The Stata Journal

The *Stata Journal*, electronic version (ISSN 1536-8734) is a publication of Stata Press, and Stata is a registered trademark of StataCorp LP.

The Stata Journal (2005)
5, Number 1, pp. 64–82

Visualizing main effects and interactions for binary logit models

Michael N. Mitchell
UCLA Academic Technology Services

Xiao Chen
UCLA Academic Technology Services

Abstract. This paper considers the role of covariates when using predicted probabilities to interpret main effects and interactions in logit models. While predicted probabilities are very intuitive for interpreting main effects and interactions, the pattern of results depends on the contribution of covariates. We introduce a concept called the covariate contribution, which reflects the aggregate contribution of all of the remaining predictors (covariates) in the model and a family of tools to help visualize the relationship between predictors and the predicted probabilities across a variety of covariate contributions. We believe this strategy and the accompanying tools can help researchers who wish to use predicted probabilities as an interpretive framework for logit models acquire and present a more comprehensive interpretation of their results. These visualization tools could be extended to other models (such as binary probit, multinomial logistic, ordinal logistic models, and other nonlinear models).

Keywords: st0081, logistic regression, predicted probabilities, main effects, interactions, covariate contribution

1 Introduction

Logistic regression models are commonly used for analyzing binary outcome variables. While such models are more appropriate than OLS models for binary outcomes, the interpretation of such models is much more complicated. We are aware of three interpretive frameworks for such models.[1]

1. Logits. We can interpret the coefficients of the logit model as the degree of change in the logit of the outcome for a one-unit change in the predictor. While this model has the advantage of expressing the impact of the predictor in a linear scale, we have a very difficult time intuitively understanding the logit scale for the outcome.

2. Odds ratios. The exponentiated logit coefficient can be interpreted as the factor change in the odds of the outcome being a 1 as compared to the odds of the outcome being a 0 for a one-unit change in the predictor. While we have met a number of people who intuitively understand this metric, we often find that people will confuse the term *odds* with *likelihood* or *probability*. Even after careful

[1]The authors wish to express their deep thanks to (in alphabetical order) Phil Ender, Brad McEvoy, and Christine Wells for their thoughtful conversations, careful reviews, and warm encouragement, as well as an anonymous reviewer for an exceptionally detailed review. We welcome correspondence regarding this article by emailing the UCLA ATS Statistical Consulting Group at *ATSstat@ucla.edu*.

explanation, we find that people commonly mistake the odds-ratio framework for a probability framework.

3. Probabilities. In this framework, we can examine the association between the values of, or changes in, predictors and the predicted probabilities of the outcome being 1. The best examples of this framework can be found in Long (1997, section 3.7) and Long and Freese (2003, section 4.6). When using the probability metric, one cannot express relationships in a linear fashion because the relationship between the predictor and the predicted probability has a nonlinear S-curve shape, but this metric still can be very intuitive. Our experience has been that researchers appreciate the intuitive nature of this interpretive framework, and we feel that by using such a framework, many are more apt to accurately interpret the results of their analyses. Thanks to the efforts of Long and Freese (2003), such interpretations are not only well explained and documented, but are also quite easily obtained via the `Spost` suite of utilities (available within Stata by typing `findit spostado`).

   To further simplify the application of these methods to models involving interactions, the first author developed `xi3` and `postgr3`, the former tool being an extension to the existing Stata `xi` command and the latter an extension of the tools by Long and Freese (2003) for creating graphs of predicted values from such models. Both of these commands are available from within Stata via the `findit` command, and documentation on them is available by visiting *http://www.ats.ucla.edu/stat/stata/* and searching for `xi3` or `postgr3`. However, we have come to believe that the graphs created by `postgr3`, while accurate, present only a small portion of the entire picture. The paper by Norton, Wang, and Ai (2004) drew us back to some fundamental thinking about the appropriateness of such graphs. As a result, we have developed tools to further extend `xi3` and `postgr3` for visualizing the patterns of predicted probabilities associated with main effects and interactions and how these are influenced by additional covariates in the model. In presenting these tools, we will first consider models with main effects and then consider models involving interactions.

2 Logit models with main effects

2.1 Models with a single covariate

Consider a logistic regression model with a binary outcome variable named y and two predictors x_1 and x_2, as shown below.

$$\text{Logit}(y) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon \qquad (1)$$

The predicted values from (1), $\widehat{\text{Logit}(y)}$, could be graphed as a function of x_1 and x_2 forming the logistic regression plane. Because this is a linear model, the plane is completely flat. One such plane is graphed in the left panel of figure 1, where $\widehat{\beta}_1 = .786$ and $\widehat{\beta}_2 = 1.031$. The $\widehat{\beta}_1$ coefficient determines the tilt of the plane on the x_1 axis, and

$\widehat{\beta}_2$ determines the tilt of the plane with respect to the x_2 axis. The slope of the x_1 axis would always be $\widehat{\beta}_1$, regardless of the value of x_2 (and likewise for $\widehat{\beta_2}$).

The right panel of figure 1 is the same as the left panel, except that the logits have been converted into probabilities, Pr(y) (see, e.g., Long [1997], for this conversion). Note how the shape of the relationship between x_1 and Pr(y) depends on the value of x_2. To highlight this, we have drawn three thick lines where x_2 is -1.26, $-.34$, and .51, corresponding to the 20th, 50th, and 80th percentiles of x_2, respectively (we will discuss our selection of these values later in the paper). When x_2 is low (at the 20th percentile), the relationship between x_1 and Pr(y) is attenuated by the predicted probabilities pushing against the floor (against 0), and as x_2 increases to a high value (the 80th percentile), the strength of the relationship between x_1 and Pr(y) is greater.

