Section A: Comparison of the Average Case and Observed Value Approaches

A1 Marginal Effect Analysis to Determine the Difference Between the Average Effect Using the Observed Value Approach and the Effect for the Average Case

Earlier we noted that because the probability density function (pdf), f, used for limited dependent variable models is nonlinear, the marginal effect for the average case is generally not equivalent to the average marginal effect calculated using the observed value approach (hereafter average effect). We expressed this in equation (7a), $f(\bar{x}\beta)\beta_k \neq \frac{1}{n}\sum_{i=1}^{n} f(x_i\beta)\beta_k$. We now show that where the average case is on the curve plays an important role in determining when the effect for the average case is greater than, equal to, or less than the average marginal effect calculated via the observed value approach.

Using the law of large numbers, for a continuous pdf, we know that the average marginal effect, $\frac{1}{n}\sum_{i=1}^{n} f(x_i\beta)\beta_k$, converges in probability to $E(\beta_k f(x\beta))$. By taking the second order Taylor series expansion of $E(\beta_k f(x\beta))$ around $\mu\beta$, where μ is the population mean of the x's, we obtain the following:

$$E(\beta_k f(x\beta)) = E(\beta_k f(\mu\beta + x\beta - \mu\beta))$$
$$= \beta_k E(f(\mu\beta) + (x\beta - \mu\beta)f'(\mu\beta) + \frac{1}{2}(x\beta - \mu\beta)^2 f''(\mu\beta) + \Psi)$$
$$= \beta_k (f(\mu\beta) + \frac{1}{2}Var(x\beta)f''(\mu\beta) + \Psi) \tag{A1},$$

where ψ represents the higher order terms.[1] The first term on the right hand side of the above

---

[1] Note that the middle term of the second line drops out since $E(x\beta - \mu\beta) = 0$. Though in the 5th edition of his classic textbook Greene (2003) suggested that the average case approach and the observed value approach are asymptotically equivalent, in the 6th edition, Greene (2008) corrects that claim, providing a calculation similar to the one above (we work from the expected value of the average effect and use different notation). We thank Professor Greene for pointing us to and sharing the relevant chapter with us. Greene (2008) does not evaluate the implications of his equation in detail and suggests that the

result is simply the expected value of the marginal effect evaluated at the mean of all of the *x*'s; i.e., the effect for the average case.[2] Thus, working with the second order approximation, by examining β_k, the variance of $x\beta$, and the second derivative of the pdf, we can determine whether the average marginal effect is substantively greater than or less than the effect for the average case. Since Var($x\beta$) will always be positive, we need to consider β_k and the second derivative of the pdf. In terms of the substantive size of the average effect using the observed value approach versus the effect for the average case, it turns out that regardless of the sign of β_k, whenever the second derivative of the pdf is negative, the average effect will be substantively smaller (i.e., smaller in absolute value) than the effect for the average case.[3] Similarly, regardless of the sign of β_k, when the second derivative of the pdf is positive, the average effect will be substantively larger than the effect for the average case. The magnitude of the difference (up to the second order approximation) between the average effect and the effect for the average case will depend on the size of β_k, the size of Var($x\beta$), and the value of the second derivative of the pdf evaluated at $\mu\beta$. To approximate when the second derivative of the pdf will be positive or negative we turn to an investigation of the probit pdf and logit pdf.

Using the probit pdf we can rewrite the equation above as:

---

differences between the two approaches are likely to be small. Contrary to that suggestion, our analysis in this section and our examples using NES data and Monte Carlo simulations show that substantial differences can result. SI Section D Table 1(see below) reports results from Monte Carlo simulations that provide additional evidence that results from the average case approach and observed value approach do not converge in large samples.

[2] That is, using the law of large numbers and the continuous pdf, $\beta_k f(\bar{x}\beta)$ converges in probability to $\beta_k f(\mu\beta)$.

[3] This is obvious for $\beta_k > 0$ because the average effect (which will be positive by virtue of β_k being positive) will equal the effect for the average case, which is also positive (since $\beta_k > 0$), plus a negative number, resulting in an average effect that is less than the effect for the average case. When β_k is negative, then the average effect (which is negative because $\beta_k < 0$) will equal a negative number (the effect for the average case is negative since $\beta_k < 0$) plus a positive number resulting in a smaller negative number (i.e. a substantively smaller effect).

$$E(\beta_k \phi(x\beta)) = \beta_k (\phi(\mu\beta) + \frac{1}{2}Var(x\beta)\phi(\mu\beta)((\mu\beta)^2 - 1) + \Psi) \qquad \text{(A2)}.$$

