

Added-variable plots with confidence intervals

John Luke Gallup
Portland State University
jlgallup@pdx.edu

Abstract. An added-variable plot is a effective way to show the correlation between an independent variable and a dependent variable *conditional* on other independent variables. For multivariate estimation, a simple scatterplot showing x versus y is not adequate to show the *partial* correlation of x with y because it ignores the impact of the other covariates. Added-variable plots are especially effective for showing the correlation of a dummy x variable with y because the dummy variable conditional on other covariates becomes a continuous variable, making the relationship easier to visualize.

Added-variable plots are also useful for spotting influential outliers in the data which affect the estimated regression parameters. Stata provides added-variables plots after OLS regressions with the `avplot` command. The new command `avciplot` adds a confidence interval and other options to the `avplot` command.

Keywords: `st0001`, `avciplot`, `avciplots`, `avplot`, added-variable plot, confidence interval, postestimation diagnostics

1 Introduction

An added-variable plot is a scatterplot of the transformations of an independent variable (say, x_1) and the dependent variable (y) that nets out the influence of all the other independent variables. The fitted regression line through the origin between these transformed variables has the same slope as the coefficient on x_1 in the full regression model which includes all the independent variables.

An added-variable plot is the multivariable analogue of using a simple scatterplot with a regression fit when there are no other covariates to show the relationship between a single x variable and a y variable. An added-variable plot is a visually compelling method for showing the nature of the *partial* correlation between x_1 and y as estimated in a multiple regression.

The plot of the fitted regression line alone does not show whether the slope of the line (the regression coefficient on x_1) is statistically significant. Including a confidence interval around the regression line in the plot clarifies this. `avciplot` includes a confidence interval in the added-variable plot, unlike Stata's built-in `avplot` command. To add a confidence interval to an `avplot` would require that almost all the code in `avplot` be reconstructed manually, hence the reason for creating the `avciplot` command.

At a chosen level of statistical significance (typically 95%), if the confidence interval does not include a slope of zero then the coefficient on x_1 is statistically significant. The

confidence interval helps the viewer see precisely how the sample data fit the partial correlation of x_1 with y .

Outliers in a simple scatterplot of x_1 versus y may no longer be outliers when other covariates are included in the model. An added-variable plot is a handy visual diagnostic for spotting influential outliers after conditioning on the other covariates in the model.

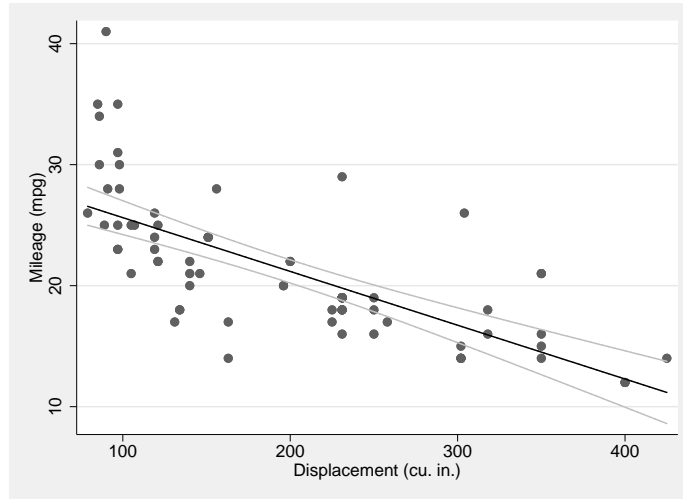
A simple scatter plot of x_1 versus y can be a misleading representation of the relationship of these variables when controlling for other covariates. Even when it is more or less representative, meaning that the other variables do not affect the partial correlation of x_1 versus y very much, it is still difficult to grasp the correlation between a dummy variable and the y variable because all the dummy variable observations are clustered at 0 and 1. Since the residual of a dummy variable conditional on the other covariates has continuous rather than discrete values, the added-variable plot has the virtue of making the partial correlation of a dummy x_1 and y easier to visualize. (See the last example using `avciplots` below).

In addition to providing confidence intervals, `avciplot` has two useful options not available in `avplot`. The `generate` option allows the user to save the values of the residuals of x_1 and y conditional on the other covariates for subsequent use in calculations or additional graphs. The `xlim` and `ylim` options allow the user to exclude extreme observations from the scatterplot (without removing their influence from the regression estimate). These options should be used with care, but can make the graph more clear when distant but inconsequential outliers affect the graph's scale as often happens with large datasets.