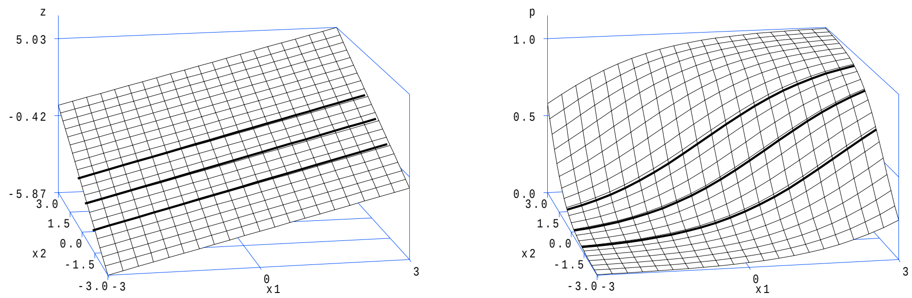

Figure 1: Logistic regression surface in logit (left) and probability (right) scale

While the right panel of figure 1 is an accurate and complete representation of the way Pr(y) varies as a function of x_1 and x_2, it can be difficult to create such three-dimensional graphs, and it is not currently possible to do so in Stata; therefore, we might want to find ways to represent this relationship in two dimensions that can easily be graphed in Stata. While this could be done using the `prgen` command (part of the `spost` suite of utilities [Long and Freese 2003, 99]), we illustrate how this could be done using `xi3` and `postgr3` below. (We use `xi3` before the `logit` command because it will be followed by the `postgr3` command. Note that the `postgr3` output is omitted to save space.)

```
. use sjvibl1, clear
. xi3: logit y x1 x2, nolog
```

(Some output omitted to save space)

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
x1	.7859427	.1864292	4.22	0.000	.4205481	1.151337
x2	1.031519	.2011148	5.13	0.000	.6373413	1.425697
_cons	-.4209601	.1722116	-2.44	0.015	-.7584886	-.0834317

```
. postgr3 x1, x(x2 = -1.26) gen(yhatl) // 20th percentile
. postgr3 x1, x(x2 =  -.34) gen(yhatm) // 50th percentile
. postgr3 x1, x(x2 =   .51) gen(yhath) // 80th percentile
. line yhatl yhatm yhath x1, sort xlabel(-3(1)3)
> legend(order(1 "x2=-1.26" 2 "x2=-.34" 3 "x2=.51") rows(1))
```

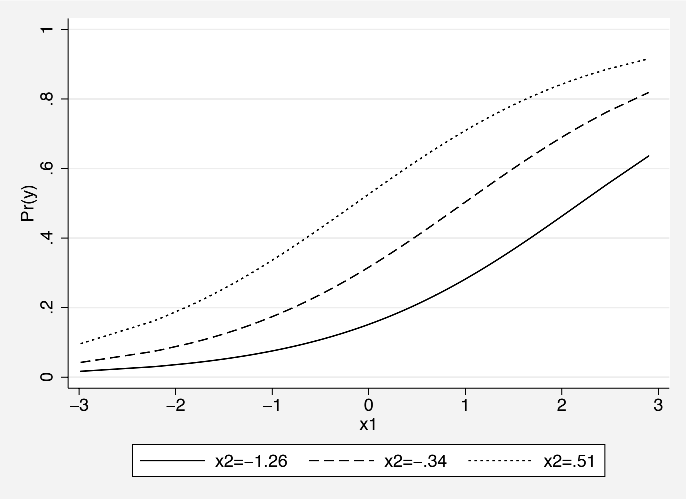

Figure 2: Two-dimensional graph of logistic regression surface in probability scale

Figure 2 is a two-dimensional representation of the right panels of figure 1 graphing the three heavy lines with x2 at the 20th, 50th, and 80th percentiles as a function of x1.[2] More importantly, the right panel of figure 1 and figure 2 convey that the shape of the relationship between x_1 and $\Pr(y)$ strongly depends on x_2. If we had portrayed this relationship with any one of the three lines alone, it would have given a misleading impression about the nature of this relationship. If we were to report a graph showing this kind of relationship between x1 and $\Pr(y)$, we would recommend a graph like figure 2 because it more completely conveys the nature of this relationship. However, it is not inevitable that the relationship between x1 and $\Pr(y)$ will strongly depend on x2, as we will see in our next example.

Consider this second example, which is based on the same model from (1) but yields the parameter estimates $\widehat{\beta}_1 = .208$ and $\widehat{\beta}_2 = .162$. Like above, we will run this model and generate a two-dimensional graph of the predicted probabilities as a function of x1 at three levels of x2 (corresponding to the 20th, 50th, and 80th percentile of x2).

[2] We use x_1 to refer to the first predictor in a hypothetical regression model and x1 to refer to the name of a Stata variable.

```
. use sjvibl2, clear
. logit y x1 x2, nolog
```
(Some output omitted to save space)

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
x1	.2080826	.0856741	2.43	0.015	.0401645	.3760007
x2	.1622411	.0753504	2.15	0.031	.0145571	.3099251
_cons	.3198027	.1425396	2.24	0.025	.0404302	.5991752

```
. postgr3 x1, x(x2=-1.43) gen(yhatl) // 20th percentile
. postgr3 x1, x(x2=  .62) gen(yhatm) // 50th percentile
. postgr3 x1, x(x2= 2.47) gen(yhath) // 80th percentile
. line yhatl yhatm yhath x1, sort xlabel(-3(1)3)
> legend(order(1 "x2=-1.43" 2 "x2=.62" 3 "x2=2.47") rows(1))
```