Since $\phi(\mu\beta)$ is always positive, the second derivative of the probit pdf, $\phi(\mu\beta)((\mu\beta)^2 - 1)$, is negative when $-1 < \mu\beta < 1$ (where the pdf is concave), zero at $\mu\beta = -1$ or $\mu\beta = 1$, and positive otherwise (where the pdf is convex). We get a clearer picture of when this occurs by transforming various values of $\mu\beta$ into probabilities. That is, by considering the probability of success associated with various values of $\mu\beta$; or, in other words, evaluating where the average case is on the curve. SI Section A Figure 1 plots the second derivative of the probit pdf by the probability of success for the average case. The figure shows that the second derivative of the probit pdf is negative across most of the range of the predicted probabilities, from 0.1587 to 0.8413.[4] Thus, only when the probability of success for the average case is in the tails will the average effect using the observed value approach be substantively larger than the effect for the average case. As noted earlier, the size of the difference depends on the size of β_k, the size of $Var(x\beta)$, and the value of the second derivative of the probit pdf; as shown in SI Section A Figure 1, the absolute value of the second derivative varies across the range of predicted probabilities.[5]

It is important to note that the above analysis has assumed that the higher order terms play a small role. Our investigation of the higher order terms indicated that there were too many

---

[4] As we discuss at the end of this section, these values are best viewed as approximations.

[5] The general conclusions derived for probit also hold for logit, though the second derivative of the logit pdf is negative over a slightly different range of predicted probabilities, from 0.2113 to 0.7887. Note that part of the second derivative of the logit pdf contains the term β_k^3. The value of β_k does not change where the second derivative is positive, negative, or zero, subject to rounding error. When β_k is positive, though the density changes, the second derivative is positive and negative over the same range of predicted probabilities regardless of the value of β_k. When β_k is negative, the series when plotted will be a reflection over the x-axis, and though the density will change, again, the second derivative will be positive or negative over the same range of values regardless of the value of β_k. Note that when β_k is negative, the second derivative will be negative from p = 0.2113 to p = 0.7887, but the average effect and the effect for the average case will also be negative resulting in an average effect that is substantively smaller than the effect for the average case (i.e. smaller in absolute value).

unknown quantities to provide further insight. That said, the pdfs considered here are bell-shaped, and the marginal effects are proportional to the pdf, so the second order approximation will be reasonable across most of the range of predicted probabilities. Moreover, our Monte Carlo simulations (see section 5) provide results consistent with the expectations from the second order approximation. Nonetheless, we err on the side of caution and suggest the values over which the effect for the average case is greater than or less than the average effect, presented above and subsequently, are best viewed as approximations, though ones we believe are quite good. As we argued in the main text, though the comparison of the results from the two approaches is interesting, the primary argument for use of the observed value approach is theoretical.

SI Section A Figure 1. Second Derivative of the Probit pdf by the Probability of Success for the Average Case

Probability of Success
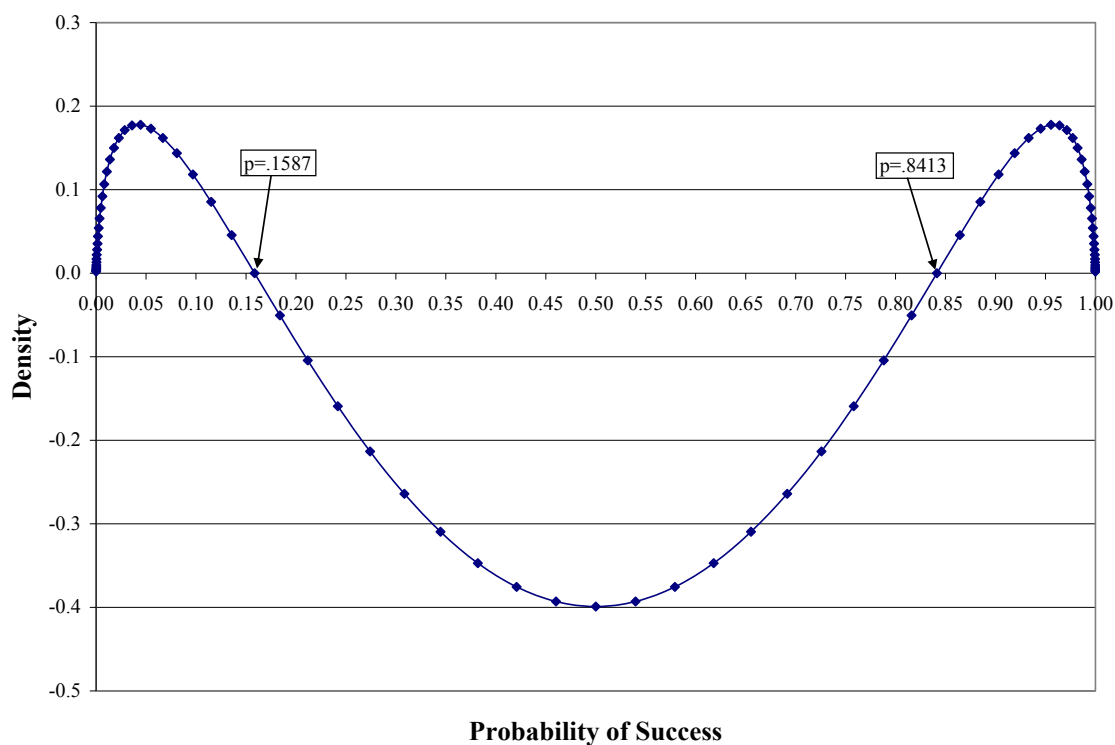
A2 Discrete Difference Analysis

