic animation speed can be adjusted by clicking on the wrench icon in the player control panel. This feature is useful to provide animation in Google Earth without running Sliver at all, or to reduce clutter in complicated plots by only showing a particular range of the plot at a time.
    - In Pillar and Connection plots, and whenever spheres are added to a Path plot, a window provides an option to display the corresponding row values of selected variables in a pop-up when an element is left-clicked in Google Earth (called *balloons*). This is a great way to see the data at any point in the Google Earth displays.
    - Whenever spheres are added to a Pillar or Path plot, the window above will also provide an option to display the corresponding row values of any selected variables in a label when a mouse-over of a sphere is performed. This is very useful for fast analysis or in finding the location of a particular value in a plot.
    - It is possible to display a mapped string rather than a number in pop-ups. This uses a string mapping file whose path is entered in the Plot Settings. If enabled, values for each variable are compared to a list of matching numbers and strings for that variable in the file, and if a match occurs the string is displayed instead of the number. This file has the format of the output string mapping file of the *Replace Strings with Numbers* function of the DataTools menu, although the string mapping file can also be created by hand (see Appendix D for the format).
    - Count-Based Animation that includes Google Earth animation now provides an option to move the Google Earth camera view to follow the icon. The range, tilt angle and relative heading of the camera from the icon as it moves can be specified. If the option to draw a line from the icon to a secondary set of coordinates is enabled, the relative heading can also be switched in sign to always be opposite that line for better viewing of it, or the heading can always be along that line.

  - Google Earth animation sometimes loses sync due to collisions between reading the position data and writing the data, which then requires right-clicking on the Position label in the left panel and choosing *Refresh*. These occurrences are greatly reduced now by implementing doubly-nested network link files.
- Adds the choice in the **Plots→Display Values on Mouse-Over** menu function to either display the numerical values or use string mapping as described above to check values against a string mapping file populated in the Plot Settings and, if found for a variable, display the string. This only applies to the main PC Plot.
- Any variable with the string *(sec-time)* anywhere in its name is automatically interpreted as having units of seconds and will be transformed into the hh:mm:ss time format in the Sliver displays.
- Updates DataTools menu functions:
  - Sliver associates commas in double quotes in CSV files as delimiters, and fails to load such files since the column count is deemed inconsistent. Implementing a real-time check on commas when a file is loaded, though, significantly slows down the input of data. A new DataTools function *Remove Commas from Strings* is added to remove these commas and the surrounding double quotes from a CSV file prior to reading it into Sliver.
  - Added a function *Extract Rows with Multi-Column Values* to extract rows based on Boolean combinations of values across multiple variables.
  - The *Replace Strings with Numbers* function, which is more significant now that mapping to the original string values is implemented, now operates only on user-selected columns. This is also useful to avoid replacing time-formatted values that Sliver can natively input.
- Adds minor improvements:
  - Restores the functionality of the line to secondary Google Earth coordinates
  - Ignores leading zeros in data
  - Accepts non-integer seconds for time-formatted (hh:mm:ss) data
  - Skips writes to the settings file when locked due to Dropbox syncs, etc.
  - Correctly propagates time format and name staggering when axes are moved or deleted
  - Implements some performance improvements

Version 1.4 and 1.4.1 bug fix (released January 15, 2017)

- Adds a new Analysis menu and moves the Brush Statistics function to it. New options in the Analysis menu are:
  - Display of histograms and mean value markings along axes, with user-selectable axes and number of bins. If Ctrl-zooming is performed while histograms and means are displayed, no scrollbars are added, so zoom out and back in or use arrow keys to center a different position in the plot (resizing the window does not introduce this restriction).
  - Entry of ranges for user-selectable axes, which simplifies setting axes to identical ranges for comparison purposes.
  - Centering of user-selectable axes on their means to show variations.
  - Centering of user-selectable axes on values of a selected line to show variations from a reference.
  - Resetting of axis ranges to their original values that span the minimum to maximum of each.
  - Automatic K-means clustering and brushing, with a selectable number of clusters *k* and number of found clusters to color brush in a rainbow palette. This requires the free R statistical software program to be installed on your PC, with the path to the Rscript.exe file entered into the new Analysis Settings window.

  - Creation and export of 3D scatterplots for automatic display in Matlab with Sliver color brushing, providing the benefits of Matlab's interactive 3D scatterplot capabilities. This requires the Matlab program to be installed on your PC, with the path to the matlab.exe file entered into the new Analysis Settings window.
- Adds a new Selections menu and moves some selection functions from the Brush menu to it. New options in the Selections menu are:
  - Selection of all unbrushed lines, which is mostly useful for assigning them to a color brush and then hiding that brush to leave other manually-brushed lines visible.
  - Thickening of selected lines, which can be re-invoked to thicken them further, to better view important data such as reference lines for axis centering. Thickened lines only appear in the PC Plot and in plots exported to PostScript/PDF where they appear thickened, opaque, and on top of the other lines. Unthickening of all visible lines is another menu option.
  - Ctrl-A now invokes the existing option to select all visible lines.
- Updates the Plots menu:
  - The Google Earth Path Plot window now includes:
    - An option to not connect geographic points farther apart than an entered distance. Previously, plotting a set of segments would add straight lines between them, but these segments can now be isolated. Note that to show a single disconnected point, the option for supports must be enabled in the Google Earth Settings.
  - The Google Earth Connection Plot window now includes:
    - An option to assume the endpoints of the connection lie on the ground. Otherwise, lines connected between distant points can pass through the Earth if the altitude reference is not set to "Clamp to Ground" in the Google Earth Settings. This also means that the altitude variable does not have to be selected, which is necessary when only ground coordinates are available. This ground connection lies on the surface of the Earth along the great circle path between the points, with width, opacity, etc., set as before in the Google Earth Settings.
    - An option to draw a vertical parabolic arc between two connections lying on the ground rather than a line. The heights of the arc for different distance (the ratio of the distance between the points and the height) is user-selectable. This arc lies above the great circle path between the points. Again, the width, opacity, supports, etc., for the arcs is set as before in the Google Earth Settings, where you probably prefer the supports to be disabled.
  - A new Google Earth Pillar Plot is introduced, consisting of vertical rods at geographical points on the surface of the earth. This is very nice for marking individual points, even densely spaced points, with Sliver color-brushing and fixed heights or variable heights. The width and opacity of the pillars is user-selectable in the pillar plot window (not in the Google Earth Settings). The height above the ground of the pillars can either be a third variable or be set as a fixed value entered in the window. There is also an option to place a sphere on top of each pillar of that color and a configurable size and opacity.
  - A new function is added to the DataTools menu to merge multiple KML files created by Sliver into a single KML file, which is useful to combine individual path, pillar and connection plots without having to keep track of their individual KML files.
- Updates the Axis menu:
  - Adds a new Axis menu option to horizontally compress or expand axes between two user-selected axes. This is very useful for spacing out axes that are of most interest while compressing others. The compression factor is also user-selectable. There is also an option to revert to equally spaced axes.

  - The option to stagger and unstagger axis names to avoid overlapping text in PC plots of many variables now staggers the minimum and maximum axis values as well. Staggering is now preserved across sessions.
- Updates the Animate menu:
  - The Count-Based Animation window is updated to make the entries more intuitive. The maximum speed of play available in the Sliver DataPlayer window that controls the playback is now user-selectable in this window. The DataPlayer window now allows looping and zigzagging.
  - The previous restriction that a Google Earth Path Plot had to be created before Google Earth animation is enabled has been removed. If selected, the variables for the geographic coordinates are now selected as part of this function. Also, the existing option to animate a white line to a secondary set of coordinates can now be optionally colored according to another selectable variable. The color of the line is white if the variable is zero and rainbow-brushed from blue to red otherwise. (The reason for assigning white to a value of zero is to support Boolean values, so white is off, i.e., 0, and red is on, i.e., 1.)
  - Two or more instances of Sliver can now run in a common Count-Based Animation. A single Sliver DataPlayer window will serve to control all instances, so it is possible, for example, to animate multiple paths in Google Earth simultaneously with their own icons. The animation is based on row number, so you will want to make sure the data used in all Sliver instances starts at the same time of day, for example, and have the same data rate. Note that the same data file can be read into multiple Sliver instances and different variables selected for the animation.
  - The maximum selectable number of lines to fade in Count-Based Animation is increased from 10 to 20. If performance lags, reduce this number.
- Updates certain DataTools functions to handle leading zeros in the data more consistently and to accept file paths that include a dot "." (previously the output file would be saved to the next higher folder).
- Restrictions on performing certain functions after zooming or panning have now been removed, although for a few functions the display will automatically revert to its original layout.
- Performance for large data files has been improved in various ways.

Version 1.3 (released October 25, 2016)

- Adds the menu option Plots→New Google Earth Connections Plot to plot a line for each row of data between a selected pair of geo-coordinate variable sets (lat/lon/alt for point 1, lat/lon/alt for point 2). Settings for the plotted lines are taken from the Google Earth settings under the Options menu just as for a Google Earth path plot.
- Adds a Brushes menu function to select visible lines by a range of variable values—no more zooming in while displaying values on mouse-over to select lines based on variable values. First, select a variable axis by pressing "Alt" while clicking on the axis name at the bottom, which highlights the name, and then select the menu option Brushes→Enter Selection Range for an Axis to bring up a window to enter the minimum and maximum values to use to select lines. To choose lines in different ranges, save each set of lines to a color brush as they are selected by the individual ranges. It's easy to create custom range brushing this way.
- Adds the menu option Axis→Stagger/Unstagger Names for PC Plots that have many axes or long axis names. Staggering the names will alternately shift axis names up and down so that overlapping names will be easier to read.
- Increases the maximum number of automatic range brush colors from 8 to 16 (this can be changed in the Plot Settings menu option), plus improves the hues of the previous range colors.
- Resizes points automatically after zooming a scatterplot window—no more having to press "S" after zooming to un-stretch points, at least in most all cases.

- Adds the ability to choose every nth row to plot in Google Earth. This option is displayed in the Google Path Plot window where the variables to plot are selected, not in Google Earth settings, and this value is not saved between sessions. Plotting fewer points in Google Earth is faster and provides greater transparency of the supports. Also, some plots from the data may require more or fewer points than other plots. And when plotting more than about 20,000 points Google Earth often does not plot any points at all.
- Adds a Brushes→Select All Visible Lines menu function—no more swiping all the lines to select them all.
- Fixes an issue where the previous Brushes→Toggle Selection menu option also selected any hidden lines, which caused problems displaying them after they were assigned to another brush. The menu option is now titled Brushes→Toggle Selection for Visible Lines.
- The DataTools menu now supports the new *Sort Rows by Column Value* function in DataTools 1.2.4.
- When selecting variables while reading in a file, previous variable selections in the same session are now automatically pre-selected more consistently than before. However, if the new file contains a different number of variables, all variables are automatically deselected to avoid inadvertent variable selections.
- The units for altitude, latitude and longitude in the Google Earth settings (feet/meters and radians/degrees) are now saved between sessions. The multipliers and offsets are reset to zero.
- Updates the user controls in the Data Player window used in Count-Based Animation to be more intuitive, and substantially decreases freezing in Google Earth during animation (if the icon does stop moving along the path, right-click on the *Position* label in the left pane of Google Earth and select *Refresh*.
- Turns on by default the setting to display values along an axis while swiping. This is a very useful feature, but if desired it can be disabled again in the Options→Plot Settings menu function.
- Fixes an issue where the displayed fraction for selected lines in the PC Plot was shifted when an axis was deleted.
- Section 2 of this User's Manual (*Data Visualizations in Sliver*) has been updated with more specific menu selections to serve as something of a tutorial.

Version 1.2.2 (released July 3, 2016)

- When selecting variables to load from an input file, users can now save/load the selections to/from a file. The selections file format is identical to that of the standalone DataTools application, so files can be shared.
- The DataTools menu is updated to the latest version 1.2.3 of the standalone DataTools application.

Version 1.2.1 (released May 16, 2016)

- New functions to manipulate CSV files are added to the DataTools menu.

Version 1.2 (released April 19, 2015)

- Data values in time formats of hh:mm:ss and mm:ss are now allowed as well as decimal numbers. Here hh and mm must be positive integers of one or two digits, and ss can be either an integer or a decimal number. The two formats can be intermixed. Sliver automatically detects this time format from the colon (:) character and displays that variable data as hh:mm:ss in all plots.
- Lines can be shifted upward or downward on an axis relative to the either the top or bottom of the axis. Lines can be shifted off the ends of an axis or even off the top or bottom of the window. This provides a way to spread lines of interest to the full length of the axis. This is particularly useful when extreme data has caused other lines of interest to be compressed at the opposite end of an axis. Rather than having to delete the input rows containing the extreme data, this feature can shift the lines for the extreme data off the end of an axis at the top or bottom to spread out the lines of interest. (See page 15)

- Range brushing a selection of lines now retains the color brushing of lines that are not part of the selection, and any brush that is hidden is not range-brushed and remains hidden. As before, range brushing an entire selected axis removes all existing brushes and assigns the lines on the axis to new color brushes based on the selected range palette. Lines that have been shifted off the axis are unbrushed (uncolored) when range brushing an entire axis, but are brushed if they are included in a range brush selection.
- Previously, lines added to a hidden brush were not hidden until the brush was shown and hidden again, but they are immediately hidden now. Also, when axis values are shown while swiping, the values did not previously account for an axis being inverted.
- The format of saved session files has been included as Appendix C. With multiple data files it may prove faster to programmatically write directly to a session file to assign colors and axis ranges to the data. Also, the last rows of this file are the axis ranges, so a saved session file can be edited in a text editor to set the axis ranges to fixed values, perhaps all the same values, and then restored into Sliver. The alternative is to edit the data itself or use the DataTools→Add Rows of Min/Max Values to set all axes to the same range.

Version 1.1 (released Jan. 3, 2015)

- Initial public release.

1. Introduction

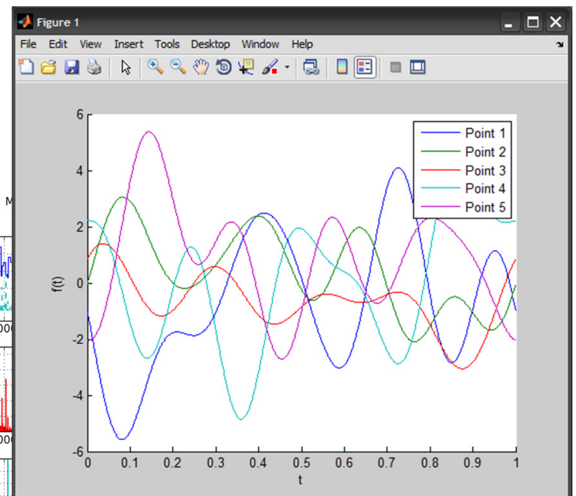
Sliver is a software application for data visualization and analysis using methods of modern exploratory data analysis and data mining (*siftware*). It incorporates parallel coordinate plots, 2D and 3D scatterplots, and Google Earth path plots, all fully linked through user-defined color brushing and data animation. These highly visual displays reveal correlations, patterns, trends and anomalies in multi-dimensional (multivariate) systems, and these in turn can significantly aid in the diagnostic analysis of sensitivities and error sources in a system.

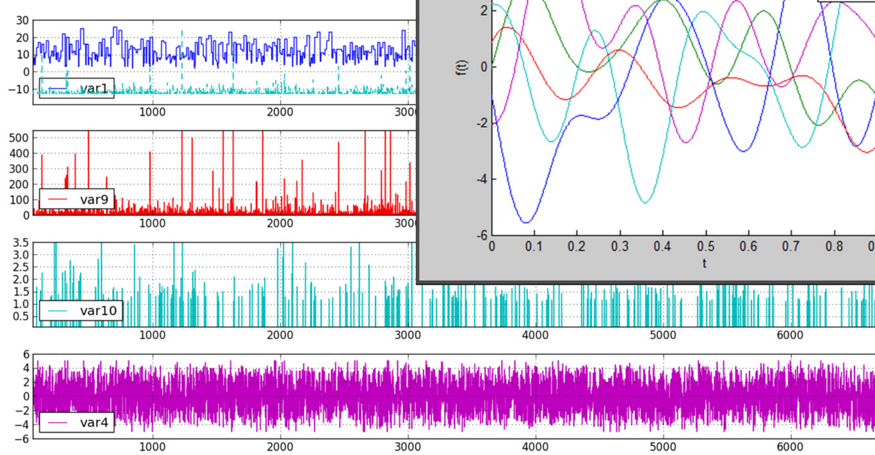
Data can be input from a comma-separated value (CSV) file or a tab-delimited text (TXT) file in either decimal/time or hexadecimal format with a header row for variable (column) names. The decimal/time format includes data in hh:mm:ss and mm:ss format as well as decimal values. Decimation (reading every *n*th row) is supported, and a set of data tools is provided to manipulate the data before being input.

Sliver is packaged as a set of files contained in a single zipped file. There is no installation required—simply extract the files into a local folder on a Windows PC and double-click to launch the *Sliver_v1.5.exe* file.

A Companion Tool to Traditional Plotting Programs

Traditional plotting programs offer 2D or 3D plots, capturing the dependency among two or three variables. Relationships among more than three variables are represented as families of individual curves or sets of plots. Understanding these relationships is difficult for four or five variables and probably impossible for 10 or 20 variables.





Consider the plots on the right of several variables vs. the data sample (or with time if data is measured linearly with time). When an event of interest occurs, these plots can be zoomed in to show the corresponding values of the variables at that time, and this may or may not be sufficient to identify the cause of an event.

However, it is difficult with this type of software to

1. Correlate several variables against time across wide time spans with many plotted values and transitions. This is important, for example, in verifying that the conditions that cause the event *only* occur at the times of the problems.

2. Correlate variables against other variables independent of time. At best, two or three variables can be plotted against each other in a scatterplot.

3. Visually identify anomalous system behavior. This capability would be useful in identifying undiagnosed software bugs that have not yet been observed in the system.

4. Study occurrences not tied to time tags (such as times in video) that require searching for multivariate conditions.

The suite of tools provided by Sliver assists in these types of analyses.

2. Data Visualizations in Sliver (a very brief tutorial)

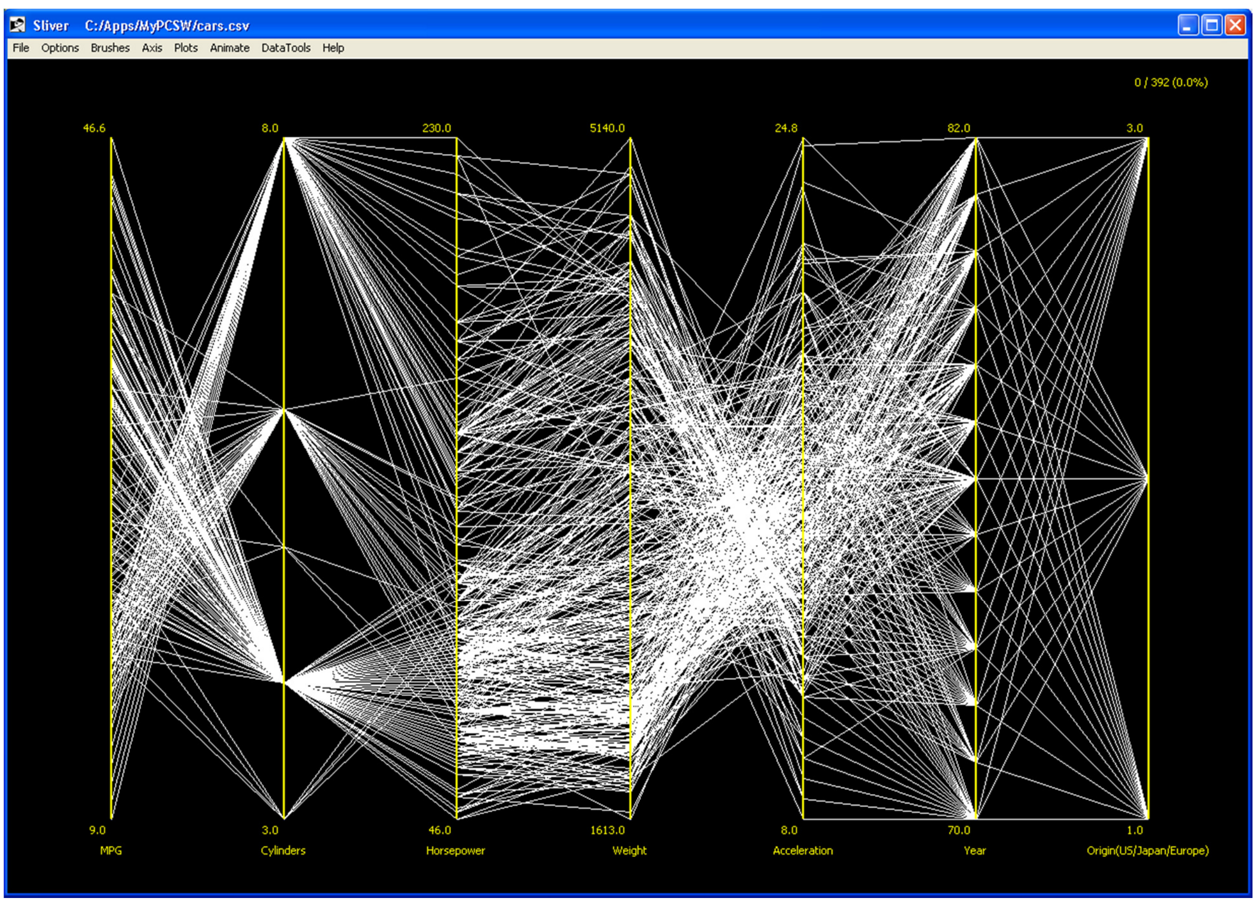
Sliver offers several forms of data presentation, all connected statically by color brushing of data ranges of interest and dynamically by animation of the data set. **Select the File→Read File→All Variables menu option and open the *cars.csv* dataset included in the Sample Data folder to follow along with this brief tutorial.**

We will first describe the types of displays available, with minimal attention to how they are launched and configured. Later we will cover the settings, menu options and hotkey assignments available for exercising the plots. **NOTE**: Some of the screen captures show an older menu layout without the new Selections and Analysis menus, but the text is correct on their use.

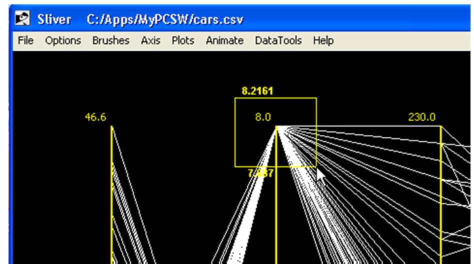
Parallel Coordinate Plot (PC Plot)

In Sliver all input data is first plotted as a parallel coordinate plot, or **PC Plot**, with all other displays created as offshoots of this main data display. The parallel coordinate plot window can be resized, zoomed and panned.

In a parallel coordinate plot, the axes of values for the variables lie parallel to each other, typically running vertically and with a linear mapping from the minimum to maximum value of each variable. Each instance of measurement of the variables (i.e., each row of the input data file) is represented by a segmented line, or polyline, that passes through the corresponding value on each variable axis. Below is a standard example shown in Sliver of data for 392 car models introduced between 1970 and 1982 (the *cars.csv* dataset). Each of the variables (MPG, Cylinders, Horsepower, Weight, Acceleration (really seconds to 60 mph), Year, and Origin (US/Japan/Europe) is assigned to an axis with values running linearly from its minimum to maximum values.

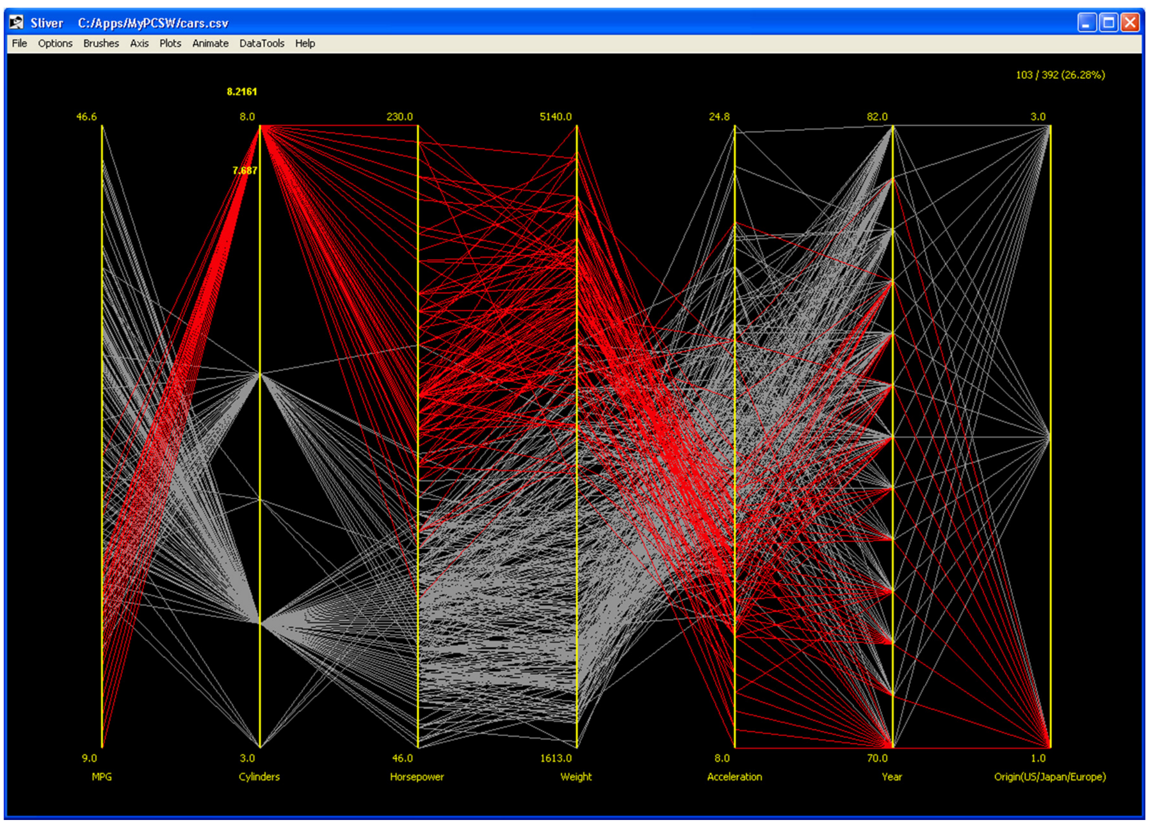


What can we tell right away? We can see that there were 3, 4, 5, 6 and 8 cylinder models introduced that generally had an inverse (criss-cross) relationship with MPG values ranging from 9.0 to 46.6 mpg, a correlation between number of cylinders and horsepower (although the 3-cylinder models did surprisingly well) and between horsepower and weight (with that one anomalous line angling downward), and a strongly inverse relationship between weight and time to 60 mph (the variable Acceleration).



To deduce more we need to isolate data. For example, we can color all the data related to 8-cylinder models by swiping the mouse to create a rectangle surrounding those lines, as shown on the right. Sliver then selects and colors those lines with the selection color (**red here but purple by default**) as shown below. We can also modify the selection by swiping other lines and combining them with AND (with the shift key pressed), OR (with the ctrl key pressed) or XOR (with both keys pressed) operations. (**NOTE**: if swiping is slow to react, try toggling the ALT key).

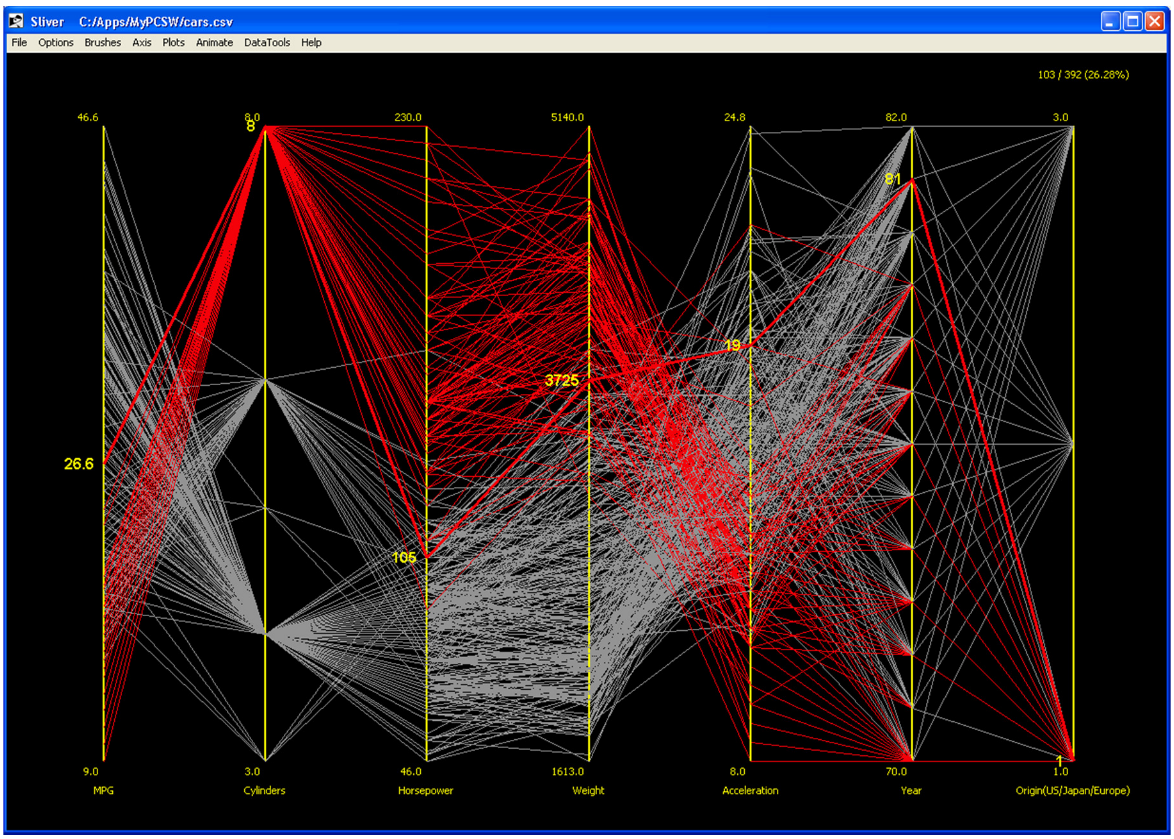
The small range values at the upper and lower ends of the selection box in the small display on the right are shown by default, but can be turned off using the **Options→Plot Settings** menu option. They can be very useful for showing ranges while swiping.



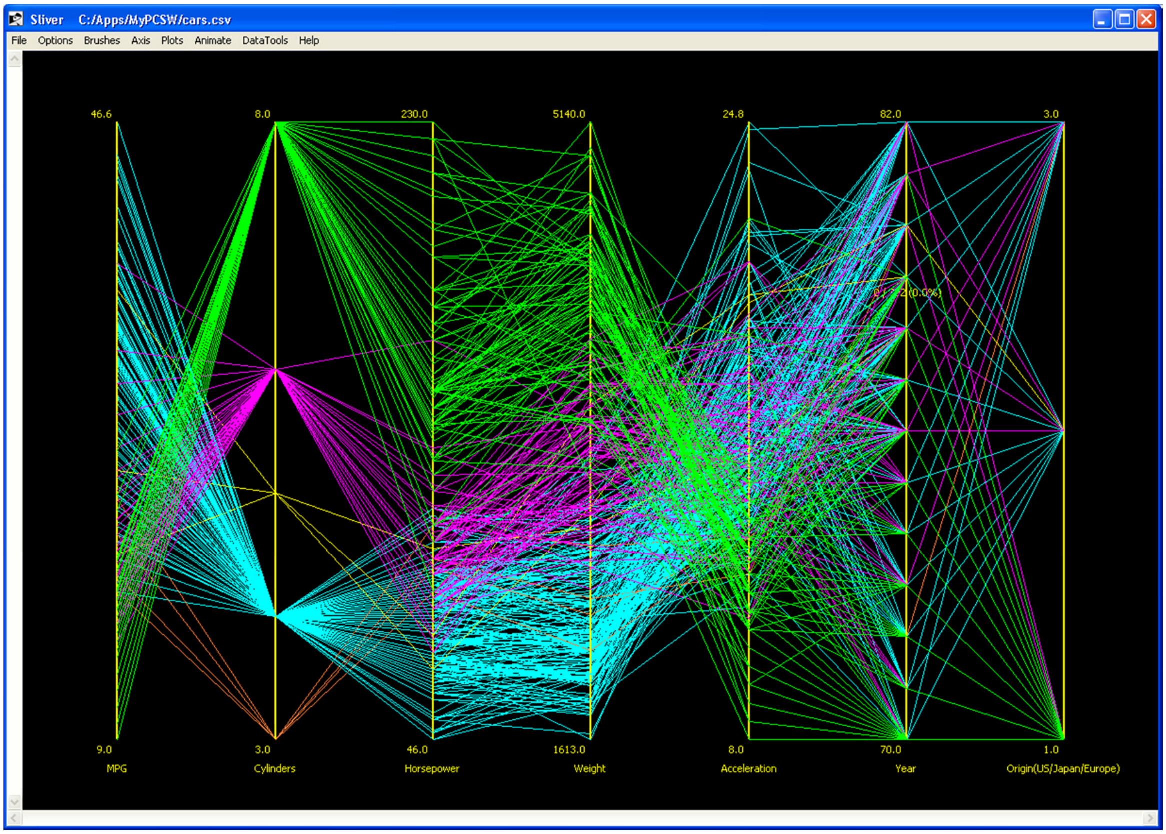
We now can see that all the 8-cylinder models were made in the US, with none made in 1980 and 1982 and only one made in 1981. The ranges of the variables in the selection can also be seen.

For more detail we can turn on a menu option to highlight the variable row values when the mouse hovers over a line (**Plots→Display Values on Mouse-Over→Without String Mapping**). An example is shown below, where the values for the 8-cylinder car with the best MPG are displayed. We can see that this 1981 model had the second-lowest horsepower (105 HP) and second-worst acceleration (19 seconds to 60 mph). Repeat the menu selection or press ESC once or twice to no longer show axis values when the mouse hovers over a line.

Sliver does not support strings, only numbers. However, the DataTools menu provides a function to replace strings in variables with unique numbers in a file to allow it to be read into Sliver (**DataTools→Replace Strings with Numbers**). This function also creates a text file listing the mapping between strings and assigned numbers. If the path to this string mapping file is entered in the **Options→Plot Settings** window, or the path to any file created in this format (see Appendix D), the menu option **Plots→Display Values on Mouse-Over→With String Mapping** can be selected. Any row numbers matched to a variable string will then display that string rather than the numerical value for that variable in the main PC Plot window.

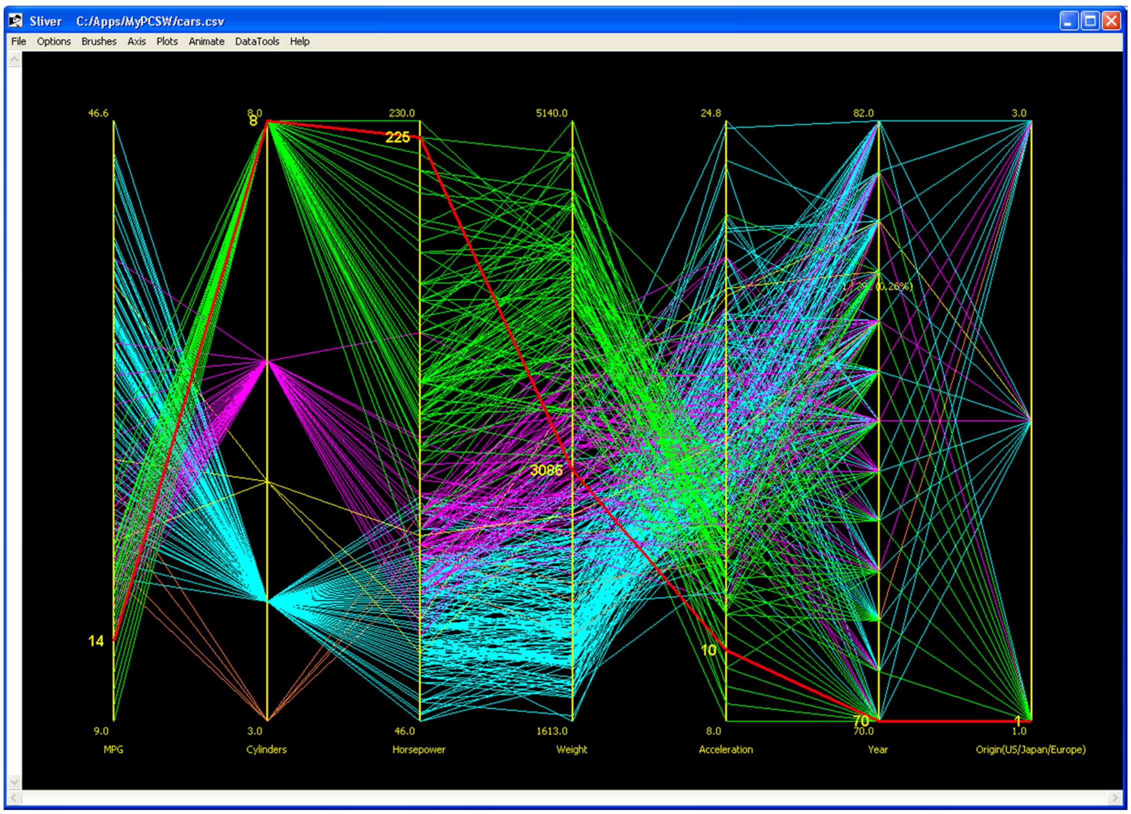


We can assign the selection to a more permanent color "brush" with the **Brushes→Assign Selection to New Brush** menu option—here we choose a particular shade of green. And we can color other selections, here selecting each cylinder number and color brushing them independently by calling the same menu option after each selection as seen below.



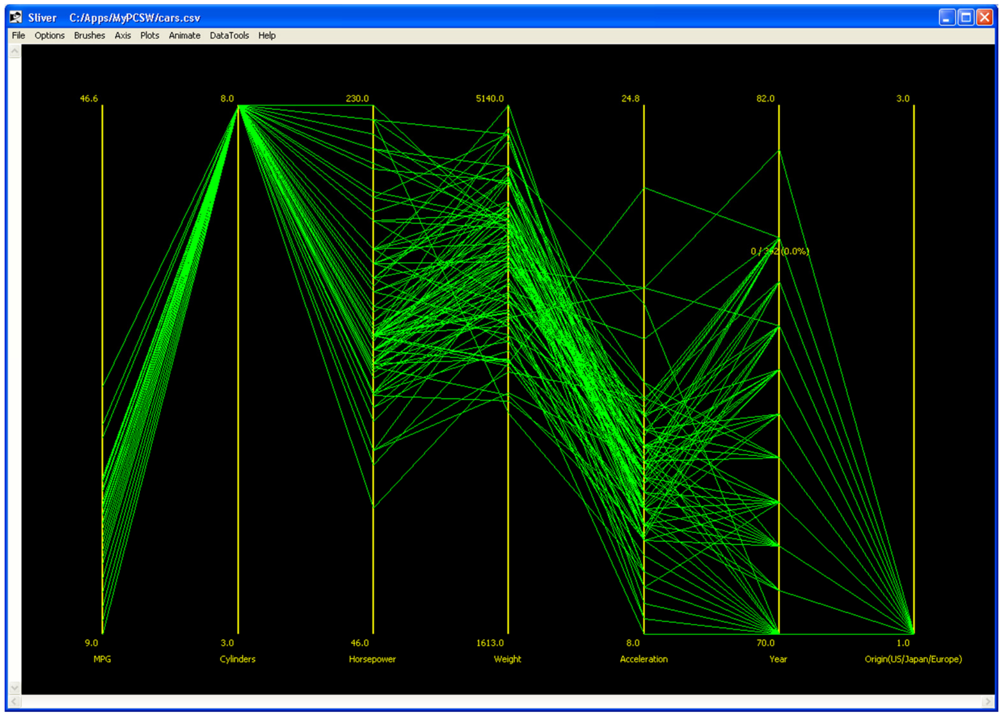
We can study other aspects of the data by selecting and coloring different sets of lines. In fact, let's look at the anomalous line we saw in the overall plot before, the diagonal line running from near the top of the Horsepower axis to the middle of the Weight axis.

We swipe that line to highlight it as a selection (we can swipe a line between axes) and we again turn on the **Plots→Display Values on Mouse-Over→Without String Mapping** menu option to show the values at each axis. We have the display below, where we see a 225 HP car from 1970 with a radically low weight of 3086 lbs. and consequently a 10 second to 60 mph rating, but it comes in at a disappointing 14 MPG. This type of anomaly, often noticed on the fly, can identify unsuspected software errors or other anomalous behavior in a system.



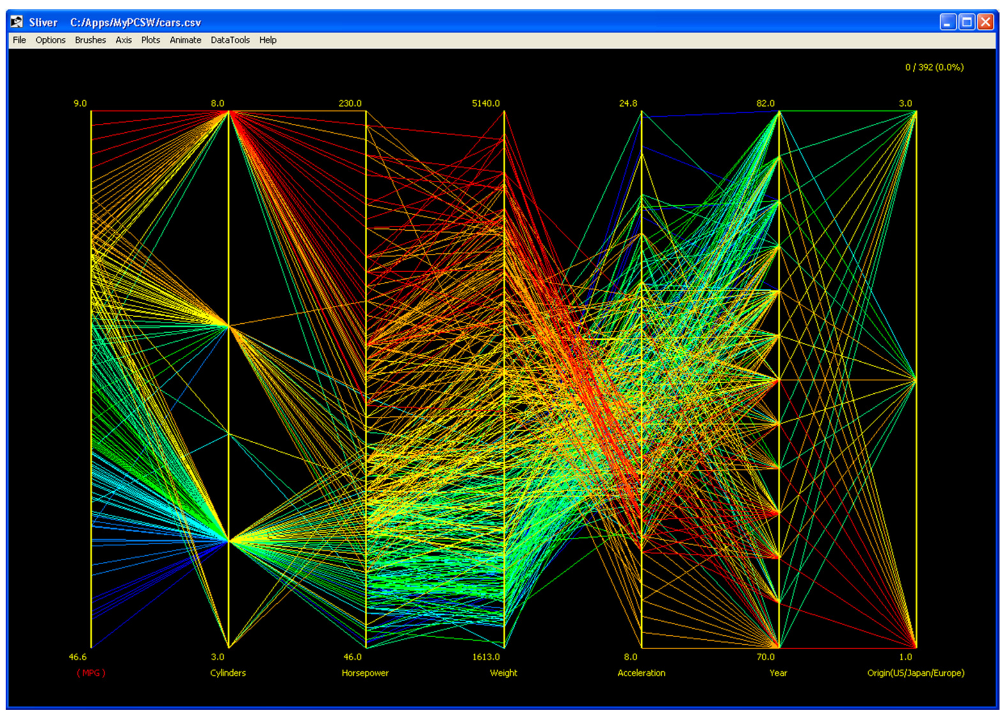
We can also easily choose our own ranges along an axis in which to apply colors. After selecting an axis, choosing the **Selections→Enter Selection Range for an Axis** menu option will pop up a window to enter minimum and maximum values, and lines that pass through this range on the axis are then automatically selected. These lines can be color brushed with the **Brushes→Assign Selection to New Brush** or the **Brushes→Add Selection to Existing Brush** menu option, and more sets of lines can be colored by selecting different ranges and color brushing them as desired.

Other menu options for selecting lines are **Selections→Select All Visible Lines** (also invoked with **Ctrl-A**), **Selections→Select All Unbrushed Lines**, and **Selections→Toggle Selection for Visible Lines**.



We can deduce more of the relationships, but the overlapping lines are a problem. This is called *overplotting*, and there are ways to reduce this effect. For example, we can choose to hide any color with the **Brushes→ Show/Hide/Recolor Brushes** menu option, and here we hide all color brushes except the green one, as shown on the right. We can then select and color subsets of these lines. This selection and trimming of data is very important in isolating data of interest for study.