2 Why do we need added-variable plots and where do they come from?

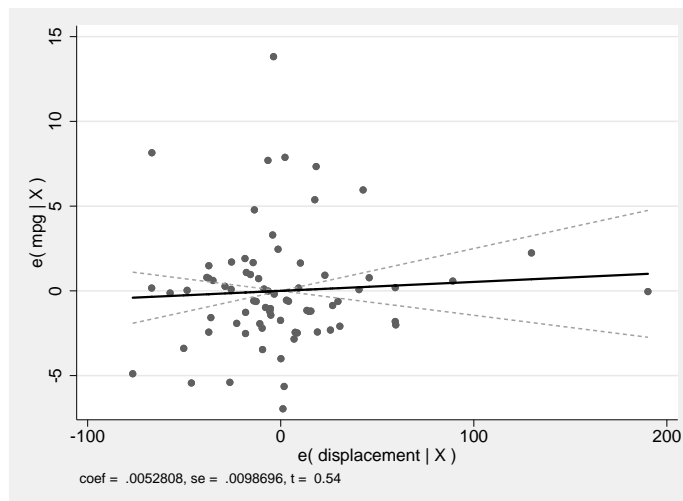
The purpose of multivariate regression is to assess the influence of each independent variable on the dependent variable while simultaneously accounting for the influence of all the other independent variables. The regression coefficient quantifies the *partial* correlation of an independent variable (x_1) on the dependent variable (y) controlling for the other independent variables (\mathbf{x}). A simple scatterplot is an effective visual presentation of the unconditional correlation of x_1 with y , but an added-variable plot is needed to display the partial correlation of x_1 with y conditional on other \mathbf{x} variables. The partial correlation typically has a different magnitude and may even have a different sign than the unconditional correlation.

For example, there is a negative correlation between automobile fuel efficiency (`mpg`) and engine size (`displacement`) in Stata's `auto` dataset. This is clear to the eye from a scatterplot of the data with a regression line:



In a multiple regression which includes the auto's weight in addition to displacement, however, displacement has a small but positive partial correlation with mpg in this sample. Presumably the problem with large engines is their weight, not their size. How can we display this relationship graphically, the partial correlation of displacement with mpg while controlling for the influence of weight? With `avciplot`:

```
. regress mpg displacement weight
(output omitted)
. avciplot displacement
```



The added-variable plot shows that the partial correlation of displacement with

`mpg` when `weight` is also included in the model is dramatically different from the unconditional correlation of `displacement` with `mpg`. The negative unconditional correlation becomes a positive correlation when it is conditioned on the regressor `weight`.

The slope of the fitted regression line though the origin in the added-variable plot is equal to the estimated coefficient on `displacement` in the preceding regression. The confidence interval bounded by the dashed lines includes the 0 line on the vertical axis. This indicates that the slope of the regression line is not significantly different from zero at the five percent level (the default), which is confirmed by the t statistic of 0.54 printed at the bottom of the graph.

The next subsections explain the statistical basis for added-variable plots. If that is not your interest, please skip to Section 3, the Stata syntax of `avciplot`, or detailed examples of its use in Section 5.

2.1 Partial Regression

The statistical basis for an added-variable plot is partial regression. Uncharacteristically, the Stata documentation for `avplot` does not explain the methods behind added-variable plots, so they are covered in detail here.

Partial regression is also known as the Frisch-Waugh-Lovell Theorem after Frisch and Waugh (1933) and Lovell (1963). Partial regression shows that the partial correlation of x_1 , one of multiple independent variables, with the dependent variable y can be found by “partialing out” the influence of the other independent variables on both x_1 and y first and then regressing the “partialed” x_1 on the “partialed” y .

Take the standard linear regression equation relating the dependent variable, y , to $K - 1$ independent variables x_1, \dots, x_{K-1} , and an intercept term and an error term ε :

$$y_i = \beta_1 x_{1i} + \dots + \beta_{K-1} x_{K-1,i} + \beta_K + \varepsilon_i.$$

where the intercept term is placed after the x variables for notational convenience.

If we draw a sample of N observations of data that conform to this relationship, we have $n \times 1$ data vectors of the dependent variable \mathbf{y} and the K independent variables (including $\mathbf{x}_K \equiv \mathbf{1}$, a vector of ones, for the intercept β_K), $\mathbf{x}_1, \dots, \mathbf{x}_K$. Combining all the independent variables into an $n \times K$ matrix \mathbf{X} , the data fit the equation

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

where $\boldsymbol{\beta}$ is a $K \times 1$ vector of unknown parameters and $\boldsymbol{\varepsilon}$ is an $n \times 1$ vector of the unobserved errors.