The analysis of the discrete differences can proceed along similar lines. Start with the following equation for the average effect using the observed value approach (hereafter average effect) of a change in x_k from $x_k = d$ to $x_k = c$, holding all other independent variables, $x_{\sim k}$, at their observed values:

$$\frac{1}{n}\sum_{i=1}^{n}(F(x_{i,\sim k}, x_k = c; \beta) - F(x_{i,\sim k}, x_k = d; \beta)) \tag{A3a}.$$

Separating out the values of x_k that represent the manipulation of interest and β_k from the rest of the x's and β's, this can be rewritten as follows:

$$\frac{1}{n}\sum_{i=1}^{n}(F(x_{i,\sim k}\beta_{\sim k} + c\beta_k) - F(x_{i,\sim k}\beta_{\sim k} + d\beta_k)) \tag{A3b},$$

where $\beta_{\sim k}$ represents all of the coefficients that are not the coefficient on the variable of interest, β_k. Using the law of large numbers, relying on the continuity of F, taking the expectation, and using a second order Taylor series expansion around $\mu\beta$ with x_k set to values of c and d gives us the following:

$$
\begin{aligned}
&E(F(x_{\sim k}\beta_{\sim k} + c\beta_k) - F(x_{\sim k}\beta_{\sim k} + d\beta_k)) \\
&= E(F(\mu_{\sim k}\beta_{\sim k} + x_{\sim k}\beta_{\sim k} + c\beta_k - \mu_{\sim k}\beta_{\sim k}) - F(\mu_{\sim k}\beta_{\sim k} + x_{\sim k}\beta_{\sim k} + d\beta_k - \mu_{\sim k}\beta_{\sim k})) \\
&= E(F(\mu_{\sim k}\beta_{\sim k} + c\beta_k) - F(\mu_{\sim k}\beta_{\sim k} + d\beta_k) \\
&\quad + (x_{\sim k}\beta_{\sim k} - \mu_{\sim k}\beta_{\sim k})F'(\mu_{\sim k}\beta_{\sim k} + c\beta_k) - (x_{\sim k}\beta_{\sim k} - \mu_{\sim k}\beta_{\sim k})F'(\mu_{\sim k}\beta_{\sim k} + d\beta_k) \\
&\quad + \frac{1}{2}((x_{\sim k}\beta_{\sim k} - \mu_{\sim k}\beta_{\sim k})^2 F''(\mu_{\sim k}\beta_{\sim k} + c\beta_k) - (x_{\sim k}\beta_{\sim k} - \mu_{\sim k}\beta_{\sim k})^2 F''(\mu_{\sim k}\beta_{\sim k} + d\beta_k))) + \Psi \\
&= F(\mu_{\sim k}\beta_{\sim k} + c\beta_k) - F(\mu_{\sim k}\beta_{\sim k} + d\beta_k) \\
&\quad + \frac{1}{2}Var(x_{\sim k}\beta_{\sim k} - \mu_{\sim k}\beta_{\sim k})((F''(\mu_{\sim k}\beta_{\sim k} + c\beta_k) - F''(\mu_{\sim k}\beta_{\sim k} + d\beta_k)) + \Psi
\end{aligned}
$$

$$\tag{A4}.$$

As was the case with the marginal effects, the average effect derived from the discrete differences is equal to the effect for the average case plus additional terms. The key to

4

evaluating the substantive difference between the average effect and the effect for the average case involves the difference between the second derivative of the cdf evaluated at $\mu_{\sim k}\beta_{\sim k} + c\beta_k$ and $\mu_{\sim k}\beta_{\sim k} + d\beta_k$.

Compared to the marginal effects, there are more moving parts, so to speak, so determining where this difference is positive and negative is more involved than the analysis of the marginal effects. SI Section A Figure A2 considers a move from $x_k = 0$ to $x_k = 1$, as might be the case for the analysis of the effect of a dummy variable, for various values of β_k for the probit cdf across the range of possible initial probabilities of success when $x_k = 0$. Though β_k plays a larger role here, as was true for the marginal effects, the second term of the Taylor series expansion for the effect using the discrete differences is negative over most of the range of possible initial probabilities of success; i.e., across most of the range of initial probabilities of success, the effect for the average case will be larger than the average effect. Although we provide an approximation for several values of β for a common scenario, our general conclusion is that the features of the data play too much of a role to draw firm conclusions regarding when the effects estimated from the average case approach will be greater than or less than the effects estimated from the observed value approach.