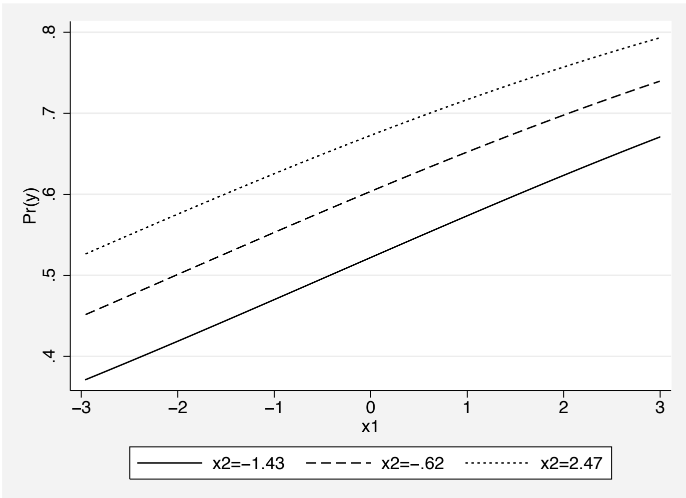

Figure 3: Logistic regression surface in probability scale, example 2

Note how in figure 2 the relationship between `x1` and $\Pr(y)$ depends highly on the value of `x2`, while in figure 3 the relationship between `x1` and $\Pr(y)$ is largely independent of `x2`. Any of the three lines shown in figure 3 would accurately portray the degree to which $\Pr(y)$ increases as a function of `x1`. For results like these, it is reasonable to portray the relationship between `x1` and $\Pr(y)$ focusing on any probable value of `x2`, with probably the most conventional value being the average of `x2`. This portrayal could be like figure 3 but contain only a single line when `x2` is held constant at the mean. Such results could also be fairly portrayed by showing a table of selected values of `x1` and the corresponding $\Pr(y)$ values. For example, we issued a series of seven `prvalue` commands like the one shown below using the `Spost` suite of utilities (Long and Freese 2003, 67), each time changing the value of `x1` from -3 to 3 in increments of 1.

```
. prvalue, x(x1=-3)
  (output omitted)
```

We then culled together the predicted probabilities from the `prvalue` commands and manually constructed this table.

x1	-3	-2	-1	0	1	2	3
Pr(y)	0.443	0.496	0.548	0.598	0.647	0.693	0.735

2.2 Models with multiple covariates

Let us now extend the strategy from section 2.1 to models with multiple covariates. It is often stated that the effects of a predictor depend on the levels of the other covariates in the model (see Hosmer and Lemeshow [2000], Long and Freese [2003], Long [1997], Norton, Wang, and Ai [2004]), but in this section we will begin to construct a framework for exploring the implications of this.

Consider this model, which now has two covariates, x_2 and x_3[3]

$$\widehat{\text{Logit}}(y) = \widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \widehat{\beta}_2 x_2 + \widehat{\beta}_3 x_3 \tag{2}$$

When we had a single covariate, we used the right panel of figure 1 to visualize the relationship between x_1 and $\Pr(y)$ while taking x_2 into account, but to extend this to a model with two covariates would require a four-dimensional graph, and a model with k covariates would require a $k+2$-dimensional graph. However, regardless of the number of covariates in the model, we can still express the relationship between x_1 and $\Pr(y)$ in two dimensions while taking the covariates into account by not considering the contribution of each individual covariate but by considering the aggregate or collective contribution of all of the covariates. Suppose that we rewrite (2) like this:

$$\widehat{\text{Logit}}(y) = \widehat{\beta}_0 + \widehat{\beta}_1 x_1 + \text{covariate contribution} \tag{3}$$

where

$$\text{covariate contribution} = \widehat{\beta}_2 x_2 + \widehat{\beta}_3 x_3 \tag{4}$$

When focusing on x_1, the covariate contribution (which we will abbreviate as CC) is the linear combination of the remaining predictors in the model multiplied by their corresponding logit coefficient.[4] Breaking the model up in this fashion allows us to see that it is not so important what the individual x_2 and x_3 values are, and in fact there

[3] Note that we are focusing on the influence of x_1 as a predictor and then treating x_2 and x_3 as covariates, but we could just as easily focus on any of the variables in the model, thus treating the remaining variables as covariates.

[4] For each variable that you would focus on, there would be a new CC. If we focused on x_2, then the CC would be $\widehat{\beta}_1 x_1 + \widehat{\beta}_3 x_3$. We do not include $\widehat{\beta}_0$ in the definition of the CC since it is not a covariate, but it still included when computing predicted probabilities.

are numerous combinations of such values that would lead to the same CC. Regardless of the individual x_2 and x_3 values, if they lead to the same CC, then their impact on the predicted logit of y in (3) will be exactly the same. Regardless of the number of additional predictors (covariates) in the model, we can represent the relationship between any given predictor and $\Pr(y)$ in three dimensions while fully taking into account the aggregate contribution of all of the remaining predictors (covariates) by replacing x_2 in the right panel of figure 1 with an axis labeled CC.[5] We can then reduce figure 1 to two dimensions like we did for figure 2, except that the multiple lines would represent multiple values selected on the CC. In figure 2, we displayed three lines corresponding to the 20th, 50th, and 80th percentiles on `x2`, we could make a similar graph with respect to the CC, presenting three lines corresponding to the 20th, 50th, and 80th percentiles on CC. Below, we introduce a suite of tools to help us create such visualizations.

2.3 Visualizing main effects of a continuous variable

Let's consider (2) and focus on the effect of x_1 and consider x_2 and x_3 as covariates. We run the logit model below.

```
. use sjvibl1, clear
. logit y x1 x2 x3, nolog
```

(Some output omitted to save space)

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
x1	.7489698	.1921057	3.90	0.000	.3724496	1.12549
x2	1.054486	.2064745	5.11	0.000	.6498038	1.459169
x3	1.224105	.358766	3.41	0.001	.5209366	1.927274
_cons	-1.080401	.2708698	-3.99	0.000	-1.611296	-.5495062

From (4) the CC can be computed with the `generate` command, as shown below. We can then use descriptive statistics commands to get an understanding of the distribution of the CCs (e.g., `summarize`, `tabulate`, `centile`). Next we use the `centile` command to determine the 20th, 50th, and 80th percentiles of the CC.[6]

[5] In fact, we view figure 1 as a simplified version of such a graph where the CC is represented by x_2 instead of $\widehat{\beta}_2 x_2$ (which would be the CC if x_2 were the only covariate in the model).