The ordinary least squares (OLS) estimator \mathbf{b} is derived by minimizing the sum of squared residuals ($\boldsymbol{\varepsilon}'\boldsymbol{\varepsilon}$ where $\hat{\boldsymbol{\varepsilon}} = \mathbf{y} - \mathbf{X}\mathbf{b}$) and solving the normal equation

$$\mathbf{X}'\mathbf{X}\mathbf{b} = \mathbf{X}'\mathbf{y}. \tag{1}$$

We can partition the \mathbf{X} matrix into $\mathbf{X} = [\mathbf{x}_1 \mathbf{X}_2]$ where $\mathbf{X}_2 = [\mathbf{x}_2 \cdots \mathbf{x}_K]$, partition \mathbf{b} into $\mathbf{b} = \begin{bmatrix} b_1 \\ \mathbf{b}_2 \end{bmatrix}$ where $\mathbf{b}_2 = \begin{bmatrix} b_2 \\ \vdots \\ b_K \end{bmatrix}$, and rewrite (1) as

$$\begin{bmatrix} \mathbf{x}'_1 \mathbf{x}_1 & \mathbf{x}'_1 \mathbf{X}_2 \\ \mathbf{X}'_2 \mathbf{x}_1 & \mathbf{X}'_2 \mathbf{X}_2 \end{bmatrix} \begin{bmatrix} b_1 \\ \mathbf{b}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{x}'_1 \mathbf{y} \\ \mathbf{X}'_2 \mathbf{y} \end{bmatrix}.$$

With some manipulation, we can solve for $b_1 = (\mathbf{x}'_1 \mathbf{M}_2 \mathbf{x}_1)^{-1} \mathbf{x}'_1 \mathbf{M}_2 \mathbf{y}$ where $\mathbf{M}_2 = (\mathbf{I} - \mathbf{X}_2 (\mathbf{X}'_2 \mathbf{X}_2)^{-1} \mathbf{X}'_2)$. Since \mathbf{M}_2 is symmetric and idempotent, we can rewrite b_1 as

$$b_1 = (\mathbf{x}'_1 \mathbf{M}'_2 \mathbf{M}_2 \mathbf{x}_1)^{-1} \mathbf{x}'_1 \mathbf{M}'_2 \mathbf{M}_2 \mathbf{y} = (\mathbf{e}'_{\mathbf{x}_1} \mathbf{e}_{\mathbf{x}_1})^{-1} \mathbf{e}'_{\mathbf{x}_1} \mathbf{e}_{\mathbf{y}} \quad (2)$$

where $\mathbf{e}_{\mathbf{x}_1} = \mathbf{M}_2 \mathbf{x}_1$ and $\mathbf{e}_{\mathbf{y}} = \mathbf{M}_2 \mathbf{y}$.

By inspecting the equation for \mathbf{M}_2 , we can see that $\mathbf{e}_{\mathbf{y}} = \mathbf{M}_2 \mathbf{y}$ is the vector of residuals from the regression of \mathbf{y} on \mathbf{X}_2 and likewise $\mathbf{e}_{\mathbf{x}_1} = \mathbf{M}_2 \mathbf{x}_1$ is the vector of residuals from the regression of \mathbf{x}_1 on \mathbf{X}_2 .

$\mathbf{e}_{\mathbf{y}}$ and $\mathbf{e}_{\mathbf{x}_1}$ can be interpreted as \mathbf{y} and \mathbf{x}_1 purged of the influence of the \mathbf{X}_2 variables. $\mathbf{e}_{\mathbf{y}} = \mathbf{y} - \hat{\mathbf{y}}_{\mathbf{x}_2}$, where $\hat{\mathbf{y}}_{\mathbf{x}_2}$ is the predicted value of \mathbf{y} from the regression of \mathbf{y} on \mathbf{X}_2 . That is, $\mathbf{e}_{\mathbf{y}}$ is what is left over when all the variation in \mathbf{y} that can be predicted by \mathbf{X}_2 has been subtracted out. Similarly for $\mathbf{e}_{\mathbf{x}_1}$. So the correlation of $\mathbf{e}_{\mathbf{y}}$ and $\mathbf{e}_{\mathbf{x}_1}$ is the partial correlation \mathbf{y} and \mathbf{x} conditional on \mathbf{X}_2 .

This decomposition gives rise to the added-variable plot. A scatterplot of the values in $\mathbf{e}_{\mathbf{x}_1}$ versus $\mathbf{e}_{\mathbf{y}}$ will show the correlation of the x_1 variable with the y variable, *controlling for the influence of the other independent variables* in the multiple regression. From equation (2), we can see that the OLS estimate of β_1 , b_1 , is the result of regressing $\mathbf{e}_{\mathbf{y}}$ on $\mathbf{e}_{\mathbf{x}_1}$. Thus the OLS linear fit of the data in the scatterplot of $\mathbf{e}_{\mathbf{x}_1}$ versus $\mathbf{e}_{\mathbf{y}}$ is equal to b_1 , the estimated partial effect of x_1 on y .