SI Section A Figure A2. Difference in the Second Derivative of the Probit CDF Changing From $x_k = 0$ to $x_k = 1$ for Various Values of β_k by the Probability of Success for the Average Case When $x_k = 0$
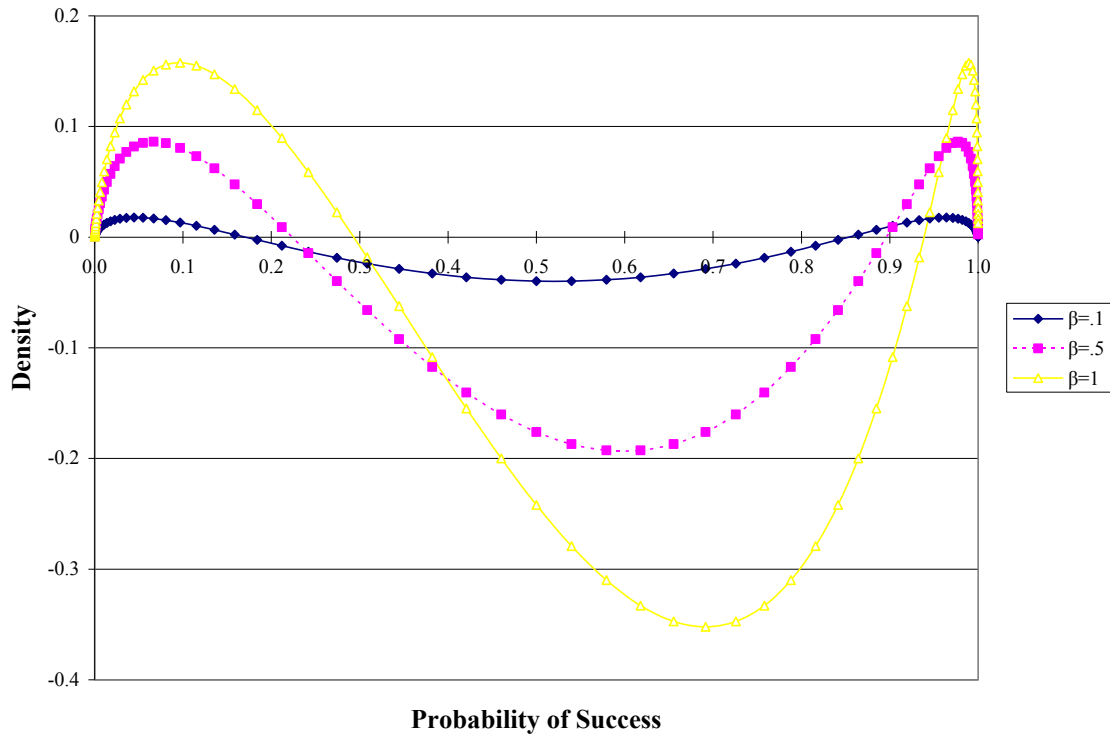
SI Section B Table 1. Probability Of Voting For George W. Bush vs. John Kerry In 2004

Variable	Coefficient	Std. Err.	p value
Constant	-2.611	0.894	0.004
Retrospective Economic Evaluations	0.871	0.273	0.001
Party Identification	0.583	0.087	0.000
Approval of Handling of Iraq War	2.130	0.380	0.000
Ideology	0.181	0.136	0.183
White	0.091	0.308	0.768
Female	-0.081	0.259	0.756
Age in years	-0.004	0.008	0.670
Education	-0.076	0.103	0.459
Income	0.014	0.027	0.597

Log likelihood = -60.14, Pseudo R^2 = 0.781
Number of observations = 383

Notes:
Data are from the 2004 NES, using respondents who first answered the standard turnout question. The variables are coded as follows:

Retrospective Economic Evaluations: -1 = Much Worse; -0.5 = Somewhat Worse; 0 = Same; 0.5 = Somewhat Better; 1 = Much Better

Party Identification: 0 = Strong Democrat; 1 = Weak Democrat; 2 = Independent Democrat; 3 = Independent; 4 = Independent Republican; 5 = Weak Republican; 6 = Strong Republican

Approval of Handling of Iraq War: 0 = Disapprove Strongly; 0.33 = Disapprove Not Strongly; 0.66 = Approve Not Strongly; 1 = Approve Strongly

Ideology: 1 = Extremely Liberal; 2 = Liberal; 3 = Slightly Liberal; 4 = Moderate; 5 = Slightly Conservative; 6 = Conservative; 7 = Extremely Conservative

White: 1 = white, 0 = other

Female: 1 = female, 0 = male

Age: years, 18-90

Education: 1 = 0 - 8 Years; 2 = High School; No Degree; 3= High School Degree; 4 =Some College, No Degree; 5 = Assoc. Degree; 6 = College Degree; 7 =Advanced Degree

Income: 1-23 where 1 = less than $2,999 and 23 = $120,000 +

SI Section B Table 2. Predicted Probability of Voting for George W. Bush vs. John Kerry in 2004, Using the Average Case and Observed Value Approaches, for Variables Not Shown in Figure 1.