[6] Given this is a new area we are exploring, we have no firm reason for picking the 20th, 50th, and 80th percentiles. In a way, what we are doing could be viewed as analogous to the way that (Aiken and West 1991, 13) explore continuous-by-continuous interactions by examining the simple effect of one variable at probe values for the other variable. They generally suggest probe values at the mean, one standard deviation above the mean, and one standard deviation below the mean. Assuming a normal distribution, this would roughly correspond to the 17th, 50th, and 83rd percentiles. We have adapted this rule of thumb for our purposes but do not wish to make any assumption about the distribution of the CC, so we instead directly specify the probe values as percentiles and have rounded them to the 20th, 50th, and 80th percentiles. While we are open to other ideas for probe values, we are concerned about selecting values that are too extreme because the estimates would appear to get increasingly unstable as they move to the frontier of the CC values and could even lead to extrapolation.

```
. generate cc = _b[x2]*x2 + _b[x3]*x3 // _b[x2] is 1.05... and _b[x3] is 1.22...
. centile cc, centile(20 50 80) normal
```

			— Normal, based on observed centiles —		
Variable	Obs	Percentile	Centile	[95% Conf.	Interval]
cc	198	20	-.9594531	-1.23664	-.6822663
		50	.2416907	.0202967	.4630848
		80	1.412609	1.141197	1.684021

We then can create a graph analogous to figure 2 that shows the relationship between `x1` and $\Pr(y)$ at three levels of the CC, $-.96$, .24, and 1.41 (corresponding to the 20th, 50th, and 80th percentiles of `x2`). To streamline this process, we have developed a program named `viblmgraph`, which stands for **vi**sualizing **b**inary **l**ogit models for **m**ain effects `graph`. This is part of a suite of `vibl` tools for visualizing binary logit models. Below we call the program specifying the intercept and the slope, and use the `ccat()` option to indicate that we want to view three lines for the CCs $-.96$, .24, and 1.41. We also specify that `x1` can range from -3 to 3. We can (and do) add additional graph options to the end of the command, and this produces the graph shown in figure 4.

```
. viblmgraph, b0(-1.08) b1(.75) ccat(-.96 .24 1.41) xmin(-3) xmax(3)
> legend(rows(1) subtitle(Covariate Contribution)) xlabel(-3(1)3)
```

(Output with predicted probabilities omitted to save space)

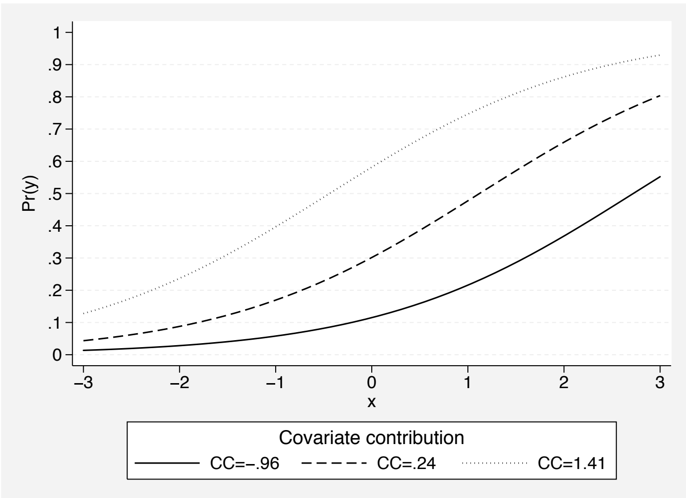

Figure 4: Predicted probabilities as a function of `x1` and CC

Examining the graph in figure 4, we can see that when the collective contribution of `x2` and `x3` is low (at the 20th percentile), the relationship between `x1` and $\Pr(y)$ is flattened because the curve is pressing against the floor value of 0, but as the CC increases to the 50th, and 80th percentiles, the relationship between `x1` and $\Pr(y)$ becomes steeper.

In the illustration above, the CCs were generated manually, followed by `viblmgraph`. We created a convenience command `viblmcc` to compute the CC and also provide summary statistics about the percentiles of the CC. Because we added the `graph` option, the `viblmgraph` command is also displayed and executed, producing a graph similar to figure 4 with the CC at three levels, at the 20th, 50th, and 80th percentiles.

```
. viblmcc y x1 x2 x3, gen(cc2) graph
Saving covariate contribution as cc2
Percentiles for Covariate Contribution
     P1    P10    P20    P30    P40    P50    P60    P70    P80    P90    P99
 -2.533 -1.443 -.9636 -.4425 -.1048  .2422  .5622  .9427  1.418   1.93   3.42
```
(Rest of output omitted to save space)

2.4 Visualizing main effects of a dummy variable

The previous example focused on `x1` which is a continuous variable. Now let's turn our focus to `x3`, which is a dummy variable. Consider the logit model below.

```
. logit y x3 x1 x2, nolog
```
(Some output omitted to save space)

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
x3	1.224105	.358766	3.41	0.001	.5209366	1.927274
x1	.7489698	.1921057	3.90	0.000	.3724496	1.12549
x2	1.054486	.2064745	5.11	0.000	.6498038	1.459169
_cons	-1.080401	.2708698	-3.99	0.000	-1.611296	-.5495062

