# Using the spost13 commands for adjusted predictions and marginal effects with binary dependent variables

Richard Williams, University of Notre Dame, https://www3.nd.edu/~rwilliam/

Last revised January 17, 2022

Note: User-written post-estimation commands sometimes get zapped when a new version of Stata gets released. This is because Stata may change the way results get stored internally, and the command, written for an earlier version of Stata, does not know that. This was true for the spost13 commands when Stata 15 was first released, but I think most issues have now been addressed. But, if a user-written command isn't running correctly, try running the estimation command with version control, e.g.

version 14: logit y x1 x2 x3

**THE SPOST13 COMMANDS**. Many of Long and Freese's spost13 commands are basically shells for `margins`. However, they can be easier to use and produce nicer looking output than does `margins`. Despite the name, most commands also work with Stata 11 and later. To install the commands, from within Stata type

```
findit spost13_ado
```

This handout will primarily focus on the `mtable` and `mchange` commands when they are used in conjunction with `logit`. However, the spost13 commands will also work after many other estimation commands. Indeed I find them even more useful after multiple outcome commands like `ologit` and `mlogit`. A separate handout will focus on those commands.

Note on using SPost9 with SPost13 (copied from https://jslsoc.sitehost.iu.edu/web_spost13/sp13_faqspost9.htm )

The postestimation commands in SPost13 are named mtable, mchange, mgen, and mlincom. They are much more powerful than the prvalue, prtab, prgen, prchange, and praccum comamnds in SPost9. To install the spost13_ado package, you must uninstall the spost9_ado packages (run: ado uninstall spost9_ado). The spost9_legacy package (run: search spost9_legacy) installs versions of the pr* commands that are compatable with SPost13. If you are trying to reproduce earlier work that used SPost9, however, the safest thing is to uninstall spost13_ado and spost9_legacy, and install spost9_ado.

**MTABLE**. The help for `mtable` says "`mtable` uses `margins` to construct tables of predictions. If the outcome has multiple categories, `mtable` automatically submits multiple margins commands for all outcomes and combines the results in the table. Results from multiple calls of `mtable` can be combined into a single table." Here are examples:

```
. * Adjusted predictions & Marginal effects
. * Requires Stata 11+ and spost13
. webuse nhanes2f, clear
. keep if !missing(diabetes, black, female, age)
(2 observations deleted)
. label define black 0 "nonBlack" 1 "black"
. label define female 0 "male" 1 "female"
. label values black black
. label values female female
. logit diabetes i.female i.black c.age c.age#c.age, nolog

Logistic regression                             Number of obs     =     10,335
                                                LR chi2(4)        =     381.03
                                                Prob > chi2       =     0.0000
Log likelihood = -1808.5522                     Pseudo R2         =     0.0953

------------------------------------------------------------------------------
    diabetes |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      female |
     female  |   .1566863   .0942032     1.66   0.096    -.0279486    .3413212
             |
       black |
      black  |   .7207406   .1266509     5.69   0.000     .4725093    .9689718
         age |   .1324622   .0291223     4.55   0.000     .0753836    .1895408
             |
 c.age#c.age |  -.0007031   .0002753    -2.55   0.011    -.0012428   -.0001635
             |
       _cons |   -8.14958   .7455986   -10.93   0.000    -9.610926   -6.688233
------------------------------------------------------------------------------

. * AAPs using margins, mtable
. margins black

Predictive margins                              Number of obs     =     10,335
Model VCE    : OIM

Expression   : Pr(diabetes), predict()

------------------------------------------------------------------------------
             |            Delta-method
             |     Margin   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       black |
   nonBlack  |   .0443074   .0021005    21.09   0.000     .0401906    .0484243
      black  |   .0847238   .0084893     9.98   0.000      .068085    .1013626
------------------------------------------------------------------------------

. mtable, at(black = (0 1)) statistics(all) dec(4)

Expression: Pr(diabetes), predict()

           |    black      Pr(y)         se          z          p         ll         ul
 ----------+---------------------------------------------------------------------
         1 |        0     0.0443     0.0021    21.0941     0.0000     0.0402     0.0484
         2 |        1     0.0847     0.0085     9.9800     0.0000     0.0681     0.1014
```

```
. * AMEs using margins, mtable
. margins, dydx(black)

Average marginal effects                        Number of obs     =     10,335
Model VCE    : OIM

Expression   : Pr(diabetes), predict()
dy/dx w.r.t. : 1.black

------------------------------------------------------------------------------
             |            Delta-method
             |      dy/dx   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       black |
      black  |   .0404163   .0087456     4.62   0.000     .0232752    .0575574
------------------------------------------------------------------------------
Note: dy/dx for factor levels is the discrete change from the base level.

. mtable, dydx(black) statistics(all) dec(4)

Expression: Pr(diabetes), predict()

 d Pr(y)        se         z         p        ll        ul
----------------------------------------------------------
  0.0404    0.0087    4.6213    0.0000    0.0233    0.0576
```