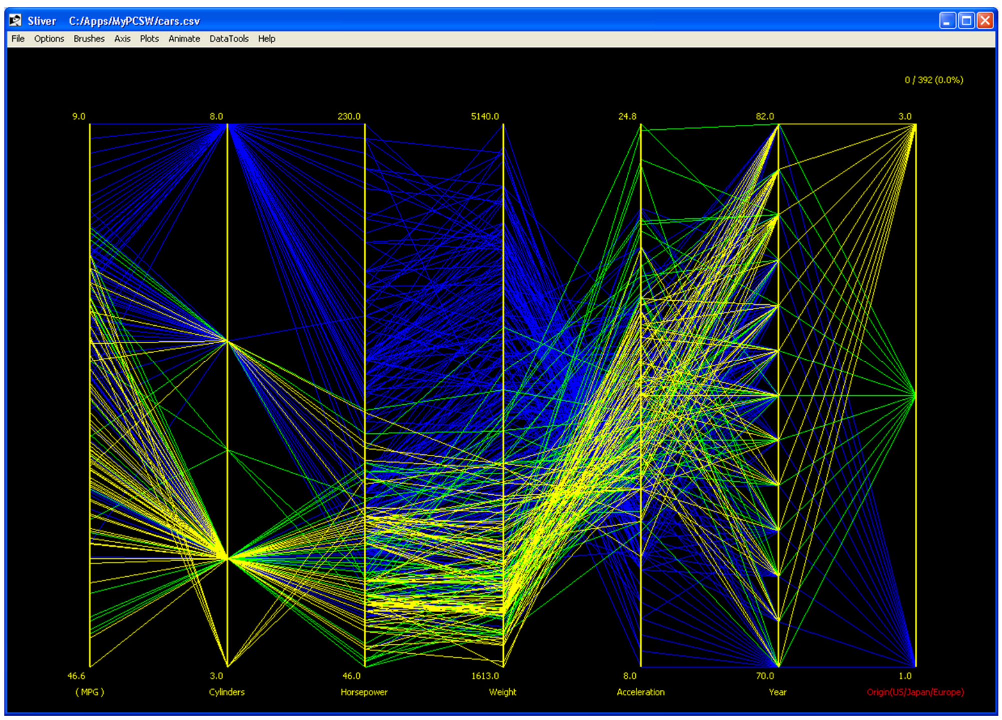
Often one variable or a combination of variables represents a measurement of error, so we might select and color-code the ranges of error. There is a menu option to *range brush* an entire axis or just a selected set of lines along an axis through a set number of up to 16 spectral colors. The default is 16 colors, but this can be changed with the **Options→Plot Settings** menu option.



Let's say we want to range-brush the entire MPG axis. First we make sure we have not selected any lines, or the range brush will span only those lines—we can do that by swiping anywhere in the display where there is no line. Then we can select the MPG axis by pressing the Alt key and clicking on the axis name, which colors the selected name. Then we press the *r* key or select the **Axis→Range Brush** menu option to automatically range brush the entire axis in a spectrum color from blue to red upward. However, we want the worst MPG to be the red end, and that's at the bottom of the axis right now, so we first invert the axis by pressing the *i* key (or using the **Axis→Invert** menu option) after the axis is selected. The axis is inverted and parentheses are added around the axis name to denote that it has a high-to-low orientation. We then perform the range brush operation and we end up with the display here. The blue lines are drawn first so they lie underneath the more red colors, but we can see correlations here. We can invert the axis again if we wish and the red end will lie at the bottom of the axis.

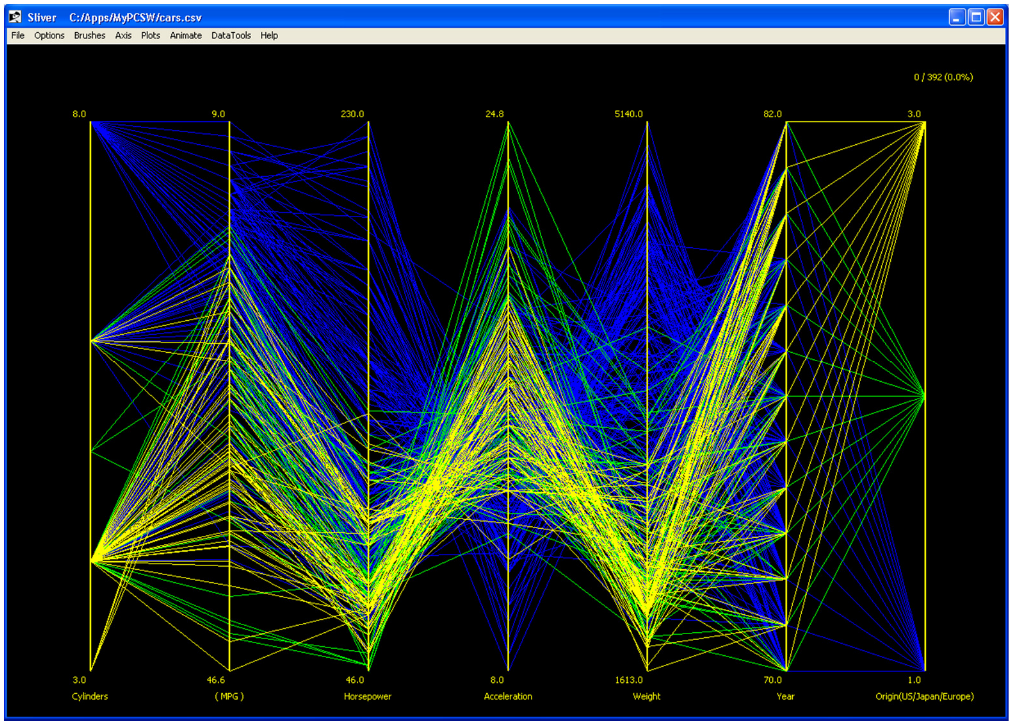
Selecting an axis and range brushing it removes all other brushes and makes all lines visible, applying the range colors to all lines that have not been shifted off the ends of the axis (see page 18 on shifting lines). However, if any lines are selected, then only the selected, visible lines are range brushed, and the colors span the selection rather than the entire axis. Unselected lines retain their visibility and their color brushes, so to range brush lines along an axis while leaving hidden lines invisible, simply select all the visible lines and then range brush the axis. If any range brush color matches a color of an unselected line the brushes are simply merged.

Another convenient use of range brushing selections rather than an entire axis is when, say, we want the value of zero on an axis to be blue but positive and negative extremes to be red. To do this, we can select the range from zero to the positive extreme (either by swiping or, to get exactly zero at the low end, by using the **Selections→Enter Selection Range for an Axis** menu option). We range brush this selection. Then we invert the axis with the **Axis→Invert** menu option, select the range from zero to the negative extreme (now at the top) by swiping or using the menu option above, and range brush this selection. Then we can invert the axis again and we end up with a color of blue at the value of zero stretching to red at both extremes.

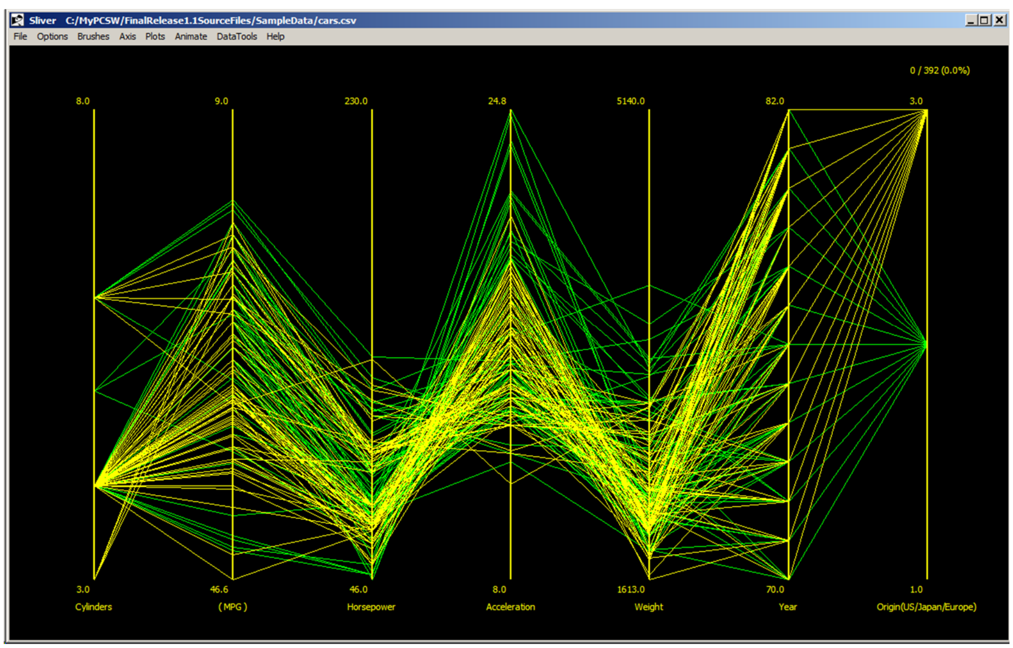


Perhaps we want to color the data according the major groupings of values of a variable. One situation that could benefit from this is a categorical variable, or in other words, a variable that has discrete values. Range brushing may not uniquely color the different values with different colors, so there is a *gap brush* option that will color lines on an axis based on the largest separations between them. Here we choose the Origin axis by clicking on the axis name while pressing the Alt key, and then we select the **Axis→Gap Brush** option, ending up with the display here. Up to 20 colors, based on gap sizes, can be applied using the up/down arrow keys to increase/decrease the colors (try it on a different axis). Here we could have individually selected and colored the lines for the three values, but the automatic option is much faster, and there are times when an axes has gaps that are not discrete but are significant. (We could have gap-brushed another categorical axis, the Year axis, but the data is not well-correlated to the year, and a cluttered display results.)

We can also re-order the axes to show more relationships between the variables. In this case, the relationship between the horsepower and weight, other than the anomalous case we observed, does not tell us much, and the relationship between horsepower and acceleration would be clearer if they were next to one another. Also, categorical variables squeeze polylines through just a few points, so it can mask relationships among other axes. For this reason, it's usually best to move categorical axes to the ends.

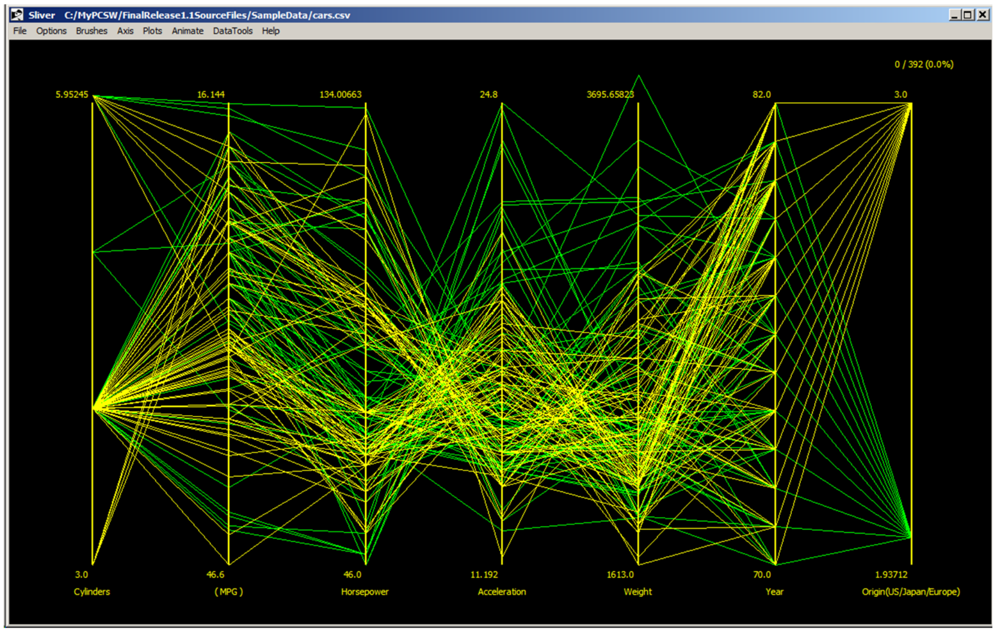


We can move an axis by selecting it (pressing Alt and clicking the name) and either pressing the *m* key or selecting the **Axis→Move** menu option. Drag the arrow to the space between other axes or at the end where you want the axis to move. Moving the Acceleration axis next to the Horsepower axis reveals a strong inverse correlation between the two variables. Moving the Cylinders axis to the left end of the display reveals a strong correlation between mileage and horsepower independent of the number of cylinders, which was not very apparent before. The resulting display is shown on the right. Note that the number of re-orderings to get all the variables as neighbors of the others is surprisingly small: the integer value of (n+1)/2 for n axes.

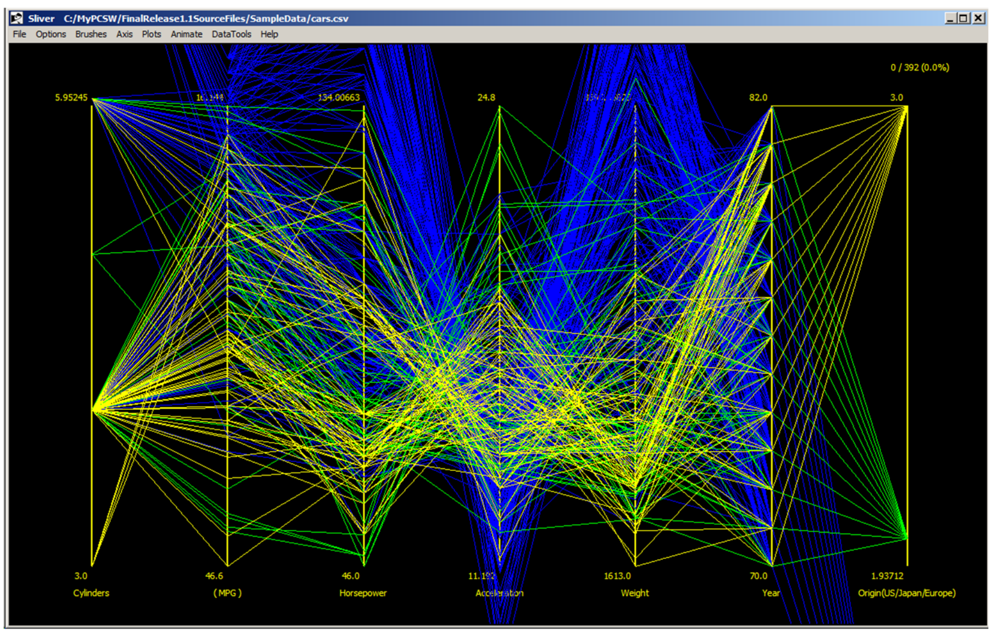


Now say we are only interested in studying the cars made in Japan and Europe, so we hide the blue brush using the **Brushes→ Show/Hide/Recolor Brushes** menu option, yielding the display on the right. It would be easier to glean behaviors if the visible lines were spread across the full spans of the axes. There are three ways to spread the lines vertically, either by saving and reloading the visible set of data to automatically change the ranges, or by shifting the lines in the display along the axes, or by selecting the **Analysis→Enter Range for Axes** menu option (not shown in these screenshots).

The first method can be done by selecting the **File→Save Visible** menu option to save the visible subset of the data. Opening that file will automatically re-scale the axes to the ranges of the saved data. However, this is not a very interactive way to visualize subsets of the data.



The second method is to retain all the data but shift the ranges of the axes to spread the visible lines along the axes of interest. We can select any axis and use the **Axis** menu options or arrow keys to shift the lines at this axis up and down from either the top or the bottom of the axis (Up/Down arrow for shifting up/down at the top, Left/Right for shifting up/down at the bottom). The ranges shown at the ends of the axes are automatically updated. The steps are discrete, so it's likely we can't get the lines to extend exactly to the end of the axes, but if we do this for all the axes we can end up with the display on the right.

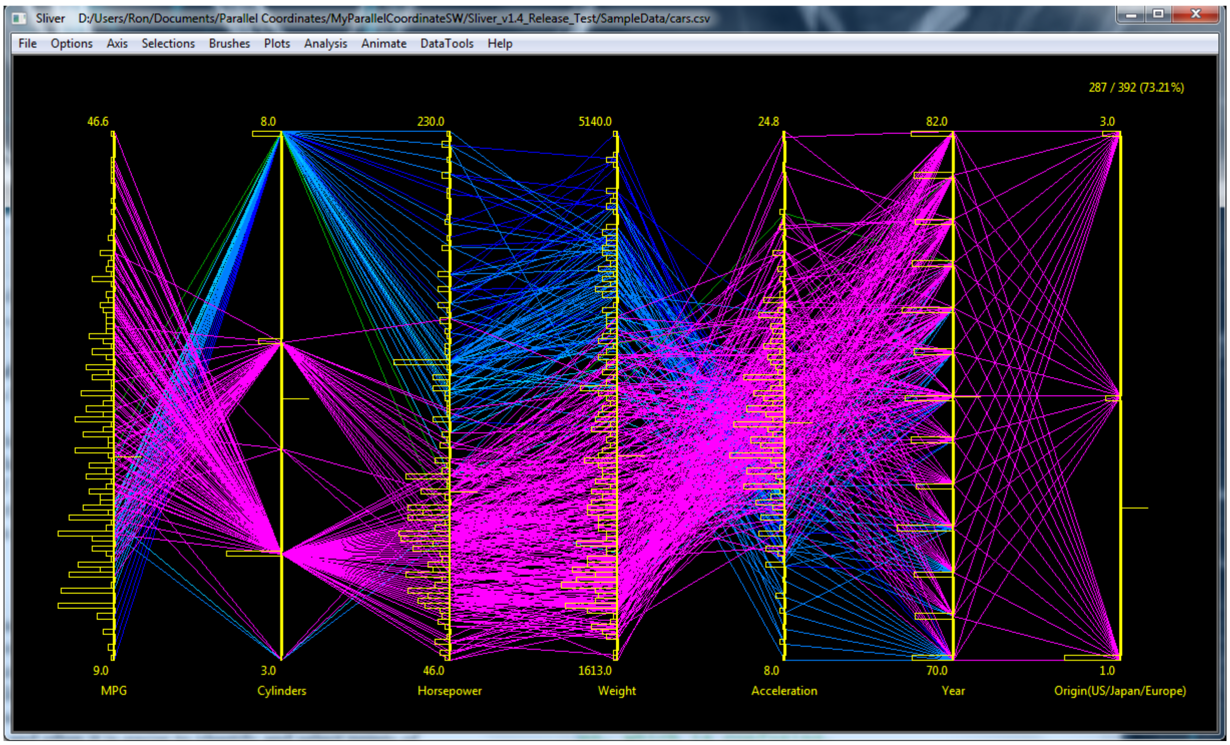


The hidden lines are still there, but shifted, as seen in the display on the right where the blue brush is again made visible through the **Brushes→ Show/Hide/Recolor Brushes** menu option. The blue lines could have been visible while the shifting was occurring. Shifting of lines can also mitigate a situation where a few extreme data values for a variable compresses the remainder of the data at the other end of its axis. Scatterplots also display any shifts when they are created, but later shifts are not dynamically updated—you have to create a new scatterplot to reflect any new shifting. In all cases, exports of the plots display any shifting that has been applied.

The third method of shifting axis ranges is to invoke the **Analysis→Enter Range for Axes** menu option, which pops up a window to select the axes to shift and then a window to manually enter the range values for these axes. Other menu functions relevant to setting axis ranges are **Analysis→Center Axes on Means**, **Analysis→Center**

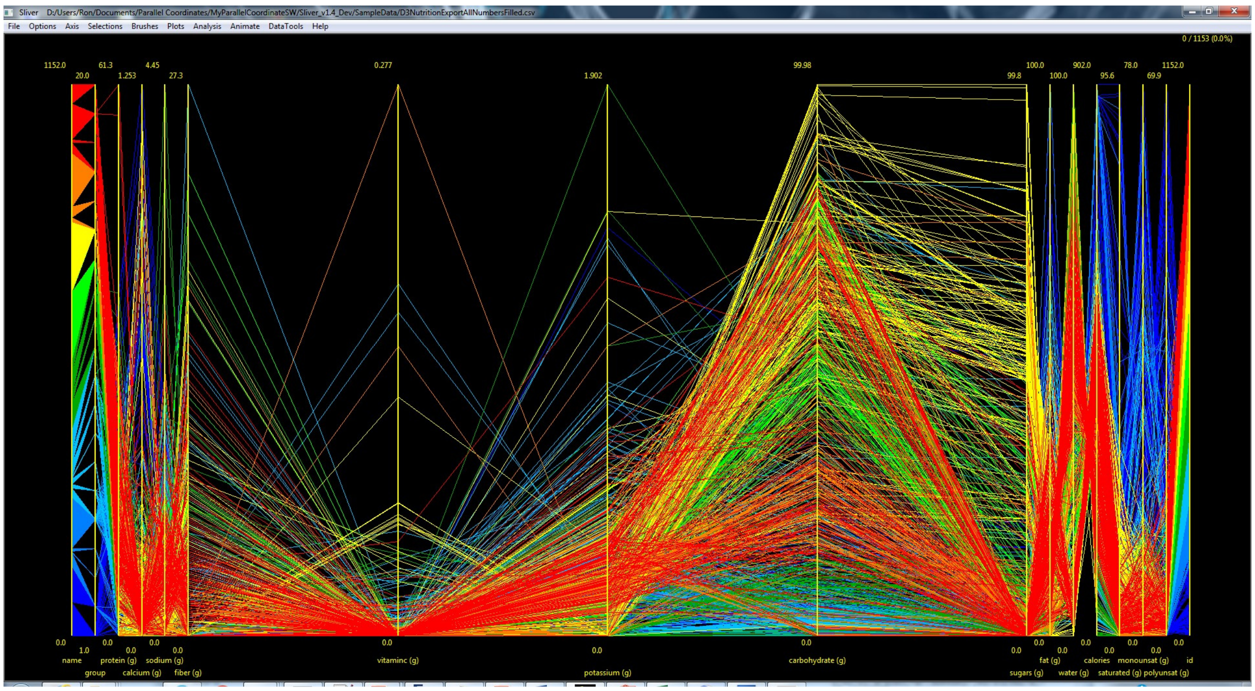
Axes on a Selected Line (useful, for example, for comparing the performance of other runners across splits with a particular runner like you!) and **Analysis→Reset to Full Axes Ranges**.

The Analysis menu provides other functions related to selections and display. The **Analysis→Show/Hide Histograms and Means** draws a user-selectable number of histogram bins along the left side of each selected axis, where the height of each bin is proportional to the number of lines intersecting the axis within the span of that bin. The mean of each axis is marked by a horizontal line on the right side of the axis. This is demonstrated in the figure below, where some lines have been swiped to better show the yellow markings. This is very useful for interpreting the clusters of data in the PC Plot. Only visible lines that intersect an axis are considered, so shifting lines off an axis removes them from calculations of the histograms and means. However, **Analysis→Center Axes on Means** resets the axes ranges so that all visible lines lie along them before centering on their means. Finally, **Analysis→Brush Statistics** shows the statistics for any selected lines and for each existing brush.



For easier viewing of PC Plot windows consisting of many variables, Sliver offers the options to vertically stagger the names and range values to avoid overlapping, and to horizontally compress or expand the spacing between two selected axes. Each are reversible, so different groups of axes can be spread out to help analyze their relationships.

The figure below shows a PC Plot in which the names and range values been staggered vertically via the **Axis→Stagger/Unstagger Names** menu option and the axes spread out between the fiber and sugars axes via the **Axis→Compress/Spread Axes** menu option. Moving axes of interest together as neighbors first will increase the benefits of horizontal spreading the axes. This spreading will also be reflected in the Transparent PC Plot Window that we will discuss shortly.



And so it goes for the PC Plot window, and we have not yet come to the other types of brush-linked plots. Correlations and clustering in the data become visually apparent by re-ordering, selecting, color-brushing and hiding combinatorial sets of data. In this way we can winnow down to the data of interest, say to data that corresponds to high levels or error or to high performance. We can also save just the selected data or just the visible data to a new CSV or TXT file, and then reload the file in order to reset the ranges of the axes to the new minimum and maximum values for greater separation of those polylines or manually shift or enter new ranges. Also, note that selections can be made in any scatterplot windows that are created (as described later), and often it is easier to identify and select points of interest in these plots than lines in the main PC Plot window. All selections and color-brushing are automatically propagated among the PC Plot and all scatterplots.

The data set used for this example is small at 392 rows (the numbers of selected and total polylines are shown in the upper right of the display). Sliver easily handles up 100,000 rows of data of, say, 16 variables with reasonable processing delays. It can be used with larger data sets with somewhat longer delays—in fact it has been used to plot up to 500,000 rows of data with a little patience and limited interactivity. One tip for larger files is to choose to click corner points rather than swiping in order to select lines. This option is set in the **Options→Plot Settings** menu option—in this case the display does not need to continuously update as you swipe with the mouse. Also, with larger data sets the lines soon become an indiscriminate mass, so range-brushing of colors and hiding of polylines are critical in reducing the display to one with visual distinctions.

The PC Plot window can be maximized by pressing the rectangle in the upper right corner of the window, or stretched by dragging the corners of the window. It can also be zoomed in or out at any point by pressing Ctrl while rolling the mouse wheel or by pressing the + and – keys (no need to press Shift). In any case the mouse must be moved into the window in order to resize the plot to the new dimensions of the window. If zoomed, panning is then accomplished by clicking the scrollbars along the side and bottom, or by using the left/right/up/down arrow keys. However, if histograms and means are drawn when a zoom is performed, the scrollbars are not added to the window and you will have to zoom out and in with the mouse in different locations to move around or simply use the arrow keys.

For a convenient listing of command keys that are available for different functions in Sliver, select the **Help→Tips** menu option. This User's Manual is also available under the **Help→User's Manual** menu option. More detailed information on Sliver operation, including the windows that appear when menu functions are selected, is found in Section 3 *Sliver Detailed Operations* of this manual.

But first continue on to view and try out more Sliver data visualizations.

Parallel Coordinate Plot with Transparency

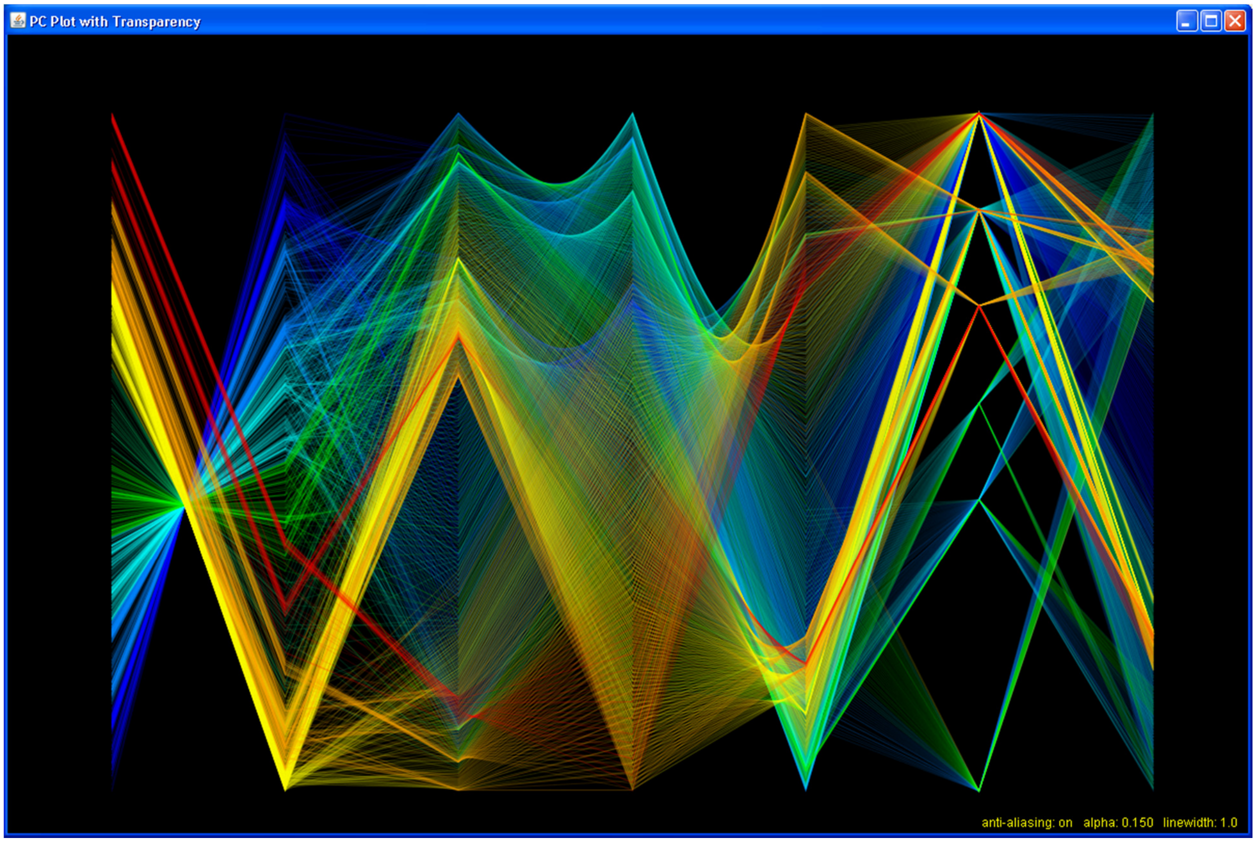
As described above, one method of interpreting the mass *overplotting* of lines in the parallel coordinate plot for large data sets is to use range brushing to color code the data and then using the other visualizations described below. Another method is to hide selected groups of lines by color brushing them and using a menu option to hide those brush colors. Also the axes can be re-scaled to spread out the lines either by shifting the lines off the ends of a selected axis using the arrow keys, or by shifting or manually setting the range of a selected axis using the Axis menu functions, or by exporting the visible lines to a new data file and loading that file.

Applying a level of pixel transparency, also called *alpha blending*, also gives an overall sense of the clustering in the data, i.e., the main arteries of the lines. The resulting plot is reminiscent of an X-ray, revealing inner structures. The colors of the lines also add when overlapping, which helps distinguish the structure even more.

Sliver does not support alpha blending in the main, interactive PC Plot window due to practical limitations on processing time. However, Sliver provides two means of viewing the parallel coordinate plot with transparency. First, a separate window of fixed size can be plotted from the Plots menu to view the PC Plot with adjustable transparency. Second, the PC Plot and any scatterplots can be exported to PostScript and then converted to PDF using Acrobat Distiller—there are many settings available in Sliver for this case, and the resulting PDF can be panned and zoomed as well.

Transparent PC Plot Window

A transparent plot can be created in a separate window as shown below by selecting the **Plots→New Transparent PC Plot** menu option. This viewing window does not dynamically update when brushing or other changes are performed on the main PC Plot. Mouse selections and other functions are not enabled in this window. It can also be slow to draw for larger datasets. The size of the window is set in the **Options→Plot Settings** menu option.



The following functions are available in this window:

1. The degree of opacity (the alpha value) can be changed by pressing numbers 0 through 9 on the keyboard, corresponding to alpha values of 0.01, 0.02, 0.05, 0.10, 0.15, 0.20, 0.25, 0.3, 0.4 and 0.5. Lower alpha values correspond to greater transparency. The UP/DOWN arrows on the keyboard fine-tune the alpha value between these major increments. The current alpha value is shown along the bottom right (the example below has an alpha value of 0.15).

2. The width of the lines can be decreased/increased with the LEFT/RIGHT keyboard arrows. The current linewidth value is shown along the bottom right.

3. Anti-aliasing of the lines can be turned ON or OFF by toggling the lowercase or uppercase "A" key. This can improve the sharpness of the lines and the quality of the overall image, and it is beneficial when a screen capture is made for presenting the plot. The example here has anti-aliasing turned on, as indicated in the lower right. Note that this very significantly slows display updates for new alpha values or linewidths, particularly for larger data files, and should normally be turned off for faster performance.

4. A Help display of these functions toggles on and off in the upper right corner of the display when the lowercase or uppercase "H" key is pressed.

When color brushing or other changes are made in the main PC Plot window, a new Transparent PC Plot can be created from the Plots menu. This terminates the current transparent window if it exists and creates a new one with the updated changes. The current values of alpha, linewidth and anti-aliasing are saved between plots and sessions, and are automatically applied to the new plot.

This brings us to the **Analysis→K-Means Clustering** menu option, which provides automatic brushing of clusters in the data using the k-means algorithm. Choosing this menu option pops up a window for selecting which variables to include when identifying clusters. It also allows the number *k* of clusters in which to partition the data, as well as the number of those clusters to brush in rainbow colors. The most significant clusters are brushed up to the total specified. The cluster having the most data points is brushed in red and the cluster with the least data points (of the number of clusters to be brushed) in blue. There is also an option to automatically update or launch the Transparent PC Window described above. This provides a means of viewing the identified clusters in the transparent view to verify that the selected number *k* of clusters and the number of clusters to color brush through the data is optimal, at which point the process can be repeated with different values. This clustering function requires that the free R statistical software be installed on the PC, with the path to its RScript.exe file selected in the **Options→Analysis Settings** menu option.

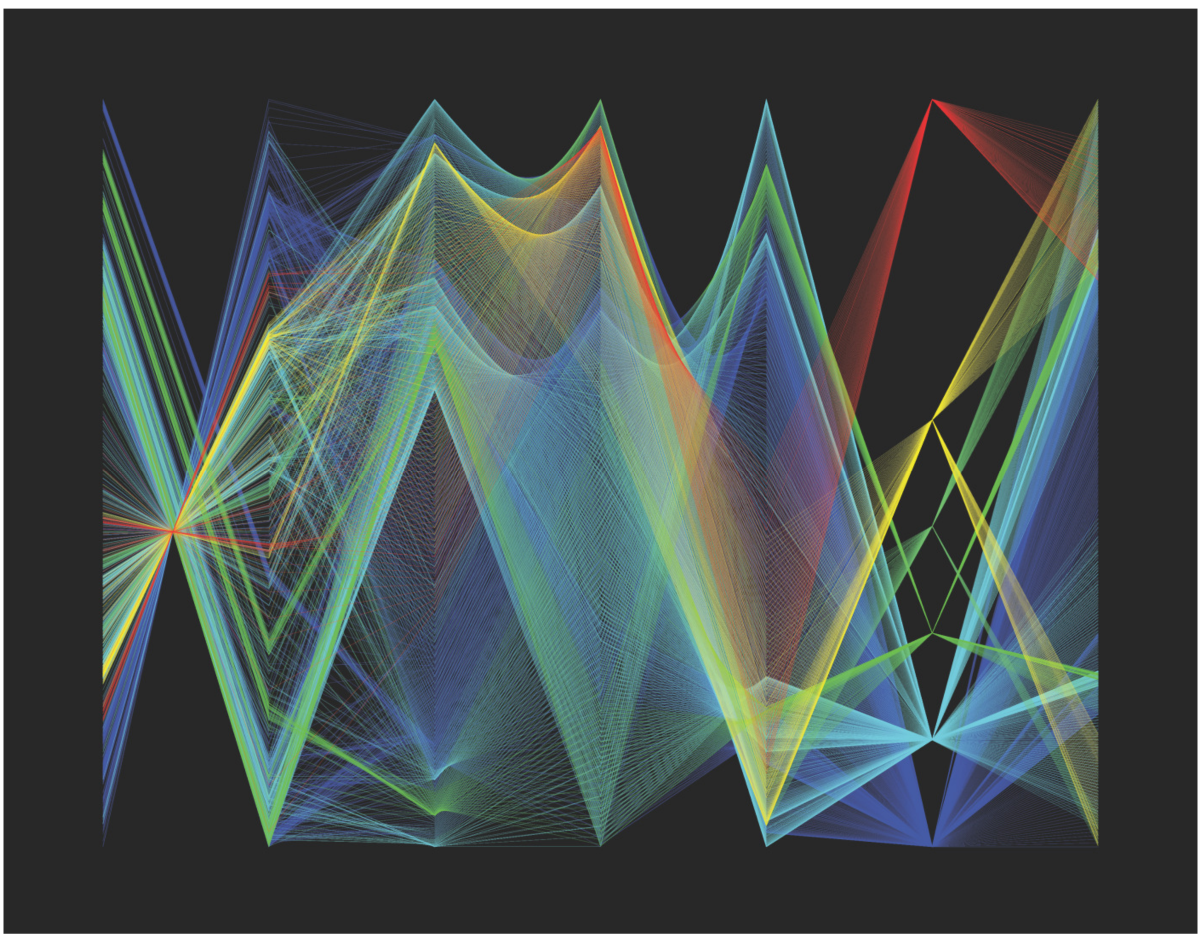
Export to PostScript and PDF

As a second option, Sliver provides a **File→Export Plots to PostScript/PDF** menu function that creates a Postscript file of the PC Plot and any scatterplots with a selectable (configurable) transparency, line width and paper size. The **Options→PostScript/PDF Export Settings** menu option configures the settings. The Adobe Acrobat Distiller application can convert this to a PDF file, and if the path to Distiller is entered into the settings the conversion will automatically occur and the PDF file will be launched for immediate viewing.

Note that there is a one-time settings change in Adobe Acrobat that must be performed to enable export with transparency. This update procedure is described on page 62.

An example of a Postscript output that was automatically converted to PDF is shown below. This plot does not appear as vivid as the screenshot of the previous transparent PC Plot window, but this depends on the alpha value and the blending mode that is selected in the PS/PDF settings. **Also, choose *Leave Color Unchanged* in the Adobe Distiller menu function of Settings→Edit Adobe PDF Settings→Color→Color Management Polices for blackest backgrounds and boldest colors.** In this case the options for including the axes and text in the output were not selected in the PS/PDF export settings, and therefore these are not shown. Acrobat automatically displays anti-aliased plots, and provides zooming, panning and much faster rendering than in the previous display window.

In general, exporting plots (including scatterplots) to PDF instead of taking screenshots is an excellent way of obtaining vector-drawn, presentation-quality graphics whether transparency is chosen or not. Because of the large number of intersections and the extensive flattening process required, printing the PDF typically requires saving it first as, say, a 600dpi PNG file.



PC Plot Matrix (PCM)

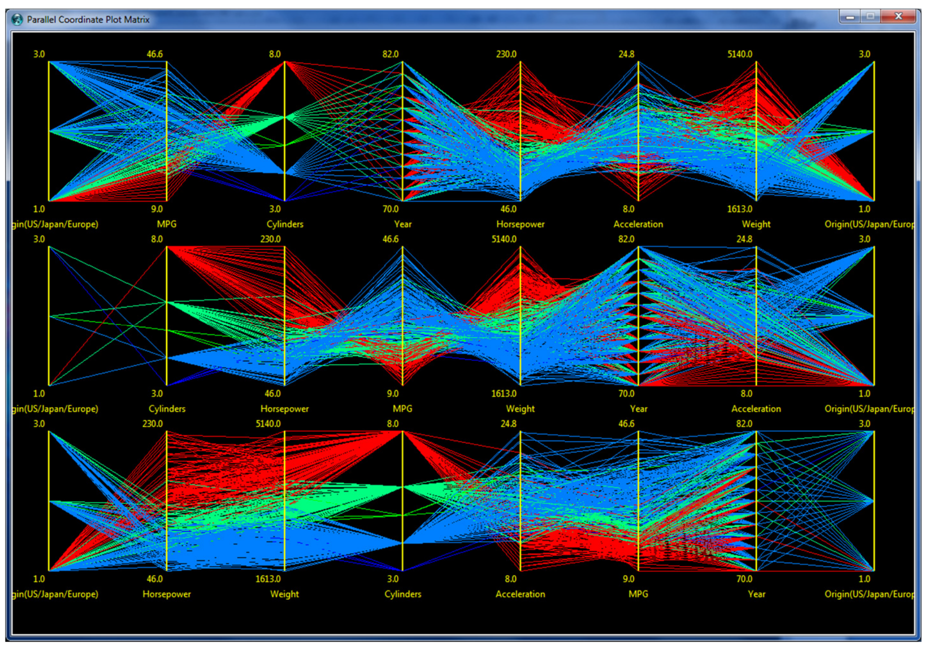
The PC Plot can only show relationships between neighboring axes. Axes can be moved to different locations to provide different orderings, but a convenient way to visualize pair-wise relationships between all variables is to create a parallel coordinates matrix, or PCM. This is similar to a scatterplot matrix, in which 2D scatterplots of all variable pairs are displayed in a matrix. Here PC plots between every pair of axes are shown. In addition, the individual plots are placed within a minimum number of rows in such a way that a continuous PC plot also occurs across all the axes in each row. Use the Plots→New PC Plot Matrix menu option to create this window. The size of the window can be configured using the Options→Plot Settings menu option, limited to just less than the size of the display.

The existing zoom methods for all windows (Ctrl plus mouse scroll or the + and – keys) works for the PC Plot Matrix as well. Again, the Alt key zooms horizontally while the Shift key zooms vertically. **Zooming may be slow for large datasets given the number of lines involved, but the zoom level is remembered between invocations.**

Shown below is an example for the cars.csv dataset. Each row represents a PC Plot across all the variables, while a neighboring pair of axes for any two variables can be found somewhere in the three rows. This provides a quick look at all the relationships in the data, and may spur you to re-arrange axes in the main PC Plot. This window is passive; selecting, brushing or hiding brushed lines cannot be initiated in this window, but any such changes in the main PC Plot are automatically applied to the PCM window. Any changes to the color scheme settings of the PC Plot window also propagate immediately to the PCM window. Axes deleted in the main PC Plot will be deleted in the PCM when the PCM is created again. Axes inverted in the PC Plot, however, will not appear inverted in the PCM. The window size is set by an option in the plot settings.

For an even number N of variables, there will be N/2 rows with an axis for each variable placed in each row. For an odd N, as in this case, there will be (N-1)/2 rows, but the axis for the last variable will be repeated at the beginning of each row in order to show all the variables neighbored pair-wise in this number of rows.

The PCM window may seem cluttered, but in fact interesting and useful relationships can be seen even when there are, say, 10 rows of plots across 20 variables. But there are ways to reduce clutter:



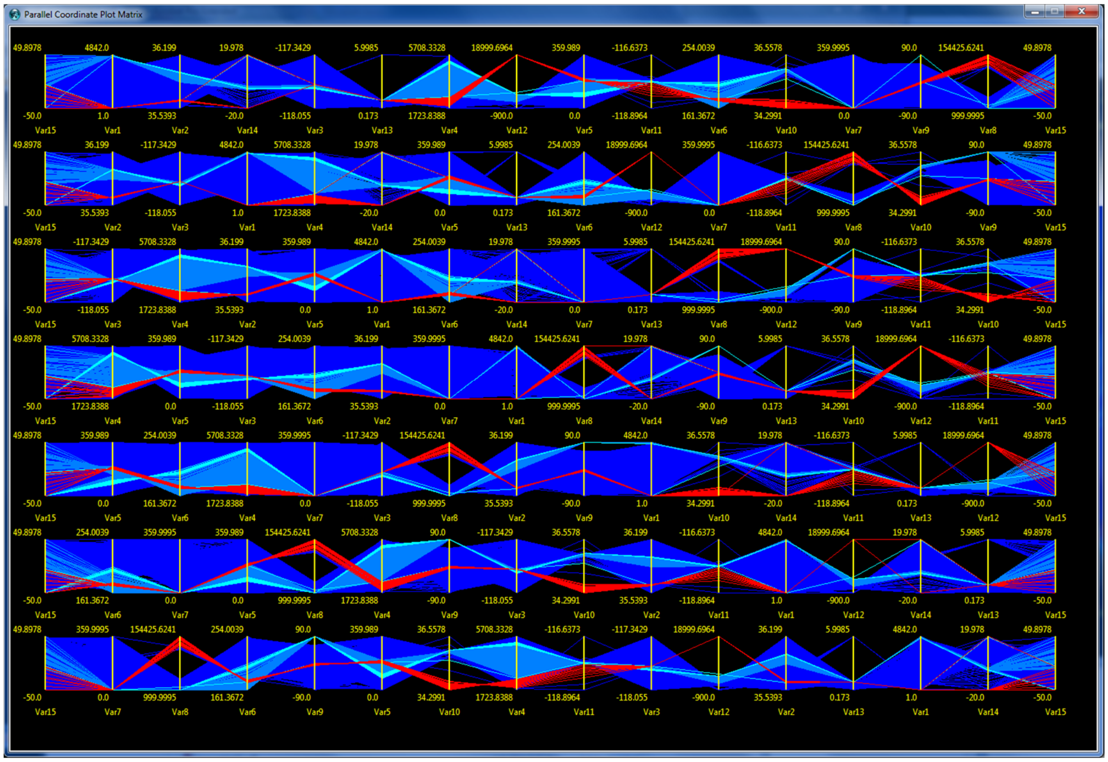
1. Different selections and brushings can be applied in the main PC Plot, and lines of any brush color can also be hidden, as mentioned above. Selected lines appear on top in the selection color.