This is what we were seeking: a way of displaying the relationship between x_1 and y , controlling for the effect of the other independent variables in the regression. Stata's built-in command `avplot` creates a scatterplot of $\mathbf{e}_{\mathbf{x}_1}$ versus $\mathbf{e}_{\mathbf{y}}$ and displays the linear fit line with a slope of b_1 . `avciplot` also creates a scatterplot of $\mathbf{e}_{\mathbf{x}_1}$ versus $\mathbf{e}_{\mathbf{y}}$ and displays the linear fit line, with the addition of confidence interval boundaries above and below the regression line. The formula for the confidence interval around the linear fit line in $(\mathbf{e}_{\mathbf{x}_1}, \mathbf{e}_{\mathbf{y}})$ space is derived in section **4 Methods and Formulas**, below.

3 Stata syntax

`avciplot indepvar [, options]`

`avciplots [, options combine_options]`

<i>options</i>	Description
Plot	
<i>marker_options</i>	change look of markers (color, size, etc.)
<i>marker_label_options</i>	add marker labels; change look or position
Regression line	
<code><u>r</u>lopts(<i>cline_options</i>)</code>	affect rendition of the regression line
<code><u>n</u>ocoeff</code>	turns off display of coefficient below graph
Confidence interval	
<code><u>c</u>iopts(<i>ci_options</i>)</code>	affect rendition of the confidence interval lines
<code><u>n</u>oci</code>	turns off confidence interval
<code><u>c</u>iunder</code>	puts confidence interval underneath the scatter
<code><u>l</u>evel(#)</code>	specifies the confidence level
<code><u>c</u>iplot(<i>plotype</i>)</code>	how to plot CI; default is <code>ciplot(rline)</code> ; a common alternative is <code>ciplot(rarea)</code>
Y axis, X axis, Titles, Legend, Overall	
<code><i>twoway_options</i></code>	any of the options documented in [G-3] <i>twoway_options</i> other than <code>by()</code>
avciplot - only options	
<code><u>x</u>lim(#[#]), <u>y</u>lim(#[#])</code>	limit the ranges of the <i>x</i> and <i>y</i> residuals displayed
<code><u>g</u>enerate(<i>exvar</i>, <i>eyvar</i>)</code>	save the values of <i>x</i> and <i>y</i> residuals in new variables
<code><u>n</u>odisplay</code>	suppress display of plot
<code><u>a</u>ddplot(<i>plot</i>)</code>	add other plots to the generated graph
avciplots - only option	
<code><i>combine_options</i></code>	any of the options documented in [G-2] <i>graph combine</i>

Description of `avciplot` and `avciplots`

`avciplot` is a post-estimation command that creates an added-variable plot (*a.k.a.* partial-regression leverage plot, partial regression plot, or adjusted partial residual plot) after the `regress` command. It differs from Stata's built-in `avplot` command by adding confidence intervals around the regression line and various options.

`avciplots` creates of a matrix of added-variable plots of all the *indepvars*.

indepvar is an independent (x) variable (*a.k.a.* predictor, carrier, or covariate) that may or may not have been included in the preceding regression. The user would choose an *indepvar* not already in the regression to evaluate whether it is worthwhile to include it.

`avciplot` shows the partial correlation between one *indepvar* and the *depvar* from a multivariate regression.

Besides showing the relationship between the *indepvar* and the *depvar* controlling for the other regressors, `avciplot` is useful for visually identifying which outlier observations have a big effect on the estimated coefficient.

After an OLS regression of x_1 and \mathbf{x}_2 on y , the plotted $\mathbf{e}(\mathbf{x}|\mathbf{X})$ values are the residuals from the regression of x_1 on the other \mathbf{x}_2 variables in the original regression, and the plotted $\mathbf{e}(y|\mathbf{X})$ values are the residuals from the regression of y on the other \mathbf{x}_2 variables.

The fitted line shown in the graph is the least squares fit between the residuals $\mathbf{e}(\mathbf{x}|\mathbf{X})$ and $\mathbf{e}(y|\mathbf{X})$. The fitted line has the same slope as the estimated coefficient on the *indepvar* in the preceding regression.

Due to their construction, the residuals $\mathbf{e}(\mathbf{x}|\mathbf{X})$ and $\mathbf{e}(y|\mathbf{X})$ each have a mean of zero, and the regression line fitted between them passes exactly through $\mathbf{e}(\mathbf{x}|\mathbf{X})=0$ and $\mathbf{e}(y|\mathbf{X})=0$. At that point, the confidence interval has zero width, giving it an unfamiliar shape.¹

Options for `avciplot`

Plot

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] *marker_options*.

marker_label_options specify whether and how markers are to be labeled; see [G-3] *marker_label_options*.

Regression line

`rlopts`(*cline_options*) affects the rendition of the regression (fitted) line. See [G-3] *cline_options*.