Variables and Categories	Predicted Probabilities	
	Average Case Approach[1]	Observed Value Approach[2]
Ideology		
Extremely Liberal	0.39	0.46
Liberal	0.46	0.47
Slightly Liberal	0.53	0.49
Moderate	0.60	0.51
Slightly Conservative	0.67	0.52
Conservative	0.73	0.54
Extremely Conservative	0.79	0.56
White		
White	0.60	0.51
Not white	0.57	0.50
Female		
Female	0.60	0.51
Male	0.63	0.51
Age		
20 years old	0.64	0.52
48 years old	0.60	0.51
65 years old	0.58	0.51
Income		
$22,000-$24,999	0.57	0.50
$40,000-$44,999	0.60	0.51
$70,000-$79,999	0.62	0.51

Notes:
Data are from the 2004 NES, using respondents who first answered the standard turnout question. Results are based on estimates from the model reported in SI Section B Table 1.
1. Predicted probabilities are computed by setting all other independent variables at their mean values.
2. Predicted probabilities are computed by setting all other independent variables at their observed values, using the coefficients in SI Section B Table 1.

SI Section B Table 3. Statistical Simulation Results for Predicted Probability of Voting for George W. Bush vs. John Kerry in 2004, Using the Observed Value Approach, with 95% Confidence Intervals

	Predicted Probability	Confidence Interval Lower Bound	Confidence Interval Upper Bound		Predicted Probability	Confidence Interval Lower Bound	Confidence Interval Upper Bound
Retrospective Economic Evaluation				**White**			
Much Worse	0.44	0.38	0.49	White	0.51	0.49	0.54
Somewhat Worse	0.49	0.45	0.52	Not White	0.51	0.46	0.56
Same	0.53	0.50	0.57				
Somewhat Better	0.59	0.53	0.65	**Female**			
Much Better	0.64	0.55	0.75	Female	0.51	0.48	0.54
				Male	0.52	0.48	0.55
Party Identification							
Strong Democrat	0.24	0.13	0.34	**Age**			
Weak Democrat	0.34	0.25	0.41	20 years old	0.52	0.48	0.57
Independent Democrat	0.44	0.38	0.50	48 years old	0.51	0.49	0.54
Independent	0.56	0.50	0.61	65 years old	0.51	0.47	0.54
Independent Republican	0.67	0.59	0.75				
Weak Republican	0.78	0.68	0.88	**Income**			
Strong Republican	0.87	0.76	0.96	$22,000-$24,999	0.51	0.47	0.54
				$40,000-$44,999	0.51	0.49	0.54
Handling of the Iraq War				$70,000-$79,999	0.52	0.49	0.55
Disapprove Strongly	0.42	0.36	0.47				
Disapprove Not Strongly	0.53	0.49	0.57				
Approve Not Strongly	0.63	0.57	0.71				
Approve Strongly	0.74	0.64	0.86				
Education							
0 - 8 Years	0.54	0.47	0.62				
High School, No Degree	0.53	0.48	0.59				
High School Degree	0.52	0.49	0.56				
Some College, No Degre	0.52	0.49	0.54				
Assoc. Degree	0.51	0.48	0.54				
College Degree	0.50	0.47	0.54				
Advanced Degree	0.50	0.45	0.55				
Ideology							
Extremely Liberal	0.46	0.35	0.54				
Liberal	0.47	0.40	0.53				
Slightly Liberal	0.49	0.44	0.53				
Moderate	0.51	0.48	0.54				
Slightly Conservative	0.53	0.49	0.57				
Conservative	0.55	0.49	0.62				
Extremely Conservative	0.57	0.48	0.68				

Notes:
Data are from the 2004 NES, using respondents who first answered the standard turnout question. Results are from statistical simulation and thus differ slightly from results reported in SI Section B Table 2.

SI Section B Table 4. Predicted Effect (First Difference) of Changes in Select Variables on the Probability of Voting for George W. Bush vs. John Kerry in 2004, Using the Observed Value Approach, with 95% Confidence Intervals

	Effect	Confidence Interval	
		Lower Bound	Upper Bound
Change in Retrospective Economic Evaluation from Same to:			
Much Worse	-0.09	-0.16	-0.03
Somewhat Worse	-0.05	-0.08	-0.02
Somewhat Better	0.05	0.02	0.09
Much Better	0.11	0.03	0.20
Change in Party Identification from Independent to:			
Strong Democrat	-0.32	-0.43	-0.20
Weak Democrat	-0.22	-0.31	-0.13
Independent Democrat	-0.11	-0.16	-0.07
Independent Republican	0.12	0.07	0.16
Weak Republican	0.23	0.15	0.30
Strong Republican	0.32	0.22	0.39
Change in Handling of the Iraq War from Disapprove Not Strongly to:			
Disapprove Strongly	-0.10	-0.16	-0.06
Approve Not Strongly	0.11	0.06	0.17
Approve Strongly	0.22	0.13	0.32
Change in Education from Assoc. Degree to:			
0 - 8 Years	0.03	-0.04	0.11
High School, No Degree	0.02	-0.03	0.08
High School Degree	0.01	-0.02	0.05
Some College, No Degree	0.01	-0.01	0.02
College Degree	-0.01	-0.02	0.01
Advanced Degree	-0.01	-0.05	0.02

Notes:

Data are from the 2004 NES, using respondents who first answered the standard turnout question. Results are from statistical simulation. All changes originate from the mean value of the variable of interest.