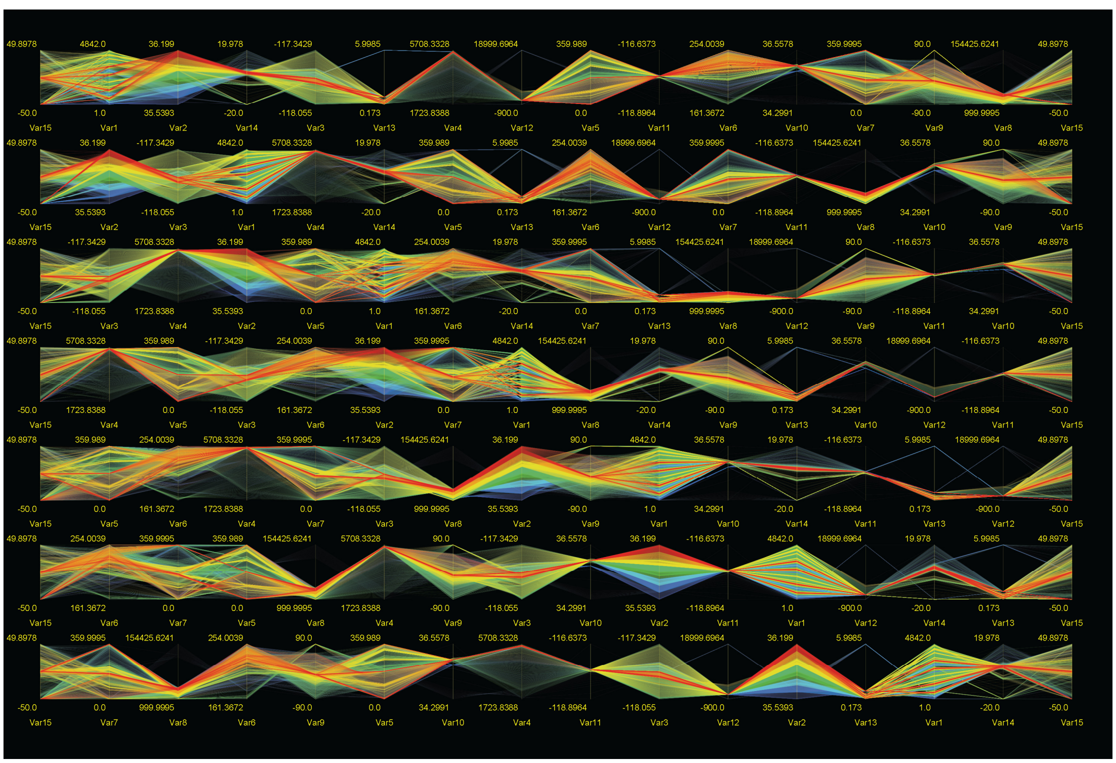
2. The plot can be zoomed and panned to reveal more details.

3. The PCM can be exported to PostScript and PDF just like the PC Plot in the previous section. A separate alpha (opacity) value for the PCM can be entered in the Options→PostScript/PDF Export Settings menu

option, and the export obeys the general settings for showing axes and text (although this is one case where including the text is very useful).

Two more PCM displays are shown below. The second has been exported to PS/PDF with 0.05 opacity.





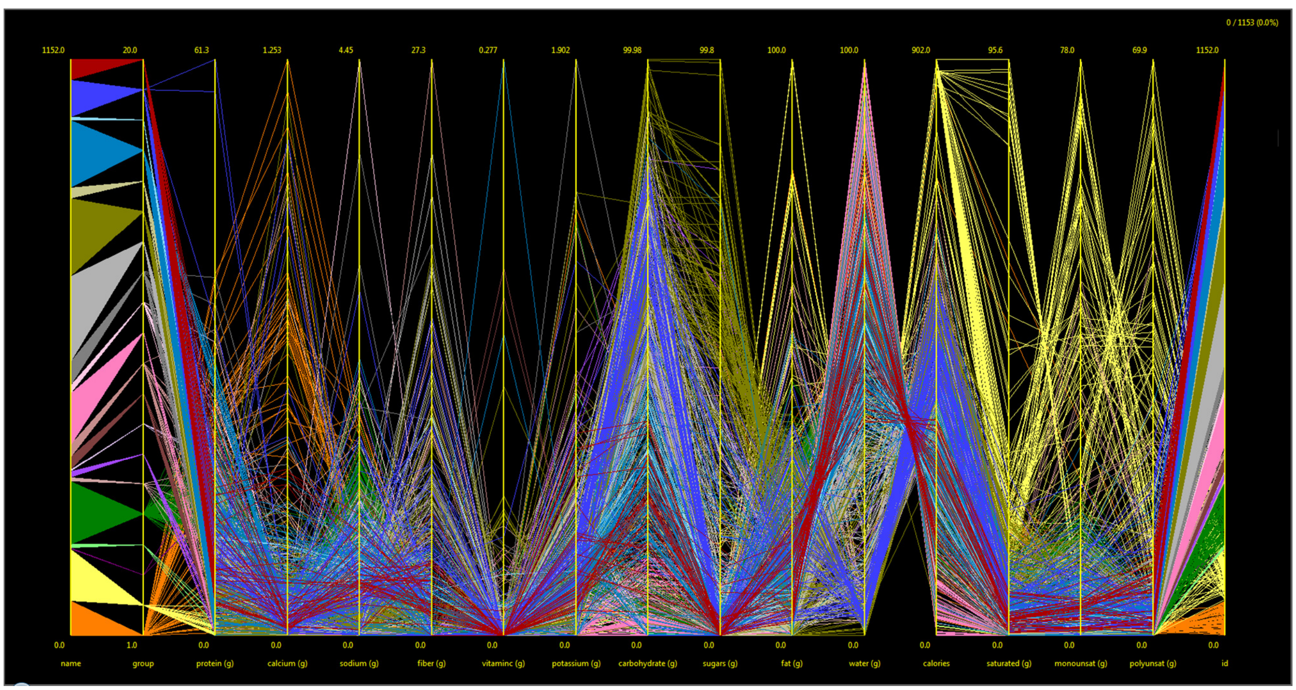
2D Scatterplots

Scatterplots are often described as the most effective visual display of data relationships between two variables. In Sliver the user can create multiple 2D scatterplots of the data. These scatterplots represent cross-sections of the data set and are used in conjunction with the parallel coordinate plot to better locate data clusters and outliers, and to identify structural (functional) relationships between the variables.

Sliver creates 2D scatterplots of any two selected variables in either Cartesian (x,y) or polar (r,θ) coordinates using the **Plots→New 2D Scatterplot** menu option. When polar coordinates are chosen, the angle can be in either degrees or radians and can increase counterclockwise from the x-axis or, as in azimuths in navigational data, clockwise from North (the y-axis).

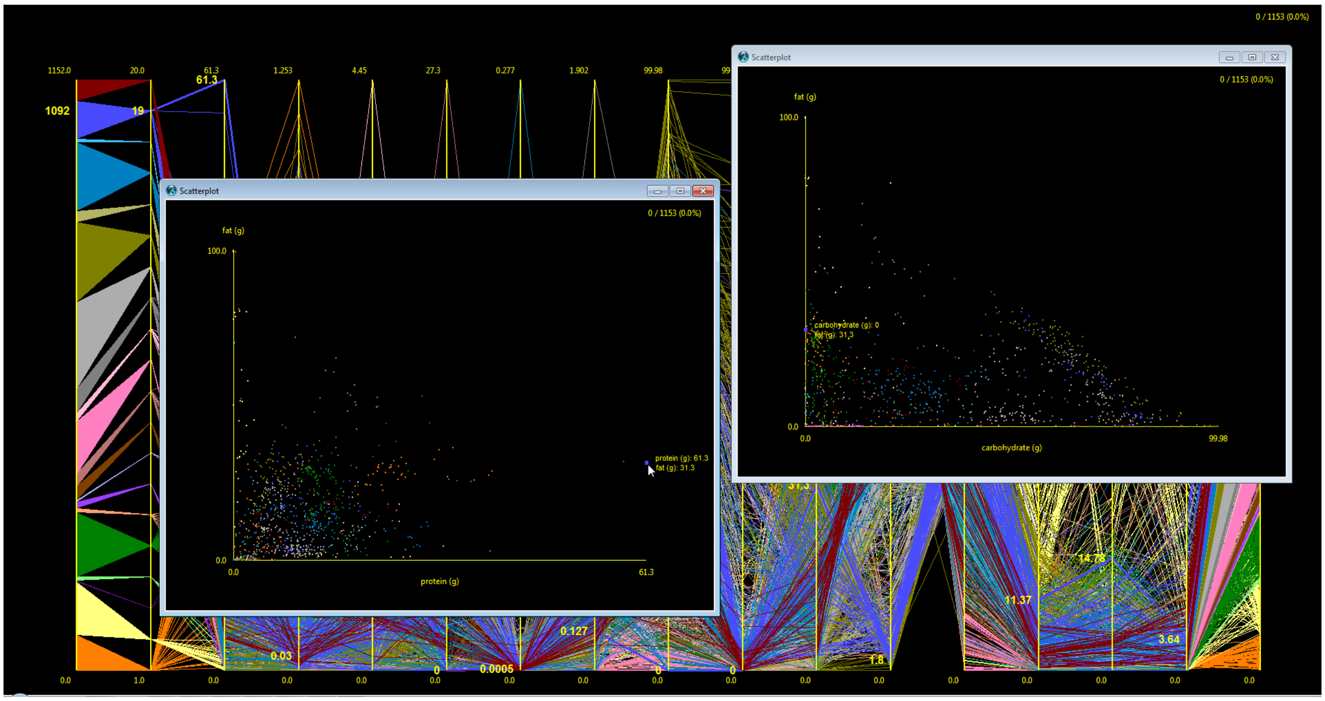
The axes of the scatterplots range from minimum to maximum values just as the axes of the PC Plot. As in that plot, each scatterplot window can be resized, zoomed and panned. A traditional plot of values vs. time can be created by selecting time (or data row number) as the variable for the x-axis and stretching the scatterplot window into a wide, narrow strip.

Below is a parallel coordinate plot in Sliver of the nutrition content per 100g of 1153 foods (left axis) grouped into 20 food colored groups (second axis from left, from which the colors have been brushed using gap brushing as described earlier). This dataset is included as *D3NutritionExportAllNumbersFilled.csv* file in the *Sample Data* folder in the Sliver distribution. There are 14 variables representing protein, calcium, sodium, fiber, vitamin C, potassium, carbohydrates, sugars, fat, water, calories, saturated fats, monounsaturated fats, and polyunsaturated fats.

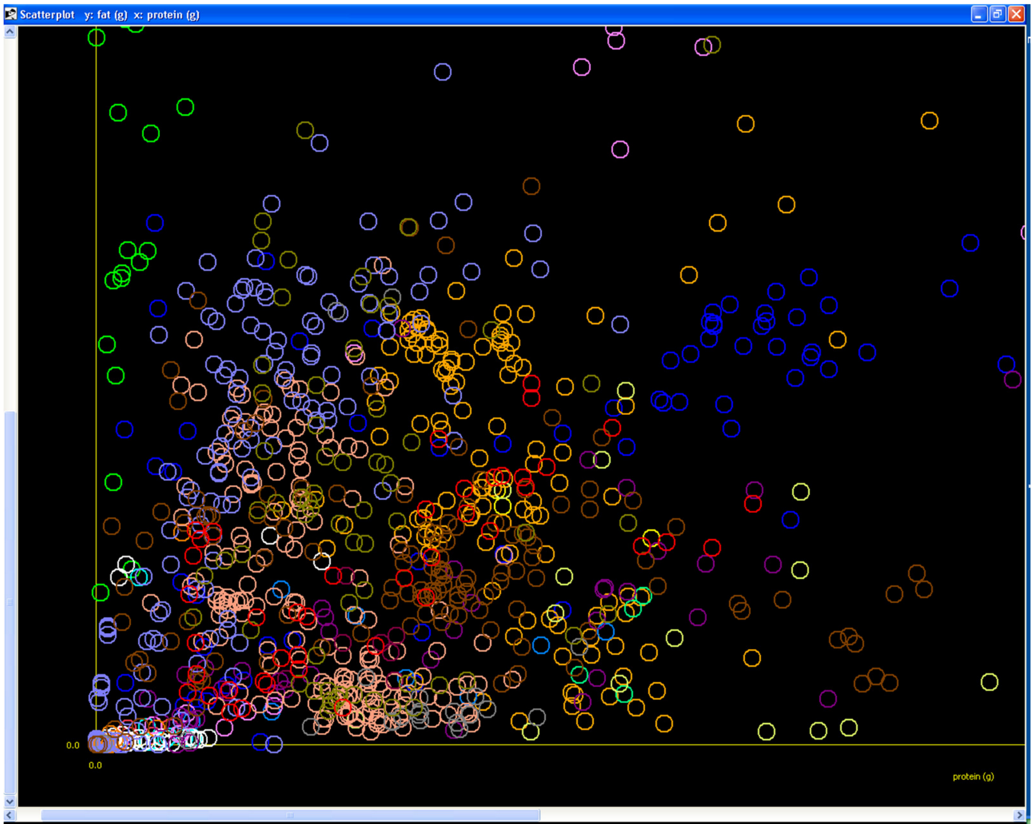


To analyze particular variable relationships, we can create scatterplots. Below are two such plots, one for fat vs. protein and one for fat vs. carbohydrates. The data in these plots share their colors with the brushed colors in the parallel coordinate plot. In fact, data can be swiped for selection and colored in any one of these three plots and the colors are automatically propagated to the other plots, and this is what is meant by *fully-linked* plots. Actually these 2D scatterplots are really 3D plots considering the use of color as a function of one or more other variables.

The colored food groups are well-clustered in the scatterplots. There are two outliers in the first scatterplot that are curious. We can turn on the option again to show axis values when the mouse is hovered using the **Plots→Display Values on Mouse-Over→Without String Mapping** menu option. This function works on scatterplots as well as the PC Plot. Placing the mouse over the rightmost outlier in the left scatterplot below highlights (increases the size of) that dot, and the data values appear in that plot as well as the other two plots. We see that this food has 61.3g of protein and 31.3g of fat per 100g! The rightmost plot shows that it has zero carbohydrates, and all the values are shown on the PC Plot. What can this food be? The leftmost axis of the PC Plot identifies the food as #1092 of the 1152 foods, and checking this against the original data assignment reveals it to be barbecue-flavored pork rinds. The other outlier is plain pork rinds.



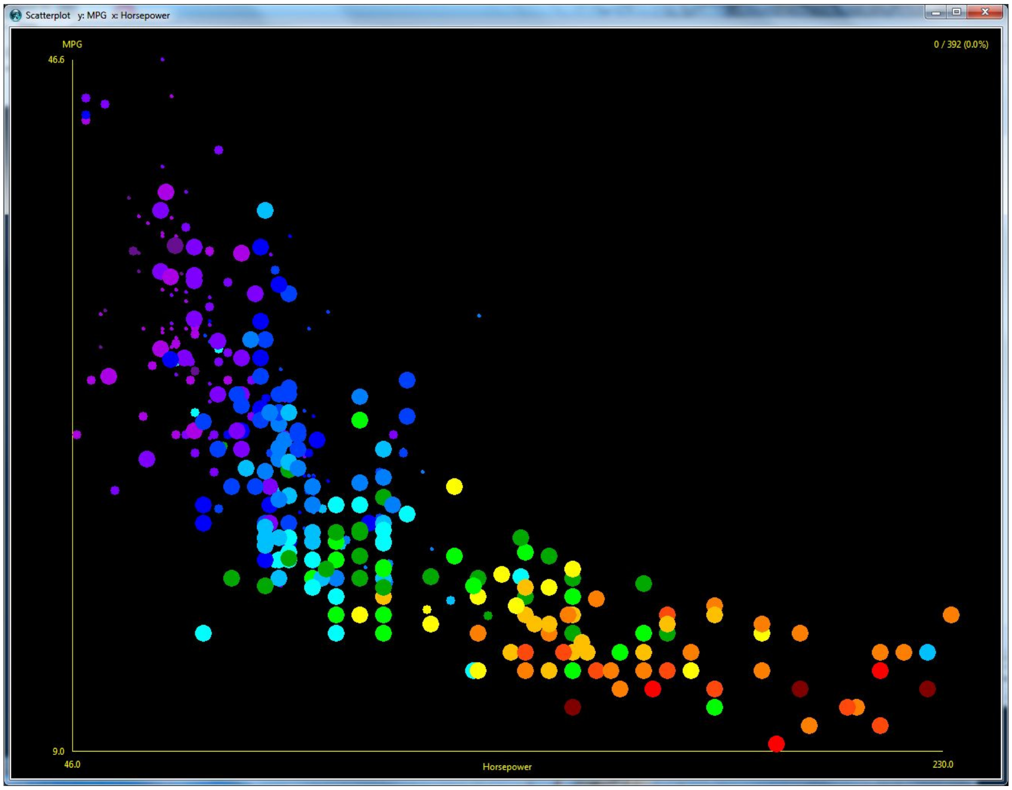
There are a number of scatterplot settings that can be applied. For example, the plotted points here can easily be seen on a monitor, but are very difficult to see in print. Using the **Options→Plot Settings** menu option, the dot size can be set to a different value, for example, and the plot can alternatively use hollow circles rather than dots. Hollow circles are useful for revealing hidden points and the density of clusters. An example of this is shown below, zoomed into a section of the fat vs. protein scatterplot.

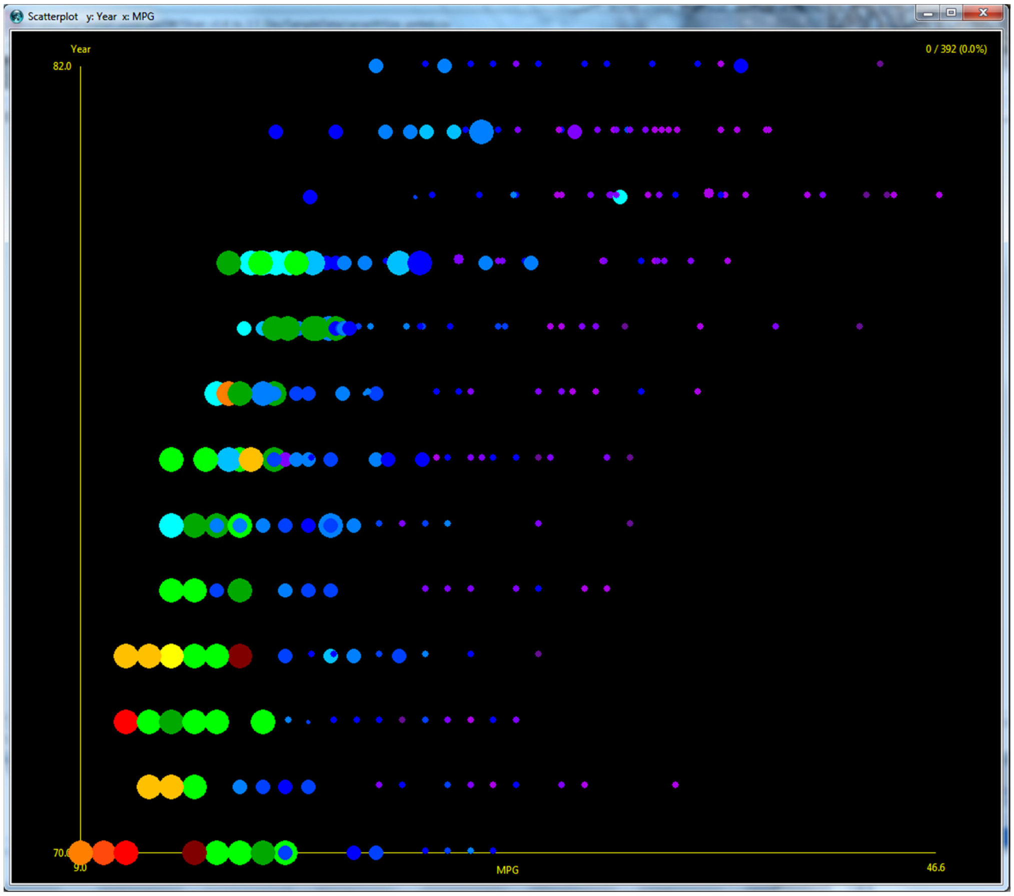


There is also an option to select a variable to define the size of each point, including a scale factor. In addition to the x-variable, y-variable and color there is now a fourth dimension of size incorporated into these "2D" scatterplots. These types of plots with variable point sizes are called *Bubble Charts*. The smaller points are drawn on top of larger points for visibility, and even greater visibility is achieved by exporting the plot to PDF with transparency as described earlier.

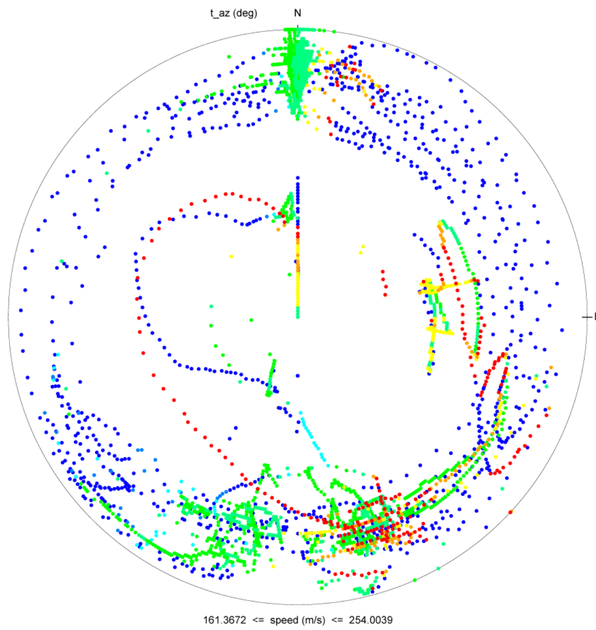
Below are two examples of a bubble chart for the *cars.csv* dataset. In the first plot a new variable was created having a large value for cars made in the US, a middling value for cars made in Japan, and a small value for cars made in Europe, based on the Origin(US/Japan/Europe) variable in that dataset. When this 2D scatterplot was created, a box was checked to include a size variable and this new variable was selected with some scale factor. The figure plots MPG vs. Horsepower, and is range brushed for Weight and sized for the Origin of the cars (the high-horsepower cars in this dataset were all made in the US).

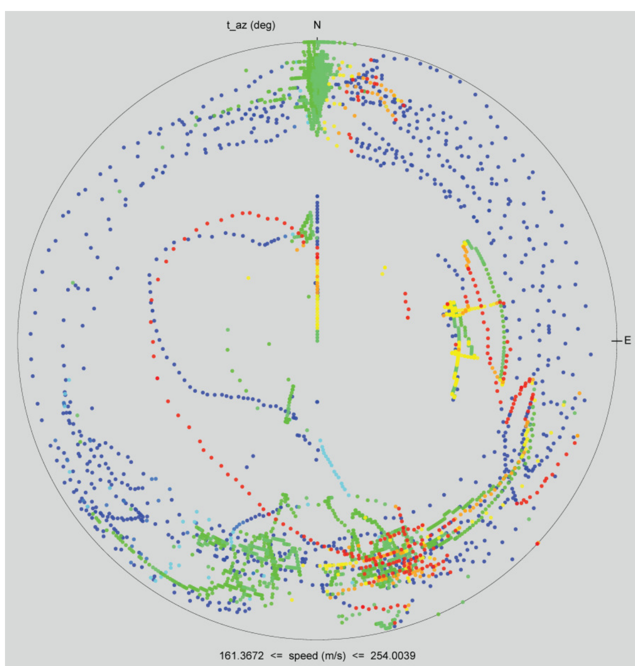
In the second plot below, the MPG variable was plotted against the Year, showing a general trend toward higher MPG as time went on. The Horsepower variable was range brushed from lowest to highest in the PC Plot, revealing some interesting outliers in the scatterplot. The Cylinders variable was chosen for the size of the points, along with a scale factor, showing that in later years there were cars with high cylinder counts but good mileage. Just as in 2D scatterplots of fixed point size, these variable size points can be selected by swiping and the **Plots→Display Values on Mouse-Over** menu options will display values on mouse-over in any plot.


Scatterplot y: MPG x: Horsepower
MPG
0 / 392 (0.0%)
46.6
9.0
46.0
Horsepower
230.0


Scatterplot y: Year x: MPG
Year
0 / 392 (0.0%)
82.0
70.0
9.0
MPG
46.6

Note that the Sliver color scheme is completely configurable with the **Options→Color Scheme** menu option. The above plots have absolutely brilliant colors on a monitor, but for print purposes a light background or solid dots may be preferable. The figures below show a polar scatterplot with two different backgrounds that were exported to PDF without transparency to provide a vector-drawn graphic of presentation and document quality.





3D Scatterplots

Sliver can either use Google Earth to display 3D scatterplots or export its data to Matlab for its interactive 3D scatterplot function.

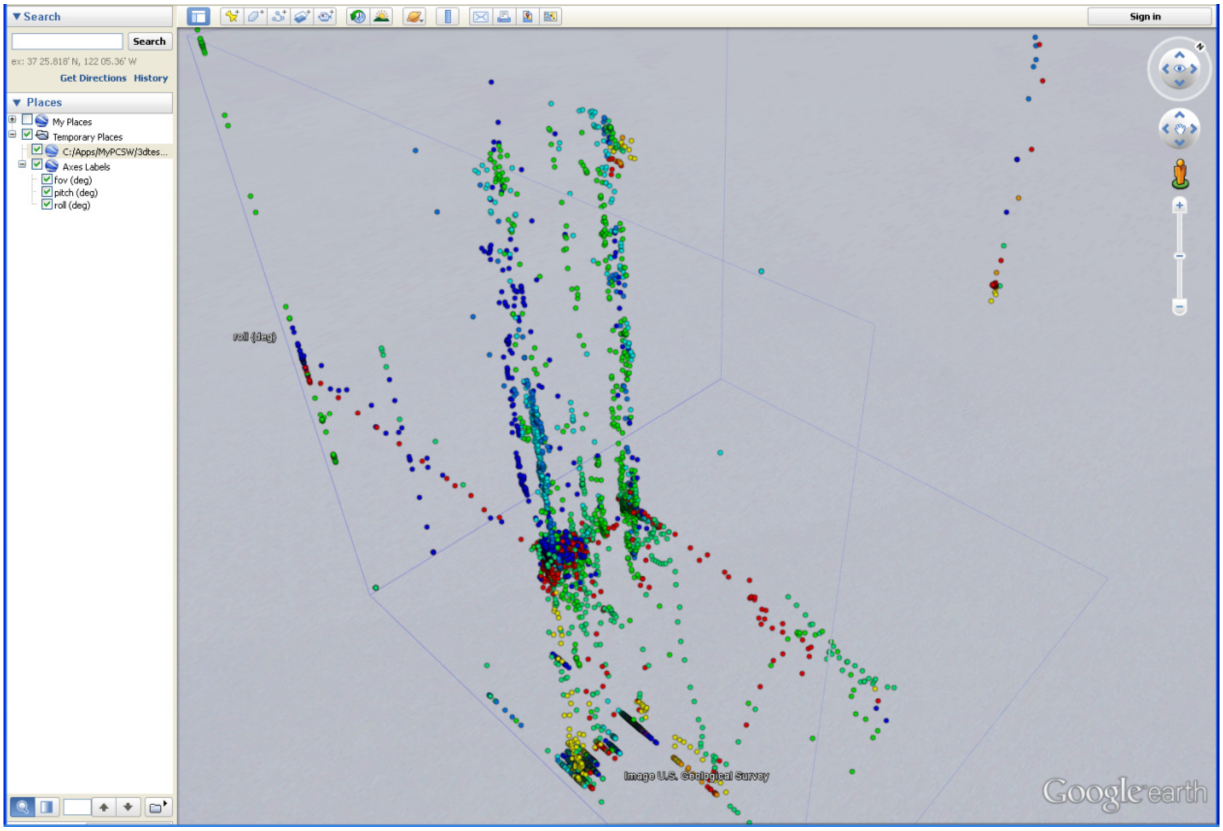
Google Earth provides extremely fast rendering of icons in a 3D setting, and it supports transparency levels that are useful when viewing many thousands of points in space. (However, Google Earth may not display a scatterplot of tens of thousands of points.) Mouse controls easily rotate, zoom and pan the view to look for structure, and the zoom control in the upper right provides precise zoom control. Spherical points of configurable size and transparency are positioned in a cubical volume of 0.1 nautical mile per side positioned above a white, remote part of Antarctica. Axes. Axes labels are drawn as well. Again, the color brushing based on one or more variables confers a 4D characteristic to the 3D scatterplot.

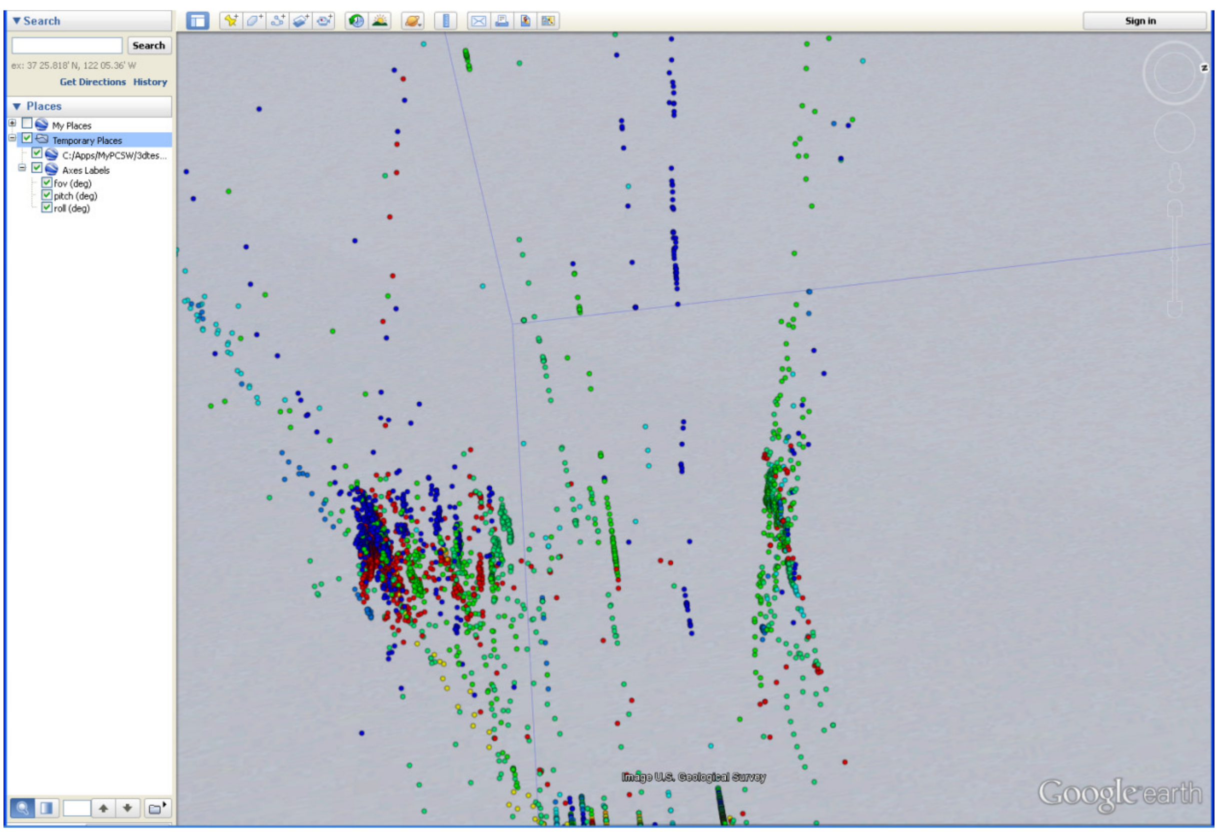
The **Plots→New 3D Scatterplot** menu option is used to create these. Three variables are selected, one for each axis for a Cartesian plot. All linear variable axes are scaled from minimum value to maximum value to expand to the bounding box. Angular variables wrap a full 0 to 360°. All the spherical points are colored according to the current color brushing of the other plots. A Google Earth KML file is generated, and if the path to the Google Earth executable is provided in the settings the file is automatically launched in Google Earth. Color brush changes in the other plots do not automatically update the Google Earth 3D scatterplot, but a new KML file is easily generated using the same menu option that can have the same or different filename. If it is saved with the same name, Google Earth automatically asks to redraw the existing plot. Once created and saved, these KML files are available to be directly loaded into Google Earth at any time, individually or in combinations. Axis labels are separate KML placemarks and can be unchecked in Google Earth to remove them if they obscure any detail.

NOTE: When moving or sending a KML file that displays spheres such as a 3D scatterplot, you must include the *dot.png* file that was added to the directory of the KML file! This PNG file provides the graphic for the spheres, and if it is missing Google Earth will complain or replace it with another shape.

Also, you can zip a KML file into a KMZ file (which needs to end in .kmz) with a size approximately 1/10 of the original KML file. You can include the *dot.png* file in this zip file. This zipped KMZ file will open correctly in Google Earth, which unzips the contents before displaying the plot. A KMZ file is a very convenient way to decrease the size of emailed Google Earth plots as well as to include needed graphic files.

Below is an example of a 3D Cartesian scatterplot generated by Sliver. The 3D aspect is not appreciated without interactively manipulating the plot by rotating, zooming and panning, so a second view is also provided here to help give a sense of what can be visualized with such a plot. Note that the point sizes are defined in units of pixels, so when zooming in the points spread apart—you are not simply zooming in closer to a fixed cluster.

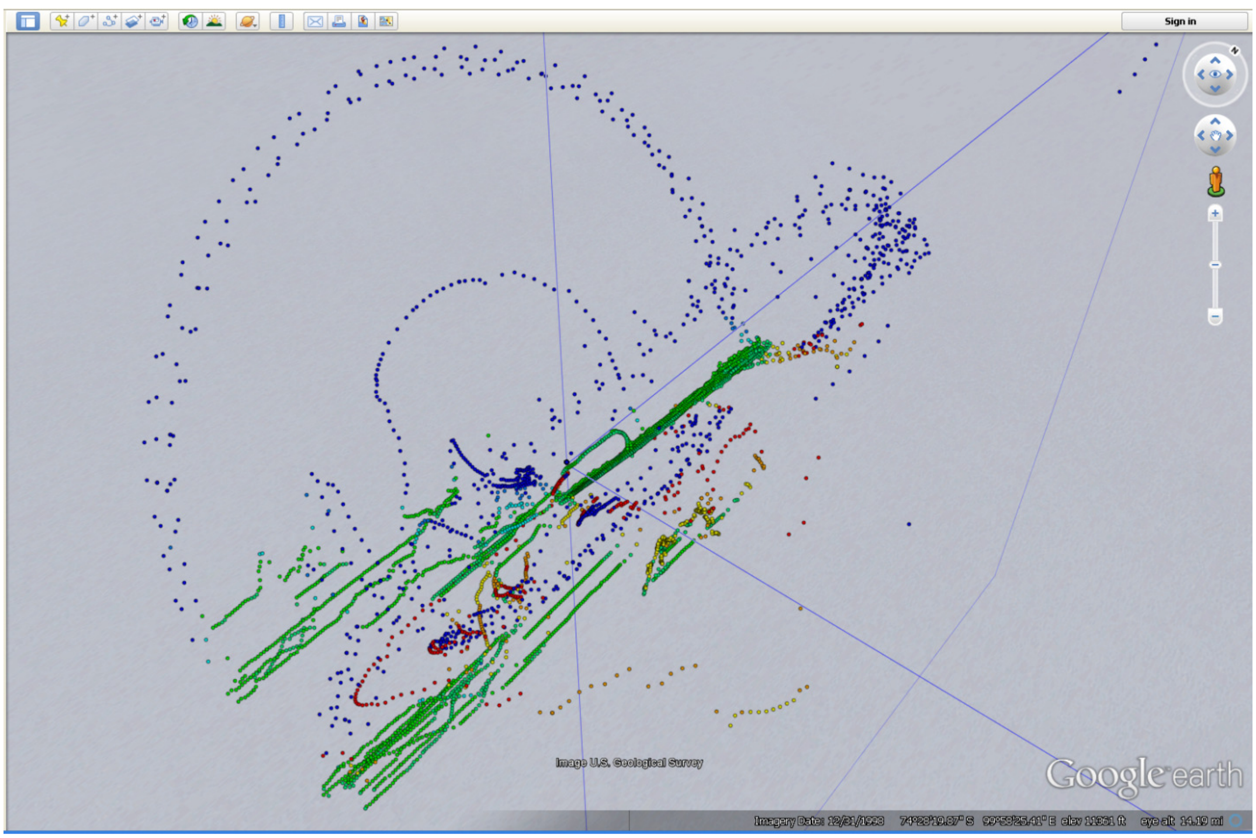

Search
Search
ex: 37 25.818' N, 122 05.36' W
Get Directions History
Places
My Places
Temporary Places
C:/Apps/MyPCSW/3dtes...
Axes Labels
fov (deg)
pitch (deg)
roll (deg)
Sign in
roll (deg)
Image U.S. Geological Survey
Google earth


Search
Search
ex: 37 25.818' N, 122 05.36' W
Get Directions History
Places
My Places
Temporary Places
C:/Apps/MyPCSW/3dtes...
Axes Labels
fov (deg)
pitch (deg)
roll (deg)
Sign in
Image U.S. Geological Survey
Google earth

In addition to Cartesian coordinates, Sliver supports Google Earth 3D plots of data in spherical and cylindrical coordinate systems, with angles specified in either degrees or radians, selectable in the same window where the variables to plot are selected. The user can also choose whether math-based or navigation-based conventions are used. The math convention for spherical coordinates (r, θ, φ) has θ increasing counterclockwise from the x-axis and φ down from the z-axis, while in navigation the azimuth θ increases CW from North and the elevation φ is relative to the horizon (the x-y plane). The North and East axes in navigation mode point to North and East in Google Earth, respectively.

NOTE: To help visualize relationships, the 3D scatterplots are scaled to fit the data in a given volume. The x, y and z axes of a plot in Cartesian coordinates are scaled so the minimum and maximum values of each variable span a single, cubical volume outlined by the coordinate planes. The angles in spherical and cylindrical plots can span a full 0 to 360° (or 0 to 2π if radian units are selected), so these plots fill a 2x2 cubical volume. Two outlines of the 2x2 cube are drawn for reference. The minimum radius in spherical coordinates lies at the origin and the maximum radius lies at some point on a sphere touching the 2x2 cube. The minimum and maximum values of z in cylindrical coordinates span the 2x2 cube vertically, with the minimum at the very bottom.

Below is a spherical plot color-brushed in concert with other plots. You can tell a great deal about a system by studying 3D data plots such as this.



As another application, a 3D scatterplot of a hiking, biking, or aircraft path given by latitude, longitude and altitude values provides an extremely clear visualization (scaled in each dimension to fit the cubical volume). This can be clearer than a standard Google Earth path plot because of the lack of a cluttered background.

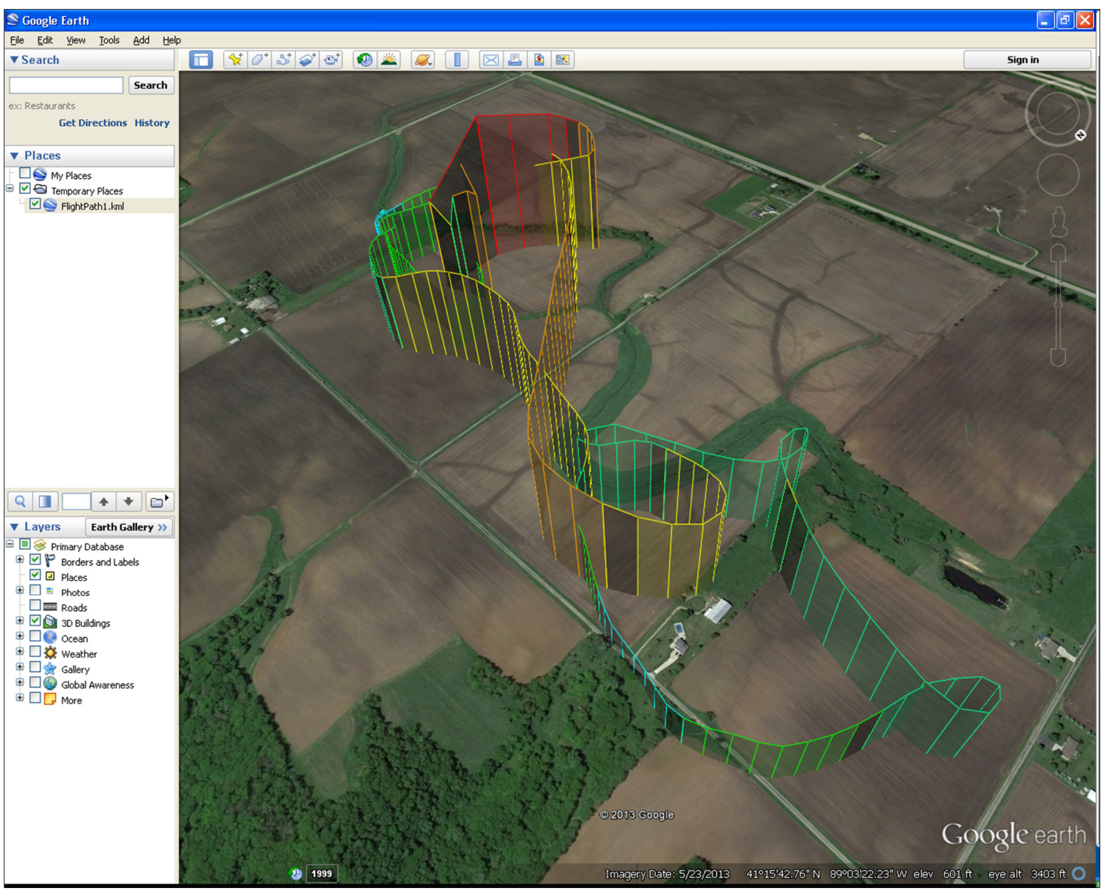
In addition to plotting 3D scatterplots in Google Earth, Sliver can export data to plot as a 3D scatterplot in Matlab. Choosing the **Analysis→Export 3D Scatterplot** menu option pops up a window to choose the three variable for x, y and z in a Cartesian coordinate system, along with the size of the points. If Matlab is installed on the PC and the path to the matlab.exe file is entered in the **Options→Analysis Settings** menu option, the 3D scatterplot will appear in a Matlab figure with the color brushing that was applied in Sliver. The interactive controls and myriad of settings of Matlab can then be applied to the plot as desired, and the plot can be saved in Matlab format if desired.

Google Earth Path Plots

Google Earth is also used to plot path data from variables representing latitude, longitude and altitude. The **Options→Google Earth Settings** menu option in Sliver provides settings for units (degrees vs. radians, feet vs. meters), path opacity and width in pixels, vertical supports extending the ground and their opacity, and the altitude reference (sea level, clamp to ground, etc.). Since the path is composed of connected data points, the path segments are colored according to the color brushing of the other plots. This is extremely useful, for example, in identifying the points along a path where particular conditions occur.

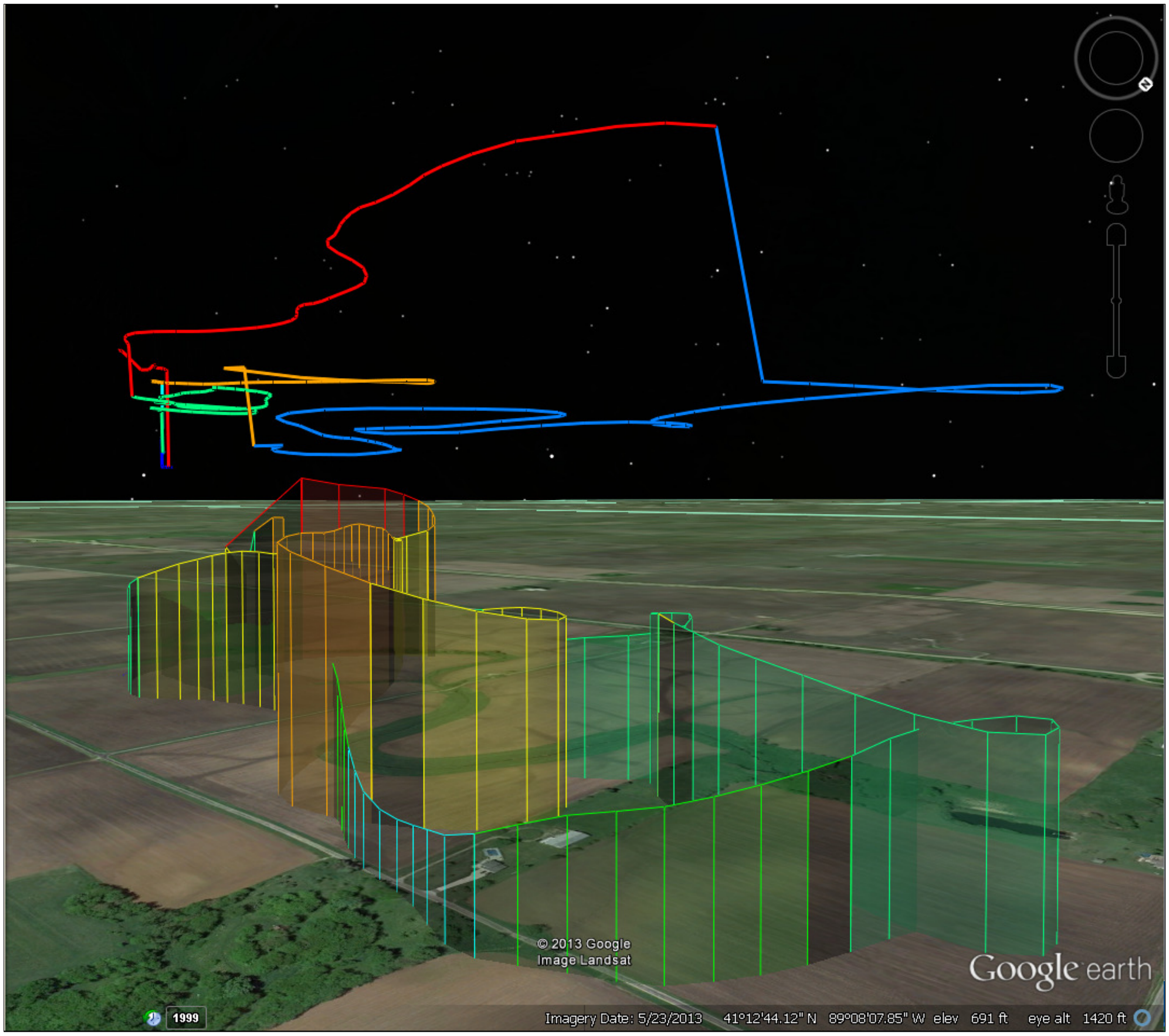
These plots can be helpful in visualizing and analyzing data for runners, bikers, hikers and so forth, particularly when data such as speed, pulse rate and such is available for study. Garmin Connect, RunKeeper and Strava, for example, offer CSV downloads of running and biking data uploaded from GPS devices. The **Plots→New Google Earth Path Plot** menu option is used to create the plot. Variables are selected for the three coordinates, and an option is presented to only plot every nth row of data. This is important because transparency of supports decreases with more points. There is also an option to not connect geographic points farther apart than an entered distance. Otherwise, plotting a set of segments would add straight lines between them, but these segments can be isolated by entering a minimum separation distance to connect. Note that to show a single disconnected point, the option for supports must be enabled in the Google Earth Settings.