`nocoef` turns off display of the values of the coefficient, standard error and t statistic from the regression line below the graph.

Confidence interval

`noci` turns off display of the confidence interval on the graph.

`ciunder` causes confidence interval to be graphed underneath the scatterplot (i.e. the data scatter is visible on top of confidence interval). This is mainly useful when

1. This also happens in a conventional regression when all the independent variables have a value of zero and there is no constant term.

graphing a solid confidence interval with option `ciplot(rarea)`.

`level(#)` specifies the confidence level, in percent, for the confidence interval around the regression line. The default is `level(95)` or as set by `set level`; see [U] **20.8 Specifying the width of confidence intervals**.

`ciopts(cline_options)` affects how the upper and lower confidence interval lines are rendered. See [G-3] *cline_options*. If you specify `ciplot()`, then rather than using *cline_options* you should specify whichever options are appropriate for the *plotype*.

`ciplot(plotype)` specifies how the confidence interval is to be plotted. The default is `ciplot(rline)`, meaning that the prediction will be plotted by `graph twoway rline`.

A common alternative is `ciplot(rarea)`, which will substitute lines around the prediction for shading. See [G-2] *graph twoway* for a list of *plotype* choices. You may choose any *plotypes* that expect two y variables and one x variable.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G-3] *twoway_options*, excluding `()`. These include titling the graph (see [G-3] *title_options*) and saving the graph to disk (see [G-3] *saving_option*).

avciplot-only options

`xlim(# [#])`, `ylim(# [#])` constrain the range of the *indepvar* and *depvar* residuals displayed. If only one number is specified, residuals with a value below that number will not be displayed in the scatterplot. If two numbers are specified, residuals below the first number and above the second number will not be displayed.

Excluding observations of the residuals does not affect the slope of the regression line in the graph. The purpose of these options is to avoid a few outlying observations dramatically extending the range of the x or y axis, obscuring display of the relationship between the variables. As usual, care should be taken to make sure that the undisplayed observations are not important to the estimated relationship and are noted in the text.

`generate(exvar eyvar)` saves the values of the x and y residuals in variables named by the user. The user must specify two variable names for *exvar* and *eyvar*. These residuals can be used for subsequent calculations or graphing commands. See section **4 Methods and Formulas** below for how to access the estimate b_1 as well as its standard error and how to calculate the regression fit and confidence intervals.

`nodisplay` suppresses display of the plot. This is mainly useful for users creating their own plots from variables created with `generate()`.

`addplot(plot)` provides a way to add other plots to the generated graph. See [G-3] *addplot_option*.

avciplots-only options

combine_options are any of the options documented in [G-2] *combine_options* for arranging a matrix of plots in a single image.

Stored results

`avciplot` stores the following in `r()`:

Scalars

<code>r(coef)</code>	the estimated coefficient of the added variable
<code>r(se)</code>	the standard error of the estimated coefficient

4 Methods and Formulas

Since `avciplot` is a `regress` post-estimation command, the preceding `regress` command will have the form

$$\text{regress } y \ x_1 \ \mathbf{x}_2 \tag{3}$$

where y is the *depvar*, x_1 is one of the *indepvars* and \mathbf{x}_2 is a vector of the other *indepvars*. This will be followed by the command

$$\text{avciplot } x_1$$

`avciplot` allows for x_1 not to be included in the preceding `regress` *indepvars*. In that case there is some adjustment to these formulas, principally to estimate the full `regress` model including x_1 for the first time as in Equation 3.

`avciplot` calculates residuals \mathbf{e}_y and $\mathbf{e}_{\mathbf{x}_1}$ in equation (2) from

```
regress y x2
predict e_y, resid
regress x1 x2
predict e_x1, resid
```

using the same weights and sample restrictions as in Equation 3.

The coefficient b_1 and its standard error $\hat{\sigma}_{b_1}$ (up to a degree-of-freedom adjustment) could be calculated by regressing the residuals $\mathbf{e}_{\mathbf{x}_1}$ on \mathbf{e}_y , but it is not necessary because they are already available from the regression in Equation 3.² By default, `avciplot`

2. This also eliminates the need to worry about heteroskedasticity corrections which may have been implemented in the preceding regression since they affect only the standard errors of the estimates, not the values of the residuals $\mathbf{e}_{\mathbf{x}_1}$ and \mathbf{e}_y . If the user is interested in verifying that the residuals are calculated correctly (consistent with the coefficient b_1 in the preceding regression), there is an otherwise undocumented `avciplot` option `debug` which calculates b_1 as the coefficient on $\mathbf{e}_{\mathbf{x}_1}$ from

```
regress e_y e_x1, nocons
```

and stores the result in `r(b_check)`. The `regress` command above will not calculate the correct standard error for b_1 . The regression in Equation 3 has $n - K$ degrees of freedom where n is the

displays a confidence interval around the predicted fit from the regression of \mathbf{e}_y on \mathbf{e}_{x_1} . The fitted values of \mathbf{e}_y are $\hat{\mathbf{e}}_y = \mathbf{e}_{x_1} b_1$. The confidence interval boundaries are $\hat{\mathbf{e}}_y \pm t_{\alpha/2[n-K]} \mathbf{e}_{x_1} \hat{\sigma}_{b_1}$, where $t_{\alpha/2[n-K]}$ is the $\alpha/2$ percentile of the cumulative t distribution with $n - K$ degrees of freedom and $\alpha = 1 - \text{level}/100$.