Section C: Sample Stata Code (Version 10)[6]

In this appendix we lay out the basic code for computing predicted probabilities and marginal effects using the observed value approach. We use a simple probit model where y represents the dependent variable, x1 is a binary independent variable, and x2 is a continuous independent variable. In the first subsection, all Stata commands are presented in **bold** type. We also include tips for other models, some of which require one to draw additional parameters.

Stata Code for First Differences and Marginal Effects

To get the predicted probability of success for each observation in the sample, after running **probit y x1 x2**, one can simply run **predict p if e(sample)**. The "if e(sample)" portion of the code simply instructs Stata calculate the prediction only for observations on which the model is run. Running **sum p** will give the average probability of success. This is usually not all that informative but is very useful as a check on user generated code to calculate the same quantity and quantities that build from this code. To generate the predicted probability of success for each case, i.e. $\Phi(x\beta)$, using one's own code run:

gen pmycode = normal(_b[_cons] + _b[x1]*x1 + _b[x2]*x2) if e(sample)

This code follows the approach advocated in the Stata manuals, with _b[_cons] representing the value of the constant that Stata stores in memory, _b[x1] representing the coefficient on x1 that Stata stores, and _b[x2] representing the coefficient on x2 that Stata stores in memory. One could type in the values of the estimated coefficients but this code is more general and flexible. As a check, running **gen difference = p – pmycode** and then **sum pmycode difference** should reveal that the results for pmycode are identical to those for p and thus, the difference variable takes on a value of 0 for each case in the analysis. If this is not the case then something is wrong with the code and you should not move forward until the problem is fixed.

---

[6] The margins command introduced in Stata 11 can also be used in many circumstances.

To get the estimated first difference for x1 setting x2 to its observed values we do the following. First, generate the predicted probability of success when x1 is set to 1 by running:

gen px1_1 = normal(_b[_cons] + _b[x1]*1 + _b[x2]*x2) if e(sample)

As can be seen, this involves just a minor modification of the code used to generate the pmycode variable. Second, generate the predicted probability of success when x1 is set to 0 by running:

gen px1_0 = normal(_b[_cons] + _b[x1]*0 + _b[x2]*x2) if e(sample)

To get the effect for each observation run:

gen effectx1 = px1_1 – px1_0

To get the average of the respective predicted probabilities and the average effect run:

sum px1_1 px1_0 effectx1

First differences for x2 can be generated similarly. But since x2 is continuous one might want to generate the marginal effect for x2 for each observation; this can be obtained by running:

gen margeffx2 = normalden(_b[_cons] + _b[x1]*x1 + _b[x2]*x2)*_b[x2] if e(sample)

To get the average marginal effect of x2 simply run **sum margeffx2**. The same basic set up can be used for other models and more complex model specifications.

Stata Code for Implementing the Observed Value Approach Via Simulation

Although, with additional code, *Clarify* can be used to implement the observed value approach for some quantities of interest, the codebook and default settings implement the average case approach; the numerous articles in our content analysis that used *Clarify* noted using the average case approach. We found that writing code in *Clarify* to implement the observed value approach was more difficult and allowed less flexibility in terms of what could be calculated and how it could be saved than starting from scratch and combining the observed

value approach and statistical simulation. Sticking with the simple probit model from above, we

offer the following sample Stata code.

*0) Set the memory to 1000 megabytes and number of variables to 32,767.

set mem 1000m
set maxvar 32767

*1) Open the data and then run the model.

probit y x1 x2

*1a) Drop observations that are not in the model so that the quantities of interest are ///
not calculated for those observations.

gen keep = 1 if e(sample)
drop if keep ~=1

*1b) Save the data as a temporary file named temp, replacing any previous file names temp.

save temp, replace

*2a) Create a mean vector for the coefficients and covariance matrix, draw 1000 sets ///
of coefficients from the multivariate normal using this info.
**Note that that random number seed is set to 99 (chosen as it is Wayne Gretzky's number) ///
using the same seed is helpful for replication purposes.
**The variable named b_x1 will contain the 1000 simulated coefficients for variable x1, ///
the variable named b_x2 will contain the 1000 simulated coefficients for variable x2, ///
and the variable named b_cons will contain the 1000 simulated coefficients for the constant term.
**Note that you can name the coefficients anything as Stata does not recognize the name as ///
meaningful, it just recognizes the order, which follows the order of the output.///
So, the first variable specified will be the coefficient on the independent variable listed first ///
and so on (be careful with the placement of the constant).
**Note that this will temporarily clear the original data set and create a new data set with the ///
simulated coefficients.