With supports turned on and full opacity in the settings, a KML file for a plot such as the one below is created and automatically launched in Google Earth if the path to the executable is entered in the Google Earth settings. The mouse can be used to zoom, rotate and pan the view as desired. The flight path below would be smoother but there was a very limited number of flight coordinates in this dataset.



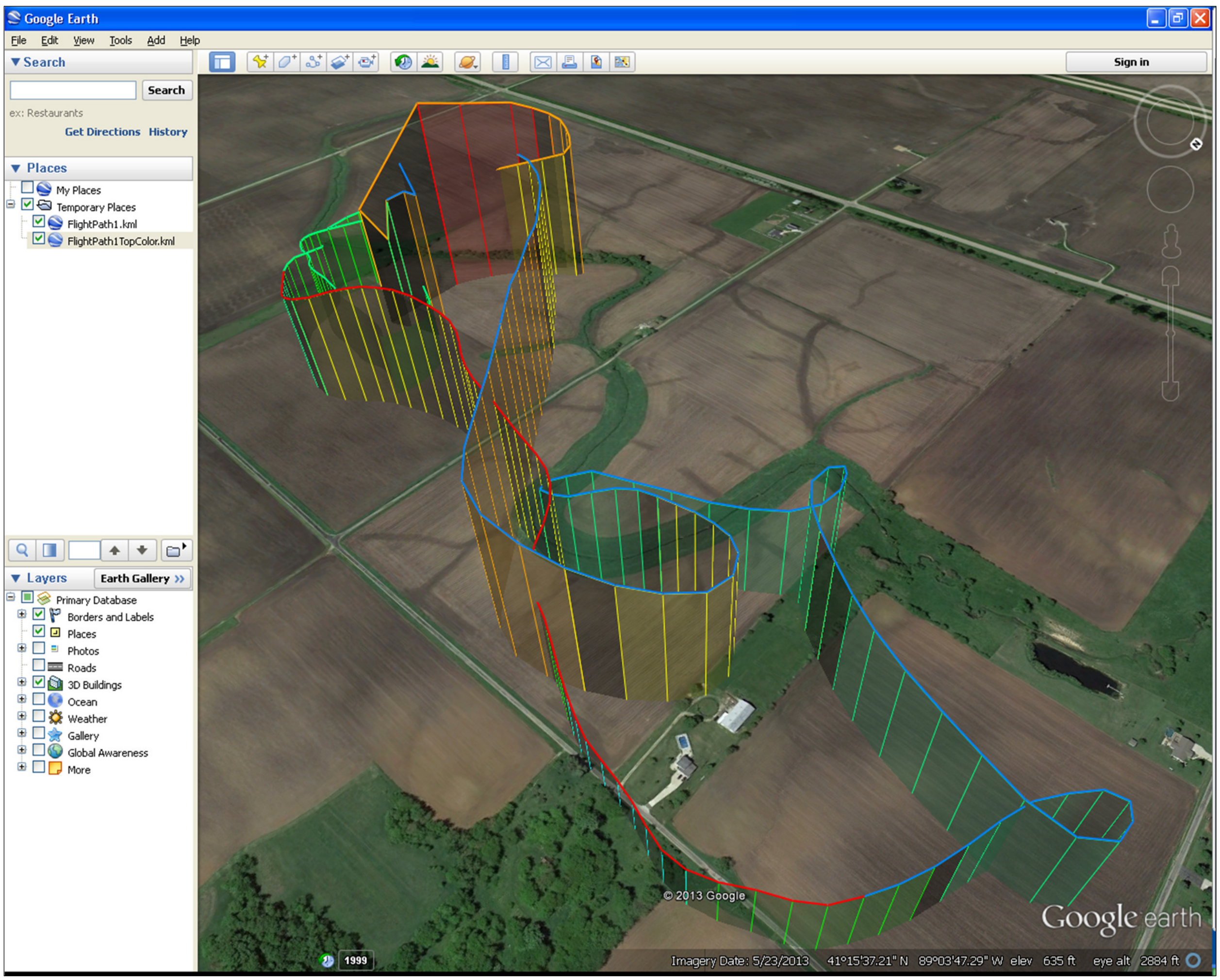
There are also settings in the **Options→Google Earth Settings** menu to apply multipliers and offsets to the latitude, longitude and altitude values. One application of this is to plot a second variable as the altitude, with an altitude offset and multiplier sufficient to plot the curve directly above the flight path. This shows values of another variable, and possibly another color brushing, above the flight path.

For example, by selecting the same latitude and longitude variables as before but choosing a different variable as the altitude, we can generate a plot of this variable at any point along the flight path, saving the KML file with a different filename. This gets added to Google Earth by Sliver, and this new plot lies directly above the existing flight path. It is much easier to visualize this by using mouse controls to pan and zoom, but one view is provided below.

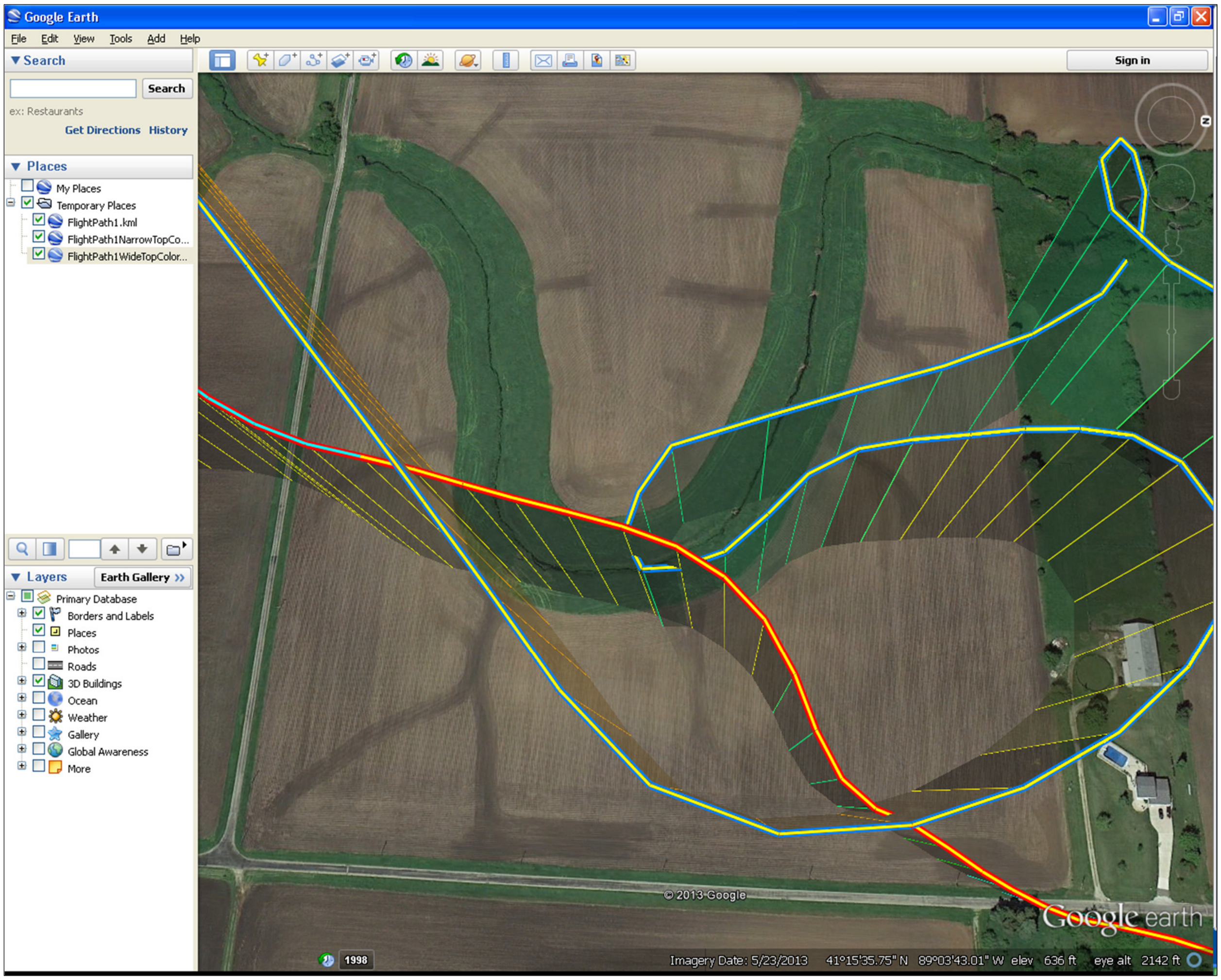


To overlay additional paths using the same geo-coordinates after performing a different color brushing possibly related to a different variable, a new Google Earth plot can be created in the same way as the first one and saved to a different KML file. Although the paths will exactly overlap when both KML files are displayed, there is a shimmering effect in the supports that will reveal two or three sets of colors as Google Earth is zoomed and rotated. Also, we can create multiple versions of the same path in separate KML files and simply toggle them off and on in the left pane of Google Earth as layers, allowing us to see differences in the path colorations as the paths appear and disappear.

Another way of visually overlaying data along a path is to create a thin path with supports that is colored (in the PC Plot) according to certain variable conditions, and then create a second path colored for a different set of conditions and placed with a small altitude offset. Then the supports show one variable dependency while the path along the top shows a different dependency, as shown below.



Or we can create a ribbon-like path along the top whose innermost colors reveal values under certain variable conditions and whose outer colors reveal other conditions. The top path width in the figure above is re-plotted after the Google Earth setting for path width is wider, say 12 pixels. Then the path width setting is changed to, say, 4 pixels and the altitude offset is increased by 10 meters, a new variable or combination of variable is color brushed, and the flight path below is produced that shows one color dependency in the supports and two color dependencies in the path along the top. Widths can be adjusted to provide easily readable colors depending on characteristics of the flight path.



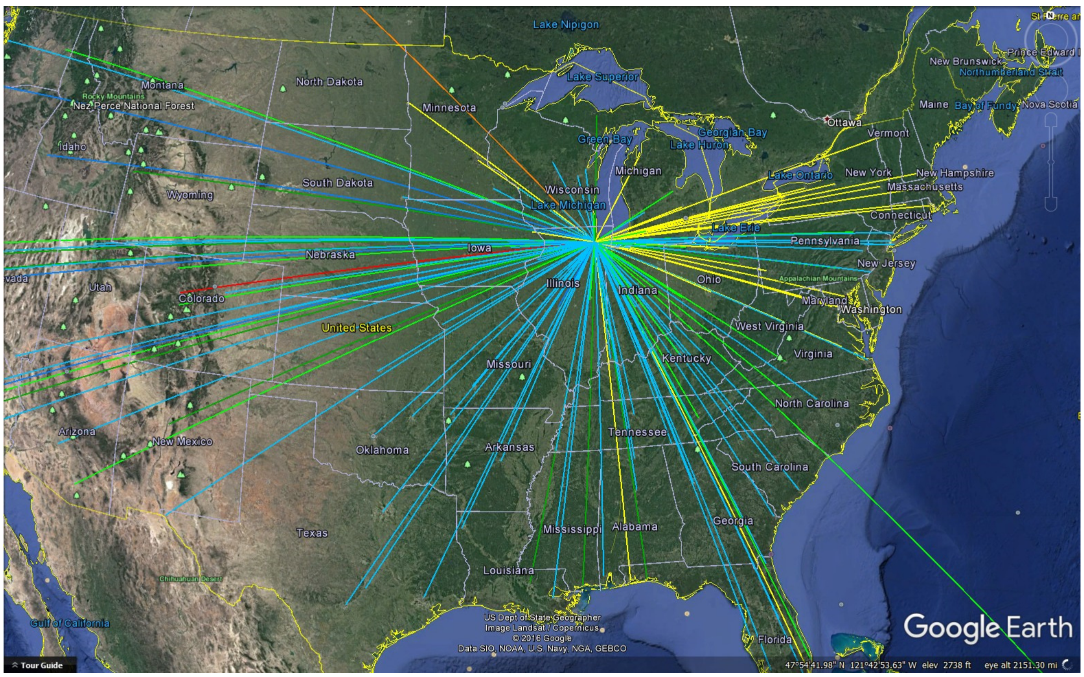
When creating a Google Earth Path Plot, options are provided to add timestamps, pop-ups and labels. These require that spheres be added along the top of the path to serve as clickable points. Since the options are common to all Google Earth plots, they are discussed later in a separate section.

A separate KML file is created every time a Google Earth plot is created, so they are always available to be loaded into that program outside of Sliver and can be mixed and matched as desired. There is also a **DataTools→Merge KML** Files menu option to combine multiple KML files generated by Sliver into a single KML file. You can also zip a KML file into a KMZ file (which needs to have a suffix of .kmz) to decrease its size by about a factor of 10.

Google Earth Connection Plots

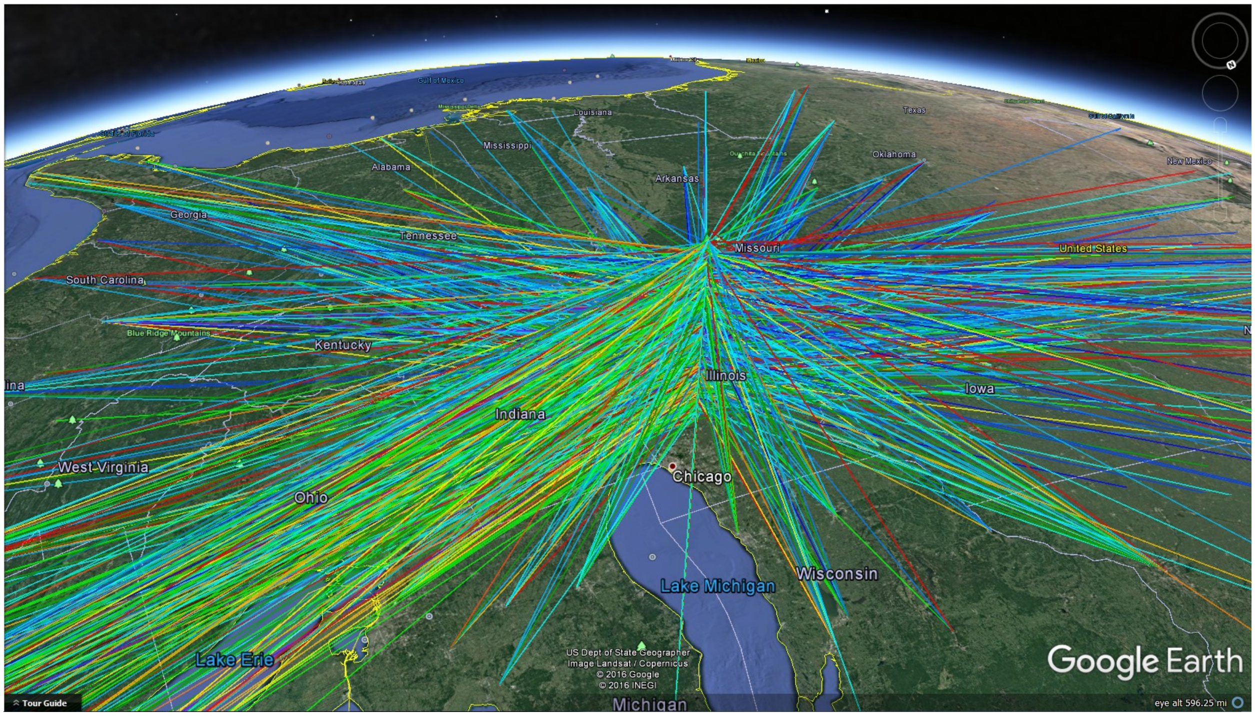
The menu option **Plots→New Google Earth Connections Plot** creates a KML file that plots lines between a selected pair of geo-coordinate variable sets (lat/lon/alt for point 1, lat/lon/alt for point 2) in each row of data, colored with the brush for that row. Plotting every nth row is also possible.

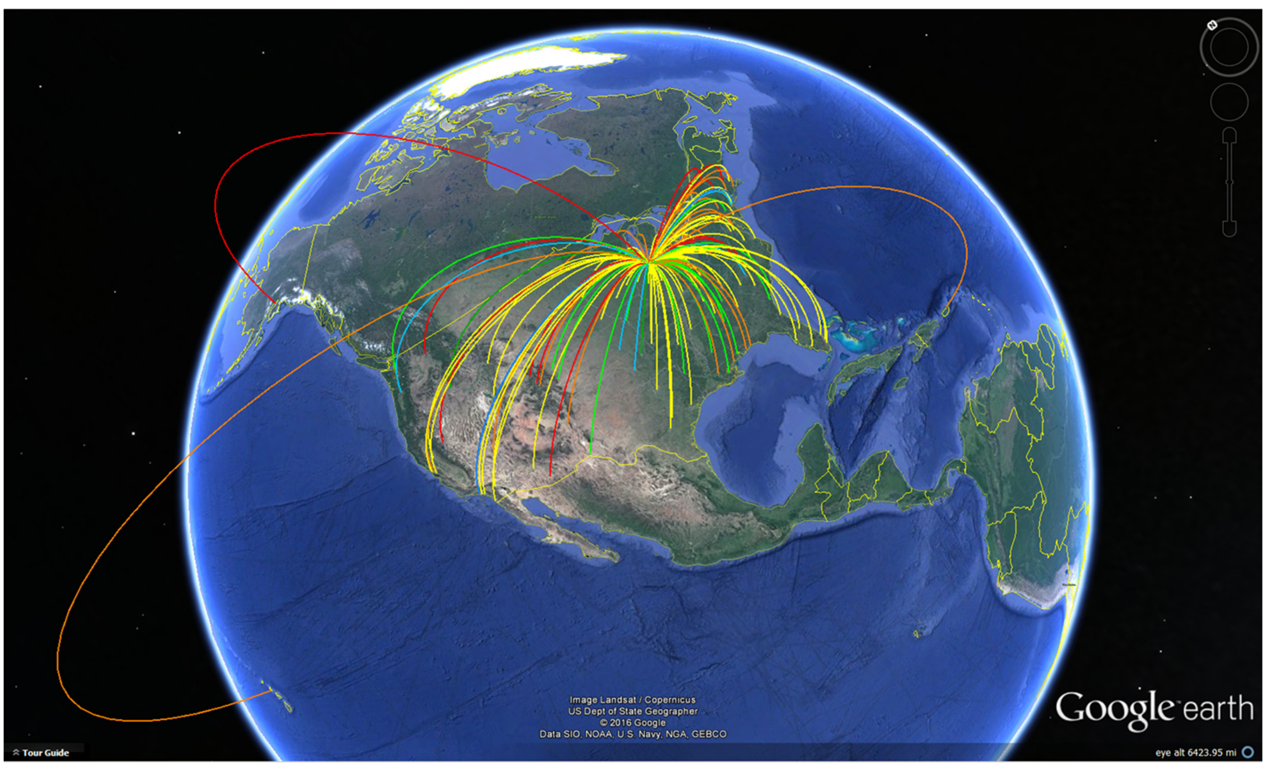
There is also an option to assume the endpoints of the connection lie on the ground. Otherwise, lines connected between distant points can pass through the Earth if the altitude reference is not set to *Clamp to Ground* in the Google Earth Settings. This also means that the altitude variable does not have to be selected, which is necessary when only ground coordinates are available. This ground connection lies on the surface of the Earth along the great circle path between the points, with width, opacity, etc., set as before in the Google Earth Settings. An example showing US-bound outgoing flights from the Chicago O'Hare Airport on January 15, 2008, color-coded for arrival delay, is shown below. Note that you can create multiple connection plots to complete grids of lines for rectangular areas, and the **DataTools→Merge KML Files** menu option can be used to combine them into a single KML file.



The file of data for this plot is found in the *SampleData* folder, although you will want to select only variables with numerical values such as Origin Lat/Lon, Destination Lat/Lon, Arrival Delay, etc. There are 972 flights in this plot, but many connections overlap because many cities have multiple flights from O'Hare in a single day. One option is to not assume the connections lie on the ground and use another variable, such as departure time during the day, as the altitude, with an appropriate multiplier in the Google Earth settings so that the altitude changes are visible. The first plot below is such a connections plot where the altitude at the O'Hare position is the departure time multiplied by such a factor. This reveals the various flights better, and can be explored using Google Earth controls to see if later flights to a city, say, are delayed more than earlier flights. Settings for the plotted lines (units, width, opacity, supports and their opacity, etc.) are taken from the Google Earth settings except that the altitude reference is not used when endpoints are chosen to lie on the surface.

You can also draw vertical parabolic arcs between two connections lying on the ground rather than a line. The heights of the arc for different distances (the ratio of the distance between the points and the height) is user-selectable. This arc lies above the great circle path between the points. Again, the width, opacity, supports, etc., for the arcs is set in the Google Earth Settings, where you probably prefer the supports to be disabled. The second plot below is of the same data as previously but with a slightly different color-coding. This plot overlaps flights.





As before, this KML file can be added to any other KML files displayed in Google Earth.

When creating Google Earth Connections plots, options are provided to add timestamps and pop-ups (but not labels) as discussed in a separate section.

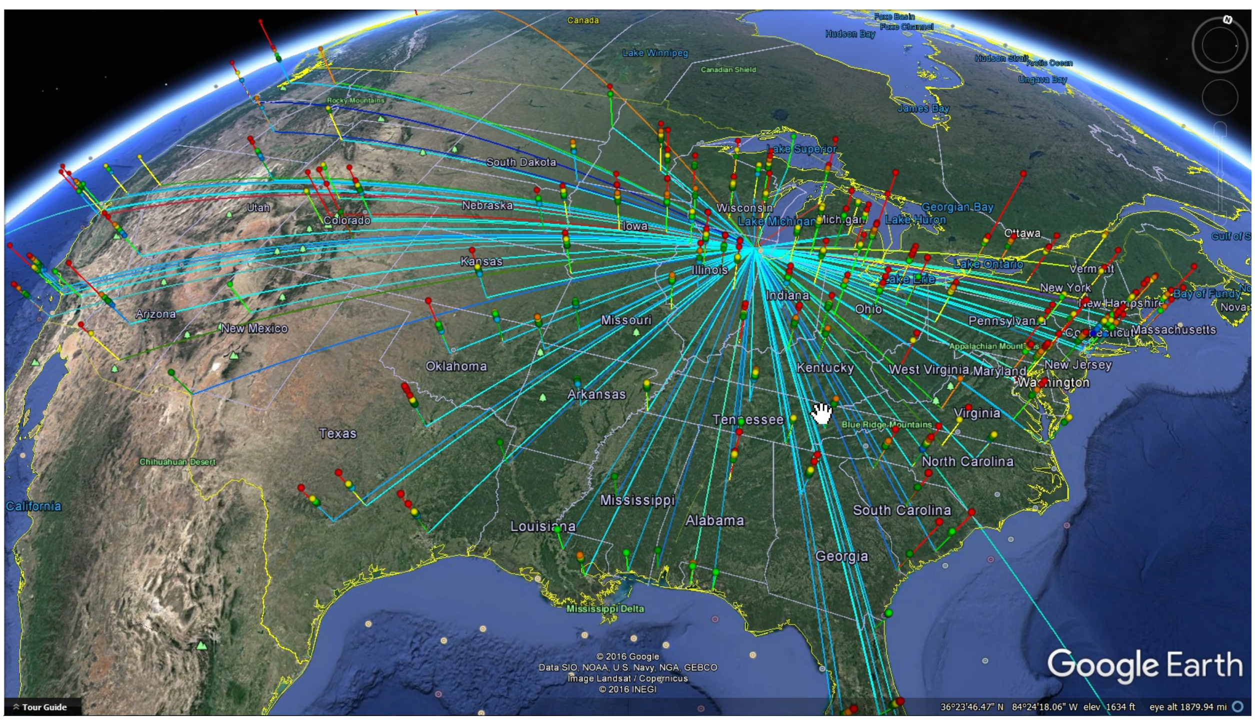
Google Earth Pillar Plots

The menu option **Plots→New Google Earth Pillar Plot** plots vertical rods at geographical points on the surface of the earth. This is very nice for marking individual points, even densely spaced points, with Sliver color-brushing and either a fixed height or variable heights. The width and opacity of the pillars is user-selectable in the pillar plot window. The height above the ground of the pillars can either be a third variable or be set as a fixed value entered in the window. There is also an option to place a sphere on top of each pillar of that color and a configurable size and opacity. These vertical rods with or without spheres are a finer alternative to the large pins and markers that Google provides by default.

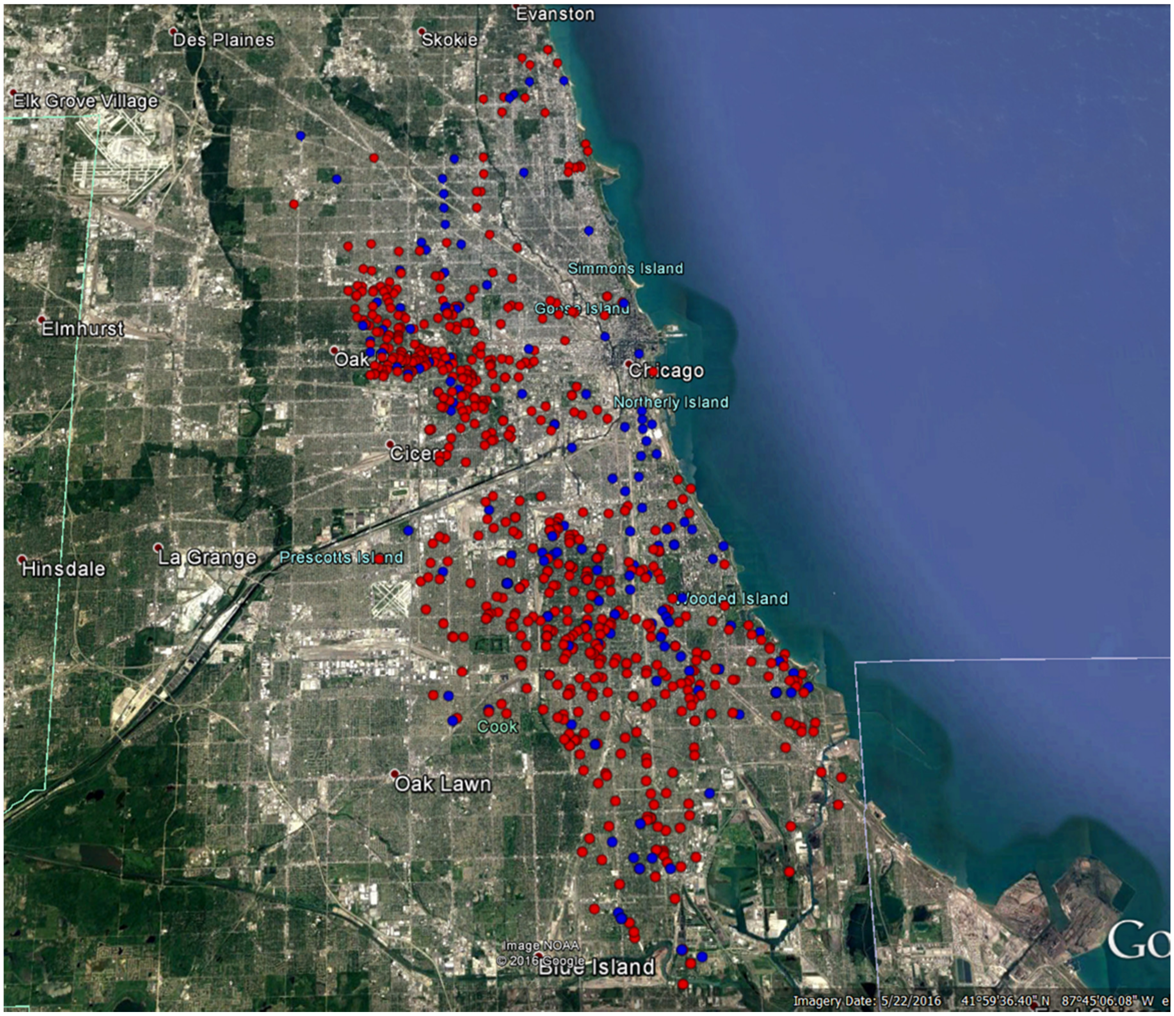
NOTE: If spheres are added to the pillars, any move of the KML file must also include the *dot.png* file added to the directory of the KML file when it was created, as it provides the graphic for the spheres. If it is missing Google Earth will complain or replace it with another shape.

Also, you can zip a KML file into a KMZ file (which needs to end in .kmz) with a size approximately 1/10 of the original KML file. You can include the *dot.png* file in this zip file. This zipped KMZ file will open correctly in Google Earth, which unzips the contents before displaying the plot. A KMZ file is a very convenient way to decrease the size of emailed Google Earth plots as well as to include needed graphic files.

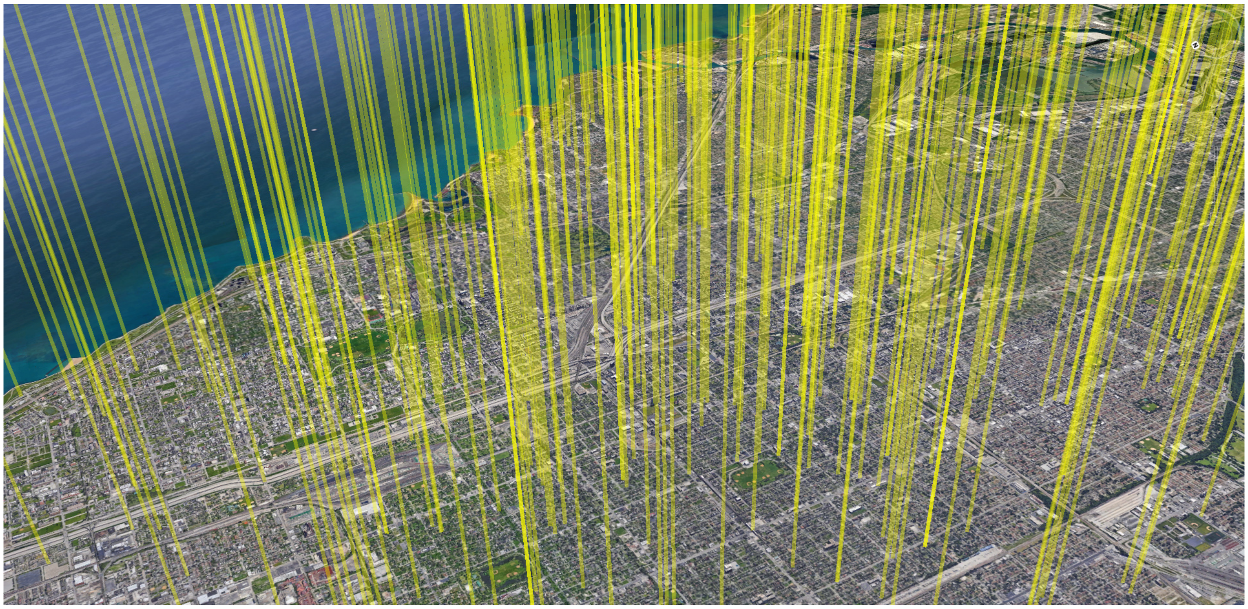
For example, to show the flight delays in the previous section, it is possible to add pillars at the destination coordinates with a height proportional to the flight delay. Where there are multiple flights to a destination, adding spheres to the tops of the multiple pillars will separate the delay information in a color-coded stack of spheres. Below is a Google Earth plot of the ground-based connections KML file from the previous section along with a new KML file of a pillar plot with latitudes and longitudes of each destination and a height that is a multiple of the arrival delay (the Google Earth Settings for altitude multiplier and offset are applied to the height in a pillar plot unless a fixed height is selected). Range brushing was first performed on the arrival delay axis, so the higher, red spheres represent flights with the longer delays. Setting a lower sphere opacity would better show overlapping spheres—however, the height could have been assigned to a different variable, such as the number of times a sphere color occurs for a destination, and then each colored sphere would sit at the height for the count of that color. Individual KML files can be turned on or off in Google Earth in the left side panel, and in fact displaying only the pillars makes a more readable graphic if the paths are not important.



Below is a pillar plot of Chicago homicide locations in 2016 through Christmas, where the data is brushed blue for the incidents where an arrest was made. This plot was made by choosing in the Pillar Plot window a fixed pillar height of zero and choosing to add a sphere on top of each pillar of size 2 and maximum opacity of 16.



Here is another pillar plot of the same data, where all the lines in the PC Plot were brushed yellow. In the Pillar Plot window here, the pillar width was increased and the opacity of the pillars decreased. The pillar height was again selected to be fixed rather than a variable value, but the height was set very high. The option to add spheres to the tops of the pillars was not chosen, but that would not make a difference since Google Earth was zoomed in enough that the tops of the pillars are not visible. Given the nature of the crimes, the ascending translucent pillars are evocative, I think, of the loss that these communities faced.



Again, the **DataTools→Merge KML Files** menu option can be used to combine multiple KML plots into a single KML file for convenience, and timestamps, pop-ups and labels are supported as described in the next section.

Google Earth Timestamps, Pop-ups and Labels

When creating Google Earth plots, options are shown to add timestamps, pop-ups and labels. These provide interactivity to the plots and to associate data with locations in the plots. All information is embedded in the generated KML file, so Sliver does not have to run in order to provide the user with an interactive Google Earth experience. Detailed information on selecting these options is provided in Section 3 *Sliver Detailed Operations*.

Timestamps

A timestamp is a value added to every plotted point that specifies a time of day. When timestamps are present in a KML file opened in Google Earth, a small player control panel automatically appears in the upper left corner of the Google Earth window, with endpoint times spanning all timestamps in the file (or multiple files if more than one is opened). Sliding the main cursor across the time scale shows the plot components at the time corresponding to their timestamp. The player also includes a second cursor to set an interval of time relative to the main cursor, and this can be adjusted to only show plot elements within the span of time between the two cursors, perhaps only one element at a time. The player controls can be used to animate the plot, only showing plot elements as the slider is manually positioned or automatically played. The automatic animation speed, which may be too fast at its default setting, can be adjusted by clicking on the wrench icon in the player control panel. A zoom button in the player controls will reduce the range of the slider and also slow down the play.

Timestamps can be applied to Path, Connection and Pillar plots. The timestamps at the points can be chosen to be the row number as seconds since midnight (simply to provide a timespan), or the values of any variable at the points as seconds. If a variable time-formatted in hh:mm:ss is selected, it is converted to seconds so the player will then show the associated time. The timestamps will roll over into new days if the number of seconds is large enough. Note that the *Convert Date/Time to Numeric* function in DataTools will convert all kinds of date and time formats to either seconds since midnight (for time intervals or times of day) or to decimal days since the start of the first date in the file.

The timestamp feature is useful to provide animation in Google Earth without running Sliver at all. It is useful to show events in the order that they happen, or to reduce clutter in complicated plots by only showing a particular range of the plot at a time. It is also possible to create one plot without timestamps and then create another one with timestamps and a short interval in order to animate motion along a visible route.

Popups

In Pillar and Connection plots, and whenever spheres are added to a Path plot, a window provides an option to display the corresponding row values of selected variables in a pop-up when an element is left-clicked in Google Earth (these are also called *balloons*). In a Path plot the spheres on top of any supports must be clicked, but Pillar spheres and bases as well as Connection lines can be clicked. The variables names and values are displayed in tabular form in the pop-up. This is a great way to see the data at any point in the Google Earth displays.

It may happen that a pop-up pointer may be a bit offset from the selected point, or that a scrollbar appears in a truncated pop-up. In either case, zooming in or out by one notch (one roll of the mouse) will correct this.

Labels

Whenever spheres are added to a Pillar or Path plot (not supported in a Connections plot), the window above will also provide an option to display the corresponding row values of any selected variables in a label when a mouse-over of a sphere is performed. These are displayed as a single row of values separated by commas, without the variable names. This is very useful for fast analysis or in finding the location of a particular value in a plot.

Displaying Strings in Pop-ups

It is possible to display a mapped string rather than a numerical value for selected variables in pop-ups. This uses a string mapping file whose path is entered in the Plot Settings, the same file used when selecting the **Plots→Display**

Values on Mouse-Over→With String Mapping menu option described earlier. If enabled, values for each variable are compared to a list of matching numbers and strings for that variable in the file, and if a match occurs the string is displayed instead of the number. This file has the format of the output string mapping file of the *Replace Strings with Numbers* function of the DataTools menu, although the string mapping file can also be created by hand (see Appendix D for the format).

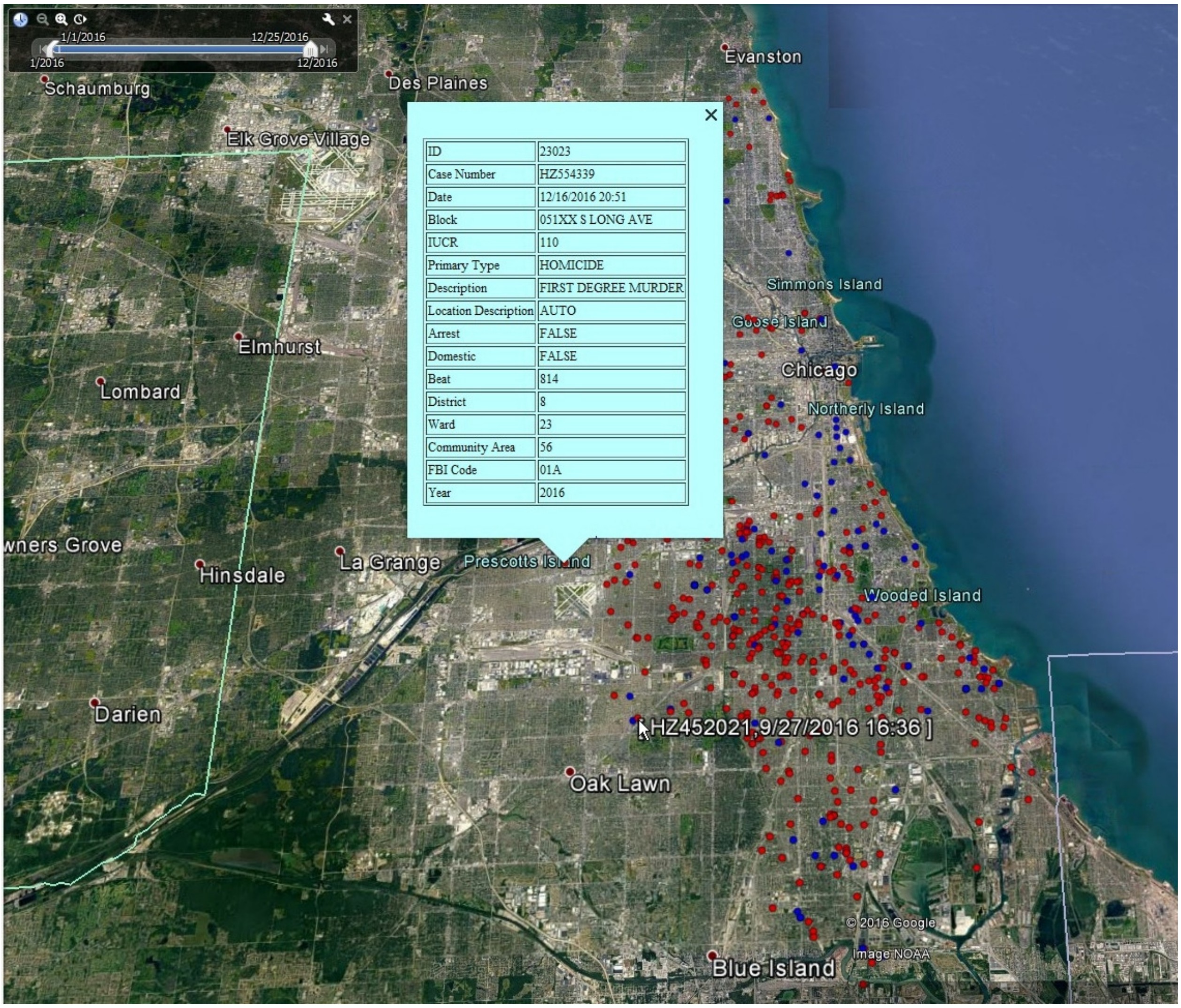
Below is the same Google Earth Pillar Plot as earlier that has timestamps, pop-ups and labels applied. The plot shows Chicago homicides in 2016 through Christmas. The data provided by the city includes strings, so the *Replace Strings with Numbers* function in the DataTools menu was used first to map any strings to unique numbers in those variables. The string mapping file that was created as part of this function was then entered in the **Options→Plot Settings** window, and the option to map numbers to strings was checked off when creating the plot. The date field was converted to seconds since the first event in the file using the *Convert Date/Time to Numeric* function in the DataTools menu and saved to a new variable, and this variable was selected to provide the timestamp in seconds. The start date for the timestamp is an entry in the window for the timestamp, and was set to 1/1/2016.

All three features are seen in this screenshot. The timestamp player is shown in the upper left corner, and it can be seen that the date converted to seconds provides the full range from 1/1/2016 to 1/25/2016. The cursors span the entire range, so all points are shown. The left cursor can be slid to the right to close the interval of visibility. The Play button is the rightmost tiny button at the top, with buttons to zoom in and out of a play range to the left of it. The animation speed can be adjusted by clicking on the wrench icon in the upper right.

Left-clicking on a point brings up the cyan pop-up with the values for that point for all variables that were selected (the latitude, longitude, and date converted to seconds were not selected for this display). The string mapping filename was entered into the Plot Settings and the box was checked to map numbers to any corresponding strings for the displayed variables, so variable values were mapped back to their original strings.

The mouse cursor in the lower right is hovering over another point, which then shows a label next to it. Labels span only one line and show only values, not variable names. Here only the Case Number and Date were selected to be included in the label. The size of the label can be adjusted in Tools→Options→3D View in Google Earth.

NOTE: Anyone with this KML file is able to fully operate these controls and displays—there is no need to have Sliver. However, the *dot.png* file added to the directory of the KML file must be moved along with any moves of the KML file, as it provides the graphic for the spheres.


1/1/2016
12/25/2016
1/2016
12/2016
Schaumburg
Des Plaines
Evanston
Elk Grove Village
ID 23023
Case Number HZ554339
Date 12/16/2016 20:51
Block 051XX S LONG AVE
IUCR 110
Primary Type HOMICIDE
Description FIRST DEGREE MURDER
Location Description AUTO
Arrest FALSE
Domestic FALSE
Beat 814
District 8
Ward 23
Community Area 56
FBI Code 01A
Year 2016
Simmons Island
Goose Island
Chicago
Northerly Island
Elmhurst
Lombard
wners Grove
Hinsdale
La Grange
Prescotts Island
Wooded Island
[HZ452021,9/27/2016 16:36]
Darien
Oak Lawn
© 2016 Google
Image NOAA
Blue Island

Animation

Sliver provides three modes of animation (motion graphics) for visualizing data:

1. **Grand Tour** – the capability to rotate *n*-dimensional data in *n*-space and project the result onto the 2-dimensional plane of the computer monitor. This dynamic mashing of data can provide visual identification of clusters of points, outliers, nonlinear relationships, and low-dimensional substructures that may not be apparent in individual two-variable or three-variable scatterplots. Color brushing is again fully synced with the other plots. This functionality is similar to that of the CrystalVision and GGobi data visualization tools listed in Appendix A. References describing the use of the Grand Tour are included in Appendix B.