5 Examples of `avciplot` and `avciplots` in use

Since `avciplot` and `avciplots` are `regress` post-estimation commands, we first load Stata's `auto` example dataset and then run the regression discussed in the introduction to relate `displacement` (engine size) and `weight` of cars to their fuel efficiency (`mpg`).

```
. sysuse auto, clear
(1978 Automobile Data)
. regress mpg displacement weight
```

Source	SS	df	MS	Number of obs	=	74
Model	1595.40969	2	797.704846	F(2, 71)	=	66.79
Residual	848.049768	71	11.9443629	Prob > F	=	0.0000
Total	2443.45946	73	33.4720474	R-squared	=	0.6529
				Adj R-squared	=	0.6432
				Root MSE	=	3.4561

mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
displacement	.0052808	.0098696	0.54	0.594	-.0143986 .0249602
weight	-.0065671	.0011662	-5.63	0.000	-.0088925 -.0042417
_cons	40.08452	2.02011	19.84	0.000	36.05654 44.11251

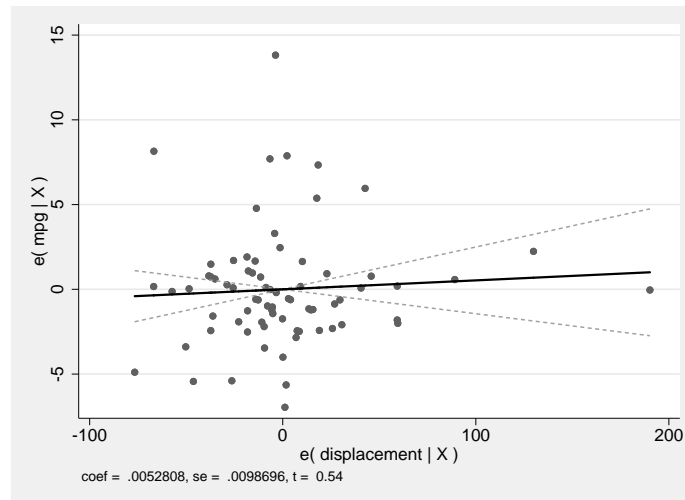
Though the simple correlation of `displacement` and `mpg` is strongly negative, if we also condition on `weight` then `displacement` and `mpg` have a positive insignificant partial correlation. This is presented visually in an added-variable plot:

number of observations and K is the number of regressors including the constant, but the regression above has only one regressor, \mathbf{e}_{x_1} . The correct standard error for b_1 can be calculated with

```
regress e_y e_x1, nocons dof(='e(df_r)')
```

where `dof()` is an undocumented `regress` option for modifying the degrees of freedom and `e(df_r)` is the degrees of freedom from the regression in Equation 3.

```
. avciplot displacement
```



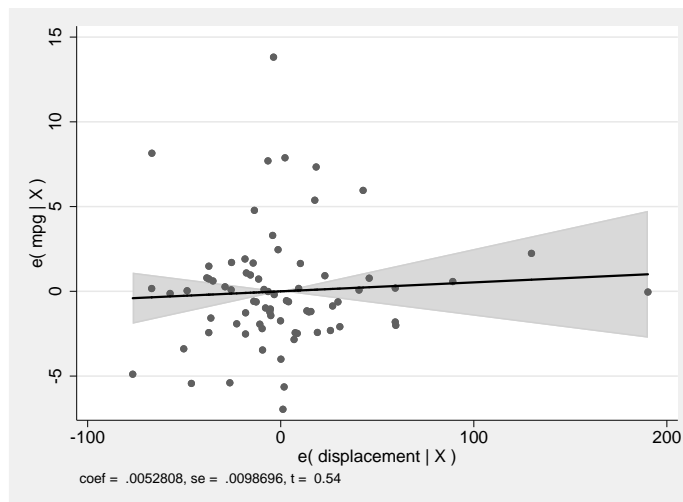
The scatter plot shows the values of the residuals e_{x_1} versus e_y . The solid line is the regression fit of these values and the dashed lines are the limits of the 95% confidence interval around the regression fit. The slope of the regression fit in the added-variable plot is equal to the coefficient on `displacement` in the preceding regression, the value of which is printed at the bottom of the graph.