We can use the `viblmcc` command to view the association between `x3` and $\Pr(y)$ when the CC is at the 20th, 50th, and 80th percentiles. By placing `x3` in the first position after the dependent variable, we indicate that `x3` is the variable of interest and that the rest of the variables are covariates. As before, we get percentiles for the CC, and because we specified the `graph` option, the `viblmgraph` command is displayed and executed, yielding the graph in figure 5. In addition, `viblmgraph` displays tables of the predicted probabilities for the two levels of `x3` for each of the levels of the covariate specified in the `ccat()` option. These predicted probabilities in these tables correspond to the predicted probabilities graphed in figure 5.

```
. viblmcc y x3 x1 x2,  gen(cc3) graph
Saving covariate contribution as cc3
Percentiles for Covariate Contribution
     P1    P10    P20    P30    P40    P50    P60    P70    P80    P90    P99
 -3.351 -2.193 -1.558 -1.046 -.5926 -.2886 -.0218  .2396  .6343  1.179  2.567
```

```
. viblmgraph, b0(-1.08) b1(1.224) ccat(-1.558 -.289 .634) xmin(0) xmax(1) xname(x3)
**For CC=-1.558279943466**
      x3
  0         1
--------------------
 0.07 (A)   0.20 (B)   (B-A) =  0.13
**For CC=-.2886011004448**
      x3
  0         1
--------------------
 0.20 (A)   0.46 (B)   (B-A) =  0.26
**For CC=.6343373894691**
      x3
  0         1
--------------------
 0.39 (A)   0.69 (B)   (B-A) =  0.30
```

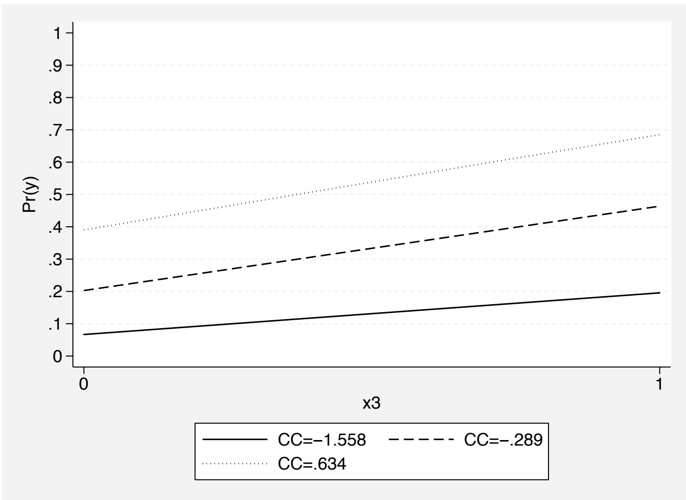

Figure 5: Predicted probabilities as a function of x3 and CC

Based on the output from viblmgraph, we can see that the difference in the predicted probabilities is .13 when the CC is at −1.558 (the 20th percentile), .26 when the CC is at −.289 (the median), and 0.30 when the CC is at .634 (the 80th percentile). In other words, the effect of x3 is mildly attenuated when the CC is at a middle value and notably attenuated when the CC is low. The tables shown in the output are visually depicted in the graph in figure 5.

While figure 5 gives us some insight into how the predicted probabilities change as a function of the dummy variable x3 and the CC, the viblmgraph command can show a second type of graph (using the type(2) option) that shows the predicted probabilities for the two levels of the dummy variable across a given spectrum of CCs. The ccmin() and ccmax() options are used to indicate the span of CCs to be considered, and the ccat() option is used to draw vertical lines at specified points (we chose the 20th, 50th, and 80th percentiles to match the values chosen in figure 5).

```
. viblmgraph, b0(-1.08) b1(1.22) ccmin(-2) ccmax(1) ccat(-1.6 -.3 .6) type(2)
```

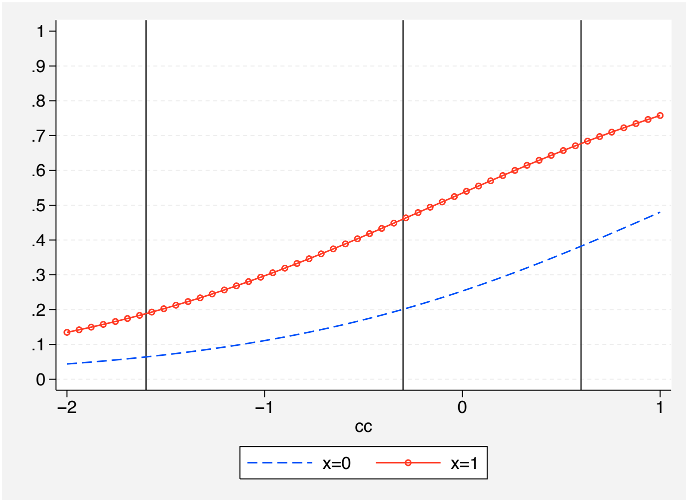

Figure 6: Predicted probabilities for x3 = 0 and x3 = 1 as a function of the CC

Besides looking at how the predicted probabilities for each level of the dummy variable change as a function of the CC, we can also visualize the *difference* in these predicted probabilities as a function of the CC. You can view such a graph using the `type(3)` option on the `viblmgraph` command, yielding the graph shown in figure 7. This graph shows how the difference between the predicted probabilities grows as the CC changes from the 20th to the 80th percentile.

(*Continued on next page*)

```
. viblmgraph, b0(-1.08) b1(1.22) ccmin(-2) ccmax(1) ccat(-1.6 -.3 .6) type(3)
```

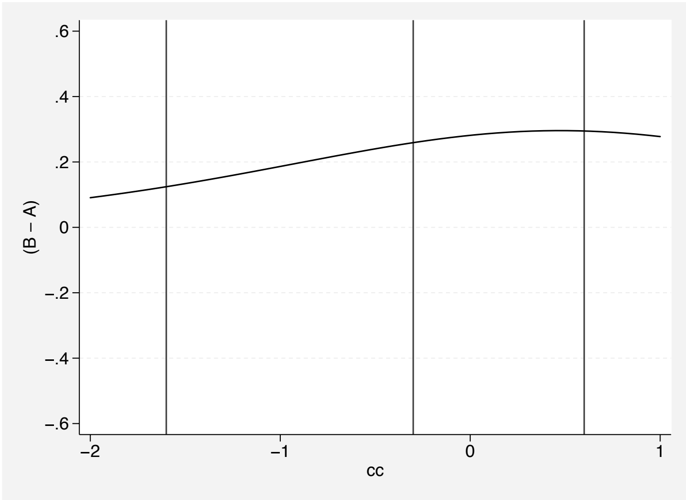

Figure 7: Difference in predicted probabilities as a function of the CC

The three graph types illustrated in figures 5, 6, and 7 are all different ways of viewing the same pattern of results. Each graph offers a unique vantage point to viewing the relationship between x3 and $\Pr(y)$ while taking the contribution of the covariates into account. However beneficial we find these graphs, they suffer from how static they are. In section 4, we introduce a tool for dynamically viewing these kinds of graphs.