2. **Count-Based Animation** – the capability to "play" the rows of data and animate the plots, including animating Google Earth plots and syncing the played data with external video player software or other software applications. This real-time correlation is particularly useful when analyzing data associated with flight video or other instrumented data. Often errors can only be detected visually in flight video, for example. Here flight video can be played or jumped forward or backward and all Sliver plots, including any Google Earth path plot of the aircraft, will follow the video, allowing visual situations to be matched with the data at that time.

   A preset selection is provided in the animation menu for playing data with the Sliver Data Player. However, any set of parameters can be entered to interact with other software or instrumentation as described in the Count-Based Animation section below.

3. **Real-Time Data** – the capability to gather and display data as it is written to a data file by an external script or instrumentation. This is useful when monitoring of data is desired over a period of time. The animation can be stopped for data analysis using Sliver functions; when restarted, it continues at the point in the data file where it left off.

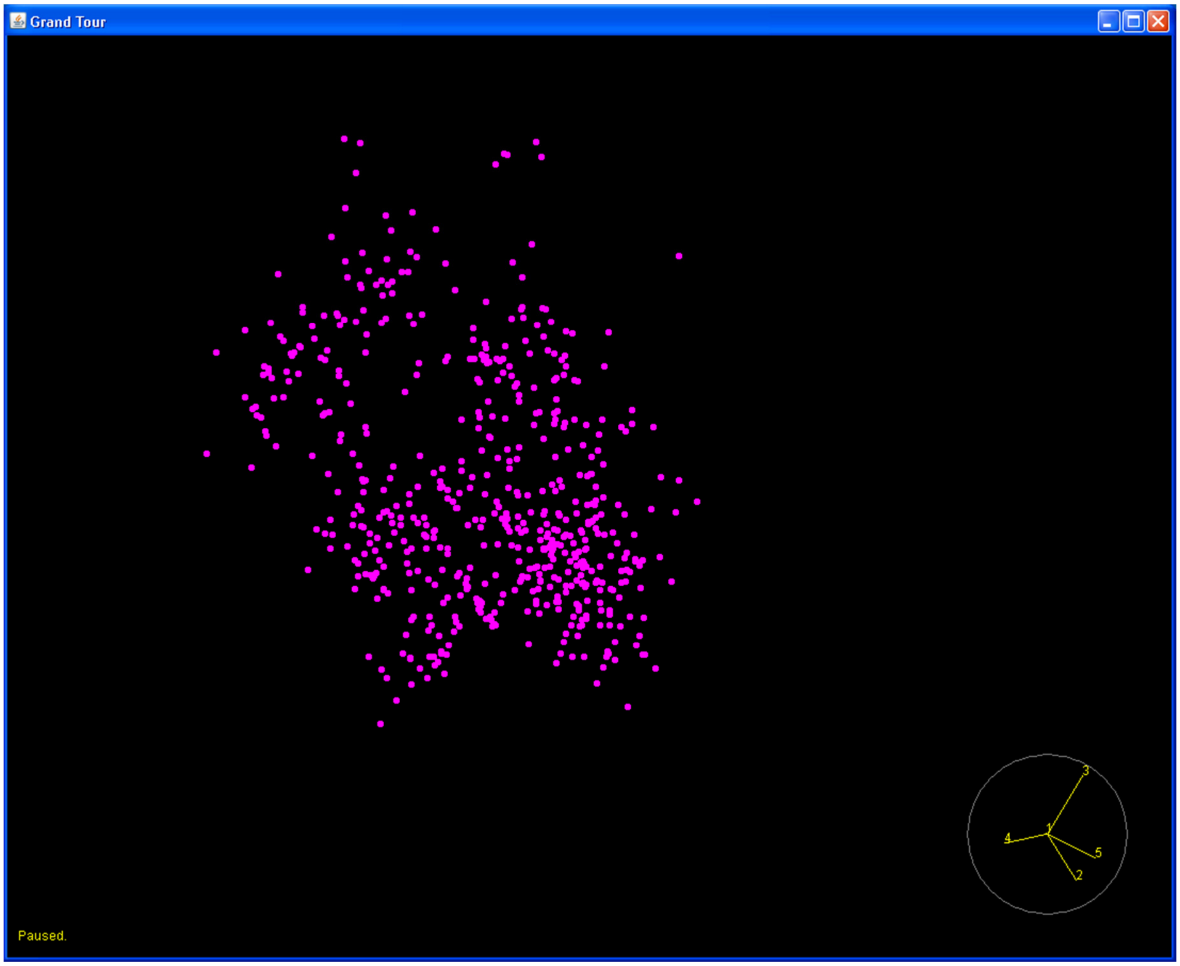
The Grand Tour

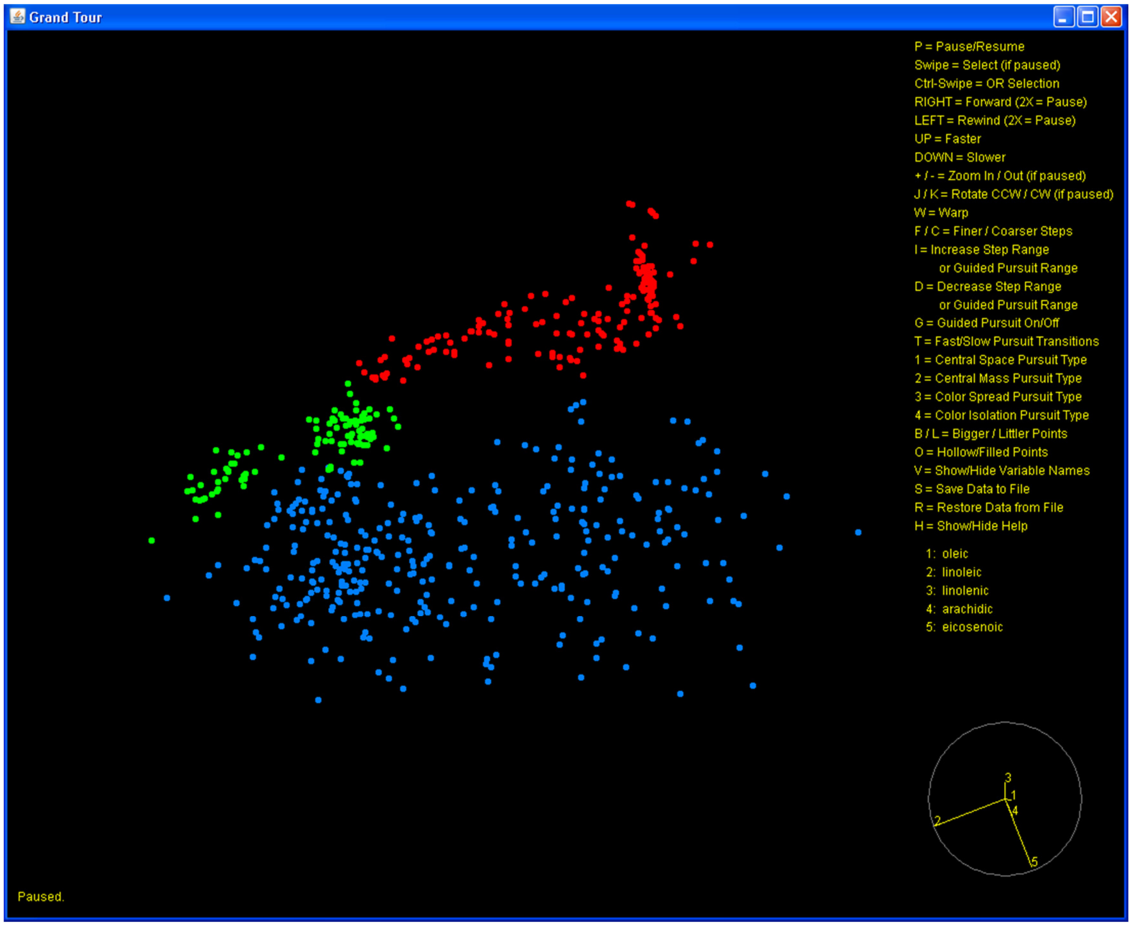
Between $n = 3$ and $n = 9$ variables can be assigned to *n* dimensions in the Grand Tour animation when the **Animate→Grand Tour** menu option is invoked. The *n* orthogonal axes rotate randomly in the Grand Tour window at a speed and direction controlled by the user. The scatterplot points shown on the display are the 2D projections of the locations of the points in *n*-space. In effect, the functions along each axis become orthogonal, linear combinations of all the variables. The dynamic behavior of these points, e.g., clusters of points moving as a group or structured along an axes, reveal data correlations. The animation can be paused for selecting individual or groups of points by brushing, and brushings among the Grand Tour window and other plots are fully synced.

Two sample views of the same data in the Grand Tour window are provided below. The point size was increased for better viewing here, but resolutions down to 1x1 pixels as well as zooming are available for larger data sets. Three clusters of data were observed to follow closely in their movement as the plot rotated in 5-dimensional space. The animation was paused and the mouse was used to swipe one of these groups. This selection was automatically propagated to the PC Plot and other 2D scatterplots. The main PC Plot Brush menu was used to assign the selection to a color brush, which propagated back to the Grand Tour display. This was repeated for the other two groups, resulting in the second image below. This second window also displays Help information on the various mouse and keypress operations available in the Grand Tour.

The circular display in the lower right shows the relative contributions (projections) of each axis onto the plane of the display. This is very useful to appreciate the effects of the variables on the distributions of points. In the first image variable 3 (shown as the variable linolenic in the text in the second image) provides the greatest contribution in the upper right direction, while variable 1 (oleic) has virtually no effect on that distribution. In the second image it can be seen that variables 2 and 5 (linoleic and eicosenoic) are the most significant contributors to the separated distribution, with eicosenoic providing the separation of the blue group from the others and linoleic instrumental in separating the green from the red groupings. This is useful in assigning color brush filters in the main PC Plot. The overall transformation (rotation) matrix can be saved to a file for more quantitative analysis if needed.

As seen in the Help text in the second image, automated pursuits (in a random walk manner) are also provided that optimize the central space (to help separate point groups), central mass, separation of colors, or separation of the color that can be most isolated. This is discussed in greater detail in a later section.

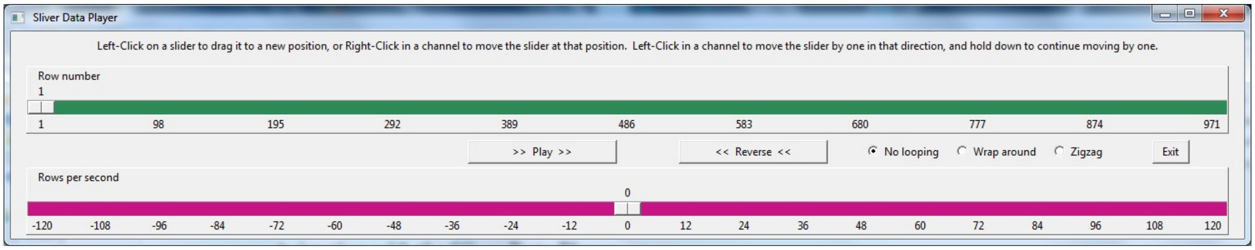



Grand Tour
P = Pause/Resume
Swipe = Select (if paused)
Ctrl-Swipe = OR Selection
RIGHT = Forward (2X = Pause)
LEFT = Rewind (2X = Pause)
UP = Faster
DOWN = Slower
+ / - = Zoom In / Out (if paused)
J / K = Rotate CCW / CW (if paused)
W = Warp
F / C = Finer / Coarser Steps
I = Increase Step Range
or Guided Pursuit Range
D = Decrease Step Range
or Guided Pursuit Range
G = Guided Pursuit On/Off
T = Fast/Slow Pursuit Transitions
1 = Central Space Pursuit Type
2 = Central Mass Pursuit Type
3 = Color Spread Pursuit Type
4 = Color Isolation Pursuit Type
B / L = Bigger / Littler Points
O = Hollow/Filled Points
V = Show/Hide Variable Names
S = Save Data to File
R = Restore Data from File
H = Show/Hide Help
1: oleic
2: linoleic
3: linolenic
4: arachidic
5: eicosenoic
Paused.

Count-Based Animation

Animation with the Sliver Data Player

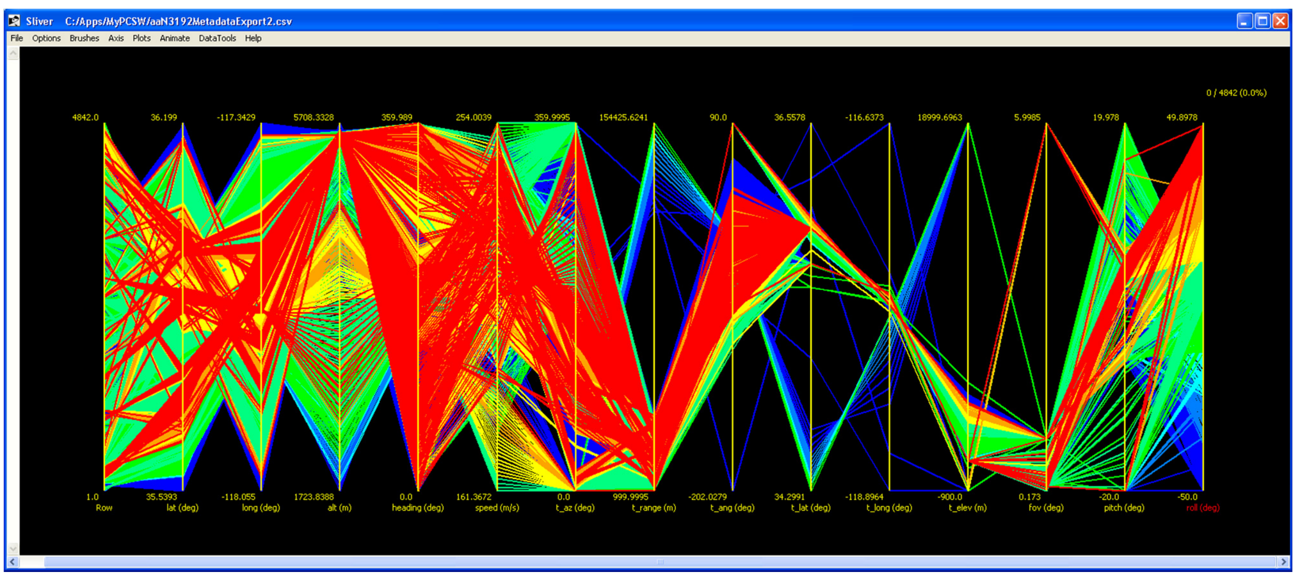
Sliver has a built-in player for data animation that sequences rows in the data file and animates the plots and optionally an external program such as a video player. When this animation mode is invoked with the **Animate→Count-Based Animation** menu option, the player interface appears:



The top green scale shows the current data row number of the total data rows loaded into Sliver. The tab can be dragged to the right to any row value, or the green bar can be left-clicked to advance the row, or the green bar can be right-clicked to immediately jump to a row. The bottom red row provides the rows per second to play, either forwards or backwards, and the Play/Pause button starts and stops the player. The maximum rows per second to set in this interface is entered in the first Count-Based Animation window. Looping or zigzagging of the data is selected with the radiobuttons in the right center of the player window.

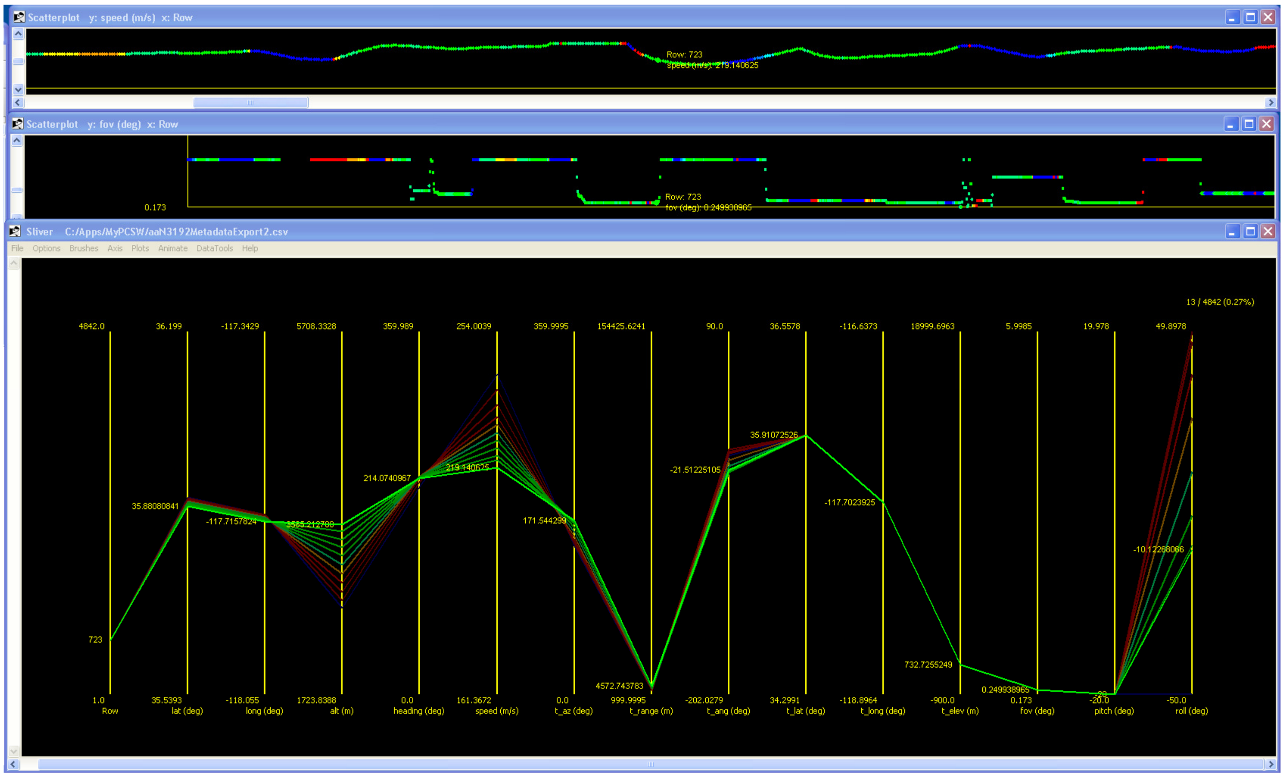
Let's begin an animation. The data file consists of small airplane data. We want to show scatterplots of variables against time, so we need to add a column with a variable for the row number. We first select the **DataTools→Add Column for Row Count** menu option described later in this manual to add the Row variable as the first column.

Then the data file is loaded into Sliver, producing the parallel coordinate plot with *Row* as the leftmost axis. The rightmost axis is selected with an Alt-Click, shown as the red name below. Then the range brush option is invoked to automatically color the lines from minimum to maximum values on that axis with 8 colors ranging from blue to red. The corners of the plot are then dragged to resize the plot to fill the width of the display.



Now two scatterplots are created using a menu function, plotting two variables vs. Row. Since the rows represent samples in time, these plots show these variables over time. The plots are resized to thin strips and placed above the parallel coordinate plot. They are then zoomed and panned horizontally and vertically (in different amounts) to best view the individual curves.

At this point the Sliver Data Player is invoked with the **Animate→Count-Based Animation** menu option, where we accept the defaults (or click on the Sliver Data Player preset which reloads the defaults) and do not check the option to animate Google Earth. The animation is then started. The PC Plot is reduced to a bright line representing the current data row values. A configurable number of trailing fading lines show previous lines to highlight variables that are rapidly changing. All the lines are shown in their colors from range brushing the full PC Plot. The two scatterplots highlight their data points for the current value of time (row number). They display their axis values if configured to do so in the window that appears when the animation is launched, and the plots roll along to maintain the current position in the center of the window unless they are very near the start or end. The sliders can be moved with the mouse to view the curves at past and future times, and when released the animation continues.



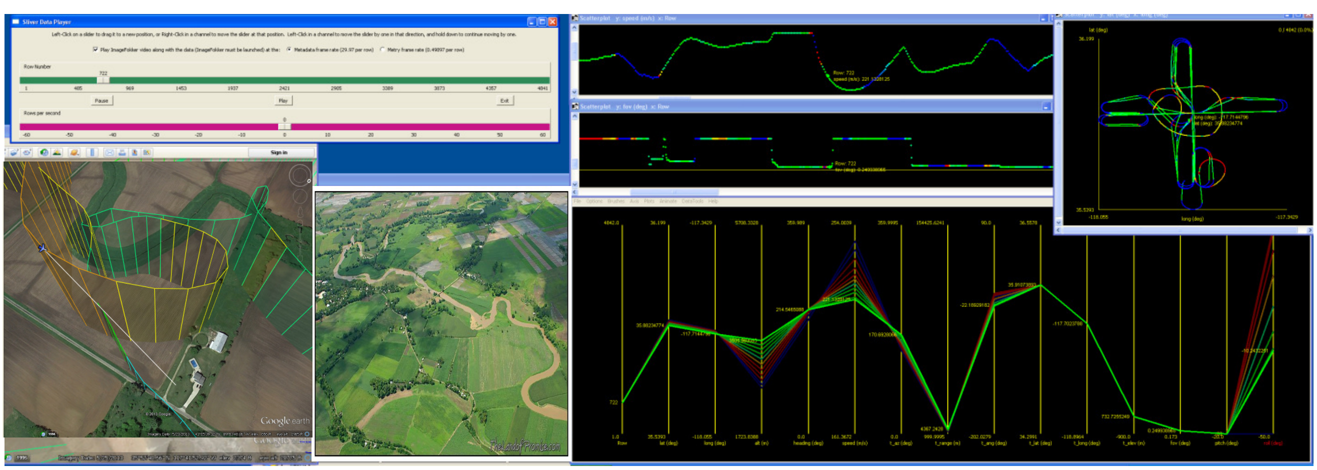
Let's stop the animation with **Animate→Stop Animation**. All plots return to their previous state. Now we create a Google Earth path plot using the Lat, Long and Alt variables. Then we again select **Animate→Count-Based Animation** and choose to also animate Google Earth and select the aircraft icon and the option to animate a line to a secondary set of coordinates (but not to color that line per another variable). The animation is restarted, and now the Google Earth path plot is also animated, showing an aircraft at the current location on the flight path pointed to its heading. An optional line from the main icon to a second set of selected coordinates (t_lat, t_long and t_elev), is also displayed. The Animate window provides another option to color this line according to the value of a different variable (white if zero, range brushed from blue to red over its range otherwise).

NOTE: If Google Earth stops updating during the animation, right-click on the "Position" label in the left pane of Google Earth and select **Refresh**. Plotting less points in Google Earth reduces the chances that it will stop updating.

There is also an option in the Animate window to move the Google Earth camera view to follow the main icon. The range, tilt angle and relative heading of the camera from the icon as it moves can be specified. If the option to draw a line from the icon to a secondary set of coordinates is enabled, the relative heading can also be switched in sign to always be opposite that line for better viewing of it, or the heading can always be along that line.

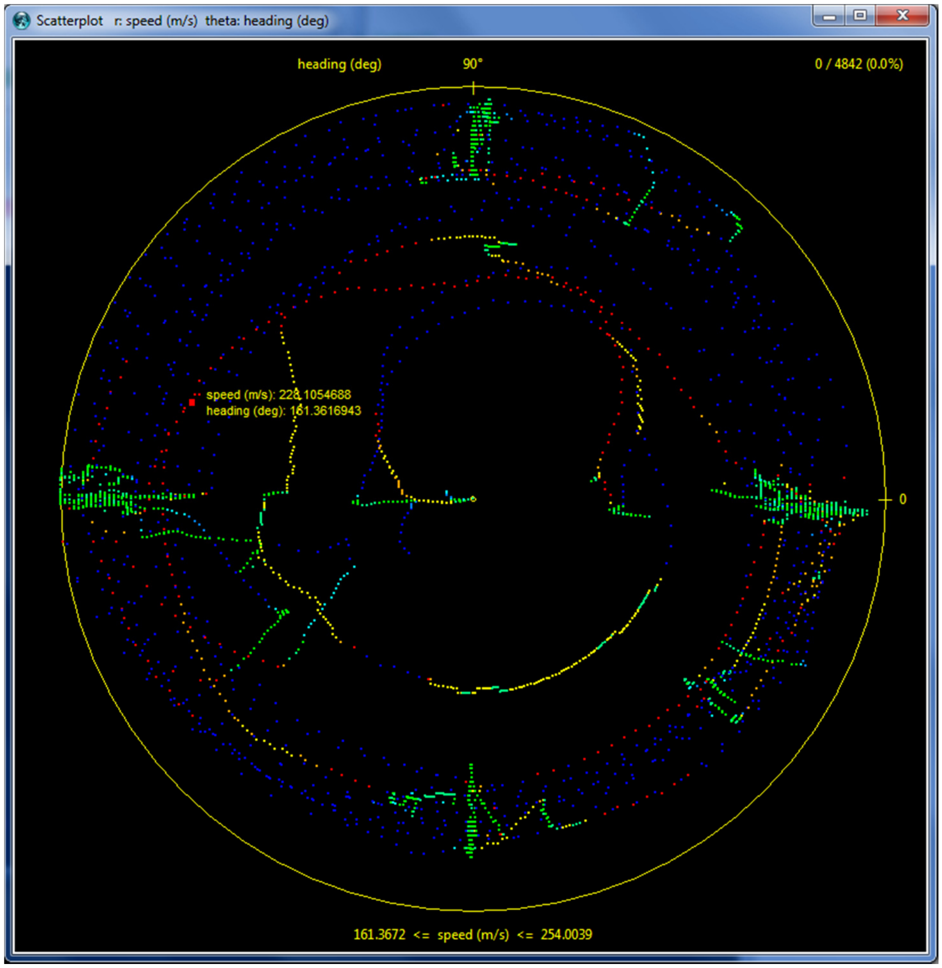
We might also launch a video player and load a video from the flight. If the frame of the video can be read or written, the video can be played in sync with all the rest of the data (see the next section). (The example flight path image here was captured during a different animation, so the coloring does not match the color brushing in the PC Plot above.)

All variable values are shown visually and numerically on the plots as the play occurs. An entire display spanning two monitors is shown below, demonstrating the data visualization possible using the animation feature of Sliver. The position in the flight path is represented here by a scatterplot of latitude vs. longitude as well as in the Google Earth path plot. More windows can be inserted by reducing the sizes of the existing ones, and in fact the scatterplot dots and parallel coordinate lines are set larger than necessary here so they are more visible when printed.



Two or more instances of Sliver can run in a common Count-Based Animation. A single Sliver DataPlayer window will serve to control all instances (creating a new player in any instance will replace the one in the previous instance), so it is possible, for example, to animate multiple paths in Google Earth simultaneously with their own icons. The animation is based on row number, so you will want to make sure the data used in all Sliver instances starts at the same time of day, for example, and have the same data rate. Note that the same data file can be read into multiple Sliver instances and different variables selected for the animation.

The animated 2D scatterplots shown above are plotted in Cartesian coordinates. However, a polar plot can provide a unique and very useful plot for animation. An example is shown on the right, where the speed is plotted as the radial distance *r* and the heading is plotted as θ. Actually, this data might be better plotted in the optional navigational format, where North lies at the top, East at the right, and θ is plotted eastward from north as the azimuth.



Count-Based Animation with Other Software Applications or Instrumentation

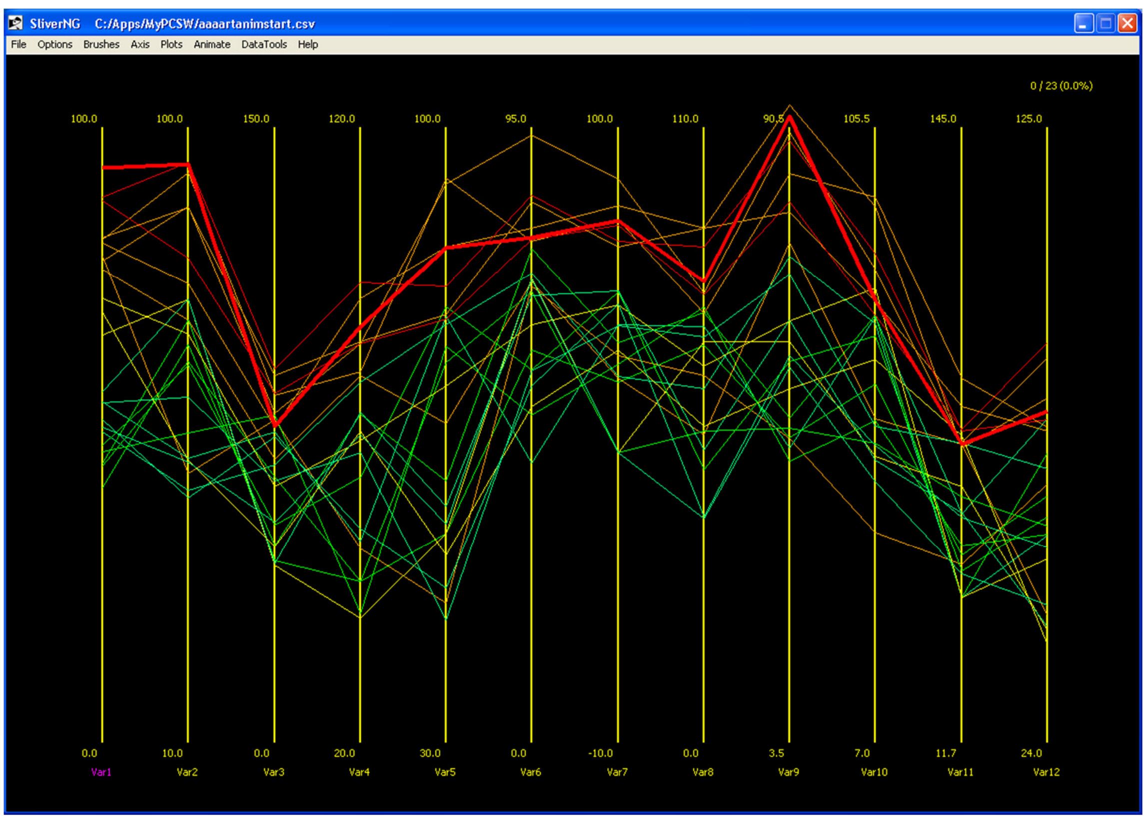
The animation of the data is based simply on whatever number is currently in the last row of a user-named text file. Syncing the data from an external program only requires a script to be written that continuously appends numbers to the end of the file based on the video frame or some other characteristic of the external program. Conversely, animating an external program can be done using the internal Sliver Data Player interface, which itself writes current row numbers to the end of a text file. The conversion of this number to the current data row is user-defined in the animation settings, including the data row offset, a multiplier to apply, the file to read the number from, and the script to launch. The file containing the count numbers is deleted between Sliver sessions to limit its size.

Real-Time Data Animation

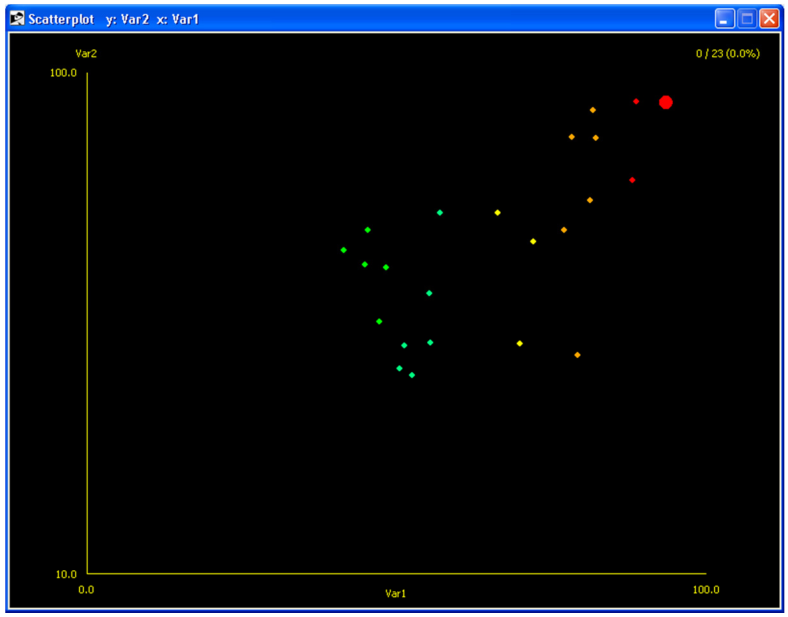
Sliver also provides real-time input and display of data in the PC Plot and 2D scatterplots. This is useful for instrumentation and monitoring operations. There is an option to display all the data or only retain the most recent n data entries (to limit clutter), and to either restore all the undisplayed data or delete it when the animation is stopped.

The interface is very simple. A file is created that has three rows, the first being the header row with the variable names and the other two rows consisting of the maximum and minimum expected values for the variables. This file is then loaded into Sliver as an ordinary data file, producing a PC Plot with scaled axes and lines across the top and bottom of the scales (maximums and minimums) corresponding to the two rows of data. The real-time data animation mode is entered through the **Animate→Real-Time Data Animation** menu option. After specifying the options listed in the previous paragraph, Sliver deletes the data from the two input rows and continuously monitors the same data file. Whenever an external script appends a new data line to the file, the PC Plot and all 2D scatterplots are populated with the data.

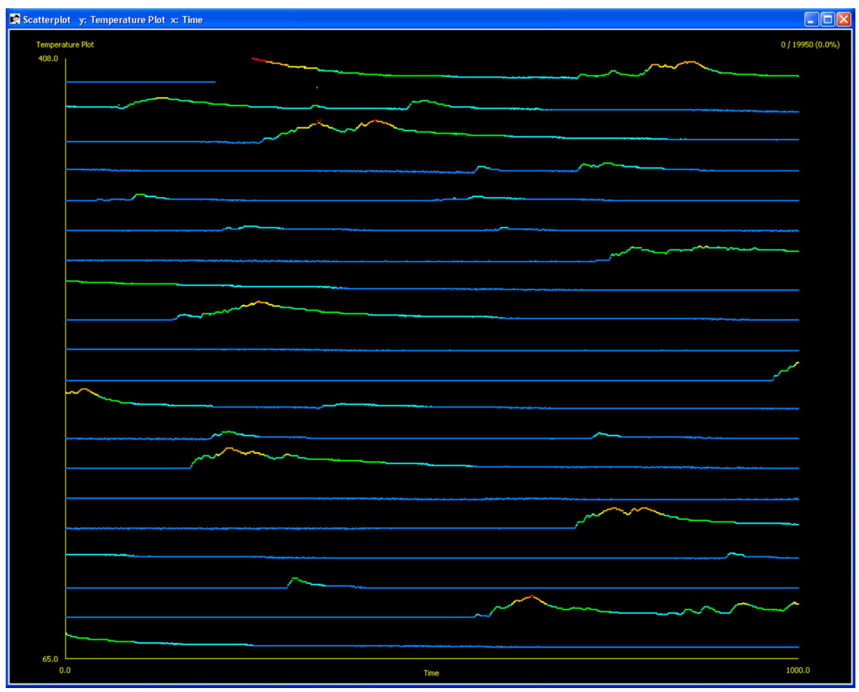
The real-time animation is exited through the **Animate→Stop Animation** menu option, and if the option to restore undisplayed data was selected all data lines are shown. At this point all functions of Sliver are available to act on the data. Restarting this animation will continue data collection at the point in the input data file where it left off—if data was written while the animation was stopped the display will very quickly update as it inputs all the new lines and catches up with the current data. A message appears whenever the display lags the input data.
Range brushing of an axis can be performed before the real-time animate mode is entered. The two data lines (maximums and minimums) that were initially entered are then colored, a good indication that the range brushing is set. Thereafter, input data is assigned to the appropriate brush color, and the PC Plot lines and 2D scatterplot points are drawn accordingly. This color brushing is useful to quickly detecting extreme data as it occurs.



The number of data points that can be input is limited only by the computer memory, but a million lines is quite reasonable. However, in this case all the data should not be restored when stopped, as Sliver cannot effectively operate on that many data rows. Retaining the last 100,000 data points is very reasonable, where perhaps 5X more points is feasible if filtering the data is performed as one of the first operations. (Because of the slow interaction for very large datasets, it is best when filtering these to choose the setting in the Options→Plot Settings menu option to click on corners of a selection rectangle rather than swiping the selected points with the mouse.)



Another example of real-time data recording is shown below. An Arduino microcontroller was used to collect temperature data from a temperature sensor connected to an analog input pin. It then processed the data and provided three data values per sample separated by commas: an x-value as the sample number mod 1000, a y-value scaled to fit 20 rows of plots of the temperature, and the actual temperature value itself in degrees Fahrenheit. Sliver was put into real-time data recording mode, where the first line of a CSV file held the variable names Time and Temperature and the next two lines were populated with min/max values of x (0 and 1000) and y (65 and 65+3 +20*17 where each plot has a range of 17 degrees between 68 and 75 except for an offset of 3 at the very bottom). The actual temperature variable was range-brushed in the PC Plot. Then the maximum number of displayed points was set to 19950 to support overwriting on wrap-around, and a 2D scatterplot was created to plot the incoming values.



A Python script monitored the serial port and appended each line to the end of the CSV file. The Arduino wrapped the plot back to the top after 20,000 samples, where the 50-point disconnect can be seen. When stopped the data was available for immediate analysis as well as later analysis by loading the CSV file.

When a large number of input data points are retained in the plots, the display can lag the data. In this event, a warning message is displayed along the top of the main PC Plot window indicating that the display is lagging the input. Since the main PC Plot was not used here, that window was minimized, leaving only the scatterplot window visible. The PC Plot is not updated while minimized, so this allowed the data rate to increase. Whenever the PC Plot is un-minimized, it is automatically updated with the latest data.

Contact me if you would like the Arduino and Python code for this example.

3. Sliver Detailed Operations

This section describes the means of creating and manipulating the data visualizations shown in the previous section. Creation is done by invoking menu functions, but the general use of the mouse and keyboard is discussed first as they are used across visualizations.

Note that selecting a set of lines is not the same as brushing a set of lines. Selections are given a selection color, but selections are temporary and are immediately replaced when a new selection is made unless an OR, AND or XOR Boolean operator is specified. Selections are used to quickly investigate correlations of data within a plot and across plots. The number of selected lines out of the total is provided in the upper right of the main window.

A selection can be saved (assigned) to a brush, however. The brush can be given any color, and these brushes are more permanent than selections. Brushes can be assigned, hidden, re-shown, re-colored and unbrushed again via the Brush menu. Selecting and then brushing lines that are already brushed will re-assign those lines to the new brush.

Selections and brushes are automatically updated across all plots except Google Earth plots, which require manual updating through creation of a new plot with the same filename.

Note: Certain functions such as inverting, deleting, moving, shifting and compressing an axis after the PC Plot has been zoomed (not simply resized) will first redraw the PC Plot to its original layout.

Mouse and Keyboard Controls

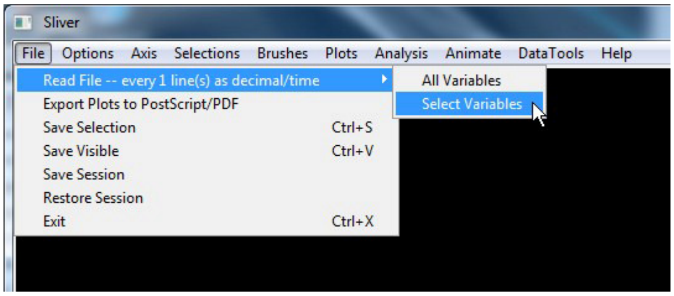
The mouse and keyboard are used for manipulating the visualizations after they are created by the menu functions.

- Use the mouse to either swipe across lines to select them (and color them the selection color) or to click the two corners of a selection box, depending on an option set in the Plot Settings. For large files, choose the setting to click for the selection rectangle corners instead of swiping, and press Enter to select or ESC to cancel. Pressing Ctrl-A selects all visible lines.
- Press Ctrl while selecting to OR a new selection with an existing one.
- Press Shift while selecting to AND a new selection with an existing one.
- Press Ctrl-Shift while selecting to XOR a new selection with an existing one.
- Press Alt while clicking on an axis name to select the axis, which will color the name with the selection color. Then invoke the Axis menu or press the i, x, m, r or g key to invert, delete, move, range brush, gap brush or shift lines along the axis. In a gap brush, the Up/Down arrow keys are used to increase or decrease the number of gaps and colors. Note that inverting, deleting, moving and shifting an axis are not allowed after a plot has been zoomed. To do these functions after a zoom has been done, save the session and reload it, or reload the original data file.
- To shift lines along a selected axis, press the Up/Down arrows to shift them up/down at the top, and press the Right/Left arrows to shift them up/down from the bottom, or use the Axis menu functions.
- Drag the corners or edges of a plot to resize it, or click on the box in the upper corner of the window to min/max it. Note that you have to move the mouse back into that window in order for the resizing to occur.
- Press Ctrl and roll the mouse scroll wheel, or press the = and – keys, to zoom a plot in and out.
- Press Ctrl-Alt and roll the mouse scroll wheel, or Alt plus the = or – keys, to zoom a plot horizontally.
- Press Ctrl-Shift and roll the mouse scroll wheel, or Shift plus the = or – keys, to zoom a plot vertically.
- Press S to resize plotted points to Plot Setting size after zooming to remove distortion or at any time to update the point size to a new setting. This will generally be automatically performed. If the point size setting was greater than 1 when the scatterplot was created, you can also press B or L for bigger or littler points, and the letter O to toggle between filled and hollow circles.
- Press K to lock (remain in the current state) or unlock a scatterplot window.
- Use the Up/Down/Left/Right arrow keys to pan a zoomed plot.
- Press H in the Grand Tour or Transparent PC Plot window to see additional mouse and key assignments.

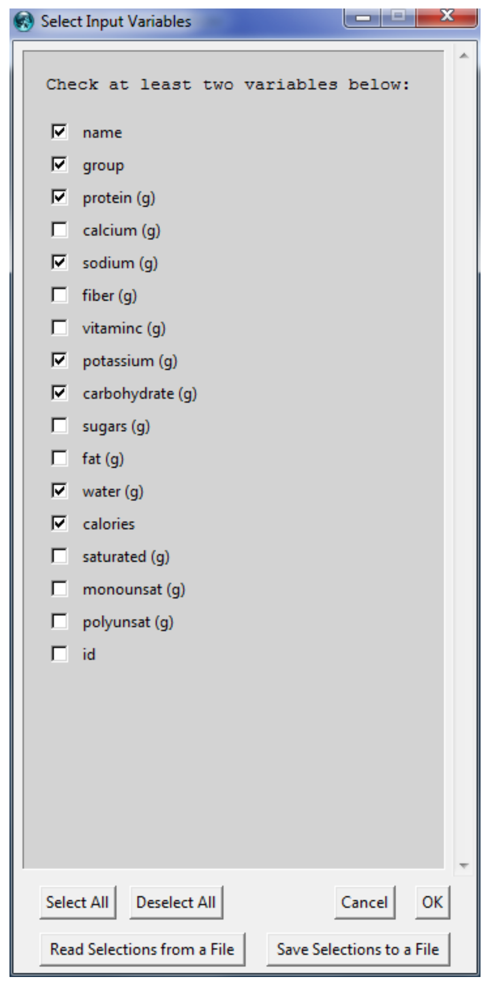
Menus

There are 10 main menu headings: File, Options, Axis, Selections, Brushes, Axis, Analysis, Animate, DataTools and Help. These menus are discussed in this section.

File Menu



The File menu provides the means to load CSV or TXT files into Sliver and to save the data for currently selected or visible lines to the same type of file. It also allows the export of visible lines to a Postscript/PDF file for viewing in an alpha-blended mode, and saving or restoring of the session.



Read File — Loads a CSV or TXT file into Sliver and creates the parallel coordinate plot in the main window. Options for decimating the input (reading in every *n*th line) and choosing a hexadecimal data format are provided in the **Options→Plot Settings** menu function, and their current values are echoed in this menu text for convenience. The decimal/time format includes any mix of decimal numbers and time data in hh:mm:ss or mm:ss format, where the seconds can be a decimal value. **Also, any variable with the string *(sec-time)* anywhere in its name will automatically be interpreted as having units of seconds and will be transformed into the hh:mm:ss time format in Sliver.** If the Select Variables option is clicked, a window is presented to check off the desired input variables as shown in the figure on the right. For convenience, the current selections can be saved to a file from this window, and these selections can be re-loaded when inputting a new file or using functions in the DataTools menu.

Export Plots to PostScript/PDF — Exports the PC Plot, PCM and/or scatterplots to a Postscript file. This produces an alpha-blended version (one with transparency) that reveals inner structure of the overplotted lines. It also provides presentation-quality vector graphics of the plots. If the path to Adobe Distiller executable file is input in the PS/PDF Settings, the Postscript file will be automatically converted to PDF and launched for viewing.