set seed 99
mat b = e(b)
mat V = e(V)
drawnorm b_x1 b_x2 b_cons, mean(b) cov(V) n(1000) clear

*Examine the coefficients and check that they are close to the original estimates; ///
if not, something is wrong. It is crucial to note that the variables need to be drawn in the ///
same order in which they appear in the model output. So, if one puts b_cons first Stata will ///
treat this as the coefficient on b1. The following command should help catch such errors.

sum b_x1 b_x2 b_cons

*2b) Merge the original data set (stored as temp) into the data set containing the ///
simulated coefficients.

merge using temp

*3) For each simulated set of coefficients (this was set to 1000 in step 2a above) create a ///
variable that contains the quantity of interest for each observation for each set of simulated ///
coefficients (i.e. loop over the 1000 sets of simulated coefficients).
**For each quantity of interest, this creates 1000 new variables with values for each ///
observation in the data set. They are structured as follows.
**Row 1 for the first new variable contains the quantity of interest using values for ///
the first observation and the first set of simulated coefficients.
**Row n for the first new variable contains the quantity of interest using values for ///
the nth observation and the first set of simulated coefficients.
**Row 1 for the second new variable contains the quantity of interest using values for ///
the first observation and the second set of simulated coefficients.
**Row n for the second new variable contains the quantity of interest using values for ///
the nth observation and the second set of simulated coefficients.
**This process continues for each of the 1000 simulated sets of coefficients.

**Then the mean of each of the new variables for a given quantity of interest is stored in a ///
new variable with a name ending in _mean.
**Row 1 for this new variable ending in _mean contains the mean across all of the ///
observations using the first set of simulated coefficients ///
(i.e. the mean of the first new variable described above).
**Row 2 for this new variable ending in _mean contains the mean across all of the ///
observations using the second set of simulated coefficients ///
(i.e. the mean of the second new variable described above).
**This continues through row 1000 which contains the mean across all of the observations ///
using the thousandth set of simulated coefficients ///
(i.e. the mean of the thousandth new variable described above).

**3a) Calculate the predicted probability of success with all variables set to their ///
observed values.
*Start by creating a variable named p_mean that is set to a missing value and is then filled ///
in after each of the 1000 variables are created.
*Next, calculate the quantity of interest for each observation and each set of simulated ///
coefficients, looping over the 1000 sets of simulated coefficients.
*Finally, fill in the p_mean variable with the mean across all observations from each set of ///
simulated coefficients.

gen p_mean = .
forvalues i = 1/1000 {

```
    gen p_`i' = normal(b_cons[`i'] + x1*b_x1[`i'] + x2*b_x2[`i'])
    summarize p_`i', meanonly
    replace p_mean = r(mean) in `i'
}
```

**Note that the p_1 through p_1000 variables are not needed after p_mean is filled in. ///
To save space these can be dropped by running drop p_1-p_1000. The same logic applies ///
for the intermediate variables calculated in subsequent commands.

**3b) Calculate the predicted probability of success when x1=1 and x2 is set to its ///
observed values.
*Start by creating a variable named px1_1_mean that is set to a missing value and is then ///
filled in after each of the 1000 variables are created.
*Next, calculate the quantity of interest for each observation and each set of simulated ///
coefficients, looping over the 1000 sets of simulated coefficients.
*Finally, fill in the px1_1_mean variable with the mean across all observations from each set ///
of simulated coefficients.

```
 gen px1_1_mean = .
 forvalues i = 1/1000 {
   gen px1_1_`i' = normal(b_cons[`i'] + 1*b_x1[`i'] + x2*b_x2[`i'])
   summarize px1_1_`i', meanonly
   replace px1_1_mean = r(mean) in `i'
}
```

**3c) Calculate the predicted probability of success when x1=0 and x2 is set to its ///
observed values.
*Start by creating a variable named px1_0_mean that is set to a missing value and is then ///
filled in after each of the 1000 variables are created.
*Next, calculate the quantity of interest for each observation and each set of simulated ///
coefficients, looping over the 1000 sets of simulated coefficients.
*Finally, fill in the px1_0_mean variable with the mean across all observations from each set ///
of simulated coefficients.

```
 gen px1_0_mean = .
 forvalues i = 1/1000 {
   gen px1_0_`i' = normal(b_cons[`i'] + 0*b_x1[`i'] + x2*b_x2[`i'])
   summarize px1_0_`i', meanonly
   replace px1_0_mean = r(mean) in `i'
}
```

**3d) Calculate the effect of x1 when moving from x1 = 0 to x1 = 1 (i.e. the first difference).
*Start by creating a variable named effectx1_mean that is set to a missing value and is then ///
filled in after each of the 1000 variables are created.
*Next, calculate the quantity of interest for each observation and each set of simulated ///
coefficients, looping over the 1000 sets of simulated coefficients.