At this point, the advantages of `mtable` over `margins` are not that `great`, if any. The key advantage of `mtable` comes with more complicated tables.

```
. * mtable -- generate several sets of adj predictions with just one command
. mtable, at (black = (0 1) age = 20 ) at (black = (0 1) age = 47 ) at (black = (0 1)
age = 74 ) dec(4) statistics(all)

Expression: Pr(diabetes), predict()

          |    black       age     Pr(y)        se         z         p        ll        ul
----------+--------------------------------------------------------------------------------
        1 |        0        20    0.0033    0.0010    3.4674    0.0005    0.0015    0.0052
        2 |        1        20    0.0069    0.0021    3.3149    0.0009    0.0028    0.0109
        3 |        0        47    0.0325    0.0028   11.7688    0.0000    0.0271    0.0380
        4 |        1        47    0.0647    0.0081    7.9553    0.0000    0.0487    0.0806
        5 |        0        74    0.1078    0.0101   10.6929    0.0000    0.0881    0.1276
        6 |        1        74    0.1990    0.0237    8.3827    0.0000    0.1524    0.2455
```

Adding labels for each set of adjusted predictions is slightly tedious but can help a lot with readability. The below option tells `mtable` to add another row to the previously generated table.

```
. * mtable, nicer formatting
. quietly mtable, at (black = 0 age = 20 ) rown(20 year old White) dec(4) statistics(all)
. quietly mtable, at (black = 1 age = 20 ) rown(20 year old Black) dec(4) statistics(all) below
. quietly mtable, at (black = 0 age = 47 ) rown(47 year old White) dec(4) statistics(all) below
. quietly mtable, at (black = 1 age = 47 ) rown(47 year old Black) dec(4) statistics(all) below
. quietly mtable, at (black = 0 age = 74 ) rown(74 year old White) dec(4) statistics(all) below
. mtable, at (black = 1 age = 74 ) rown(74 year old black) dec(4) below statistics(all)
```

```
Expression: Pr(diabetes), predict()

                   |    Pr(y)        se         z         p        ll        ul
 ------------------+------------------------------------------------------------
  20 year old White |   0.0033    0.0010    3.4674    0.0005    0.0015    0.0052
  20 year old Black |   0.0069    0.0021    3.3149    0.0009    0.0028    0.0109
  47 year old White |   0.0325    0.0028   11.7688    0.0000    0.0271    0.0380
  47 year old Black |   0.0647    0.0081    7.9553    0.0000    0.0487    0.0806
  74 year old White |   0.1078    0.0101   10.6929    0.0000    0.0881    0.1276
  74 year old Black |   0.1990    0.0237    8.3827    0.0000    0.1524    0.2455

Specified values of covariates

          |    black       age
 ----------+-------------------
     Set 1 |        0        20
     Set 2 |        1        20
     Set 3 |        0        47
     Set 4 |        1        47
     Set 5 |        0        74
   Current |        1        74
```

**MCHANGE**. The `mchange` command can also be useful. It lets you mass-produce a bunch of statistics you may be interested in. According to the help, "`mchange` uses `margins` to compute marginal effects for a regression model.  Both marginal changes (partial derivatives) and discrete changes can be computed. Discrete changes of one, a standard deviation, from 0 to 1, over the range, over a trimmed ranged, or for a specific increase from the observed value can be computed." Here is an example. The `commands` option tells you what commands mchange is executing.

```
. * mchange
. mchange, dec(6) commands statistics(all)
Command: margins female if e(sample)==1 , at()   pwcompare predict(pr)
Command: margins black if e(sample)==1 , at()   pwcompare predict(pr)
Command:  margins  if  e(sample)==1,  at(age=gen(age))  at(  age=gen(age+1))  post
predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, at(age=gen(age)) at( age=gen(age+17.21751728311454))
post predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, dydx(age) predict(pr)

logit: Changes in Pr(y) | Number of obs = 10335

Expression: Pr(diabetes), predict(pr)

                   |    Change    p-value         LL         UL    z-value    Std Err       From         To
-------------------+----------------------------------------------------------------------------------------
female             |
    female vs male |  0.006907   0.094660  -0.001193   0.015006   1.671314   0.004132   0.044626   0.051533
black              |
 black vs nonBlack |  0.040416   0.000004   0.023275   0.057557   4.621323   0.008746   0.044307   0.084724
age                |
                +1 |  0.002130   0.000000   0.001601   0.002658   7.898455   0.000270   0.048283   0.050412
               +SD |  0.039164   0.000142   0.018986   0.059342   3.804134   0.010295   0.048283   0.087447
          Marginal |  0.002111   0.000000   0.001613   0.002610   8.304949   0.000254         .z         .z

Average predictions

             |         0          1
-------------+----------------------
  Pr(y|base) |  0.951717   0.048283
```