Save Selection — Saves the data rows (polylines) that are currently selected into a new data file. Lines are selected by mouse swipes (or by clicking on corners of a selection rectangle and hitting Enter if configured in the Plot Settings), and they appear in the color configured as Selected Line Color in the Color Scheme. This is a convenient way of quickly filter the data in the data file. Loading the saved file also rescales the axes to fit just the selected data, spreading out the scales to cover only the necessary range.

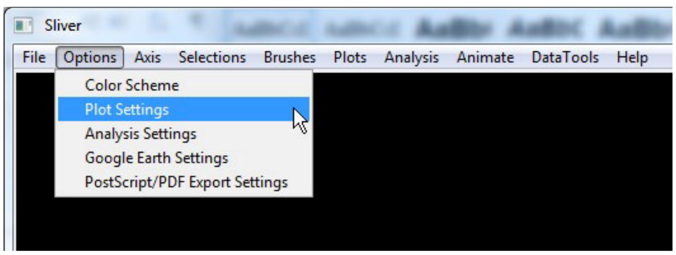
Save Visible — Saves all data rows (polylines) that are currently visible into a new data file. Lines are hidden by assigning them to one or more brushes and then hiding those brushes through the Brush menu.

Save Session — Saves the data, the current selection, the color brushing and the hidden/visible line states of the PC Plot to a user-named file.

Restore Session — Restores the data, current selection, color brushing and visible/hidden line states of the PC Plot from a saved session file.

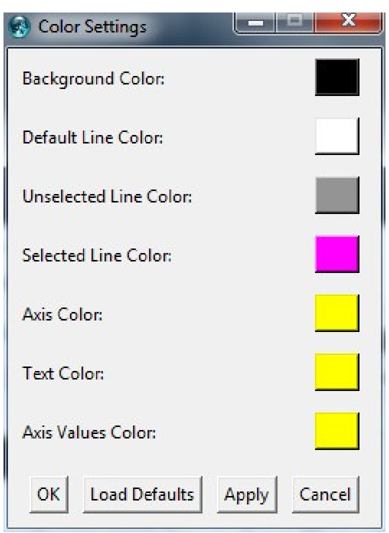
Exit — Exits Sliver. The "X" in the upper right of the main window works equally well.

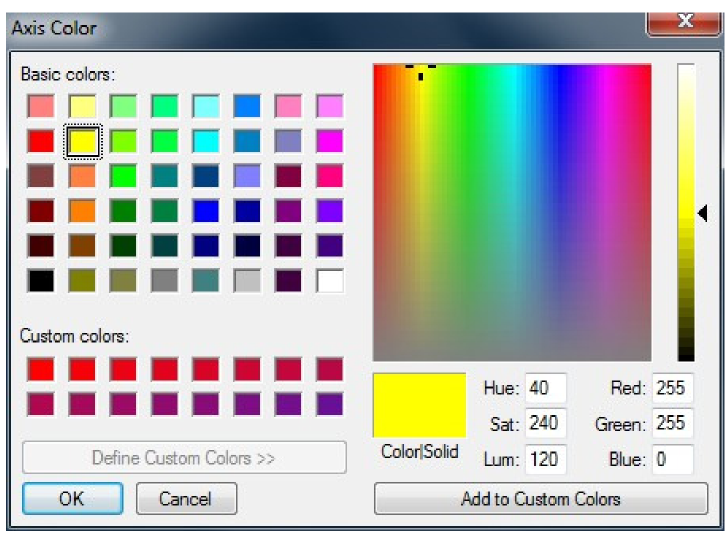
Options Menu



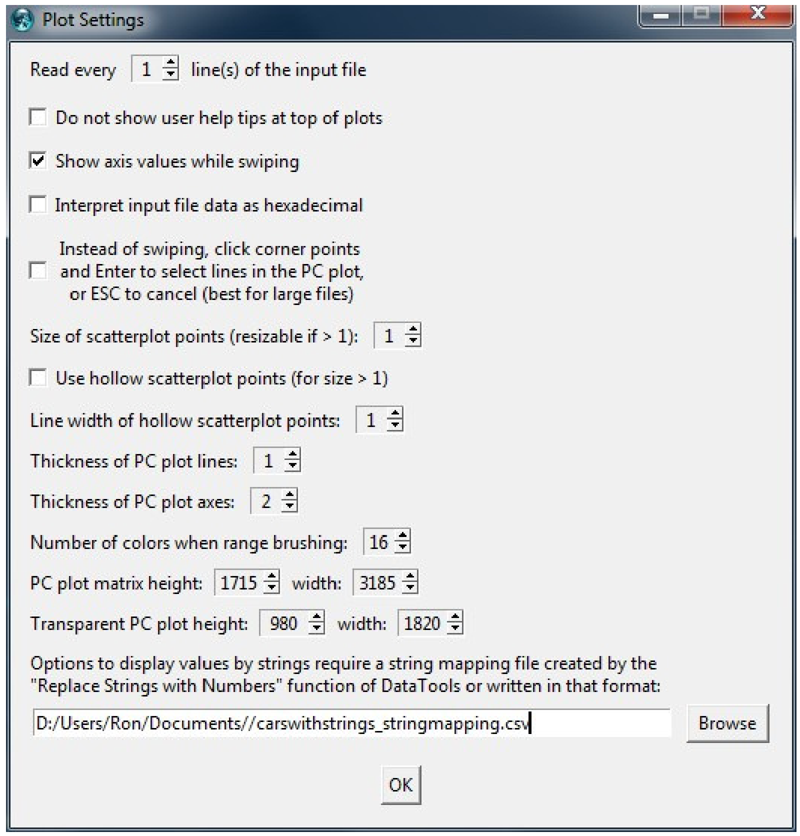
The Options menu provides pop-up windows to change Sliver settings for display colors, scatterplots, Google Earth plots, and exports to Postscript/PDF. Note that all settings are saved between sessions and restored when Sliver NG is relaunched.

Color Scheme — Provides a window to customize the color scheme of Sliver. Clicking on the assigned color in the main window brings up a Color Chooser window to change it.





Plot Settings — Provides a window to customize the various types of plots.



A very important option at the top is the decimation of the file when read in (reading every nth line), by default 1 here. This setting is echoed in the **File→Read File** menu option for convenient reference when loading files. Increase this number for better performance when reading in large files, say over 100,000 rows for a handful of variables or over 50,000 rows for 20 variables.

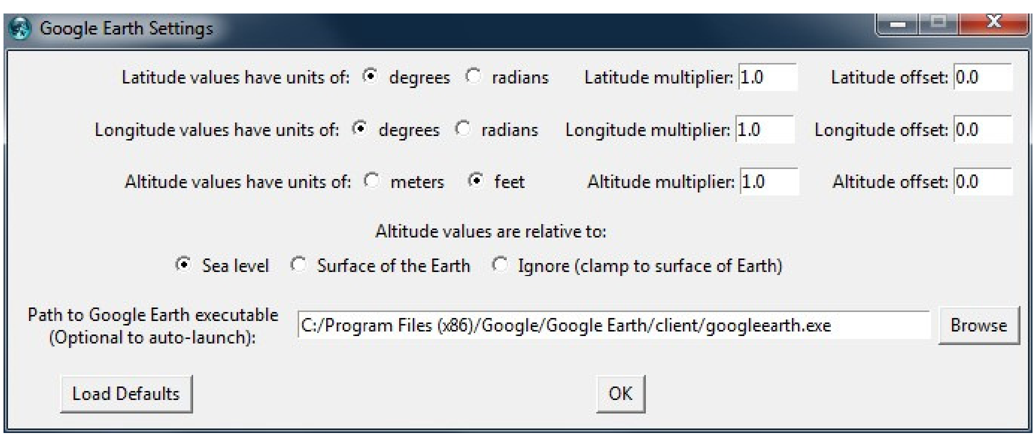
The sizes of the PC Plot Matrix and Transparent PC plot can be adjusted to best fit your desires and monitor size.

Note that scatterplot points cannot be resized with the B (bigger) or L (littler) keyboard keys, or toggled between hollow and filled with the O key, unless the size of the scatterplot points is greater than one when the plot is created.

The default thicknesses of the PC Plot lines and axes are 1 and 2, respectively, but can be changed here.

There is also a field at the bottom to browse and enter the path to a file to map variable values to strings when the **Plots→Display Values on Mouse-Over→With String Mapping** menu function is selected, or when the string mapping option is selected for values in Google Earth pop-ups.

Google Earth Settings — Provides a window to provide general Google Earth settings.



Postscript/PDF Export Settings — Provides a window to customize the export of the parallel coordinate plot to Postscript and PDF with alpha-blending (transparency). These settings are very useful in optimizing the output for the two main reasons for exporting to PS and PDF:

1. To obtain an "X-ray" view of the major line clusters (correlations) by assigning opacities to the lines of the PC plot, which are often obscured by the overplotting from even moderately-sized data sets. (The other action to take in this case is to range brush or gap brush an axis to color brush sets of lines, with the option to hide one or more brushes.) Setting an opacity value less than one for the scatterplot points also helps in seeing overlaid points.

2. To obtain presentation-quality printouts and graphics. For large datasets, there may not be sufficient memory for Adobe Acrobat to flatten the file to send it to a printer, and in this case it can be saved in Acrobat as a high-resolution (say, 600 dpi) PNG file, which can then be sent to the printer.

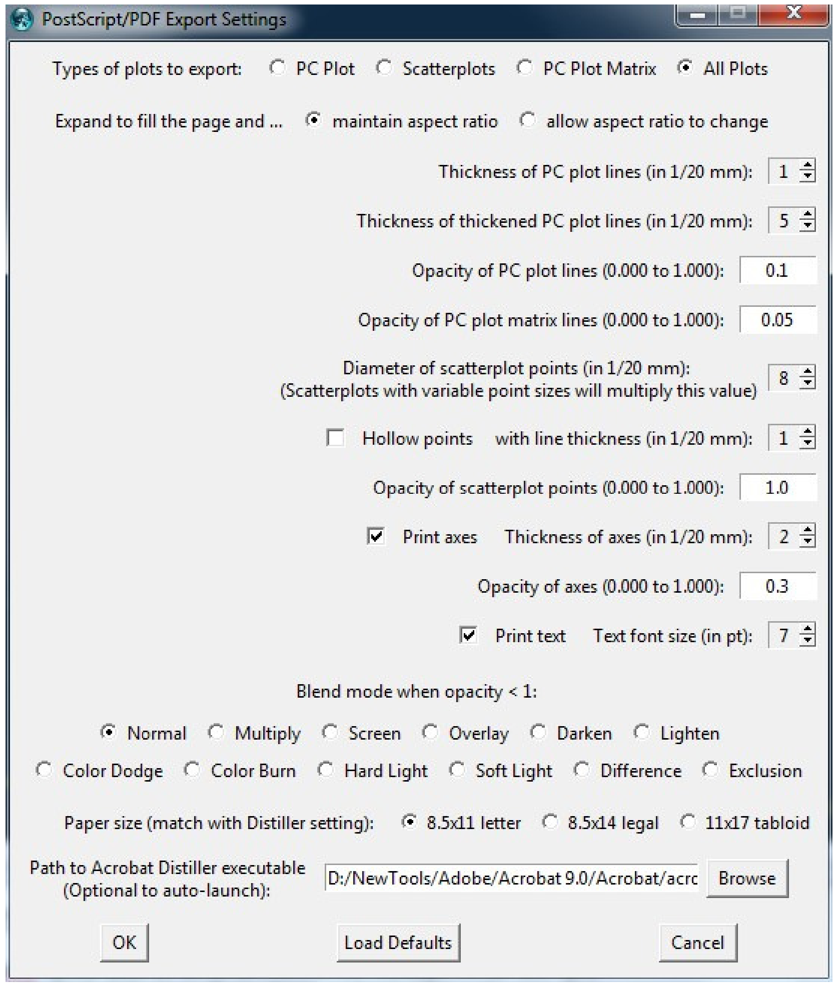
NOTE: ONE-TIME REQUIRED SETUP FOR PDF EXPORT:

The paper size and AllowTransparency option must be set in the Adobe PDF Settings. To do this, open Adobe Distiller and choose an option for Default Settings (Press Quality or High Quality Print is recommended). Then select the Distiller menu option **Settings → Edit Adobe PDF Settings**. In the main window, change the Default Paper Size to match the paper size selected in the Sliver Postscript/PDF Export Settings (in Landscape mode)—for example, for the "8.5x11 letter" setting in the screenshot shown here the Width would be set to 11.0 and the Height to 8.5 with Units of "Inches." Then click on the "Save As…" button, give your options file a new name and remember the folder where Distiller will save it, and then click Save. Then go to that folder, open the new options file, find the line with the phrase "AllowTransparency" near the top, and change the value from "false" to "true" and save the file. This is the settings file that should always be selected in Distiller.

You may also want to choose *Leave Color Unchanged* in the Adobe Distiller menu function of **Settings→Edit Adobe PDF Settings→Color→Color Management Polices** for the blackest backgrounds and boldest colors.

In lieu of Acrobat Distiller, it is possible to open the PostScript file with the free GSView for Windows software. GSView provides vector-drawn displays and printing but does not provide the transparency (the opacity will always be 1.0). However, this is highly preferable to screenshots when presentation-quality plot images are desired.

Working our way down the settings list, we first have the option to export just the PC Plot, just the (2D only) scatterplots, just the PCM, or all of them. The plots appear on separate pages in the PDF.



There is also an option to either maintain the aspect ratio of the current window, which may have been resized, or to expand the plots in both directions to fill the page. In either case, the entire page background is colored with the current background color of the plot.

The line and point settings for PS/PDF export are independent of the plot settings in Sliver due to the different characteristics of a monitor display. Here we can also set the thickness and opacity of the PC plot and the PCM lines. The scatterplot point diameters and opacities can also be set, and if hollow points are checked off the thickness of the outer line is also specified. If variable sized points are selected for a scatterplot, the size will multiply the diameter entered here so the exported plot also has variable sized points, although this may require some testing to achieve the desired sizes.

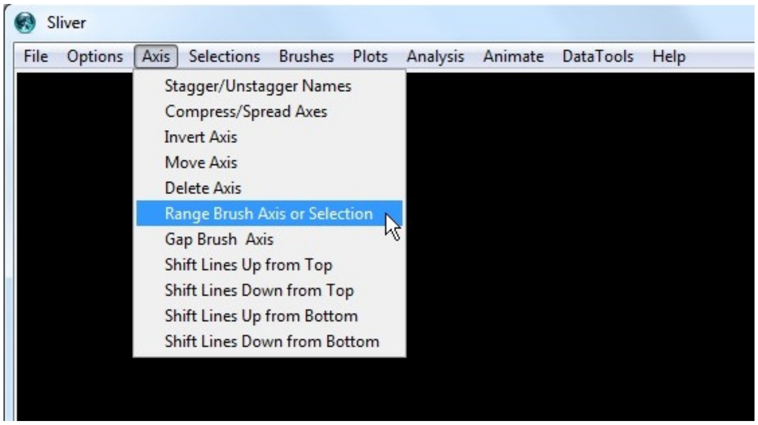
There is a choice to print or not print the axes, and their thickness and opacities if they are printed. There is also a choice to print or not print the text (axes names, etc.), and a setting for the font size.

There are 12 modes of alpha-blending available in Adobe Acrobat for the entered opacities. The Normal blend mode works quite well, but with different backgrounds or desired effects it may be beneficial to try out the different modes. These are the short descriptions of the modes listed in Adobe document PDF 32000-1:2008 (Document Management—Portable document format—Part 1: PDF 1.7), reproduced on the right.

The paper size (in Landscape) of the PostScript output is also selectable. As mentioned above, the page size setting in Adobe Distiller should be set to the same value and in Landscape mode or the output will not be centered on the page. If the path to the Adobe Distiller executable (*acrodist.exe*) is populated, the PostScript file will be automatically converted to PDF.

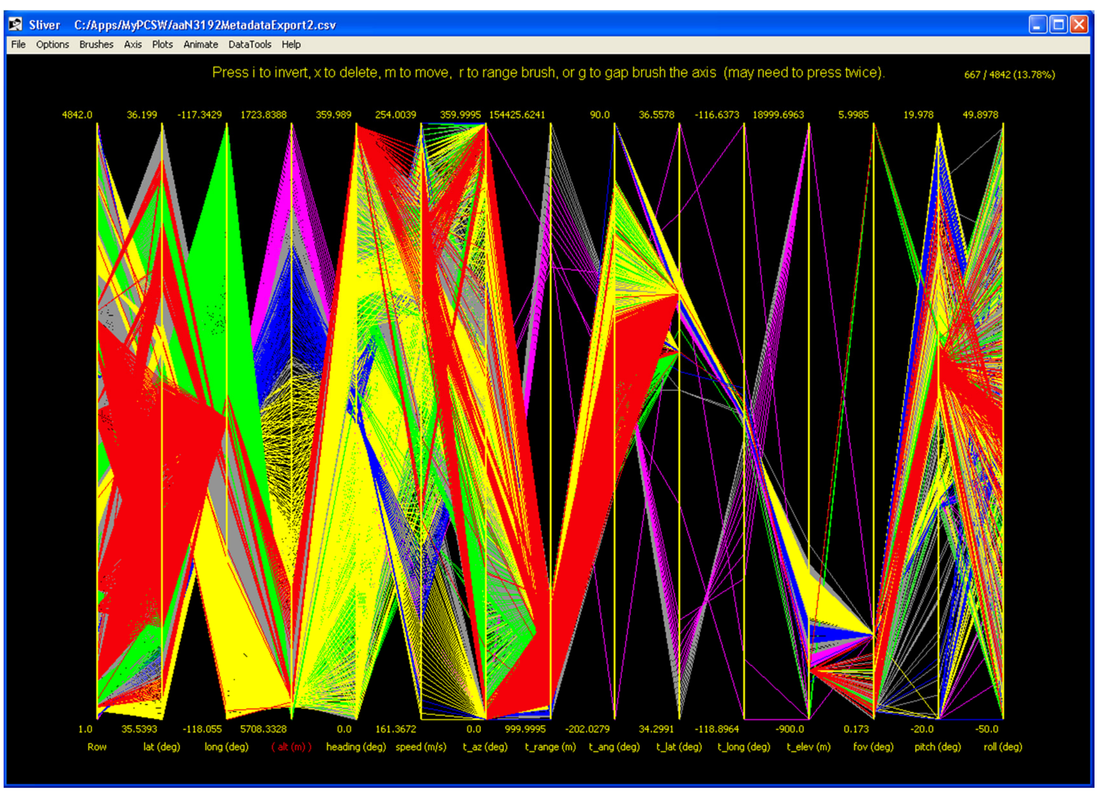
Name	**Result**
Normal	$B(c_b, c_s) = c_s$ NOTE Selects the source colour, ignoring the backdrop.
Multiply	$B(c_b, c_s) = c_b \times c_s$ NOTE 1 Multiplies the backdrop and source colour values. NOTE 2 The result colour is always at least as dark as either of the two constituent colours. Multiplying any colour with black produces black; multiplying with white leaves the original colour unchanged. Painting successive overlapping objects with a colour other than black or white produces progressively darker colours.
Screen	$B(c_b, c_s) = 1 - [(1 - c_b) \times (1 - c_s)] = c_b + c_s - (c_b \times c_s)$ NOTE 3 Multiplies the complements of the backdrop and source colour values, then complements the result. NOTE 4 The result colour is always at least as light as either of the two constituent colours. Screening any colour with white produces white; screening with black leaves the original colour unchanged. The effect is similar to projecting multiple photographic slides simultaneously onto a single screen.
Overlay	$B(c_b, c_s) = \text{HardLight}(c_s, c_b)$ NOTE 5 Multiplies or screens the colours, depending on the backdrop colour value. Source colours overlay the backdrop while preserving its highlights and shadows. The backdrop colour is not replaced but is mixed with the source colour to reflect the lightness or darkness of the backdrop.
Darken	$B(c_b, c_s) = \min(c_b, c_s)$ NOTE 6 Selects the darker of the backdrop and source colours. NOTE 7 The backdrop is replaced with the source where the source is darker; otherwise, it is left unchanged.
Lighten	$B(c_b, c_s) = \max(c_b, c_s)$ NOTE 8 Selects the lighter of the backdrop and source colours. NOTE 9 The backdrop is replaced with the source where the source is lighter; otherwise, it is left unchanged.
ColorDodge	Brightens the backdrop color to reflect the source color. Painting with black produces no changes. $B(c_b, c_s) = \begin{cases} 0, & c_b = 0 \\ 1, & c_b \geq (1 - c_s) \\ (c_b/(1 - c_s)), & otherwise \end{cases}$
ColorBurn	Darkens the backdrop color to reflect the source color. Painting with white produces no change. $B(c_b, c_s) = \begin{cases} 1, & c_b = 1 \\ 0, & (1 - c_b) \geq c_s \\ 1 - ((1 - c_b)/c_s), & otherwise \end{cases}$
HardLight	$B(c_b, c_s) = \begin{cases} \text{Multiply}(c_b, 2 \times c_s) & \text{if } c_s \leq 0.5 \\ \text{Screen}(c_b, 2 \times c_s - 1) & \text{if } c_s > 0.5 \end{cases}$ NOTE 12 Multiplies or screens the colours, depending on the source colour
SoftLight	$B(c_b, c_s) = \begin{cases} c_b - (1 - 2 \times c_s) \times c_b \times (1 - c_b) & \text{if } c_s \leq 0.5 \\ c_b + (2 \times c_s - 1) \times (D(c_b) - c_b) & \text{if } c_s > 0.5 \end{cases}$ **where** $D(x) = \begin{cases} ((16 \times x - 12) \times x + 4) \times x & \text{if } x \leq 0.25 \\ \sqrt{x} & \text{if } x > 0.25 \end{cases}$ NOTE 13 Darkens or lightens the colours, depending on the source colour value. The effect is similar to shining a diffused spotlight on the backdrop.
Difference	$B(c_b, c_s) = \lvert c_b - c_s \rvert$ NOTE 14 Subtracts the darker of the two constituent colours from the lighter colour. NOTE 15 Painting with white inverts the backdrop colour; painting with black produces no change.
Exclusion	$B(c_b, c_s) = c_b + c_s - 2 \times c_b \times c_s$ NOTE 16 Produces an effect similar to that of the **Difference** mode but lower in contrast. Painting with white inverts the backdrop colour; painting with black produces no change.

Axis Menu



The Axis menu offers the ability to shift up or down (stagger) every other axis name and range values to better view overlapping names when many axes exist. It also provides a function to compress or spread a set of axes horizontally for easier viewing of axes of interest. In addition, it offers other operations on a single axis after the axis has been selected by pressing Alt while clicking on the variable name located under the axis. This action colors the name and produces helpful text along the top that lists keystrokes for inverting (*x*), moving (*m*), deleting (*x*), range brushing (*r*), gap brushing (*g*) or shifting (arrow keys) the axis. The example below shows the display after the *alt (m)* (for altitude in meters) axis is selected (and colored in the selection color) and then inverted. In addition to the maximum value being at the bottom rather than the top of an axis, the parentheses around an axis name indicates that the axis has been inverted.

The Invert, Move, Delete and Shift axis functions will cause the PC Plot to revert to its original layout if it has been zoomed.



An alternative to using a hotkey to perform an axis function is to use the menu functions under this Axis menu.

Stagger/Unstagger Names — Shift up or down (stagger) every other axis name and range views to better view overlapping names when many axes exist.

Compress/Spread Axes — Expand or compress the horizontal interval between two selected axes, with the compression factor selectable as well.

Invert — Inverts the axis and places parentheses around the name. Inverted axes are shown non-inverted in the PCM.

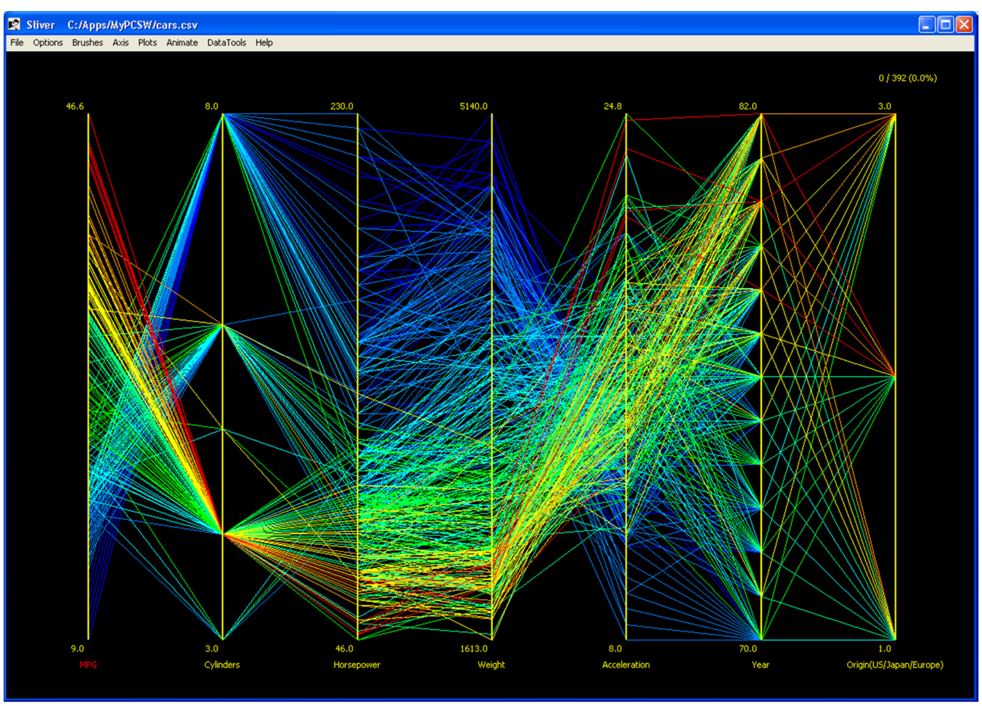
Move — Initiates a move of the axis. An arrow is drawn from the axis name. To move the axis, drag the arrow to a new location to the left or right of another axis and release the mouse. Pressing ESC cancels the move.

Delete — Deletes the axis and all scatterplots that use this variable. The remaining axes are redistributed to take up the same overall space. Deleted axes are removed from the PCM the next time it is created.

Range Brush Axis or Selection — Clears all existing brushes and assigns new color brushes to any selected lines in different ranges of the axis according to a rainbow scheme, blue at the bottom to red at the top. This allows a set of lines to be selected based on ranges of one variable while range-brushing those lines according to values of a different variable. If there are no lines selected, all lines along the axis are brushed. The number of ranges to split the axis into is defined in the Plot Settings, with a maximum of 16 colors, with a default of 16. To make the red end (often interpreted as high error) occur at the lower values, first invert the axis, range brush it, and invert it again.

Range brushing just a selection of lines retains the color brushing of lines that are not part of the selection, and any brush that is hidden is not range-brushed and remains hidden. Range brushing an entire selected axis removes all existing brushes and assigns the lines on the axis to new color brushes based on the selected range palette. Lines that have been shifted off the axis are unbrushed (uncolored) when range brushing an entire axis, but are brushed if they are included in a range brush selection.

The image below shows the PC Plot after the MPG axis has been range brushed.



Gap Brush — Clears all existing brushes and assigns new color brushes to all lines (selected or not) based on the largest gaps in the values along the axis. This is more useful than range brushing when coloring different values of a categorical variable (i.e., one with discrete values). The groups of lines between the 10 largest gaps are initially brushed with 10 different colors. The UP/DOWN arrows can then be used to change the gap sizes to increase or decrease the number of colors from 2 up to 20 colors. Pressing ENTER exits this adjustment phase.

Line Shifting Operations:

The shifting operations described below can be used to spread lines along a selected axis to zoom into specific lines of interest. This is particularly useful when extreme data has caused other lines of interest to be compressed at the opposite end of an axis. Shifting the lines is an alternative to saving only the lines of interest to a file through the **File→Save Selected** or **File→Save Visible** menu function and reloading the saved file. **The arrow hotkeys can be used instead of these menu functions.**

NOTE: Re-select the axis if the arrow cycles through menus instead of shifting the lines.

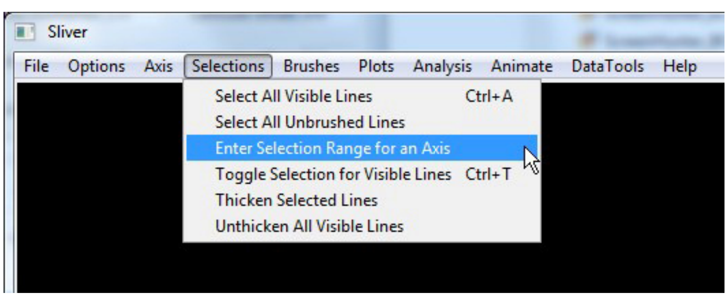
Shift Lines Up from Top — Reduces the axis top range to shift the lines at the selected axis upwards at the top. The UP arrow can also be used after the axis is selected.

Shift Lines Down from Top — Increases the axis top range to shift the lines at the selected axis downwards at the top. The DOWN arrow can also be used after the axis is selected.

Shift Lines Up from Bottom — Reduces the axis bottom range to shift the lines at the selected axis upwards at the bottom. The RIGHT arrow can also be used after the axis is selected.

Shift Lines Down from Bottom — Increases the axis bottom range to shift the lines at the selected axis downwards at the bottom. The Left arrow can also be used after the axis is selected.

Selections Menu



The Selections menu provides functions to select lines other than swiping them with the mouse, and to thicken any lines that have been selected.

Select All Visible Lines — Selects all visible lines, a faster alternative to swiping across all lines on the screen.

Select All Unbrushed Lines — Selects all visible line that have not been assigned to a brush. This is often useful to select all uninteresting lines that have not already been brushed, and then assign these newly selected lines to a new brush and hide that brush to remove the lines from the visible display to better show the other lines.

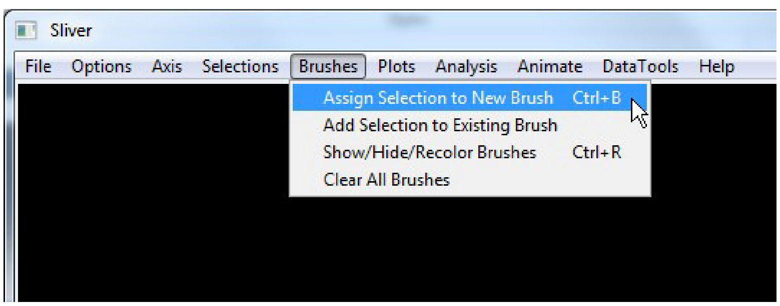
Enter Selection Range for an Axis — Pops up a window to allow a minimum value and a maximum value to be entered once an axis has been selected with an Alt-Click on the axis name. All lines lying on or between these values will be automatically selected (and any other selected lines to be unselected). This is a far more convenient way of selecting a particular line or range of lines compared to zooming in and finding those lines using the **Plots→Display Values on Mouse-Over** menu option. By repeatedly using this window, different sets of lines can be selected and each assigned to a brush in order to create custom range-brushed colors. For time-formatted axes having range values such as 04:12:53, enter the values for the range in seconds.

Toggle Selection for Visible Lines — Performs a NOT operation on the current selection, toggling it to the currently unselected, visible lines. This complements the OR, AND and XOR selection functions from keyboard commands.

Thicken Selected Lines — Increases the line widths of the selected lines by one pixel. This menu option can be repeated to thicken the selected lines further. This is useful to highlight particular lines of interest and for comparing other lines to particular lines. When exported to PostScript/PDF, the thickened lines remain thickened and opaque, and are drawn on top of the other lines.

Unthicken All Visible Lines — Reverts the linewidths of all visible lines to the value entered in the **Options→Plot Settings** window.

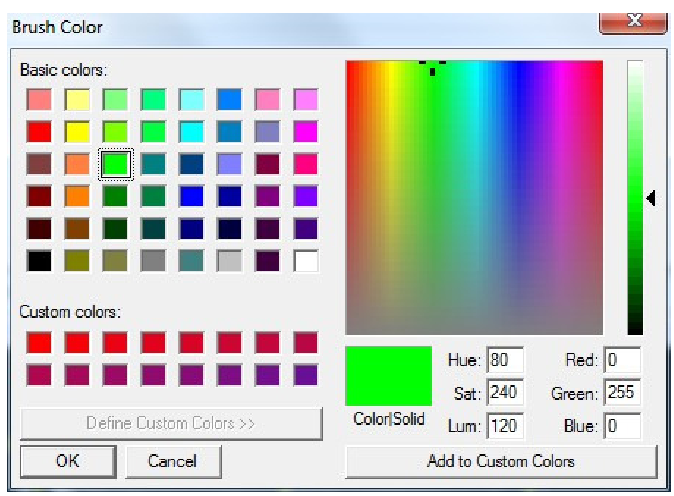
Brushes Menu



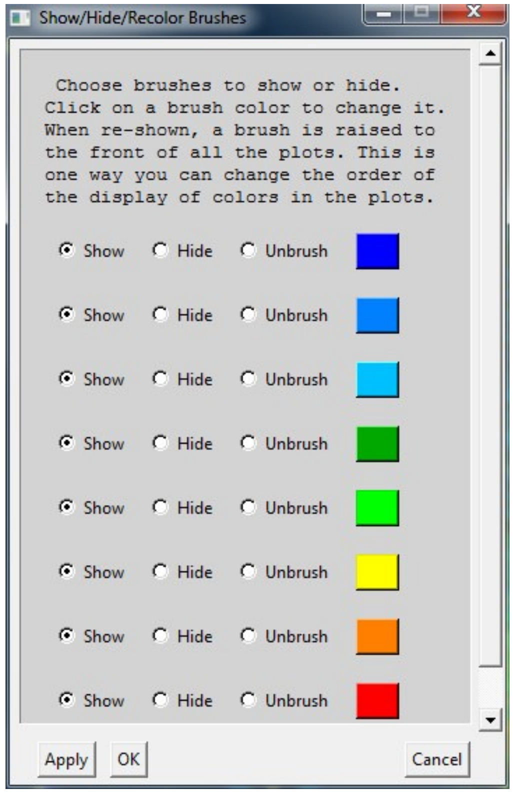
The Brushes menu provides operations on selections made with the mouse, including assigning the current selection to a color brush and giving the ability to hide, show, re-color or remove one or more brushes.

Assign Selection to New Brush — Assigns the currently selected lines (the current selection) to a new colored brush. No two brushes can have the same color.

Add Selection to Existing Brush — Adds the currently selected lines (the current selection) to an existing colored brush selected in a window.



Show/Hide/Re-Color Brushes — Provides the ability make the lines in a brush visible or hidden. This is useful when brushing and removing uninteresting sets of lines to better view the ones of interest. Also, re-showing hidden brushes brings them to the front, so hiding and showing brushes provides a means of re-ordering the brushes in the plots.

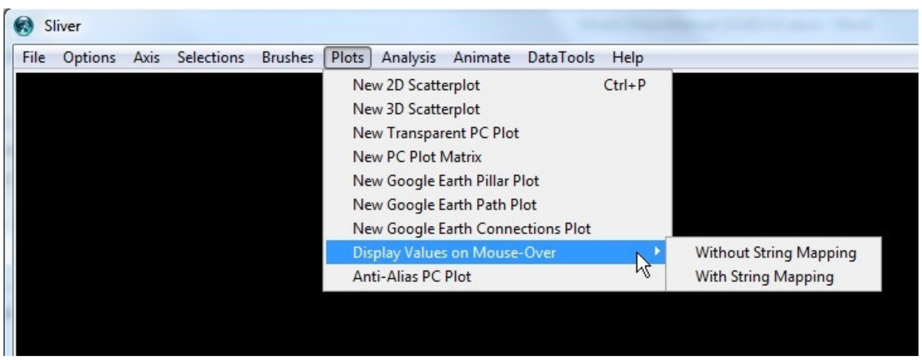


The Unbrush selection removes the brush and returns all the lines in that brush to an unselected color.

Clicking on color rectangle of a brush brings up a Color Chooser window to select a new color to assign to the lines of that brush.

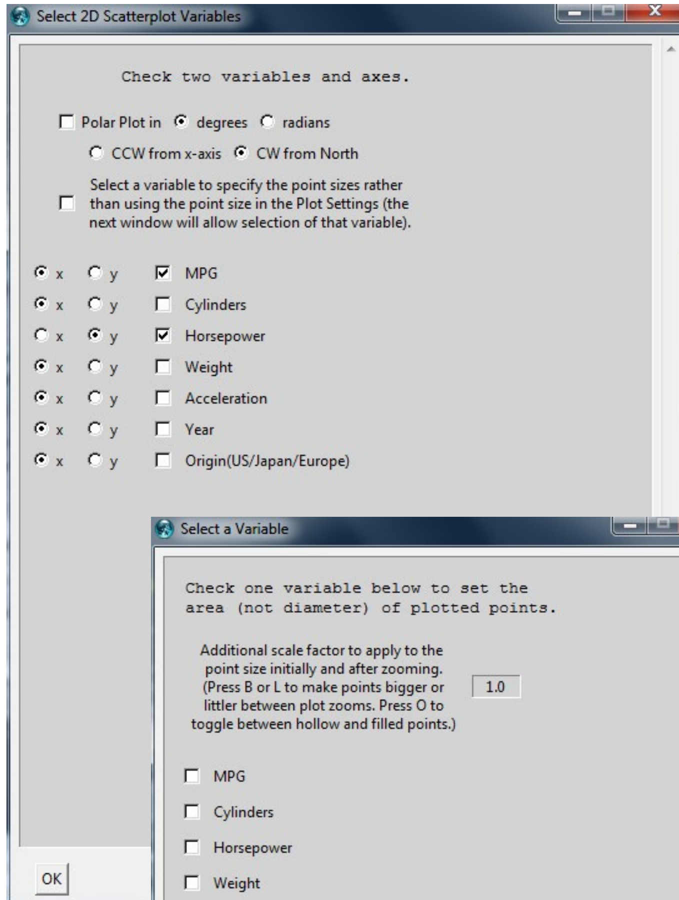
Clear All Brushes — Deletes all brushes and returns all lines to the unselected state.

Plots Menu

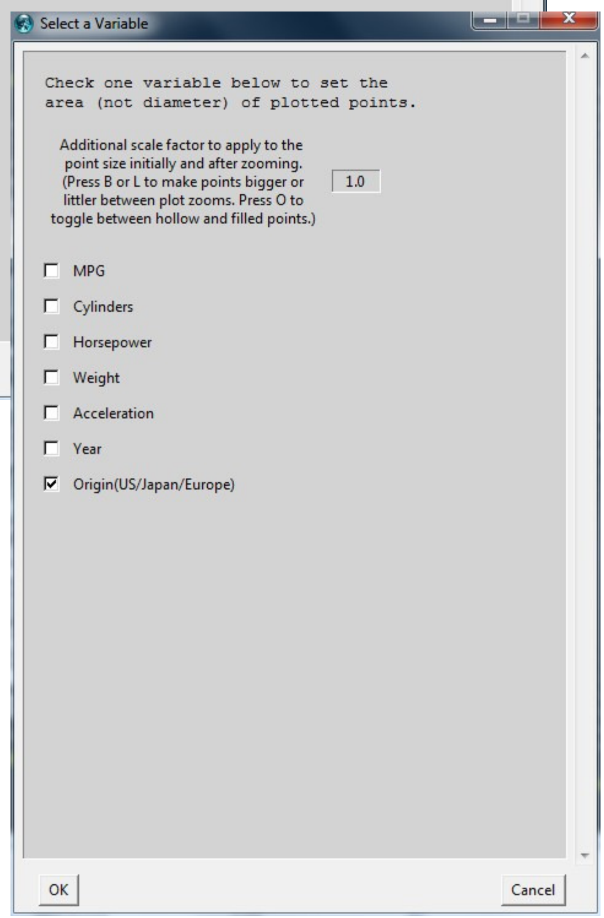


The Plots menu is the means of creating new scatterplots, 3D scatterplots and Google Earth pillar, path and connection plots based on settings under the Options menu. It also invokes the display of values along each axis when the mouse is hovered over a line (with or without mapping values to a string mapping file), which is not the default action. Finally, there is an option to temporarily anti-alias the PC Plot for cleaner screen captures—any other action removes this effect.

New 2D Scatterplot — Creates a new 2D scatterplot with x and y axis variables selected in a pop-up window as shown in the upper right. Note that two variables need to be che cked off, with one assigned as x and the other assigned to y, in order for the plot to be created. There is no limit to the number of scatterplots, and they can be created and deleted at will. If a polar plot is chosen, the axis names change to *r* and *theta*.



Selections made in any scatterplot are propagated to the PC Plot and all other scatterplots. Selections in scatterplots are always made by swiping an area with the mouse—the option to click corner points of a rectangle only applies to the PC Plot, which is slower to respond to swipes for large files than a scatterplot.

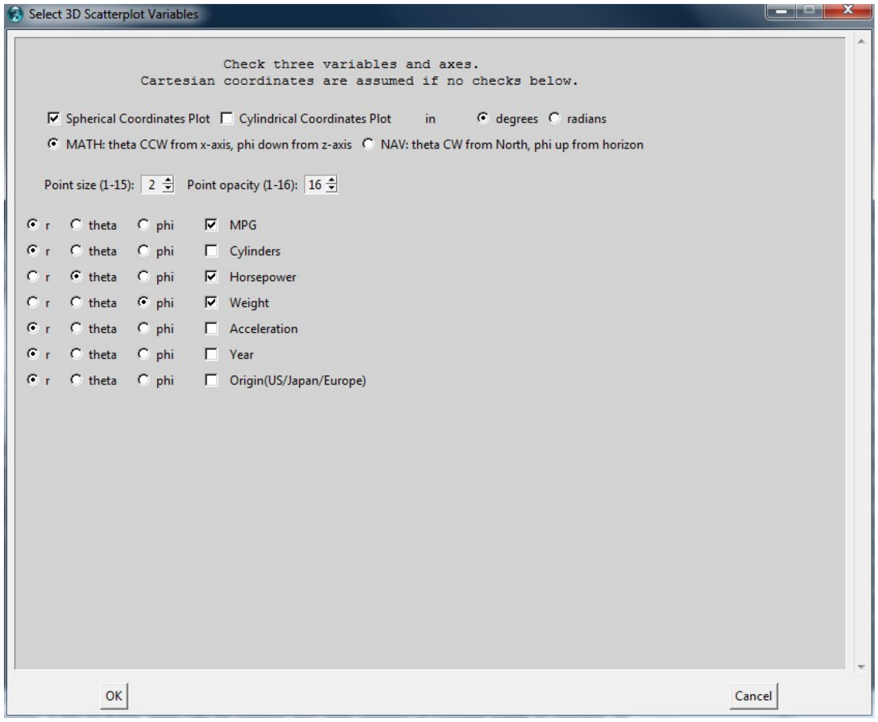


The settings for the plot (point sizes, hollow or not) are taken from the **Options→Plot Settings** menu option by default. However, there is a checkbox in this window to size the areas of the points according to values of a variable instead, either a third variable or one of the two selected ones. Checking this box will bring up a new pop-up window when OK is clicked, as shown on the lower right. The selected variable specifies the area of the point corresponding to each row of the data. There is also a field to apply a scaling factor to that variable to adjust its effect without having to edit the original data file.

Different scatterplots can be created with different settings. If the size of the points is changed in the Plot Settings after the plot is created, click on a scatterplot to make it the active window and press the *s* key, which re-sizes the points in just that plot according to the new size setting. Pressing the *s* key also removes distortions in the points due to zooming different amounts in the horizontal and vertical directions, although this should be done automatically.

If the point size in the Plot Settings was greater than one when the plot was created, or if variable point size is selected, pressing B or L will make the points bigger or littler and pressing the letter O will toggle between filled and hollow points.

Pressing *k* locks the currently active scatterplot from any changes even when the PC Plot is changed. This is useful in comparing a scatterplot against another scatterplot after color brushing, hiding of lines, or other changes are done. Pressing *k* again unlocks the scatterplot, which then updates to the latest configuration.



New 3D Scatterplot — Creates a new 3D scatterplot in Google Earth. Here three variables must be checked off in the pop-up window, one allocated to each coordinate. Cartesian coordinates (x,y,z) are assumed unless spherical or cylindrical coordinates are checked. In the latter two cases, there is a choice between math-based or navigation-based conventions (otherwise there are two default choices named N/A). The math convention for spherical coordinates (r, θ, φ) has θ increasing counterclockwise from the x-axis and φ down from the z-axis, while for navigation the azimuth θ increases CW from North and the elevation φ is relative to the horizon (the x-y plane). The North and East axes in navigation mode point to North and East in Google Earth, respectively.

Options are provided to input the (fixed) size and opacity of the points in the 3D scatterplot. The path to auto-launch Google Earth to display the 3D scatterplot is taken from the Google Earth settings under the Options menu.