We can see that the partial correlation of `displacement` with `mpg` is not statistically significant (at the 5% level) since the confidence interval includes zero.

The added-variable plot shows the correlation of the x_1 variable, `displacement`, conditional on all the other independent variables in the regression, with the y variable, `mpg`, also conditional on all the other regressors. That is, it shows the correlation of one x with y , netting out the influence of all the other covariates.

`avciplot` can display the confidence interval as a solid area (similar to the `lfitci` graphs) by using the option `ciplot(rarea)` rather than the default of delineating the interval by two dashed lines. The `ciunder` option causes the scatter plot to be superimposed on the confidence interval rather than vice versa, so that data points within the interval are still visible:

```
. avciplot displacement, ciplot(rarea) ciunder
```



Added-variable plots are a good diagnostic for finding outlier observations which influence the partial correlation of a regressor of interest, in this case `displacement`.

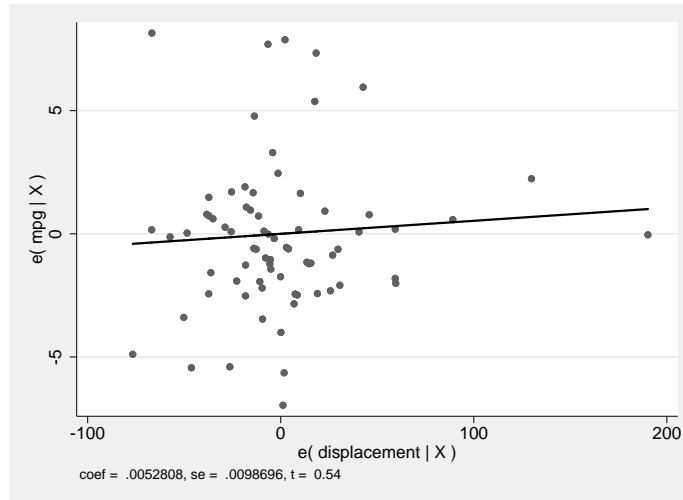
There is a clear outlier in the $e(\text{mpg} | X)$ vertical axis with a value of about 14. It is also clear that this outlier has little affect on the slope of the regression line because it has a value $e(\text{displacement} | X)$ of about zero. Including this outlier shrinks the rest of the graph vertically, which makes it harder to see the estimated relationship.

We can exclude the display of this observation with the option `ylim(-10 10)` to magnify the rest of the graph. The lower limit of `ylim` has no affect because there are no $e(\text{mpg} | X)$ values below -10. The `ylim` and `xlim` options are not available in `avplot` so this could be a reason to use `avciplot` even if you don't want to display a confidence interval.

Another difference between `avplot` and `avciplot` is the ability to save the values of e_{x_1} and e_y (*i.e.* $e(x|X)$ and $e(y|X)$) for later use with the `generate` option, perhaps to create a more complicated graph after running the `avciplot` command.

The following command implements the `ylim`, `generate` and `noci` (no confidence interval display) options:

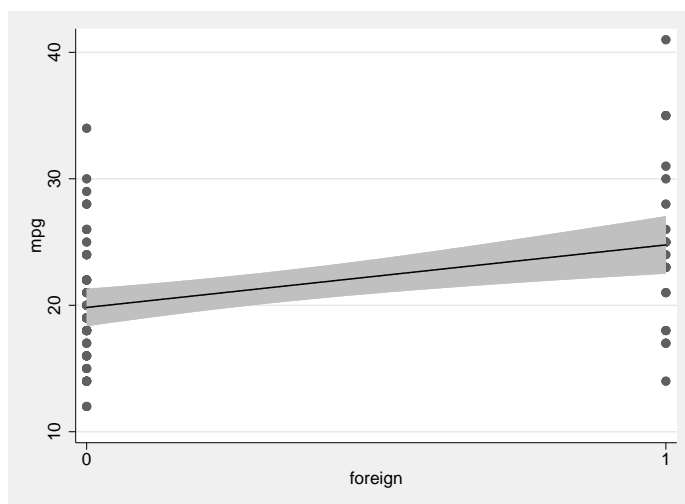
```
. avciplot displacement, ylim(-10 10) noci generate(ex ey)
```



The new variables `ex` and `ey`, containing the residuals of the x and y variables conditional on all the other regressors, are added to the dataset in memory.