2.5 Further thoughts

The extent to which the pattern of the predicted probabilities depends on the other covariates in the model will vary from situation to situation. For example, the results from figure 2 show that the effect of x1 depended highly on the levels of x2; however, figure 3 portrays a different kind of example where the effect of x1 is largely independent of x2. While there are numerous differences between these two examples, the most salient distinction is that the predicted probabilities in the second example were largely confined to be between .2 to .8, while many of the predicted probabilities from the first example were outside of this range. When the predicted probabilities are between .2 and .8 (or more conservatively .25 to .75), the relationships between predictors and predicted probabilities are largely linear (Long 1997, 64), so the model behaves similar to a linear model where the relationship between the predictor and outcome does not strongly depend on the values of other predictors in the model.

3 Logit models with dummy-by-dummy interactions

3.1 Visualizing interactions for logit models

What we have seen with respect to main effects applies equally well to interactions. When graphing the predicted probabilities associated with an interaction, it would be wise to explore a reasonable range of CCs. As we saw with main effects, the pattern of predicted probabilities associated with an interaction may be similar over the range of CCs, or the pattern might vary considerably across the range of CCs. For simplicity, we will focus on dummy-by-dummy interactions; however, the tools we provide could be used for dummy-by-continuous interactions, as well. We will use a hypothetical data file named `sjvibl3`. Consider the logit model, which predicts `y` (a dichotomous outcome) from the dummy variables `x1` and `x2` and their interaction named `x1x2`, as well as a number of additional predictors (covariates) named `x3` to `x9`.

```
. use sjvibl3, clear
. logit y x1 x2 x1x2 x3 x4 x5 x6 x7 x8 x9, nolog
```

(Some output omitted to save space)

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
x1	.7282436	.1621386	4.49	0.000	.4104578	1.046029
x2	1.29437	.2773787	4.67	0.000	.7507173	1.838022
x1x2	.9872251	.4423858	2.23	0.026	.1201649	1.854285
x3	.8052569	.0804086	10.01	0.000	.6476588	.9628549
(Output with coefficients for x4 *to* x9 *omitted to save space)*						
_cons	-3.099149	.6502432	-4.77	0.000	-4.373603	-1.824696

Let's explore the effect associated with `x1x2`. Let's start by computing the CC and getting a sense of its range using the `viblicc` command, which is the equivalent of the `viblmcc` command but for models with interactions. If we include the **`graph`** option, the **`vibligraph`** command is displayed and executed, showing the predicted probabilities broken down by `x1` and `x2` with the CC held at the median. As shown in figure 8, the predicted probabilities increase faster across `x1` when `x2` is 1 than when `x2` is 0. A corresponding table of the predicted probabilities is shown broken down by `x1` and `x2` with the CC held at the median. This table includes the simple differences of `x1` at each level of `x2`, as well as the difference of differences, abbreviated as $(\mathtt{D} - \mathtt{C}) - (\mathtt{B} - \mathtt{A})$ (in Norton, Wang, and Ai [2004], this was called the interaction effect). This table shows a difference of differences of 0.13 when the CC is at the median, but the size of this value could depend on the CC, as we saw in the main effects examples.

```
. viblicc  y x1 x2 x1x2 x3 x4 x5 x6 x7 x8 x9, gen(cc) graph
Saving covariate contribution as cc
Percentiles for Covariate Contribution
     P1    P10    P20    P30    P40    P50    P60    P70    P80    P90    P99
  .2561  .9605  1.302  1.631  1.887  2.178  2.461  2.757  2.998  3.338  4.156
```

```
. vibligraph, b0(-3.099) b1(.728) b2(1.294) b12(.987) ccat(2.178) ccmin(1.302)
> ccmax(2.998) x1name(x1) x2name(x2)
**For CC=2.177959084511**
         |      x1
      x2 |  0         1
---------+-------------------
       0 |  0.28 (A)    0.45 (B)    (B-A) =  0.17
       1 |  0.59 (C)    0.89 (D)    (D-C) =  0.30
                      (D-C) minus (B-A) =  0.13
```