Like Google Earth path plots, 3D scatterplots are saved in KML files that are selected or named by you. Once these files are created, they can be opened in Google Earth (or double-clicked to launch them in Google Earth), independent of Sliver. Therefore, a set of KML files can be created and later opened in Google Earth as needed, or combined into a single KML file with the **DataTools→Merge KML Files** menu option. **Note that when moving or sending the KML file, the dot.png file created in the original directory with the KML file must also be moved with the KML file, as it is the graphic used to create the spheres.**

New Transparent PC Plot — Creates a new window containing the PC Plot with no axes or text but with an alpha value (opacity) applied as described earlier. Hotkeys can change the alpha value (0-9 corresponds to 0.01, 0.02, 0.05, 0.10, 0.15, 0.20, 0.25, 0.3, 0.4 and 0.5, with UP/DOWN arrows to fine-tune the value), changing the linewidths (LEFT/RIGHT arrows), toggling anti-aliasing (the A key) and displaying help (the H key). Creating a new plot destroys the older plot, but settings are preserved and applied to the new plot.

The size of the Transparent PC Plot window can be set in the **Options→Plot Settings** menu option.

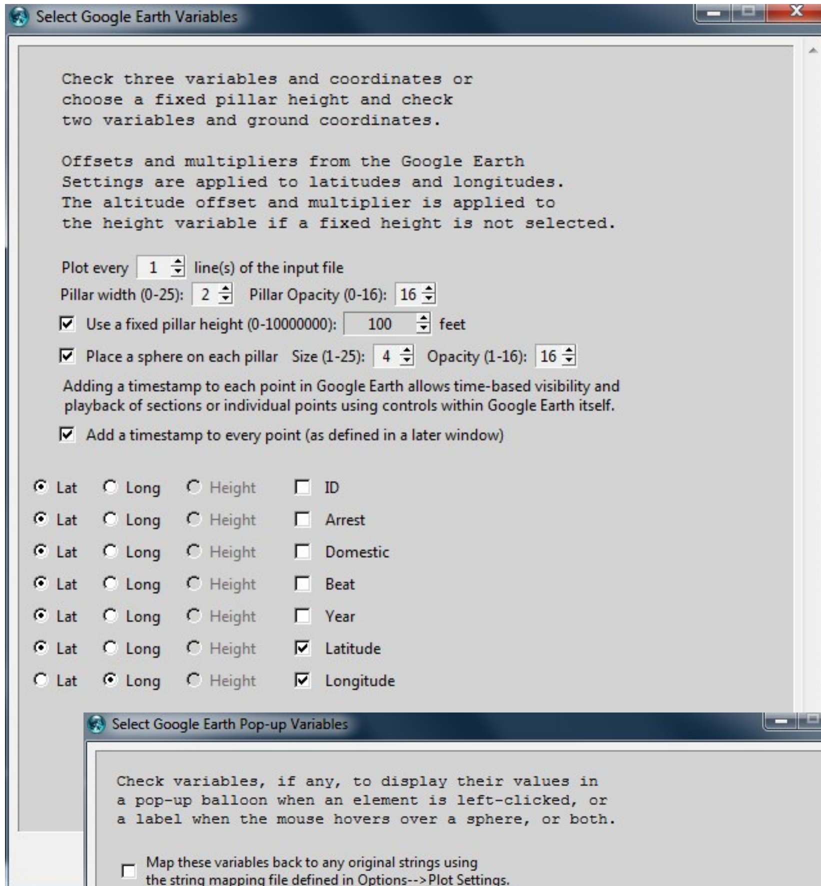
Note that when anti-aliasing is on the response to changes in the alpha value will be considerably slower. Larger data can make the anti-aliasing function so slow it is essentially inoperable.

NOTE: Initiating a Transparent PC Plot will exit all Java programs, including a Grand Tour window if present.

New PC Plot Matrix — Creates a new window containing the PC Plot Matrix (PCM). If the PCM window exists, it will be deleted and a new window created. This is sometimes necessary to remove an axis that is deleted in the main PC Plot.

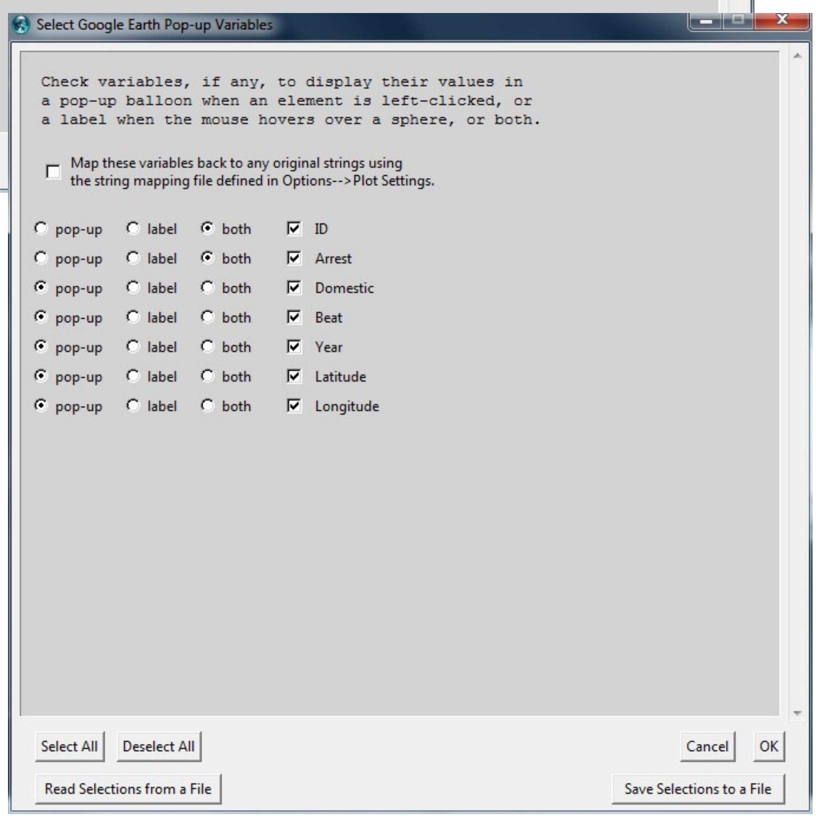
The size of the PC Plot Matrix window is set in the **Options→Plot Settings** menu option. The window cannot be resized once it is created, but it can be zoomed and panned.

New Google Earth Pillar Plot — Creates a new pillar plot in Google Earth. If the height above ground of the pillars is the value of a variable, then three sets of variables and coordinates must be checked off. If a fixed height is chosen, only two sets (latitude and longitude) need to be checked off.



There is a setting in the pop-up window to plot every nth row of the file—a lower number of plotted rows (a higher n) provides better transparency.

There are also options for the width of the pillar and its opacity, as well as the option for a fixed height and a field for entering the height in the units selected in the **Options→Google Earth Settings** menu option. The units of the fixed height are taken from the units selected in the Google Earth Settings. Offsets and multipliers for the latitude and longitudes are also taken from the Google Earth Settings, and the associated multiplier and offset for the altitude is applied to the height above ground here if a fixed height is not selected.

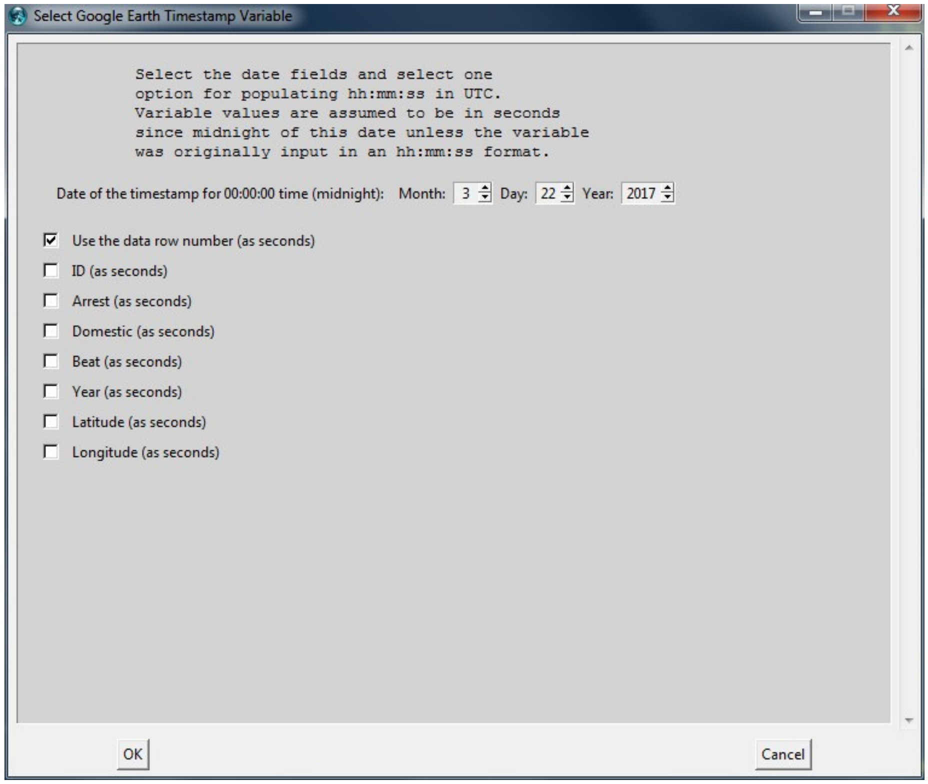


There is also an option to place a sphere on top of each pillar with a specified size and opacity.

Finally, there is an option to add a timestamp in Google Earth to every point. As described earlier, timestamps produce a player control panel in Google Earth that will span the timestamps and allow time-based selection of visible points. Choosing this option brings up a new pop-up window when OK is clicked, as shown on the right.

Variables can be selected to be displayed in Google Earth either in a pop-up (balloon) when left-clicked, or in a one-line label on mouse-over, or both. Pop-ups appear when either a pillar or a sphere is clicked, while labels require a mouse-over of a sphere and the option only appears if spheres were added to the tops of the pillars. Pop-ups display a table of the variable names and values, while labels display only the comma-separated values on a single line.

If the top box is checked, the variables in pop-ups and labels can also be replaced with strings for any values that have a string assigned to them in a string mapping file entered in the Plot Settings. The format of this file is that produced in a string mapping file when running the *Replace Strings with Numbers* function of DataTools. It can also be manually created in the format given in Appendix D.

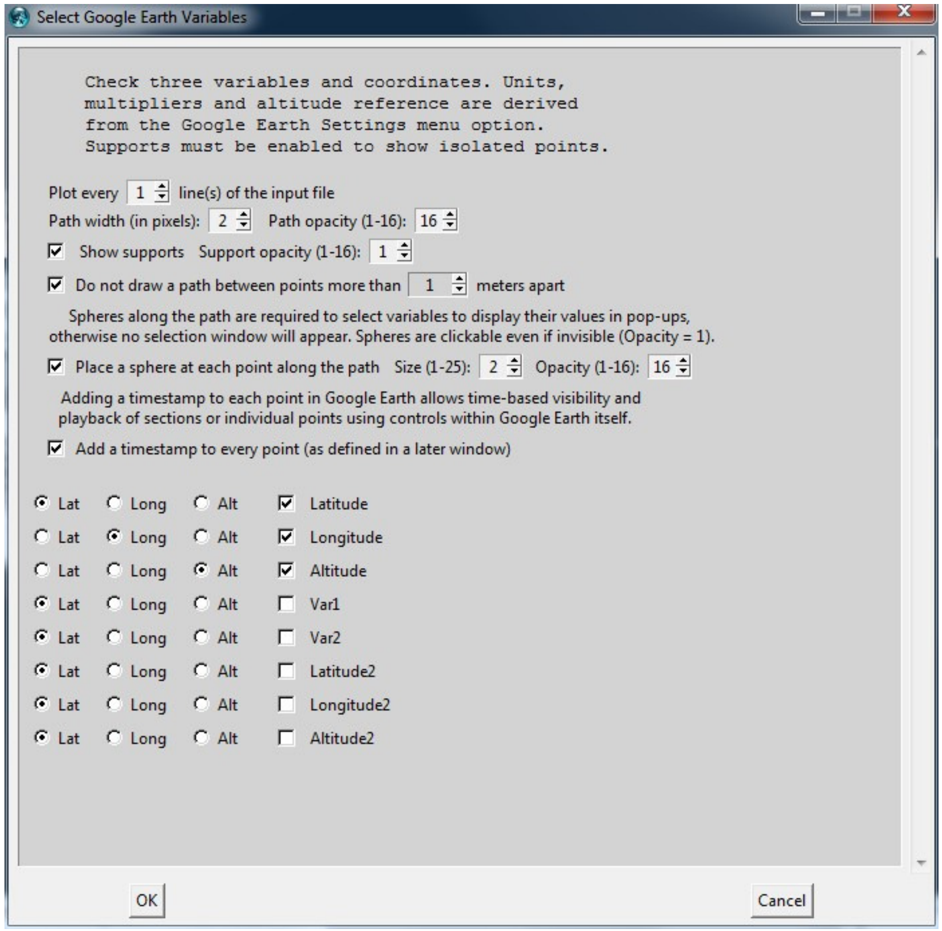


Clicking OK in the pop-up/label selection window brings up the timestamp selection window if that option was checked off in the original window. This window is shown on the right.

The timestamp is applied as seconds since the start of a date entered at the top of this window, set to the current day by default. This date can be changed to match an actual date of the data, and this is more important if the variable selected for the times in seconds is an actual time of day. Otherwise it is a convenient starting point for some kind of timestamp.

The first selection is to simply use the data row number as the number of seconds. This provides a convenient form of a time interval, one second per line, over which to operate with the player control panel in Google Earth. Alternatively, another variable can be selected to provide the offset in seconds from the date above. A time-formatted variable in hh:mm:ss will be interpreted correctly as a timestamp. All timestamps are rounded to one second.

Detailed information and examples of these options can be found in the section in this manual titled *Google Earth Timestamps, Pop-ups and Labels*.



New Google Earth Path Plot — Creates a new path plot in Google Earth. Here three variables must be checked off in the pop-up window, one each allocated for the latitude, longitude and altitude in Google Earth. As seen earlier, it can be useful to use a non-geographic variable as the altitude to plot a variable along a flight path.

There is a setting in the pop-up window to plot every nth row of the file—a lower number of plotted rows (a higher *n*) provides better transparency. As a general rule of thumb, a few thousand points at most along a path is generally sufficient.

There is an option to not draw a path between points more than an entered distance. This prevents a flight path between segments from having a connecting line between them, which is important if they are widely separated or unrelated. Also, connected path segments of a single color act as a group when timestamps are added to the path plot—the entire colored segment has a single timestamp. To enable individual points along the path to have separate timestamps, check this box and enter zero in the field so no points are connected along the path. This requires that supports be drawn, however, in order to see individual points.

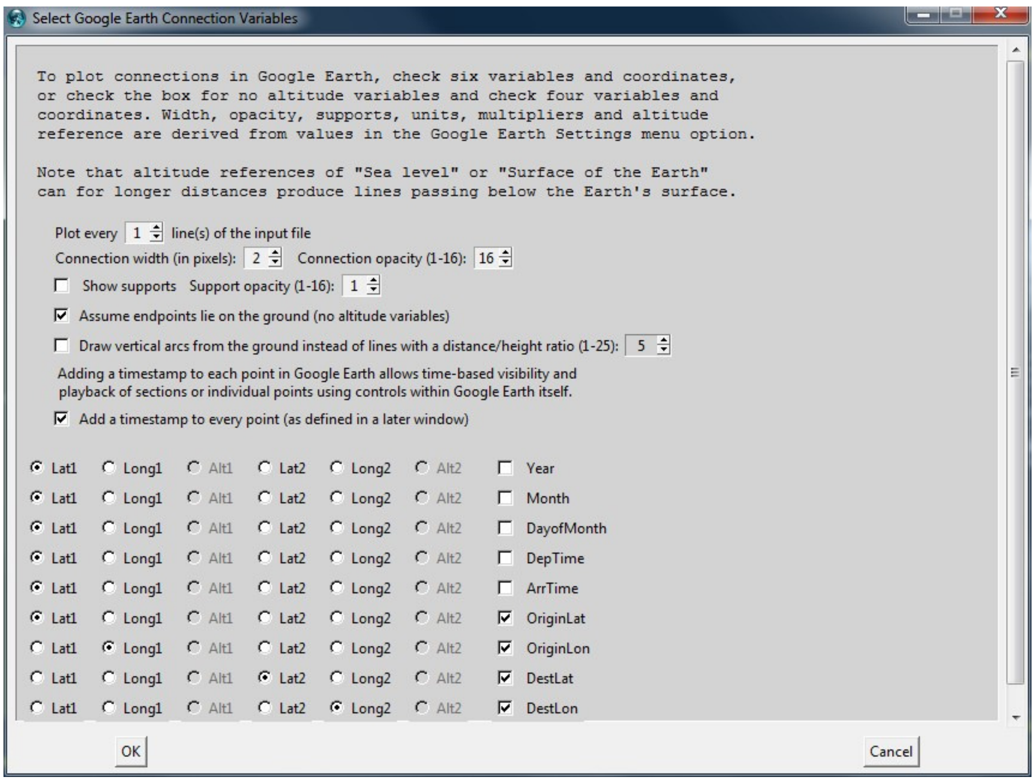
The path width and opacity, and the option to show supports down to the ground and their opacity, are entered in this window. Settings for the units of the variables, multipliers and offsets for each variable, the reference for the altitude, and an optional path to auto-launch Google Earth to display the path are taken from the **Options→Google Earth Settings** menu option.

Spheres can be added to the points along the path, sitting on the supports if they are enabled. However, their opacity can be set to one so they are not visible. **Spheres are important because pop-ups and labels require them as clickable or mouse-over points—the supports will not work in this regard.**

As in the Google Earth Pillar Plot, there is an option to insert timestamps for the path elements.

After clicking OK, a window appears to select variables for pop-ups and labels if spheres are enabled. Another window for choosing the variable or row number for the timestamps appears if timestamps are enabled. This windows are identical to those for Google Earth Pillar Plots—please refer to that section for details on the window entries and to the section in this manual titled *Google Earth Timestamps, Pop-ups and Labels* for detailed information and examples of their operation.

Again, the KML file is named by you, and these KML files can be opened in Google Earth independent of Sliver. Also, these files remain by default in the Temporary Places listing in Google Earth, and can be independently hidden or displayed by checking or unchecking them in that listing. This allows multiple plots created in Sliver to be displayed at the same time, allowing the kind of compound displays seen earlier in this manual. The **DataTools→Merge KML Files** menu option provides a means of combining multiple KML files into a single KML file for convenience. Waypoints and other data can be entered manually within Google Earth and displayed as well. Remember to include the *dot.png* file in the KML directory if the KML file is moved or sent.



New Google Earth Connections Plot — Creates a new connections plot in Google Earth. Here by default six variables must be checked off in the pop-up window, one each allocated for the latitude, longitude and altitude of the first point and the latitude, longitude and altitude of the second point. Again, there is an option to plot every nth row.

This will plot a line (a connection) in Google Earth between the pair of points in each plotted row. The width and opacity of the connections, and the option to show supports down to the ground and their opacity are entered here. The units of the variables, multipliers and offsets for each variable, and the reference for the altitude are taken from the **Options→Google Earth Settings** menu option just as for a Google Earth Path Plot.

There is an option to assume that the endpoints lie on the ground, and checking this option grays out the altitude radiobuttons, therefore requiring only four variables to be checked off. This is very useful if ground coordinates are all that are available. The lines will lie on the ground along the great circle between the two positions. If that box is checked, as in the figure here, a second option to draw vertical parabolic arcs between the ground positions rather than the lines is enabled. The ratio of arc height to distance (actually the inverse distance to height ratio) can be entered. The arc will lie over the great circle between the two points on the surface of the Earth.

As in the Google Earth Pillar Plot and Google Earth Path Plot, there is an option to insert timestamps for the path elements.

After clicking OK, a window appears to select variables for pop-ups, which will appear when a connection is left-clicked. **Labels are not enabled** because connections do not respond to mouse-overs and spheres are not an option for connection plots. Another window for choosing the variable or row number for the timestamps appears if timestamps are enabled. This windows are identical to those for Google Earth Pillar Plots—please refer to that section for details on the window entries and to the section in this manual titled *Google Earth Timestamps, Pop-ups and Labels* for detailed information and examples of their operation.

Display Values on Mouse-Over — Toggles the mode in order to display values. When the mouse hovers over a line in the PC Plot or over a point in any 2D scatterplot, the values are displayed in every plot. This can take a few seconds for large files. The line in the PC Plot is highlighted and the values appear next to the axes, and the corresponding point is highlighted and values displayed in each 2D scatterplot.

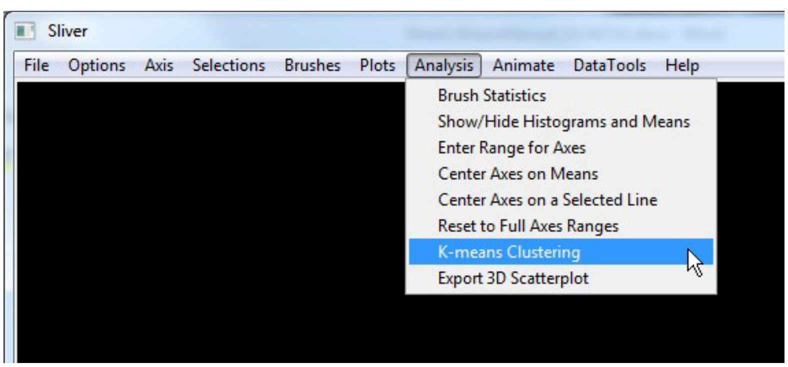
There are two options when choosing this menu function. Choosing *Without String Mapping* simply displays the numerical values of the variables along the line the mouse is hovering over. Choosing *With String Mapping* compares the numerical value for each variable to a string mapping file entered in the Plot Settings, and if a match is found for that value of a variable, the string is displayed instead of the number. This will only happen in the main PC Plot window, not in any scatterplots, which will always show the number. The format of the mapping file is that produced in a string mapping file when running the *Replace Strings with Numbers* function of DataTools. It can also be manually created in the format given in Appendix D.

Pressing ESC twice or re-selecting this menu function will exit this mode. Exiting the mode will increase the speed of updates to the display.

Anti-Alias PC Plot — Performs an operation to smooth the lines in the main parallel coordinate plot to some degree, making them less jagged. This can make screen captures more readable and professional-looking. The screen captures in this manual were not anti-aliased.

Any operation that affects the main PC Plot turns off the anti-aliasing of the lines.

Analysis Menu



The Analysis menu provides a variety of functions that can help in analyzing the relationships among the data.

Brush Statistics — Displays statistics of the data of each brush as well as the current selection. This includes the minimum and maximum values, average value, variance and standard deviation for each variable of all brushes and for the current selection if one exists.

Show/Hide Histograms and Means — Draws a user-selectable number of histogram bins along the left side of each selected axis, where the height of each bin is proportional to the number of lines intersecting the axis within the span of that bin. The mean of each axis is marked by a horizontal line on the right side of the axis. This is very useful for interpreting the clusters of data in the PC Plot. Only visible lines that intersect an axis are considered, so shifting lines off an axis removes them from calculations of the histograms and means.

Enter Range for Axes — Allows axes to be chosen and then pops up a window to allow a minimum value and a maximum value of a range to be entered. The ranges of the chosen axes will be updated to this set of values. This is very useful when a consistent range is best across axes, as in split times for races where comparisons with previous splits is important.

Center Axes on Means — Allows axes to be chosen and then centers them on the means of each selected axis in order to visualize variations from the mean. If some lines are off-axis, this function resets the axes ranges so that all visible lines lie along them before centering on their means.

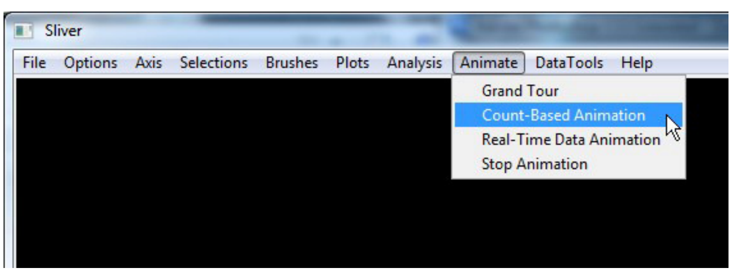
Center Axes on a Selected Line — Allows axes to be chosen and then centers them on a particular selected line in order to visualize variations from a particular row of data.

Enter Range for Axes — Returns the ranges of chosen axes to their original values that span the range of values for that axis.

K-Means Clustering — Provides automatic brushing of clusters in the data. Choosing this menu option pops up a window for selecting which variables to include when identifying clusters. It also allows the number *k* of clusters in which to partition the data, as well as the top number of those clusters to brush in rainbow colors, with the cluster having the most data points brushed in red and the cluster with the least data points (of the number of clusters to be brushed) in blue. There is also an option to automatically update or launch the Transparent PC Window described above. This provides a means of viewing the identified clusters in the transparent view to verify that the selected number *k* of clusters and the number to color brush through the data is optimal, at which point the process can be repeated with different values. This clustering function requires that the free R statistical software be installed on the PC, with the path to the RScript.exe file selected in the **Options→Analysis Settings** menu option.

Export 3D Scatterplot — Pops up a window to choose the three variable for x, y and z in a Cartesian coordinate system, along with the size of the points. If Matlab is installed on the PC and the path to the matlab.exe file is entered in the **Options→Analysis** Settings menu option, the 3D scatterplot will appear in a Matlab figure with the color brushing that was applied in Sliver. The interactive controls and myriad of settings of Matlab can then be applied to the plot as desired, and the plot can be saved in Matlab format if desired.

Animate Menu



The Animate menu provides the capability to

1. Initiate a Grand Tour animation, rotating multi-dimensional data to identify clusters, outliers and structures.

2. Initiate a count-based data animation to auto-sequence through the data rows at a given rate in a given direction. This is useful when syncing data to video playing in a video player or with other applications or instrumentation, or simply to traverse through the data while syncing the data among the various plots, including Google Earth plots.

3. Initiate real-time data collection.

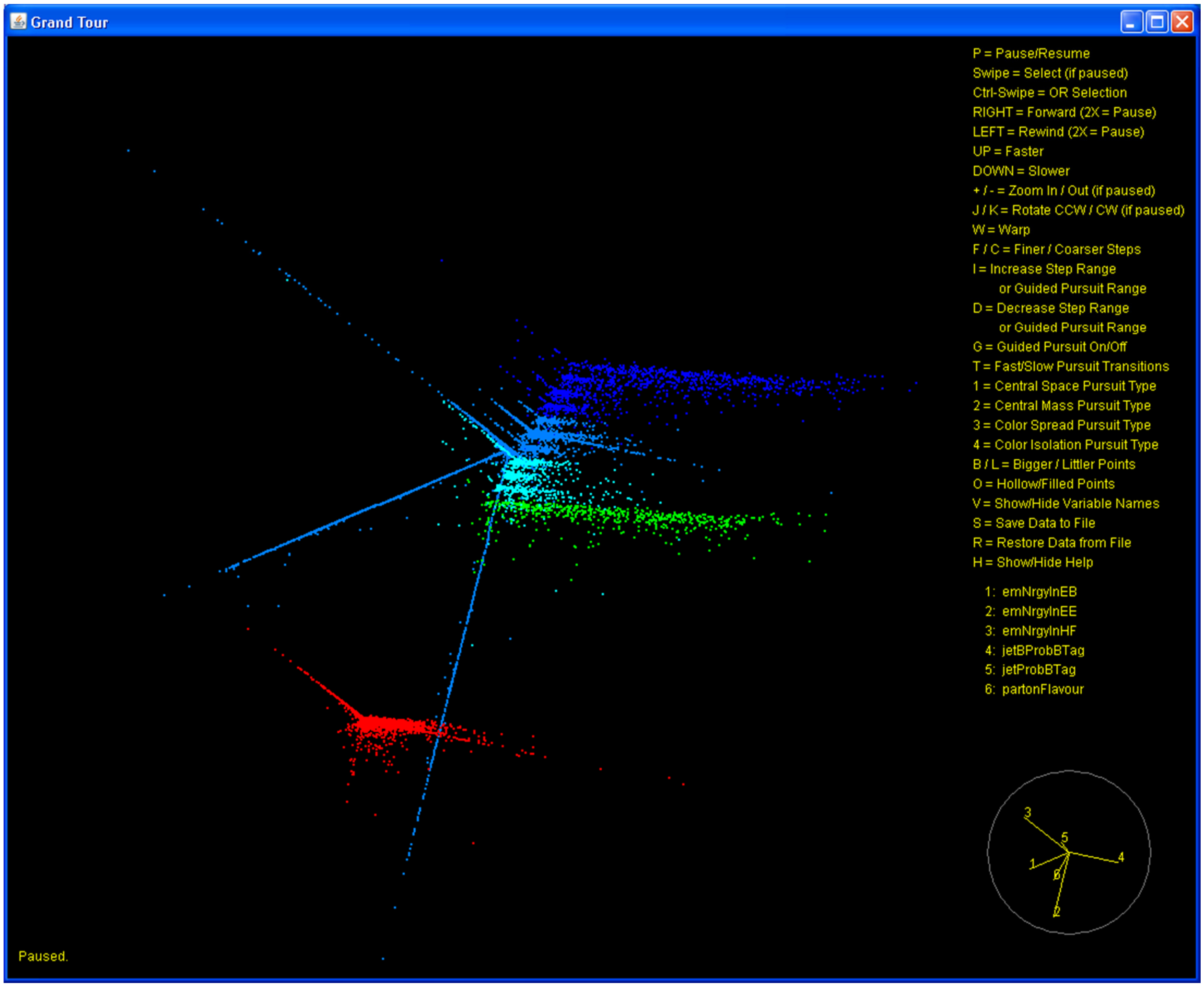
NOTE: Initiating a Grand Tour will exit all Java programs, including the Transparent PC Plot window if present.

Grand Tour Animation — Presents a window to select between 3 and 9 variables and launch a separate window for a Grand Tour as shown earlier. The displayed points represent the projections of the point locations onto the 2D plane of the monitor as the axes are rotated through *n*-space, as described earlier. Data clusters and geometries can be observed as random (or guided) rotations occur, as the points will retain their relative position while uncorrelated points will move independently. The Grand Tour can be paused and points brushed, and these selections will sync with the PC Plot and any scatterplots.

Performing Principal Component Analysis (PCA) on the data set prior to using the Grand Tour may be a useful exercise. The PCA function is available in MatLab, the statistical package R, and other programs. In PCA the coordinate axes remain orthogonal but represent linear combinations of the variables. The axes (i.e., the new variables) are created in such a way that the first axis lies along the direction of greatest variance of the data, the second axis is orthogonal along the direction of the next lower variance, and so forth. Using the modified data set often optimizes the separations of the data clusters. Also, variable axes that have low variance can be eliminated from the Grand Tour since they represent "noise" that doesn't significantly affect the distribution of clusters, structures or outliers. Dimension reduction in this way is a primary use of PCA, and is helpful in the Grand Tour as well.

The Grand Tour window is launched nearly full-screen, although the layout assumes that the tour will be conducted in the left portion of the display. The right edge of the window can be resized to reduce the window to the size shown here, or the entire window can be used by using the hotkey for zooming out and the hotkeys for shifting the origin of the plot. This can be very useful to distinguish points and clusters in larger datasets. The small circle showing the projections of the axes onto the plane of the monitor, as well as the Help text if displayed, remain in their original locations.

Pressing the H (for Help) key in the Grand Tour window displays a list of available mouse operations and keypress assignments available in the Grand Tour, as shown in the image below. (The Z/X and Y/U options were added to the Help list after this screenshot was taken.)



Data from DataViewer software distribution (Benjamin.Radburn-Smith) – see References.

The following actions are supported in the Grand Tour window:

P: Pause or resume the **P**lay of the animation.

Mouse Swipe: When play is paused, swiping the mouse while the left button is pressed draws a rectangle, and on release any points within that rectangle are colored in the selection color setting of the main PC Plot. The corresponding lines and points in other scatterplots are colored accordingly. Assigning the selection to a color brush in the PC Plot window will cause the points to be brushed in the Grand Tour window as well. In fact, any selections and brush activities, including hiding lines in the PC Plot, are synced among all the plots whenever the Grand Tour is paused. This syncing operation may take a couple of seconds to propagate.

Ctrl + Mouse Swipe: If the Ctrl key is pressed when a second selection is swiped, the resulting selected points are added (OR'd) to the existing selected points. Otherwise the previous selected points become unselected and return to their original color.

Left/Right Arrows: Pressing the right arrow key causes the Grand Tour to play in the forward direction, randomly generating new rotations in *n*-space. Pressing the left arrow key rewinds the Grand Tour play at a slower speed, allowing the exact position of interest to be located. Pressing a left or right arrow twice in a row results in pausing the play, and pressing an arrow again then resumes the play in the direction assigned to that arrow.

Up/Down Arrows: Pressing the Up arrow speeds up the play, and holding it continuously will speed up the play to the maximum speed possible for the number of points and dimensions. Pressing the Down arrow slows the play, which is useful for the study and isolation of interesting features.

Z / X keys: The Z key shifts the origin of the plot to the left, and the X key shifts the origin to the right.

Y / U keys: The Y key shifts the origin of the plot downward, and the U key shifts the origin upward.

+ / - **keys**: The + / - keys (or the shifted versions of them) zoom the display in or out for optimum viewing.

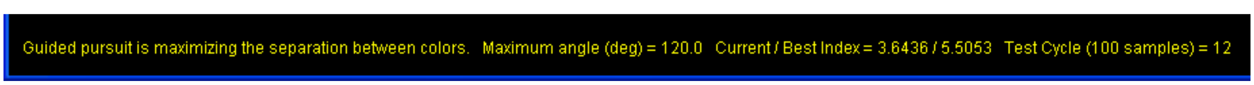
J / K keys: When paused, the J key rotates the display counterclockwise, and the K key rotates it clockwise. This is very useful when a set of points need to be selected but they lie at an angle, which might require a number of OR'd selections to enclose them all in selection rectangles while excluding others. By rotating the display, the bulk of the points can be lined up horizontally or vertically for more efficient use of the selection rectangle.

W: The W key **W**arps the view to a new location in *n*-space during play or when paused. This can be useful to jump out of an uninteresting neighborhood. This is also useful when an automated, guided pursuit is operating, which may find a local maximum for a selected projection but needs to be warped to another location to find other local maximums.

F / C keys: During play or pause, the F and C keys select a **F**iner or **C**oarser angle step in the animation. A finer step produces smoother play, but for larger data sets or dimensions a coarser angle step may be needed to speed up the animation. Generally, the Up/Down and F/C keys are used to optimize the animation for a given data set.

I / D keys: During play or pause, the I / D keys **I**ncrease or **D**ecrease the maximum possible angle changes across the dimensions. During normal play, this results in longer or shorter times that the points move without shifts in directions. During guided pursuit described below, these keys result in increased or decreased ranges of angles in which the algorithm will search for more optimum criteria. It is often useful to decrease the search range once a reasonably good pursuit solution is found in order to fine-tune the search to find a local maximum. During the guided pursuit the angle range is shown along the bottom of the display.

G: Pressing the G key toggles the tour into and out of the **G**uided pursuit mode. The guided pursuit mode randomly searches the angle space to find new sets of rotation (a transformation matrix) that optimizes a projection index for the selected pursuit type. When the guided pursuit mode is invoked, a line of text along the bottom of the display lists the selected pursuit type, the current search angle range as defined through the I/D keys, the current and maximum projection index found for the selected pursuit type, and the number of tests and samples per test that have been run, as shown below.



T: Pressing the T key toggles between faster (instant) and slower (the default) **T**ransition speeds when the guided pursuit finds a more optimized solution and rotates to the new set of angles. The slower transitions can provide insight into the structure of the data, but they can be distracting when they occur frequently or when there are many dimensions.

1: Pressing 1 selects a projection pursuit type that maximizes the central space in the plot. Points are scored by the total distance of all points from the plot origin, inversely weighted exponentially. This pursuit type may help in separating data clusters for selection, but may not be helpful in some cases.

2: Pressing 2 selects a projection pursuit type that maximizes the central mass in the plot. Points are scored by the inverse of the total distance of all points from the plot origin, weighted exponentially. This pursuit type may help in isolating outliers around the outside for selection.

3: Pressing 3 selects a projection pursuit type that maximizes the separations of the centroids of the different colors of points in the plot. Points are scored by the total distance among all centroids. This pursuit type may help in separating color brushed clusters for additional selections, or to simply find the transformation matrix that best separates clusters.

4: Pressing 4 selects a projection pursuit type that finds the maximum separation of any one centroid of the different colors of points in the plot. Points are scored by the total distance of any one centroid from all the others. This pursuit type may help in best isolating one color brushed cluster for viewing.

B / L keys: Pressing B or L results in **B**igger or **L**ittler points.

O: Pressing the letter O toggles the point styles between filled and empty **circles**.

V: Pressing V produces a line of text along the bottom of the display that lists the **V**ariables in the Grand Tour and their numbered assignments in the projection components circle in the lower right. The variable names and number assignments are also listed at the bottom of the Help text when H is pressed, as shown in the screenshot above.

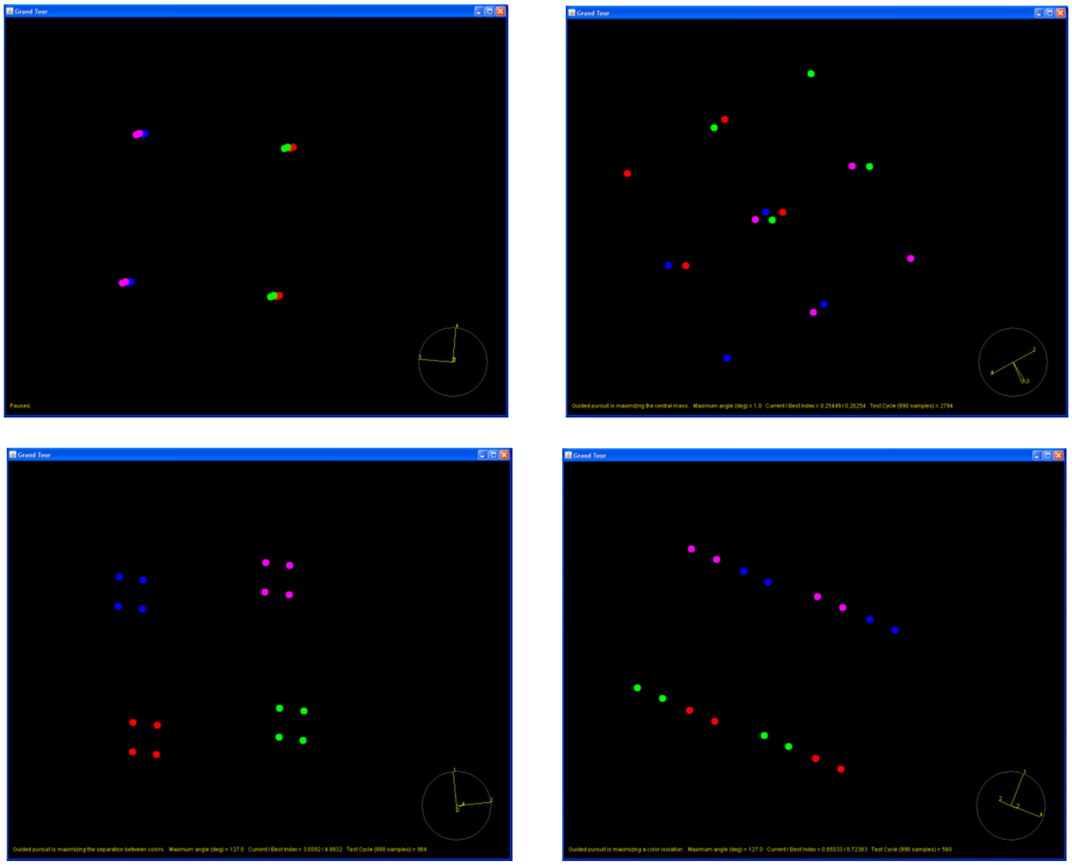
S: Pressing S brings up a file browser dialogue to **S**ave the current data to a text file in CSV format. This file contains

1. the original data for the variables,
2. the data input to the Grand Tour from the PC Plot (the data after subtracting the mean values of the variables and scaling each variable to approximately fill the display) and their original colors,
3. the mean of each variable and the scaling applied to it,
4. the current data and colors, and
5. the overall transformation (rotation) matrix that relates the input data to the current display

R: Pressing R brings up a file browser dialogue to **R**estore a previously saved data file. This involves extracting the transformation matrix and multiplying the input data of the current session by that matrix to arrive at the same display (other than color brushing) as when the data was saved.

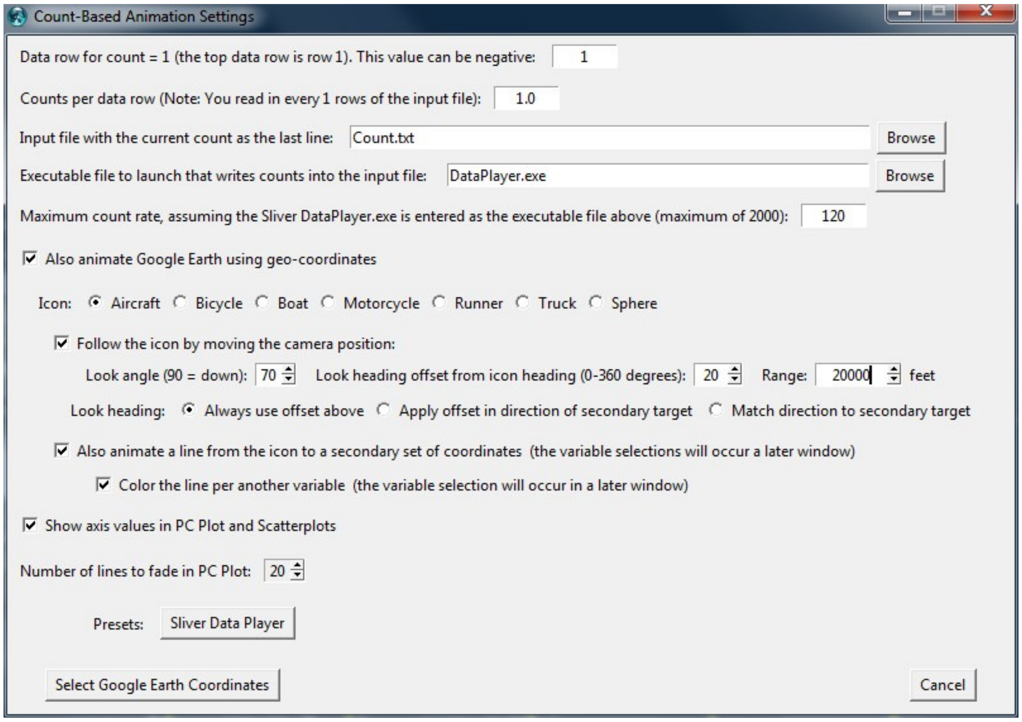
H: Pressing H toggles the **H**elp text display on and off.

For illustrative purposes, below are screenshots of the four projection types operating on a 4-dimensional hypercube. Type 1 in the upper left optimizes the central space. Type 2 in the upper right optimizes the central mass. Type 3 in the lower left optimizes the separation of color centroids (and is not unique for a 4D cube). Type 4 in the lower right optimizes the best single color centroid separation (and is also not unique in this case).



Grand Tour screens are not included in PostScript/PDF exports. Simple screenshots are the current method of recording the results.

Count-Based Animation — Initiates an animation based on settings in a pop-up window as shown below.



The current row of data to animate is determined by the current value in the last row of a text file. The first field is the row of the data file that will correspond to a value of 1 for the count, thus providing for an offset. This value can be negative, and no data will be animated until the count reaches this minimum value.

The next field contains the ratio of counts per data row. For animation using the Sliver Data Player this would normally be set to 1.0 even if the file was decimated when read in, but a different external program may require a different rate.

The next field gives the name of the text file whose last row will contain the current count, with a Browse button to locate it. Sliver will continuously read the last line and animate the corresponding row of data based on that number. Leave this filename as Count.txt to use the built-in Sliver Data Player since it saves counts into that filename.

The next field contains the name of the executable script that will be called by Sliver to start appending the count to the count file. There is a built-in script for the Sliver Data Player, and clicking on a preset button will automatically populate these fields for that player.

Then there is an option to also animate the Google Earth path with an icon, where the variable selections for latitude, longitude and altitude will be made in the next window. If this box is checked, an option will appear as shown here to move the Google Earth view "camera" to follow the moving icon at a given look angle, heading offset and range. Otherwise, the Google Earth view will remain fixed unless the view is manually changed with the mouse controls. **NOTE: If the motion following the icon is sluggish or jerky, or there is a feeling of motion sickness, change the fly-to speed to the fastest slider value in the Google Earth Tools→Options→Navigation→Fly-To Speed menu selection.**

Also, if the box to animate the Google Earth path with an icon is checked, an option will appear as shown here to draw a line from that icon to a secondary geographic location, and checking this box will show yet another window before the animation begins to allow the latitude, longitude and altitude variables of the secondary set of positions to be selected. And if this box is checked, there will also be an option to color that line according to another variable rather than the default white color, and there will be yet again a later window to choose this variable. The color of the line is white if the variable is zero and rainbow-brushed from blue to red otherwise. (The reason for assigning white to a value of zero is to support Boolean values, so white is off, i.e., 0, and red is on, i.e., 1.)