For the variables female and black, `mchange` gave you the marginal effects (since they can only change from 0 to 1). For the continuous variable age, you not only got the marginal effect, you got the amount of change that would be produced if every member of the sample was one year older and the amount of change there would be if every sample member were one standard deviation older (about 17.22 years).

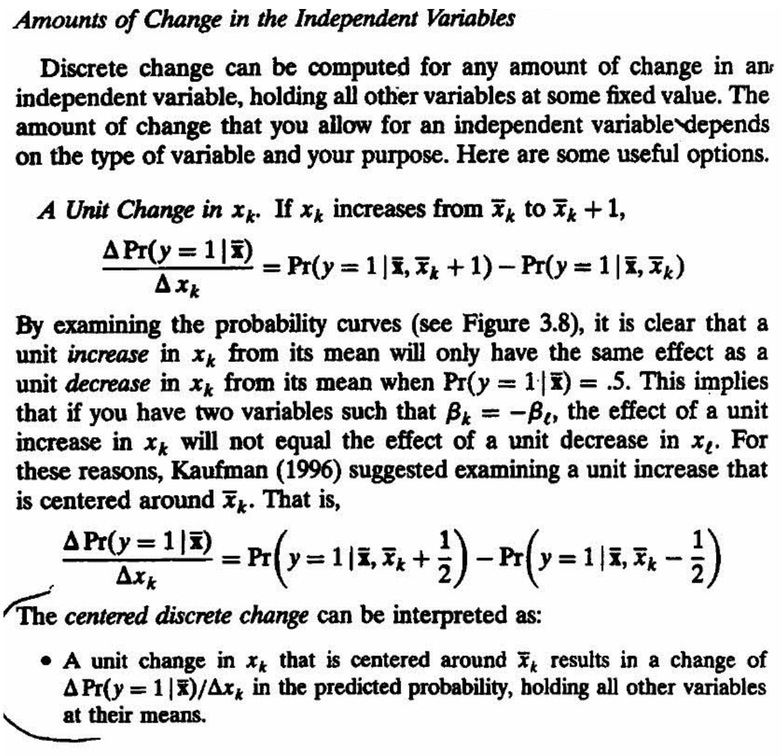
See the help for the numerous other options that are available.

By default, `mchange` uses uncentered changes. The optional appendix explains what uncentered changes are and why you might prefer centered changes instead.

# Appendix (Optional):
# Why look at the effects of centered as opposed to uncentered changes?

Note: This appendix assumes knowledge of the `mchange` command that is part of Long & Freese's `spost13` package of commands.

By default, Long & Freese's spost13 `mchange` command uses uncentered changes. This is a reversal from the earlier `prchange` command (and also from the early beta versions of `mchange`), which defaulted to centered changes. Why would you want to center changes when computing the effect of a 1 unit (or sd, or user-specified delta) change (e.g. why would you look at change between x-.5 and x + .5 as opposed to change between x and x + 1)? Long (1997, p. 77) says the following:



***Amounts of Change in the Independent Variables***

**Discrete change can be computed for any amount of change in an independent variable, holding all other variables at some fixed value. The amount of change that you allow for an independent variable depends on the type of variable and your purpose. Here are some useful options.**

***A Unit Change in $x_k$.*** **If $x_k$ increases from $\bar{x}_k$ to $\bar{x}_k + 1$,**

$$\frac{\Delta \Pr(y=1\mid\bar{\mathbf{x}})}{\Delta x_k} = \Pr(y=1\mid\bar{\mathbf{x}},\bar{x}_k+1) - \Pr(y=1\mid\bar{\mathbf{x}},\bar{x}_k)$$

**By examining the probability curves (see Figure 3.8), it is clear that a unit *increase* in $x_k$ from its mean will only have the same effect as a unit *decrease* in $x_k$ from its mean when $\Pr(y=1\mid\bar{\mathbf{x}}) = .5$. This implies that if you have two variables such that $\beta_k = -\beta_\ell$, the effect of a unit increase in $x_k$ will not equal the effect of a unit decrease in $x_\ell$. For these reasons, Kaufman (1996) suggested examining a unit increase that is centered around $\bar{x}_k$. That is,**

$$\frac{\Delta \Pr(y=1\mid\bar{\mathbf{x}})}{\Delta x_k} = \Pr\left(y=1\mid\bar{\mathbf{x}},\bar{x}_k+\frac{1}{2}\right) - \Pr\left(y=1\mid\bar{\mathbf{x}},\bar{x}_k-\frac{1}{2}\right)$$