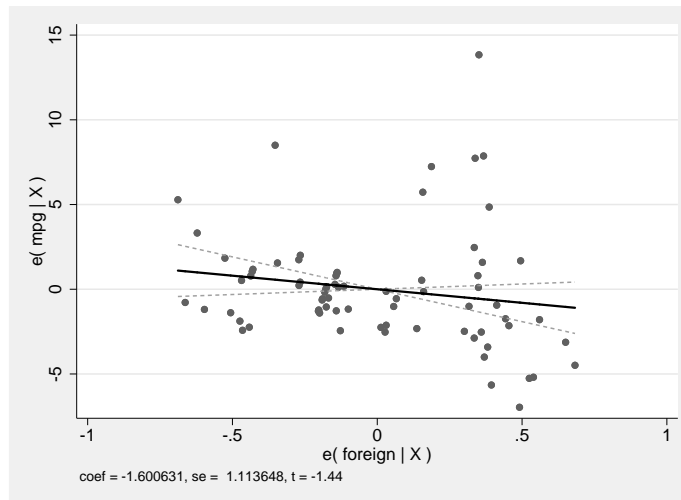
An added-variable plot is an especially effective way of showing the (conditional) relationship of a dummy variable with the dependent variable. A simple scatter plot of a dummy variable like `foreign` versus `mpg` only displays two values on the horizontal axis, making it difficult to discern the relationship visually:

```
. twoway scatter mpg foreign          ///  
>   || lfitci mpg foreign, clpattern(solid) ///  
>   ||, legend(off) xlabel(0 1) ytitle(mpg) xtitle(foreign)
```



In contrast, the added-variable plot of `foreign` versus `mpg` graphs the residual of the dummy variable conditional on the other x variables, which is continuous even though the dummy variable itself only has two discrete values. `avciplot` can take an *indepvar* which has not yet been included in the regression, making it a useful tool for exploring the influence of new variables. To see the partial correlation of the new variable `foreign` added to the existing regression, use

```
. avciplot foreign
```



This controls for all the existing variables in the previous regression, which in our example are `displacement`, `weight` and an intercept. The added-variable plot of `foreign` could help the user decide whether to add it as a new variable to the regression.

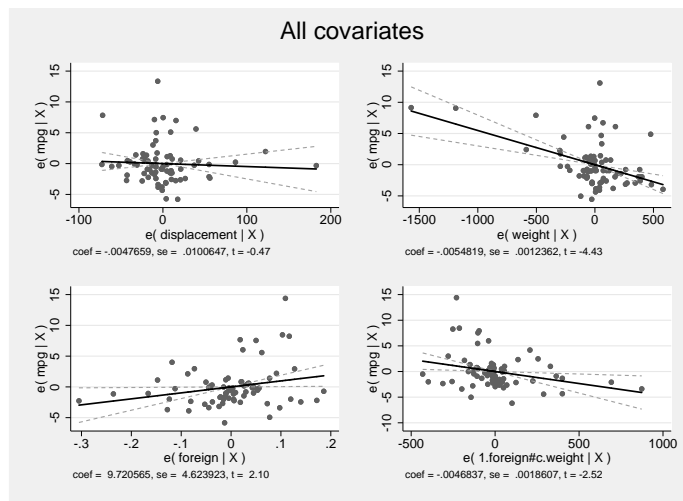
As a final example, the companion command `avciplots` shows the added-variable plots of *all* regressors in the preceding regression in a single graph. We include an interaction term between `weight` and `foreign` in the regression and then show all the partial correlations:

```
. regress mpg displacement weight foreign c.weight#i.foreign
```

Source	SS	df	MS	Number of obs	=	74
Model	1689.00001	4	422.250003	F(4, 69)	=	38.62
Residual	754.459446	69	10.9341949	Prob > F	=	0.0000
Total	2443.45946	73	33.4720474	R-squared	=	0.6912
				Adj R-squared	=	0.6733
				Root MSE	=	3.3067

mpg	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
displacement	-.0047659	.0100647	-0.47	0.637	-.0248444 .0153126
weight	-.0054819	.0012362	-4.43	0.000	-.007948 -.0030158
foreign	9.720565	4.623923	2.10	0.039	.4960906 18.94504
foreign#c.weight					
Foreign	-.0046837	.0018607	-2.52	0.014	-.0083957 -.0009717
_cons	39.12487	2.510917	15.58	0.000	34.11573 44.13402

```
. avciplots, title(All covariates)
```



`avciplots` provides a quick way of understanding the coefficient estimates after any linear regression. The graph shows the strength and significance of the partial correlations of all the independent variables, as well as helps to highlight outlier observations which affect each correlation.

The examples above show how `avciplot` and `avciplots` can be used to present the relationship between independent and dependent variables graphically when there are multiple covariates in a regression. The inclusion of confidence intervals in the `avciplot` graphs makes it possible to see the precision and statistical significance of the estimated coefficients as well as their magnitude.

6 References

About the author

John Luke Gallup is an associate professor in the Department of Economics at Portland State University. He wrote the Stata command `outreg` among others.