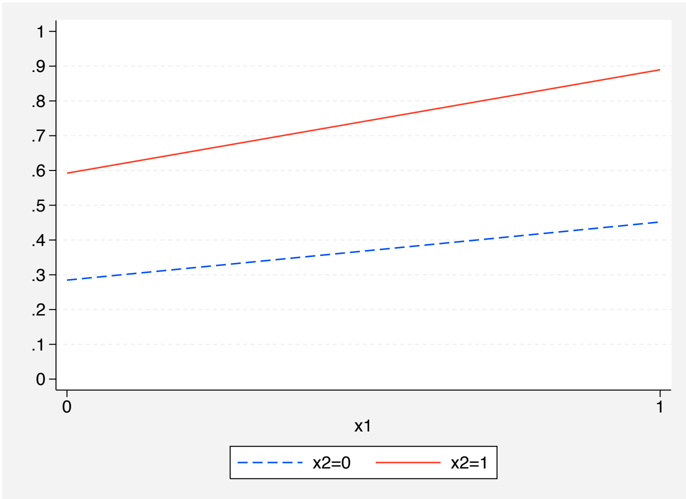

Figure 8: $\Pr(y = 1)$ by `x1` and `x2` when CC is at the median

To explore the role of the CC for this interaction, let's repeat the above `vibligraph` command and use the `ccat` option to make a graph where the CC is set at a low value (the 20th percentile, 1.302) and then repeat the command specifying a high CC value (the 80th percentile, 2.998). We save each graph and show them combined together in figure 9.

```
. vibligraph, b0(-3.099) b1(.728) b2(1.294) b12(.987) ccat(1.302) abcd
> title(CC low) name(g1)
**For CC=1.302**
         |      x1
      x2 |  0         1
---------+-------------------
       0 |  0.14 (A)    0.26 (B)    (B-A) =  0.12
       1 |  0.38 (C)    0.77 (D)    (D-C) =  0.39
                      (D-C) minus (B-A) =  0.27
```

```
. vibligraph, b0(-3.099) b1(.728) b2(1.294) b12(.987) ccat(2.998) abcd
> title(CC high) name(g2)
**For CC=2.998**
         |      x1
      x2 |  0         1
---------+-------------------
       0 |  0.47 (A)   0.65 (B)   (B-A) =  0.18
       1 |  0.77 (C)   0.95 (D)   (D-C) =  0.18
                      (D-C) minus (B-A) =  0.00
. graph combine g1 g2
```

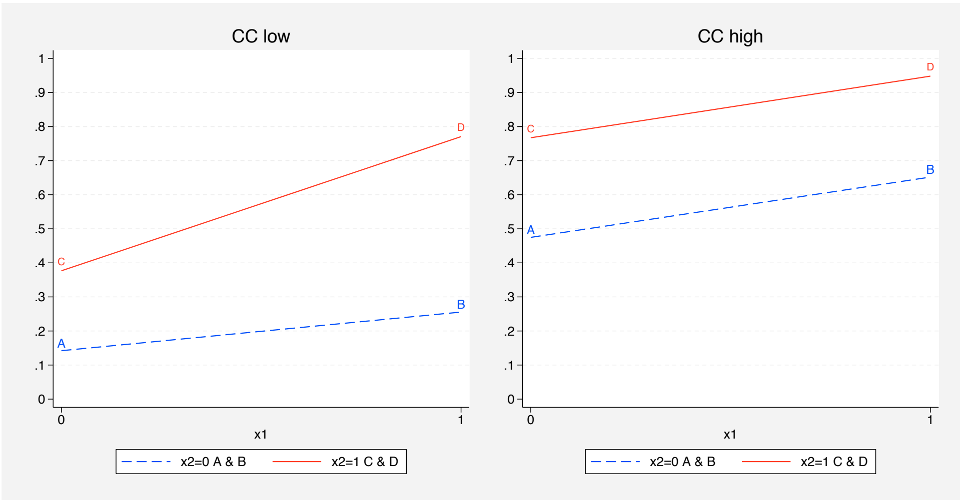

Figure 9: Pr(y = 1) by `x1` and `x2` when CC is at the 20th (left) and 80th (right) percentile

Figures 8 and 9 show that, in this example, the pattern of predicted probabilities is very sensitive to the CC. In figure 9, the `(D-C)` difference is much greater than the `(B-A)` difference when the CC is low, but these two differences are much the same when the CC is high. Further inspection reveals some insight into why these differences vary across the levels of the CCs. For example, consider the right panel of figure 9, and focus on cells C and D. Both cells are pressing against the ceiling, but D is being influenced more strongly. As the CC went from the 20th to the 80th percentile, the tables of the predicted probabilities show that the `(D-C)` difference was attenuated considerably (.39 to .18) while the `(B-A)` difference actually grew somewhat (.12 to .18).

Previously we graphed the predicted probabilities of the two levels of a dummy variable across a spectrum of CC values to illustrate its main effect, see figure 6. We can make an analogous graph with `vibligraph` by adding the `type(2)` option to create figure 10. We also specify multiple `ccat()` values corresponding to the 20th, 50th, and 80th percentiles and specify the minimum and maximum values for the x-axis containing the CC via the `ccmin()` and `ccmax()` options. This graph represents the predicted probabilities at the CC values used in figures 8 and 9 but also shows the predicted probabilities for all of the CC values in between. The perspective in figure 10 shows how the CC affects the predicted probabilities across a spectrum of CC values, in this case illustrating how the predicted probabilities for cell D are pressing against the ceiling as the CC increases, leading to the attenuation of the (D-C) difference.

```
. vibligraph, b0(-3.099) b1(.728) b2(1.294) b12(.987) ccat(1.302 2.178 2.998)
> ccmin(1) ccmax(3) type(2)
```

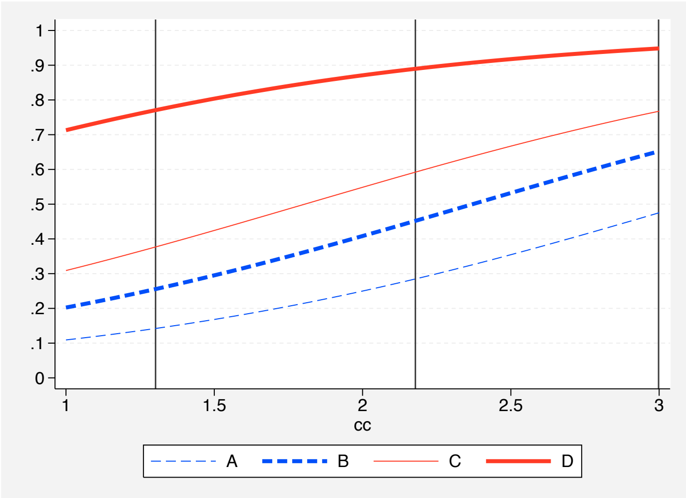

Figure 10: Pr($y = 1$) by x1, x2, and CC with lines at the 20th, 50th, and 80th percentiles

We can also make a type 3 graph that shows the differences in (B-A) and (D-C) as a function of the CC; see the left panel of figure 11.

