Graphing: Scatterplots

We have already used histograms and box plots to examine the distribution of individual variables, now we will move on to using graphs to visually examine the relationships between two variables. The scatterplot is the standard graph for presenting bivariate relationships. Scatterplots can be created with the Graphics menu or through syntax by typing either `scatter` or `twoway scatter`. Similar to other bivariate commands in Stata, the first variable specified is the dependent variable and the second variable listed is treated as the independent variable. As with histograms and other types of graphs, there are almost endless options for modifying the way Stata sets up and displays scatterplots.

Syntax for scatterplot:
```
twoway scatter depvarname indvarname
```

Example:
```
twoway scatter co2 gdp
```

A line of best fit can also be added to the basic scatterplot with the `lfit` command. "A line of best fit" is a straight line that overall best represents the data contained in the scatter plot. This line may pass through some of the points, none of the points, or all of the points. A bivariate linear regression line is the line of best fit for two variables.

Syntax for scatterplot with line of best fit:
```
twoway (scatter depvarname indvarname)(lfit depvarname indvarname)
```

Example:
```
twoway (scatter co2 gdp) (lfit co2 gdp)
```

A lowess smoother can also be added to a scatterplot. The lowess smoother is non-parametric (unlike a linear regression line) and represents the relationship between the data points in a specific section of the scatterplot.

Syntax for lowess smoother:
```
lowess depvarname indvarname
twoway (scatter depvarname indvarname)(lowess depvarname indvarname)
```

Example:
```
lowess co2 gdp  
lowess co2 gdp, bwidth(.1)  
twoway (scatter co2 gdp) (lowess co2 gdp)
```

The bandwidth for a lowess smoother can also be specified using the `bwidth` option. This is a similar concept to specifying the bandwidth for the kernel density line we plotted for individual variables. Specifying a smaller bandwidth will average the relationship over a shorter section of data points than a larger bandwidth.
You can examine bivariate scatterplots for a group of variables using the `graph matrix`. The order in which the variables are specified in this list is not important.

Syntax for graphing a matrix:

```
graph matrix co2 gdp femec hdi
```

**Modifying Scatterplots**

A number of different schemes and options can be specified for scatterplots. Titles for axes and the entire chart can be added, colors and layout types can be changed, etc. The command `graph query, schemes` will display multiple customizable layout options for your graph. You may want to spend some time looking through the different options you have for customizing scatterplots generally by looking at the online help files for scatterplots. You can do this with the syntax `help scatter` or `help graph twoway scatter`.

Options for customizing graphs are specified after commas, in the same way options are specified for other commands we are already familiar with. Remember that only one comma is used in the syntax, even when multiple options are being specified for one graph.

**Labeling Data Points**

To help identify patterns or outliers, individual data points can be labeled on the scatterplots by the identifying variable in the dataset. As the graphs in the scatterplot matrix are rather small, it might be helpful to shorten the identifying variable to two or three characters in some cases (i.e. if the original variable contains more than a few characters). This can be done quickly for text variables by using the `substr` function.

Syntax for `substr` function:

```
gen str3 nat = substr(nation,1,3)
```

The above syntax tells Stata to create a new string variable with 3 characters and name it “nat”. Set the new variable “nat” equal to a shortened version of the text variable “nation” starting at the first character of “nation” and continuing for 3 characters.

The scatterplot matrix can be re-graphed with the shorted nation labels rather than generic symbols. The `mlabel()` option labels the points with the names in the “nat” variable. The `msymbol()` option specifies which, if any, symbol markers should be attached to the data points. By default, Stata will add dots to all the data points; here we are specifying that we do not want any symbol markers.

Syntax for graphing a matrix with labeled data points:

```
twoway scatter co2 gdp, msymbol(none) mlabel(nat)
```
Graphing: Line Graphs
Although not as commonly used a scatterplots, line graphs can also be used to display bivariate relationships. Unlike scatterplots, line graphs give you a line connecting all the data points. Producing a line graph involves an important intermediate step that scatterplots do not--your data must be sorted on the independent (x) variable. If your data is not sorted, Stata will give you a graph that looks like scribbles. Likewise, if you are trying to produce a line graph for your data and end up with an unintelligible mess, the first solution you should try is sorting the data according to the independent variable.

Syntax:

```
sort indvarname
graph twoway line depvarname indvarname
```

Example:

```
sort gdp
graph twoway line co2 gdp
```

Data Description with Multiple Variables
While examining properties of individual variables is important for understanding variable contents and distributions, we are often most interested in the relationship between two (or more) variables. Scatterplots are graphical representations of bivariate relationships, and there are several other analyses that can be conducted to assess these relationships.

Correlation
Correlation is a measure of linear association between two variables. The `correlate` command displays the correlation matrix for a group of specified variables. If a variable list is not specified, the correlation matrix includes all variables in the dataset.

Syntax:

```
correlate[varlist] [if] [in] [weight] [, correlate_options]
```

Example:

```
correlate co2 gdp
```

The `pwcorr` command will also display all pairwise correlations for the variables specified and has the allowable option `sig`, which displays the p-value for a t-test of the correlation coefficients. The `obs` option tells how many cases were included in each correlation.

Syntax:

```
pwcorr[varlist] [if] [in] [weight] [, options]
```

Example:

```
pwcorr co2 gdp hdi, sig obs
```

Note: `pwcorr` omits cases between pairs of variables where data is missing on at least one of the two variables in the pair. This is called “pairwise deletion” and explains why the numbers of observations differ in each cell of the matrix. If you have missing data and plan to do further analyses, you may want to omit all cases that are missing for any variable in the correlation matrix. This is “casewise deletion” and is how missing data are handled with the `correlate` command.
Regression

Linear regression examines the relationship between a dependent and two (or more) independent variables. The outcome or dependent variable should be continuous for ordinary least squares regression. Independent variables are continuous or binary. For the `regress` command, the dependent variable is listed first, followed by the independent variable(s) it is to be regressed on.

**Syntax:**
```
regress depvarname indvarname(s) [if] [in] [weight] [, options]
```

**Example:**
```
regress co2 gdp
regress co2 gdp, beta
```

You can also ask that Stata return the standardized beta coefficients in addition to the slope coefficients for each independent variable by specifying the `beta` option.

In addition, the width of the confidence interval (or the level of confidence) can be specified by using the `level` option. The default confidence level in Stata is always 95%.

**Syntax for specifying the confidence interval:**
```
regress depvarname indvarname(s), level(#)
```

**Example:**
```
regress co2 gdp, level(90)
```

Note: You can retype the estimation command without arguments to redisplay the most recent estimation results. For instance, after fitting a model with `regress`, you can see the estimates again by typing `regress` by itself. You do not have to do this immediately – any number of commands can occur in between the estimation and the replaying. In fact, you can even replay the last estimates after the data have changed or you have dropped the data altogether.

**Graphing the regression line**

For bivariate regression, the least squares line can be examined visually by adding a line of best fit to a scatterplot.

**Syntax:**
```
twoway (scatter xvar yvar) (lfit xvar yvar)
```

**Example:**
```
twoway (scatter co2 gdp) (lfit co2 gdp)