**The *centered discrete change* can be interpreted as:**

- **A unit change in $x_k$ that is centered around $\bar{x}_k$ results in a change of $\Delta\Pr(y=1\mid\bar{\mathbf{x}})/\Delta x_k$ in the predicted probability, holding all other variables at their means.**

Let's see what that means in practice. First, we will examine the effect of uncentered changes in age, for 1 year and for 10 years. Then, we will compute a variable called xage which is the negative of age, and compute the effects of uncentered changes in it.

```
. webuse nhanes2f, clear
. quietly logit diabetes i.black i.female age c.age#c.age, nolog
. est store age
. mchange age, dec(6) uncentered delta(10) commands
Note: uncentered is the default; use option centered for centered change.
Command: margins if e(sample)==1, at(age=gen(age)) at( age=gen(age+1)) post predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, at(age=gen(age)) at( age=gen(age+10)) post predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, dydx(age) predict(pr)

logit: Changes in Pr(y) | Number of obs = 10335

Expression: Pr(diabetes), predict(pr)

             |    Change    p-value
-------------+---------------------
age          |
          +1 |  0.002130   0.000000
      +delta |  0.022481   0.000000
    Marginal |  0.002111   0.000000

1: Delta equals 10.

. gen xage = -age
. quietly logit diabetes i.black i.female xage c.xage#c.xage, nolog
. est store xage
. mchange xage, dec(6) uncentered delta(10) commands
Note: uncentered is the default; use option centered for centered change.
Command: margins if e(sample)==1, at(xage=gen(xage)) at( xage=gen(xage+1)) post predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, at(xage=gen(xage)) at( xage=gen(xage+10)) post predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, dydx(xage) predict(pr)

logit: Changes in Pr(y) | Number of obs = 10335

Expression: Pr(diabetes), predict(pr)

             |    Change    p-value
-------------+---------------------
xage         |
          +1 | -0.002092   0.000000
      +delta | -0.018889   0.000000
    Marginal | -0.002111   0.000000

1: Delta equals 10.
```

As we can see, even though xage is simply the negative of age, the effects of uncentered changes are different for xage and age. It may seem a little odd to imply that the effect of a variable depends on whether it is coded from low to high or high to low. As Long notes, we would see similar things if we had two variables with equally large effects that were opposite in sign. These differences result from the fact that in one case, you are looking at how changes in X move a case up the S curve, while in the other case you are looking at how changes in X move a case down the S curve. In general these amounts will not be the same. So if, say, we were interested in comparing the effects of two variables that had equally large effects that were opposite in sign, it really wouldn't be a fair comparison because we would be comparing effects that go in opposite directions.

Now let's see what happens when we use centered effects:

```
. est restore age
(results age are active now)
. mchange age, dec(6) delta(10) centered commands
Command: margins if e(sample)==1, at(age=gen(age-.5)) at( age=gen(age+.5)) post predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, at(age=gen(age-5)) at( age=gen(age+5)) post predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, dydx(age) predict(pr)

logit: Changes in Pr(y) | Number of obs = 10335

Expression: Pr(diabetes), predict(pr)

             |    Change    p-value
-------------+----------------------
age          |
 +1 centered |  0.002111   0.000000
+delta cen~d |  0.021005   0.000000
    Marginal |  0.002111   0.000000

1: Delta equals 10.

. est restore xage
(results xage are active now)

. mchange xage, dec(6) delta(10) commands
Command: margins if e(sample)==1, at(xage=gen(xage-.5) ) at( xage=gen(xage+.5)) post predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, at(xage=gen(xage-5) ) at( xage=gen(xage+5)) post predict(pr)
Command: mlincom 2 - 1
Command: margins if e(sample)==1, dydx(xage) predict(pr)

logit: Changes in Pr(y) | Number of obs = 10335

Expression: Pr(diabetes), predict(pr)


             |    Change    p-value
-------------+----------------------
xage         |
     +1 cntr | -0.002111   0.000000
 +delta cntr | -0.021005   0.000000
    Marginal | -0.002111   0.000000

1: Delta equals 10.
```

We see that the centered effects for age and xage are equal in magnitude but opposite in sign. Thus, by looking at centered changes, we avoid the possibly misleading comparisons that might result from comparing effects of variables that were opposite in sign or that were coded in opposite directions.