(*Continued on next page*)

```
. vibligraph, b0(-3.099) b1(.728) b2(1.294) b12(.987) ccat(1.302 2.178 2.998)
> ccmin(1) ccmax(3) type(3)
```

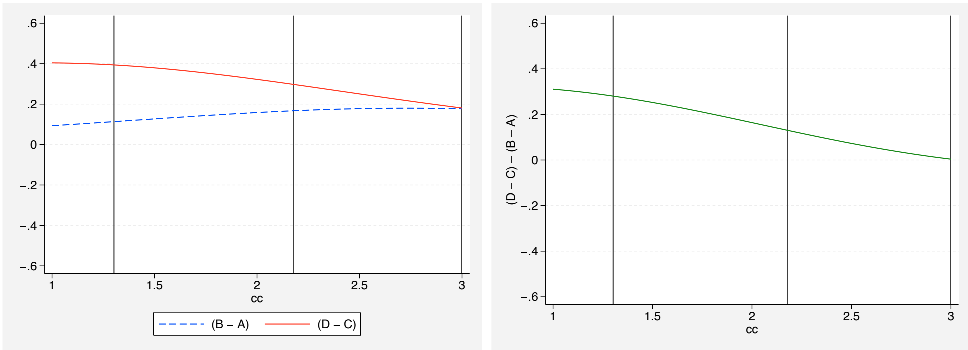

Figure 11: Differences in $\Pr(y=1)$ (left panel) and difference in difference (right panel) by CC with lines at the 20th, 50th, and 80th percentiles

We can also make a type 4 graph that shows the difference in differences `(D-C)` minus `(B-A)` as a function of the CC; see the right panel of figure 11.[7]

```
. vibligraph, b0(-3.099) b1(.728) b2(1.294) b12(.987) ccat(1.302 2.178 2.998)
> ccmin(1) ccmax(3) omitint(1) type(4)
```

This illustrates how the difference in differences in the predicted probabilities, `(D-C)` − `(B-A)`, depends considerably on the CC, being largest when the CC is low and diminishing to zero when the CC is high. For this kind of result, a single graph showing the predicted probabilities as a function of `x1` and `x2` at a single CC would not be sufficient to accurately portray the results. Either multiple graphs like figures 9 and 10 or a single graph like the left or right panel of figure 11 would be needed to account for the role of the covariates.

4 Interactive tools for visualizing binary logit models

The examples we have seen so far have used the `viblmgraph` command for visualizing main effects in logistic models and the `vibligraph` command for visualizing interactions in logistic models. While both commands easily draw a graph for a given set of parameters, these commands do not allow you to quickly and easily alter these parameters to interactively explore the behavior of these graphs across different values of coefficients or CCs. To help you interactively visualize main effects, we created an interactive point-and-click, dialog-box-driven program called `viblmdb`, which stands for **vi**sualizing **b**inary **l**ogit models for **m**ain effects **d**ialogue **b**ox, and likewise created `viblidb` for vi-

[7]This is an alternative visualization strategy to that of figure 3 in Norton et al. (2004), where we use CC instead of $\Pr(y)$ on the x-axis.

sualizing interaction effects. Due to space considerations, these tools are not discussed here but are discussed on the UCLA ATS web site, as described in the following section.

5 Additional online information

While the `vibl` suite of tools as described in this article are available from the Stata Journal web site, you can obtain the most up-to-date version of the tools and data files from the UCLA Academic Technology Services web site using the following commands.

```
. net from http://www.ats.ucla.edu/stat/stata/ado/analysis/
. net install vibl
. net get vibl
```

We acknowledge the brief coverage of the syntax and options available in the `vibl` suite of tools. However you can access such additional details, as well as illustrations of the interactive tools `viblmdb` and `vibidb`, by visiting our online seminar titled *Visualizing Main Effects and Interactions for Binary Logit Models in Stata* on our web site at *http://www.ats.ucla.edu/stat/seminars*.

6 Conclusion

We have found that researchers are very comfortable using the Long and Freese (2003) tools, as well as our `xi3` and `postgr3` commands, for interpreting the results of logit models using a predicted probability framework. We hope that the concept of the covariate contribution combined with the visualization techniques provided by the suite of `vibl` tools will be a useful extension to this framework by allowing researchers to interpret main effects and interactions via the probability metric while using the covariate contribution to account for the role of the covariates. In addition, we hope that the `vibl` tools, especially the interactive `vibl` tools, would be useful for teaching logistic regression to help students develop skills in visualizing the behavior of such models. If the `vibl` suite of tools prove useful, we envision extending them to multinomial logit models (e.g., `viml` tools), ordinal logit models (e.g., `viol` tools), and binary probit models (e.g., `vibp` tools) as well as other kinds of nonlinear models, such as Poisson and negative binomial models. We would welcome collaborative efforts in this regard.

7 References

Aiken, L. S. and S. G. West. 1991. *Multiple Regression: Testing and Interpreting Interactions*. Thousand Oaks, CA: Sage.

Hosmer, D. W., Jr. and S. Lemeshow. 2000. *Applied Logistic Regression*. 2nd ed. New York: Wiley.

Long, J. S. 1997. *Regression Models for Categorical and Limited Dependent Variables*. Thousand Oaks, CA: Sage.

Long, J. S. and J. Freese. 2003. *Regression Models for Categorical Dependent Variables Using Stata*. rev. ed. College Station, TX: Stata Press.

Norton, E. C., H. Wang, and C. Ai. 2004. Computing interaction effects and standard errors in logit and probit models. *Stata Journal* 4(2): 154–167.

About the Authors

Michael Mitchell is a statistical consultant and manager of the UCLA Academic Technology Services Statistical Consulting Group. He envisioned the UCLA Statistical Computing Resources web site and has written hundreds of web pages about Stata and statistical computing on that site. He is the author of *A Visual Guide to Stata Graphics* and of the forthcoming book *Data Management Using Stata*.

Xiao Chen is a statistical consultant of the UCLA Academic Technology Services Statistical Consulting Group.