*Finally, fill in the effectx1_mean variable with the mean across all observations from each ///
set of simulated coefficients.

```
 gen effectx1_mean = .
 forvalues i = 1/1000 {
   gen effectx1_`i' = px1_1_`i' - px1_0_`i'
   summarize effectx1_`i', meanonly
   replace effectx1_mean = r(mean) in `i'
}
```
*Note that this could also be done by creating a new variable equal to px1_1_mean minus
px1_0_mean.

**3e) Calculate the marginal effect of x2, when x1 is set to its observed values.
*Start by creating a variable named margeffx2_mean that is set to a missing value and is then ///
filled in after each of the 1000 variables are created.
*Next, calculate the quantity of interest for each observation and each set of simulated ///
coefficients, looping over the 1000 sets of simulated coefficients.
*Finally, fill in the margeffx2_mean variable with the mean across all observations from each ///
set of simulated coefficients.

```
gen margeffx2_mean = .
 forvalues i = 1/1000 {
   gen margeffx2_`i' = normalden(b_cons[`i'] + x1*b_x1[`i'] + x2*b_x2[`i'])*b_x2[`i']
   summarize margeffx2_`i', meanonly
   replace margeffx2_mean = r(mean) in `i'
}
```

*4) For each of the quantities of interest report the mean across the 1000 sets of simulated ///
coefficients.

sum p_mean px1_1_mean px1_0_mean effectx1_mean margeffx2_mean

*5) For each of the quantities of interest report the 95% confidence intervals across the 1000 ///
sets of simulated coefficients.

centile p_mean px1_1_mean px1_0_mean effectx1_mean margeffx2_mean, centile(2.5 97.5)

Tips for Other Models

Depending on the model choice, researchers might need to simulate additional parameter
estimates. The best and safest way to see these estimates is to run the **vce** command after running
the model in Stata. Doing so will present the variance-covariance matrix of the most recently
run model. For example, it will include cutpoint estimates after an ordered model, which are
necessary for estimating the predicted effects and thus need to be simulated. The parameters
need to be specified in the drawnorm line. For a 4-category ordered dependent variable and 2
independent variables the command would be:

drawnorm b_x1 b_x2 cut1 cut2 cut3, mean(b) cov(V) n(1000) clear).

Finally, we offer a reminder that it is crucial to draw the parameters in the order in which they appear in the model output as Stata does not connect the name provided in the drawnorm line to the saved results. Examples for a variety of models can be found at:
http://www.bsos.umd.edu/gvpt/hanmer/

Section D: Additional Monte Carlo Simulation Results

Appendix D Table 1 shows that whether one uses a sample of 1,000 or 100,000 the results from the average case and observed value approaches differ. As can be seen, the results when n = 1,000 are nearly identical to those when n = 100,000.

Appendix D Table 1. Monte Carlo Simulations Showing That the Average Case and Observed Value Approaches Do Not Converge as the Sample Size Increases (Using True Models)[1]

Panel a. Marginal Effects for x_2 and x_3 for True Models

Model:	$y* = -1 + -1x_1 + 1x_2 + 0.5x_3 + e$				$y* = -2.8 + -1x_1 + 1x_2 + 0.5x_3 + e$			
	Sample Size = 1,000		Sample Size = 100,000		Sample Size = 1,000		Sample Size = 100,000	
Variable	Average Case[2]	Observed Values[3]	Average Case[2]	Observed Values[3]	Average Case[2]	Observed Values[3]	Average Case[2]	Observed Values[3]
x_2	0.400	0.230	0.399	0.230	0.076	0.135	0.078	0.135
x_3	0.201	0.115	0.200	0.115	0.038	0.067	0.039	0.067

Panel b: Predicted Probability of Success Across Values of x_1 for True Models

Model:		$y* = -1 + -1x_1 + 1x_2 + 0.5x_3 + e$				$y* = -2.8 + -1x_1 + 1x_2 + 0.5x_3 + e$			
		Sample Size = 1,000		Sample Size = 100,000		Sample Size = 1,000		Sample Size = 100,000	
Variable	Category	Average Case[4]	Observed Values[5]	Average Case[4]	Observed Values[5]	Average Case[4]	Observed Values[5]	Average Case[4]	Observed Values[5]
x_1	1	0.841	0.747	0.841	0.748	0.210	0.297	0.212	0.297
	2	0.500	0.500	0.500	0.500	0.035	0.115	0.036	0.115
	3	0.158	0.252	0.159	0.253	0.003	0.031	0.003	0.031

Notes:
1. True model is: $y* = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + e$, where x_1 has three categories and x_2 and x_3 are continuous.
2. Marginal effects are computed by setting all other independent variables to their sample means.
3. Marginal effects are computed by setting all other independent variables to their observed values in the sample.
4. Predicted probabilities are computed by setting x_1 to its respective values and all other independent variables to their sample means.
5. Predicted probabilities are computed by setting x_1 to its respective values and all other independent variables to their observed values.