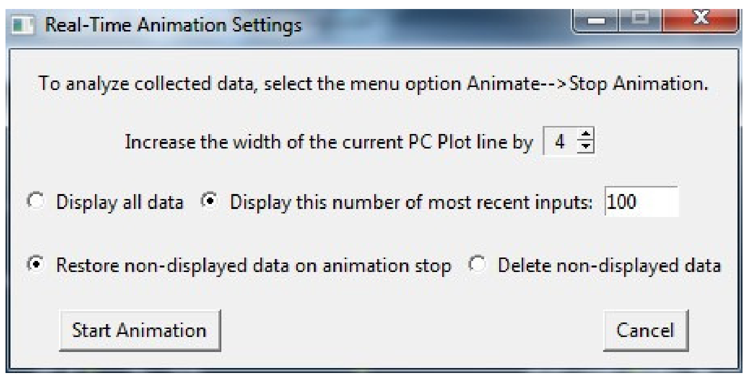
The next checkbox selects whether to display the data values next to the animated line or point in the PC Plot and 2D scatterplots as the animation proceeds.

Finally, the last item selects the number of fading trailing lines to show in the PC Plot (0 to 20). This number can be decreased for faster performance.

Initiating the animation after the subsequent selection windows are completed produces the syncing operation as described in the earlier section on data visualization modes. Selecting the Sliver Data Player results in the display of the player controls window shown earlier.

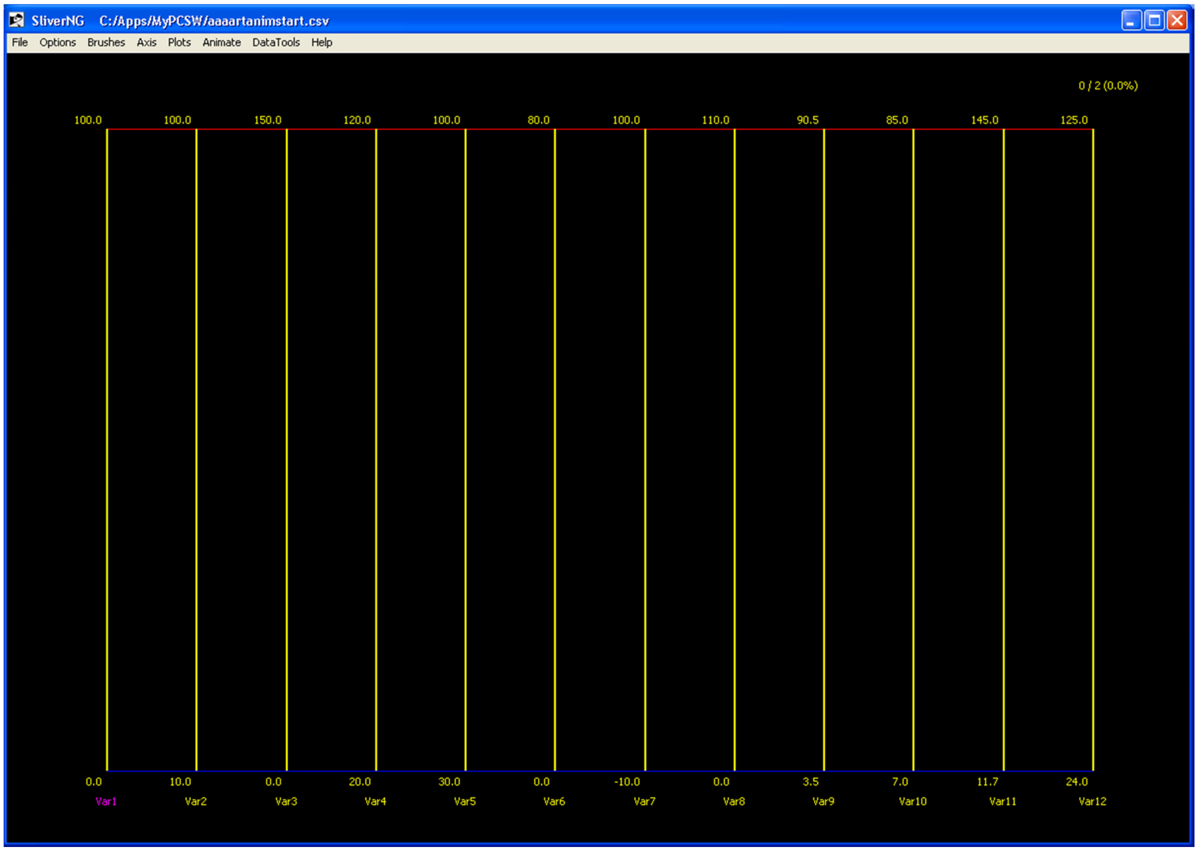
Two or more instances of Sliver can run in a common Count-Based Animation. A single Sliver DataPlayer window will serve to control all instances (creating a new player in any instance replaces the one in the previous instance), so it is possible, for example, to animate multiple paths in Google Earth simultaneously with their own icons. The animation is based on row number, so you will want to make sure the data used in all Sliver instances starts at the same time of day, for example, and have the same data rate. Note that the same data file can be read into multiple Sliver instances and then different variables selected for the animation.

Real-Time Data Animation — Presents a window of options as shown on the right. The width of the PC Plot line for the current data can be increased for greater visibility. There is an option to display all the data as it is accumulated or to just display the most recent data to lessen clutter. In the latter case there is an option to either display all the data that has been undisplayed when the animation is stopped, or to delete all the lines except the number being displayed.



Before real-time data animation is selected, it is expected that a file has been read into Sliver that consists of a header row with the variable names and two data rows that consist of the maximums and minimums of each variable. As with any input file, the variable axes are displayed, and the two lines appear along the top and the bottom of the axes since they represent the minimum and maximum extent of each axis. An axis can be range brushed so that the incoming data is color brushed, and in this case a PC Plot like the one below is produced. In this case the input data file consisted of these three lines:

```
Var1,Var2,Var3,Var4,Var5,Var6,Var7,Var8,Var9,Var10,Var11,Var12
0,10,0,20,30,0,-10,0,3.5,7,11.7,24
100,100,150,120,100,95,100,110,90.5,105.5,145,125
```

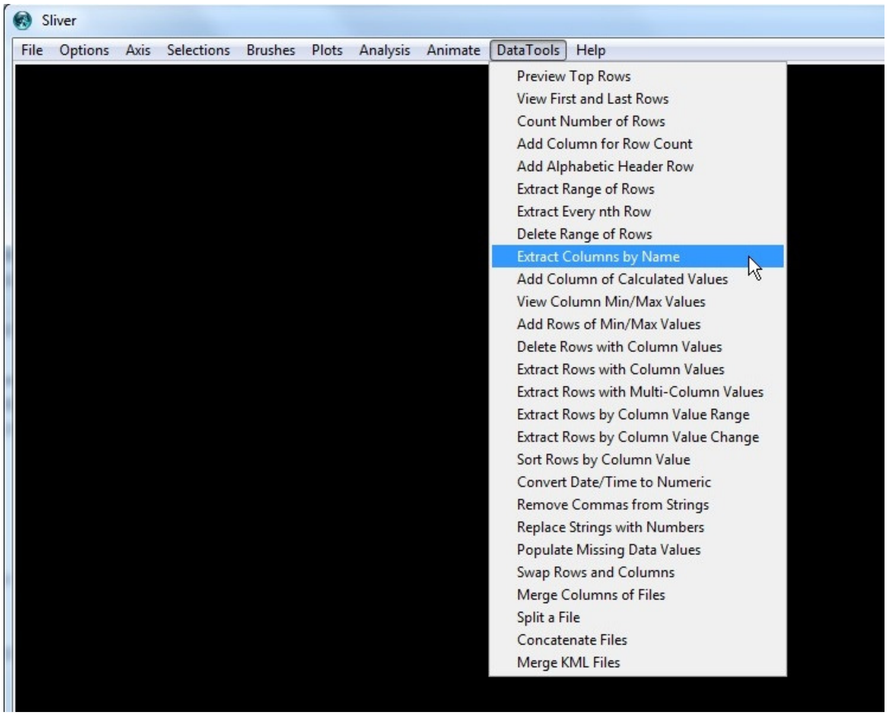


Any number of 2D scatterplots (Cartesian or polar) can be created before the animation is started. At this point all plot windows should be resized as needed for the PC monitor, as they cannot be resized while the animation is occurring. Then the Real-Time Data Animation menu option can be selected and the options set in the window shown above. The real-time data collection begins when the Start Animation button is pressed in that window. At this point the two previously entered lines are deleted and additional data is plotted as data rows are added to the original data file, as shown in the screenshots provided on pages 56-57. The animation mode is exited through the Stop Animation menu option described below. The data file is no longer monitored, and undisplayed data is then displayed if that option was selected. Sliver can then act on the collected data in any of the ways described in this manual. Restarting the animation mode continues at the point where it had been exited, quickly populating data generated while it was stopped. To start afresh, stop the animation mode and load in a new data file comprising a header and two data lines for the maximum and minimum ranges.

Stop Animation — Ends the current animation mode. In the case of a Grand Tour, that window is exited. In the case of Count-Based Animation, the PC Plot is reconfigured to again display all the lines and the executable file that was called to populate the count into the count file is exited. In the case of Real-Time Animation, undisplayed data is displayed again if that option was selected, and all other functions of Sliver are enabled.

Since the Grand Tour animation typically consumes CPU bandwidth, this type of animation exits faster if it is paused first.

DataTools Menu



The DataTools menu provides a set of tools for preparing data in a CSV or tab-delimited text file prior to loading the data into Sliver, plus one function to merge multiple KML files into a single KML file. Using these tools is generally much faster and more convenient than opening data files in Excel and manually performing the operations, particularly for large files or for files too large to load into Excel. In fact, these tools are excellent alternatives at times to working with Excel generally, and therefore a standalone DataTools application also exists.

Selecting a tool from the main Tool Selection Window in Figure 1 produces a new window that describes the operation of that tool and the required input fields, as well as listing the first 25 lines of the output file when the operation is completed. An example is shown in Figure 3. The type of file (CSV or tab-delimited TXT) is determined by the input file suffix, with a default of CSV if a different suffix occurs. The tools operate on a data file with or without a header row, depending on a checkbox setting. For tools that have the option of choosing variables (such as *Extract Columns by Name*), the selected variables can also be saved to a text file or read from a text file that lists one variable per line (see the screenshot on the next page).

The functions provided by the DataTools program are:

1. **Preview Top Rows** – Display the top 25 rows of an input file.

2. **View First and Last Rows** – View the first and last rows of one or more files, with an option to select which column variables to display the data for assuming all the files have the same header row. This is very useful when a large file has been split into smaller files and you need to find out which file contains a particular value of a variable that is sequential, such as time or a frame counter. There is also an option to show the row numbers in the output, which is useful for determining the number of rows in a file.

3. **Count Number of Rows** – Display the number of data rows of one or more input files.

4. **Add Column for Row Count** – Insert a new first column with a specified variable name that contains the row number, with the first data row being row 1. This is useful, for example, as the variable to assign to the x-axis of a scatterplot in order to plot a variable against the sample or time.

5. **Add Alphabetic Header Row** – Add a header row to a file of pure data so the columns have variable names. An alphabetic scheme such as in Excel columns (A-Z, AA-AZ, etc.) is used. This function may also be needed in order to perform other functions that involve selecting specific columns.

6. **Extract Range of Rows** – Extract a range of rows between start and end values entered here and store them into an output file.

7. **Extract Every nth Row** – Extract every nth row and store them into an output file (also called *decimating* a file).

8. **Delete Range of Rows** – Delete a range of rows between the start and end values entered here and store the result into an output file.

9. **Extract Columns by Name** – Extract columns of data from an input file based on one or more selected variable names (see the screenshot on the right) and then save them in an output file.

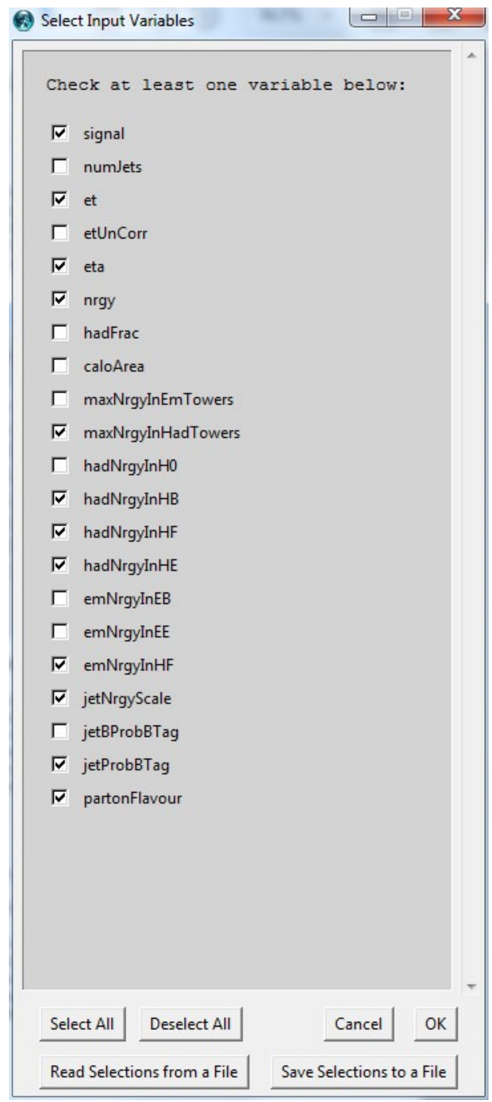
10. **Add Column of Calculated Values** – Add a user-named variable column with values calculated from one or more existing columns using math functions including arithmetic, powers, roots, and logarithmic and trigonometric functions.

11. **View Column Min/Max Values** – View the minimum and maximum values of columns of data from the file based on one or more selected variable names.

12. **Add Rows of Min/Max Values** – Add two rows to the top of the data (after the header row), the first containing the minimum values of the columns of data and the second containing the maximum values. Selecting any single variable in the variable selection window will populate the rows with the minimum and maximum values of each individual column. Selecting more than one variable will produce rows that will share the common minimum and maximum values of these selected variables, while the other values will remain the minimum and maximum values of the individual columns.

    This function is useful to set automatic ranges of a plot such as a parallel coordinate plot such that the axes of the selected variables all have the same range. The two additional rows will appear as extra data points/lines at the ends of the ranges in the plots and may be deleted after plotting the data.

    NOTE: Because two rows of min/max values are added at the top of the output file, do not use this output file for statistical analysis of the data.



Variable Selection Window.

13. **Delete Rows with Column Values** – Delete all rows for which one or more selected variables have the entered numerical value or string.

14. **Extract Rows with Column Values** – Extract all rows for which one or more selected variables have the entered numerical value or string.

15. **Extract Rows with Multi-Column Values** – Extract all rows for which one or more selected variables satisfy Boolean combinations of numerical values or strings across multiple variables.

16. **Extract Rows by Column Value Range** – Extract all rows for which the numerical values of the chosen variable lie within specified minimum and maximum values. A screenshot of this function is shown at the end of this section.

17. **Extract Rows by Column Value Change** – Extract the first row and all rows for which the numerical value or string of a selected variable changes. A new column is added on the left to describe the change, such as "0 to 1".

18. **Sort Rows by Column Value** – Sort all the rows according to the values in a selected column. The sort order can be chosen as increasing or decreasing, and the sort type can be chosen as

    a. Numeric—values are sorted as floating point numbers, so the column must contain numbers in any format
    b. ASCII—values are sorted in Unicode code-point collation order (Example: 12 8 Dog Snake dog horse snake x10y X11y x9y)
    c. Dictionary—case is ignored except as a tie-breaker and any embedded numbers are treated as integers rather than ASCII characters (Example.: 12 8 Dog dog horse Snake snake x9y X10y x11y)

19. **Convert Date/Time to Numeric** – Convert standard date and time formats, which can include time zone mnemonics, in user-selected columns to numeric values, which can then be plotted. The output format can be chosen as the number of days since the oldest date across all selected columns, or seconds relative to the start of today (which is also useful for time intervals such as 01:24:00).

    Acceptable input formats include hh?:mm?:ss?? with or without mm/dd?/yy? or dd monthname ?yy? or day or dd monthname yy or ?CC?yymmdd or ?CC?yy-mm-dd or dd-monthname-?CC?yy. So, for example, *10 December 2014* or *12/10/14* or *12/10/2014* or *141210* or *14-12-10* or *01:32:45* or *01:32:45 12/10/14* are allowed, but for a CSV file the comma delimiter cannot be used, disallowing *December 10, 2014*. If only a time is specified (hh?:mm?:ss??), the current date is assumed. If the string does not contain a time zone mnemonic, the local time zone is assumed. The default year is the current year. If the entered year is less than 100, the years 00-68 are treated as 2000-2068 and the years 69-99 as 1969-1999. Time relative to the current time, such as *4 hours* is also converted, with acceptable units of *year*, *fortnight*, *month*, *week*, *day*, *hour*, *minute* (or *min*), and *second* (or *sec*). The relative units can be singular or plural, as in *3 weeks*, and can be modified with *tomorrow*, *yesterday*, *today*, *now*, *last*, *this*, *next* or *ago*.

20. **Remove Commas from Strings** – Remove commas and any surrounding double quotes from a CSV file. Sliver associates commas in double quotes in CSV files as delimiters, and fails to load such files since the column count is deemed inconsistent. Implementing a real-time check on commas when a file is loaded, though, significantly slows down the input of data, so this DataTools function exists to remove any commas in double quotes, as well as the quotes, from a CSV file prior to reading the file into Sliver.

21. **Replace Strings with Numbers** – Most data analysis tools, including Sliver, can only read numerical data. Using this function, any variable in selected columns that contains strings will have unique numbers assigned to the strings. Certain categorical strings are automatically assigned logical values of 1 and 0 if available for that variable (such as True and False, Yes and No, On and Off, Low and High). Operating only on selected columns allows time-formatted data in variables, which Sliver can read, from being interpreted as strings and replaced with numbers.

Along with the new output file that now contains no strings, this function also creates a text file that lists the string mapping to the numbers. If this string mapping file is entered into the Plot Settings of Sliver, then displays of values on mouse-over, as well as the pop-ups and labels in Google Earth plots, can optionally display the original strings for any mapped value in a variable.

22. **Populate Missing Data Values** – Sometimes data is simply unknown for certain data samples and the value is left empty. Most data analysis tools will not allow such a file to be loaded. This function has options to fill all missing data in a file with a given value or to linearly interpolate between the known values for each variable.

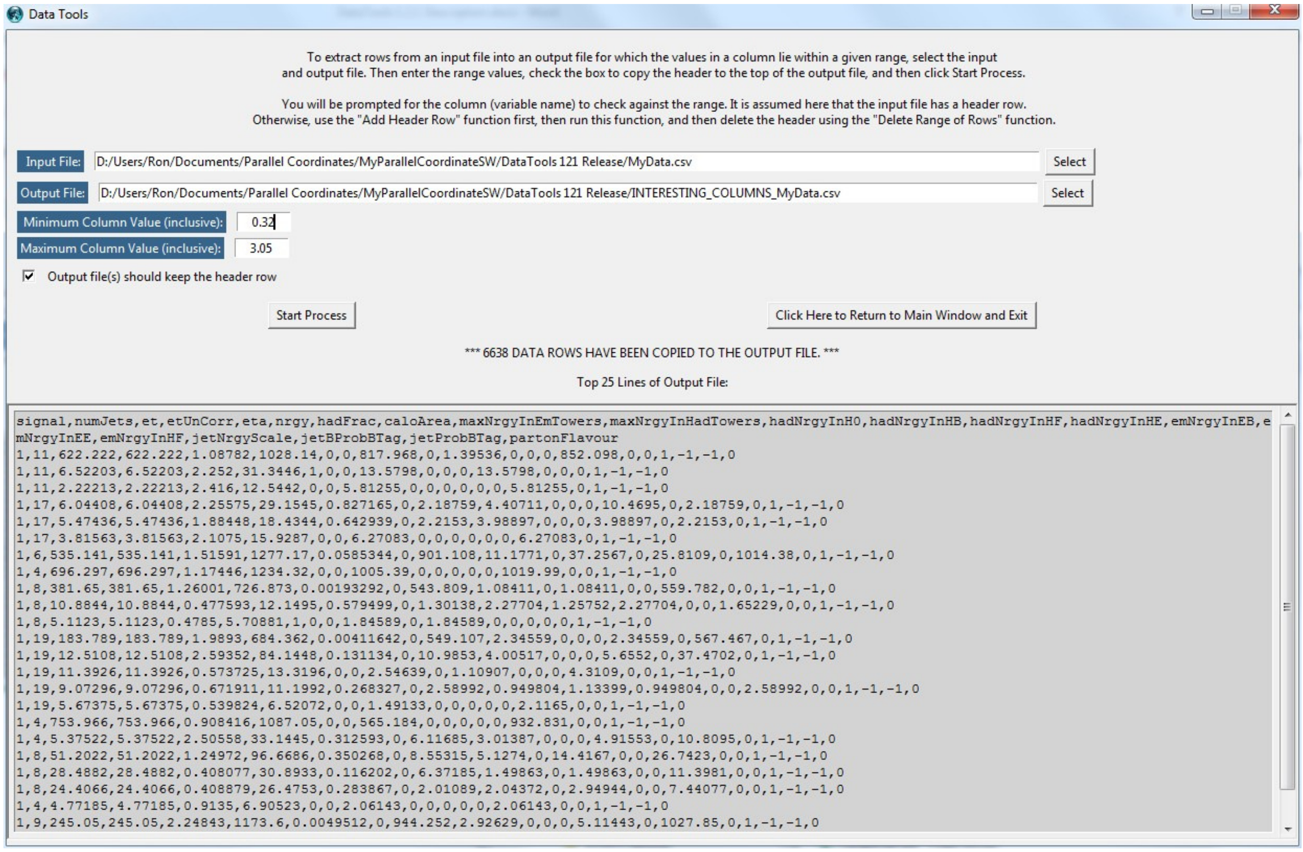
23. **Swap Rows and Columns** – Swap the rows and columns of an input file and store the result into an output file. A header row in the input file will accordingly lie in the first column of the output file.

24. **Merge Columns of Files** – Horizontally (left-right) merge columns from two different files into an output file. The rightmost set of columns has a user-specified prefix added to the variable names in case the two files contain the same kind of data. This function is useful to analyze two sets of data together whose data are row-aligned or have been made row-aligned by deleting rows.

25. **Split a File** – Split a file into multiple files having a given maximum number of lines. This is useful when a file has too many rows to analyze as a group.

26. **Concatenate Files** – Choose up to 20 files and the order in which to concatenate them into an output file. For greater numbers of files, repeat using previous concatenations.

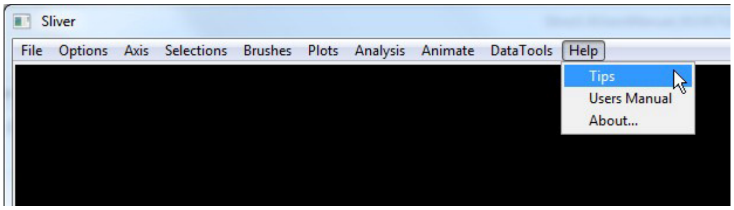
27. **Merge KML Files** – Choose any number of KML files to combine into a single output KML file. This is convenient when multiple Google Earth plots are generated in Sliver that should be displayed as one plot. This merge will work for Sliver-generated KML files and likely for KML files generated by other programs. Alternatively, KML files and images can be zipped together into a KMZ file (with a .kmz suffix), which also merges the files and typically provides a 10x reduction in file size.



Extract Rows by Column Value Range window.

(The variable/column selection window appears after "Start Process" is clicked.)

Help Menu

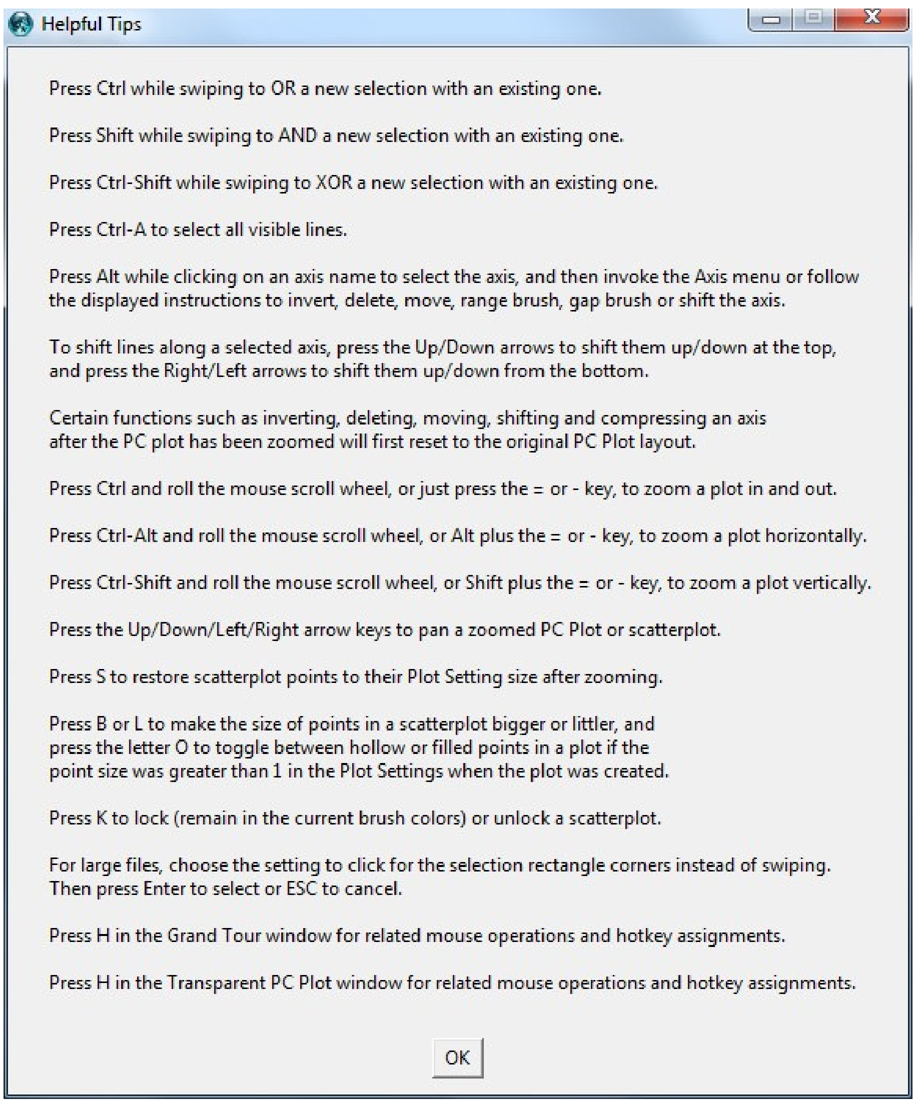


The Help menu provides general information on using the Sliver application, as well as the version being used.

Tips — Displays a window listing the mouse and keyboard interfaces of Sliver as shown below.

Users Manual — Displays this Users Manual in the default PDF reader of your PC.

About… — Displays the version number of this release of Sliver.



Appendix A: Other Parallel Coordinates Software

There are other excellent, free software applications that support parallel coordinates. Some of these run faster than Sliver for large data files, and a few offer features such as built-in, real-time alpha-blending instead of the separate transparency window or export to an alpha-blended Postscript/PDF file that Sliver provides. None offer the animation or Google Earth plotting support, for example, that Sliver provides.

1. Mondrian (*www.theusrus.de/Mondrian/*)

   - An excellent software application. Highly recommended for very large files: fast, handles up to a million records, and natively supports alpha-blending of unselected/uncolored lines.
   - Supports linked Histograms, Boxplots, Scatterplots, Barcharts, Mosaic plots, Missing Value Plots and Parallel Coordinates
   - Rather bland color options
   - Can interface with the R statistical software
   - No installation required—a single JAVA executable
   - Free software under GNU GPL, written by Martin Theus of the University of Augsburg
   - No decimation, zooming/panning, gap brushing, animation, 2D polar scatterplots, 3D scatterplots, Google Earth plots, clustering or Grand Tour that Sliver supports.

2. Ggobi (*www.ggobi.org/*)

   - Supports linked 1D/2D Scatterplots, Barcharts, Time Series Plots and Parallel Coordinates
   - Offers variable transformations and can handle missing data fields
   - No support for alpha-blending
   - Provides a 1D/2D Grand Tour with pursuits as well as manual guidance
   - Offers the R package rggobi that interfaces with GGobi (see their book on the site)
   - Currently in maintenance mode only as the authors develop an implementation directly in R
   - Installation required
   - Free software originated at the University of Washington
   - No alpha-blended export, decimation, zooming/panning, gap brushing, animation, 2D polar scatterplots, 3D scatterplots, clustering or Google Earth plots that Sliver supports.

3. CrystalVision (*ftp://www.galaxy.gmu.edu/pub/*)

   - Supports linked Scatterplot Matrix, Parallel Coordinates, 3D & "4D" Scatterplots
   - Natively supports alpha-blending (no export required as in Sliver)
   - Supports zooming and panning, and sizing of dots
   - Colors are additive, e.g., overlap of red and green lines are yellow, of red, green and blue are white
   - Provides a unique automated Grand Tour animation
   - Inputs tab-delimited TXT (numerical data only) with a minor header change
   - A CPU processing hog, but usable
   - Supports numerical data only
   - Installation required
   - Free software by Edward Wegman of George Mason University
   - No saving, decimation, range brushing, gap brushing, animation, 2D polar scatterplots, 3D scatterplots, clustering or Google Earth plots that Sliver supports.

4. DataViewer (*http://benjamin.web.cern.ch/benjamin/DataViewer.html*)

- A fast data visualization tool under active development for Windows, Mac OSX and Linux
- Natively supports alpha-blending of colored or uncolored lines using the additive color scheme and the alpha-channel (no export required as in Sliver)
- Supports linked Scatterplots and Parallel Coordinates, with horizontal or vertical orientation of the latter. Displays axes values on mouse-over of both types of plots, and a logging function to save actions and statistics of the data.
- Can produce a best order of the axes based on calculated correlation coefficients to give the user the best chance of finding interesting patterns.
- Supports moving and deleting axes. Can also renormalize them for new maxima and minima following data deletions.
- No installation required
- Written by Benjamin Radburn-Smith of CERN
- Actively being developed; at present the released Windows version does not support decimation, zooming/panning, gap brushing, animation, 2D polar scatterplots, 3D scatterplots, Google Earth plots, clustering or Grand Tour that Sliver supports.

5. XmdvTool (*http://davis.wpi.edu/xmdv/*)

- Supports linked Scatterplot Matrix, Parallel Coordinates and Pixel-Based Graphs
- Selection is performed by resizing a band through the parallel coordinate plot
- Automatic or no color brush—no manual brushing other than the current selection
- Provides a true, customizable clustering algorithm rather than just alpha-blending
- Supports numerical data only
- No cropping or hiding of data
- Installation required
- Free software by Matt Ward, et al, of Worcester Polytechnic Institute
- No compound Boolean selections, cropping (hiding), saving, decimation, range brushing, gap brushing, animation, 2D polar scatterplots, 3D scatterplots or Google Earth plots that Sliver supports.

6. Parvis (*www.mediavirus.org/parvis/*)

- Supports Parallel Coordinates only
- Automatic alpha-blending works well
- Manual color brushing by axis range or novel brushing of lines within angles
- Anti-aliased lines. Brushes can be "fuzzy", giving a cool wet look to the clusters.
- Can save currently-brushed line numbers
- Clusters can be formed from AND/OR of brushes and saved with names and colors (Brush List) for sequencing
- Can AND/OR clusters into new clusters with a new color, but cannot show more than one cluster color at one time
- Can zoom into lines at axes to spread out a range of interest
- Internally supports strings
- No installation required—a single JAVA executable
- Free open-source (GPL) software by Florian Ledermann of the VRVis Research Center in Vienna
- No decimation, range brushing, gap brushing, animation, 2D polar scatterplots, 3D scatterplots, Google Earth plots, clustering or Grand Tour that Sliver supports.

Appendix B: References

There are a large number of references on using parallel coordinate plots linked with scatterplots to visualize and analyze multivariate systems. The following are very readable overviews of the subject.

1. Inselberg - *Parallel Coordinates Chapter in **Handbook of Data Visualization***
2. Inselberg - *Multidimensional Detective*
3. Malik - *Interactive Graphics: Exemplified with Real Data Applications* (uses Mondrian)
4. Theus - *Interactive Data Visualization using Mondrian*
5. Inselberg - *Parallel Coordinates: Interactive Visualization for High Dimensions*
6. *Hauser – Angular Brushing of Extended Parallel Coordinates* (uses Parvis)
7. *Cook – Grand Tours, Projection Pursuit Guided Tours, and Manual Controls*

Please visit www.sliversoftware.com to find out how to obtain these and other references.

Appendix C: Format of Sliver Saved Session Files

It may be useful to programmatically assign brushes and states to data prior to opening the data in Sliver. For a large number of data sets this can replace many swipes and button clicks. This preformatting can be done be populating a Sliver session file, which is normally saved after Sliver editing is performed, and restoring the session file rather than opening the original data file. This appendix describes the format of these session files.

Saved session files should have a .csv suffix. They are ASCII text files that can be opened in any text editor. The format of the text is given below, followed by an example.

1) The first line must identify the file as a session file with the text "Saved Session File"

2) The second line is the number of data rows (N).

3) The third line is a listing of whether each axis is inverted or non-inverted, separated by commas, where 0 means inverted and 1 means non-inverted. The values are displayed in the order that the axes are ordered at the time of the session save, left to right. This is the original order of the variables in the input data file only if no axes were moved.

4) The fourth line is a listing of the axes variable names separated by commas. Again, the order of the variable is that of the axes at the time of saving the session file, which will not be the order in the original data file if any axes have been moved.

5) The next N lines consist of the comma-separated data for the axes in the order of the axes at the time of the save.

6) The next line is the number of color brushes assigned (B). This could be 0.

7) The next B lines are the brush colors in the order shown in the Brushes→Show/Hide/Recolor Brushes menu option. The format is "#" followed by two-digit hex values for RGB, such as #00ff00 for green. NOTE: Always use lowercase hex letters.

8) The next line is again the number of color brushes assigned (B).

9) The next B lines contain either 0 or 1 for a brush in the earlier brush order, where 0 means the lines with that brush color were hidden and 1 means they were visible when the session was saved.

10) The next line contains the number of polylines that were color brushed (C). This could be 0.

11) The next C lines list are the indexes into the N lines of step 5 for only those polylines that are color brushed, starting at 0 for the first data row listed in step 5. Polylines that are not color brushed are skipped.

12) The next line is again the number of polylines that were color brushed (C).

13) The next C lines give the colors for each of the lines listed in step 11. The format is again "#" followed by two-digit hex values for RGB, such as #00ff00 for green. NOTE: Always use lowercase hex letters.

14) The next line is the number of polylines that were selected (in purple by default) when the session was saved (S). This could be 0.

15) The next S lines are the indexes into the N lines of step 5 for only those polylines that are selected, starting at 0 for the first data row listed in step 5. Polylines that are not selected are skipped.

The remainder of the lines are optional, but written by Sliver:

16) The next line is the number of axes (A).

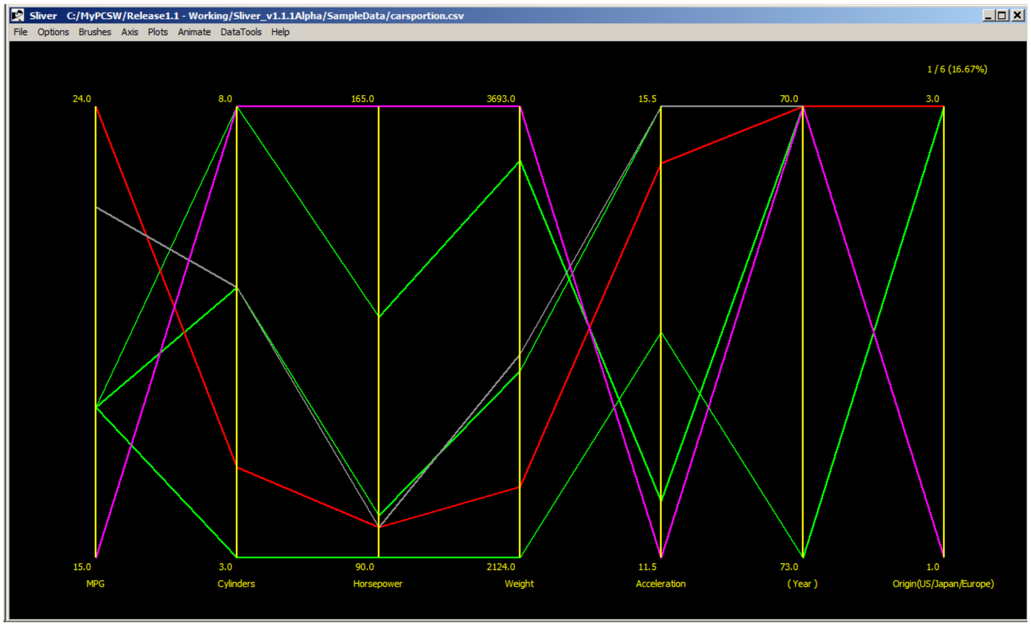
17) The next A lines are comma-separated *current* (after any range adjustments were made) minimum and maximum values for each axis, regardless of whether the axis is inverted. NOTE: There must be some range, so if all data for a variable is the same (say, n), make the range something like (n-1) to (n+1). This is what Sliver does in that case when it reads in a file.

18) The next line is the number of axes (A) again.

19) The next A lines are comma-separated *original* values (before any range adjustments were made) of the minimum and maximum values for each axis as well as the mean and sum of the values for the axes, regardless of whether the axis is inverted. NOTE: There must be some range, so if all data for a variable is the same (say, n), make the range something like (n-1) to (n+1). This is what Sliver does in that case when it reads in a file.

20) The next line is the number of thickened lines (L).

21) The next L lines list are the indexes into the N lines of step 5 for only those polylines that are thickened, starting at 0 for the first data row listed in step 5, and their difference in linewidth from the default value set in the **Options→Plot Settings** menu option, separated by a comma. Polylines that are not thickened are skipped.



Saved Session File	Title Text
6	6 data rows (polylines)
1,1,1,1,1,0,1	Only the 6^{th} axis is inverted
MPG,Cylinders,Horsepower,Weight,Acceleration,Year,Origin(US/Japan/Europe)	Axis names in left-to-right order
18,8,130,3504,12,70,1	6 rows of polyline data
15,8,165,3693,11.5,70,1	
24,4,95,2372,15,70,3	
22,6,95,2833,15.5,70,1	
18,6,97,2774,15.5,70,1	
18,3,90,2124,13.5,73,3	
2	2 color brushes are assigned
#00ff00	First color brush is pure green
#ff0000	Second color brush is pure red
2	Again, 2 color brushes are assigned
1	Polylines of the first color are visible
1	Polylines of the second color are visible
4	4 lines are color brushed
0	The 1st polyline data above is brushed
2	The 3rd polyline data above is brushed
4	The 5th polyline data above is brushed
5	The 6th polyline data above is brushed
4	Again, 4 lines are color brushed
#00ff00	The 1st polyline data above is green
#ff0000	The 3rd polyline data above is red
#00ff00	The 5th polyline data above is green
#00ff00	The 6th polyline data above is red
1	One polyline is selected (in purple)
1	The 2nd polyline data above is selected
7	There are 7 axes
15,24	7 rows of current min/max axis values
3,8	
90.0,165.0	
2124,3693	
11.5,15.5	
70,73	List minimum first even if axis inverted
1,3	

7	There are 7 axes
15,24,19.83333,119	7 rows of original min/max/mean/sums
3,8,35,5.833333	
90.0,165.0,672,112	
2124,3693,17300,2883.33333	
11.5,15.5,83,13.833333	
70,73,423,70.5	List minimum first even if axis inverted
1,3,10,1.66667	
0	There are 0 thickened lines

Appendix D: Format of String Mapping Files

The *Replace Strings with Numbers* function of DataTools creates a new string mapping file as a reference for retrieving the original strings for those replaced by numbers. These can be found manually by simply opening the text file and finding the variable name and its assignments. In addition, if this file is entered into the Plot Settings in Sliver, the **Plots→Display Values on Mouse-Over→With Strings** menu option will access this file and display any original strings instead of values when hovering over lines in the PC Plot. The pop-ups and labels in Google Earth can also be set to access the file and display any original strings.

It's possible, and often useful when strings can be related to values of a variable, to manually create a string mapping file. The format is very simple, and a sample file is shown below. Variables are on their own row and end in a colon without a space. If there are mapped strings, each number and corresponding string immediately follow the row with that variable name, where the number and string are separated by a colon with a space before and after it. The number can be a decimal value instead of an integer, and the number-string pairs can be in any order. There can be many thousands of these pairs for a large dataset.

A blank line then separates this block from a new variable name. If no number-string pairs exist after a variable name, then no values are mapped to strings for that variable, and in fact there is no reason to list that variable at all (they are shown here because the *Replace Strings with Numbers* function of DataTools does show variable names that have no mapped strings). There must be at least one blank line at the end of the file.

```
ID:

Beat:

Case Number:
0 : HZ131333
1 : HZ129324
2 : HZ129627
3 : HT626172
4 : HZ216949

Block:
0 : 030XX S WABASH AVE
1 : 132XX S RIVERDALE AVE
2 : 000XX S LEAMINGTON AVE
3 : 060XX S FRANCISCO AVE

Arrest:
0 : FALSE
1 : TRUE

Domestic:
0 : FALSE
1 : TRUE

Year:
<make sure there is at least one blank line at end of the file!>
```

Copyright Notice and Disclaimer

This program can be used for personal, educational or business purposes, including the publication of output data and any screenshots or other pictorial information produced by it. However, the program cannot be mass-distributed nor incorporated into other products or collections of products without the express consent of the author. The author bears no responsibility for any loss of information, incorrect rendering of data or results, or any other adverse effects of using the Sliver application.

REQUISITE LEGAL STATEMENTS

PROPRIETARY RIGHTS
Sliver contains proprietary and confidential information that is protected by applicable intellectual property and other laws. Except as expressly permitted by applicable law or authorized by Sliver Software, you agree not to modify, rent, lease, loan, sell, distribute or create derivative works based on Sliver, in whole or in part, without express written approval.

DISCLAIMER OF WARRANTIES
Sliver Software and its author expressly disclaim all warranties of any kind, whether express or implied, including, but not limited to the implied warranties of merchantability, fitness for a particular purpose and non-infringement. Sliver Software and its author make no warranty that (i) the software will meet your requirements, (ii) the software will be error-free, (iii) the results that may be obtained from the use of the software will be accurate or reliable, (iv) the quality of the software will meet your expectations, and (v) any errors in the software will be corrected. No advice or information, whether oral or written, obtained by you from Sliver Software or its author shall create any warranty not expressly stated here.

LIMITATION OF LIABILITY
Sliver Software and its author shall not be liable to you for any direct, indirect, incidental, special, consequential or exemplary damages, including, but not limited to, damages for loss of profits, goodwill, use, data or other intangible losses (even if Sliver Software has been advised of the possibility of such damages), resulting from: (i) the use or the inability to use the software; (ii) the cost of procurement of substitute goods and services resulting from any goods, data, information or services obtained or messages received or transactions entered into through, from or as a result of the software; (iii) unauthorized access to or alteration of your data; (iv) statements or conduct of any third party on the software; or (v) any other matter relating to the software.
You agree to indemnify and hold Sliver Software and its author harmless from any claim or demand, including reasonable attorneys' fees, made by any third party due to or arising out of content you make available through the use of the software, or your use of the software.

CONTRIBUTIONS TO SLIVER SOFTWARE
By submitting ideas, suggestions, documents, and/or proposals ("Contributions") to Sliver Software through its suggestion or feedback web pages or otherwise, you acknowledge and agree that: (a) your Contributions do not contain confidential or proprietary information; (b) Sliver Software and its author are not under any obligation of confidentiality, express or implied, with respect to the Contributions; (c) Sliver Software and its author shall be entitled to use or disclose (or choose not to use or disclose) such Contributions for any purpose, in any way, in any media worldwide; (d) Sliver Software and its author may have something similar to the Contributions already under consideration or in development; and (e) you are not entitled to any compensation or reimbursement of any kind from Sliver Software or its author under any circumstances.