SOCY498C—Introduction to Computing for Sociologists
Neustadtl

Regression Post-Estimation: `adjust`

After you have created a dataset, examined your variables, constructed and estimated a model, the real work begins. Stata has a number of *post-estimation* commands that can be used to assess model assumptions as well as provide additional results to support your analysis. One useful post-estimation command is `adjust`. In Stata version 11 `adjust` has been replaced with `margins`, but `adjust` still works, for now.

`adjust` (`help adjust`)

- After an estimation command, adjust provides adjusted predictions of **xb** (the means in a linear-regression setting), probabilities (available after some estimation commands), or exponentiated linear predictions. The estimate is computed for each level of the `by()` variables, setting the variables specified in `[var[= #]...]` to their mean or to the specified number if the `= #` part is specified. If `by()` is not specified, `adjust` produces results as if `by()` defined one group. Variables used in the estimation command but not included in either the `by()` variable list or the adjust variable list are left at their current values, observation by observation.
- There are many options including:
    - `xb` (linear prediction; the default),
    - `se` (display standard error of the prediction),
    - `stdf` (display standard error of the forecast),
    - `ci` (display confidence or prediction intervals), and
    - `level(#)` (set confidence level).

Creating the Dataset

Assuming that you have downloaded the GSS subset data file (GSS-98-08.dta) from the course Web page and placed it in a directory called "C:\data" the following program creates a dataset used for these examples. You will probably need to make some changes to reflect how your computer is setup.

```
/* Create subset of the GSS data for this example */
#delimit ;
use year
    prestg80
    educ
    age
    sex
    race
    marital using "C:\data\GSS-98-08.dta" if year==2008 & race<3, clear
;
#delimit cr

/* Create 0/1 indicator variable */
rename sex female
rename race black
drop year
```

Now, we can 1) estimate our regression model, 2) create a new variable containing the predicted values for each observation $\left(\hat{Y}\right)$, and 3) examine the predicted values. To estimate this model I used the `xi:` prefix command to create dummy variables. This is a great Stata shortcut for working with dummy variables and interaction terms in regression models. (see `help prefix` or `help xi`).

Figure 1.

```
. xi: regress prestg80 educ age i.female i.black
i.female          _Ifemale_1-2        (naturally coded; _Ifemale_1 omitted)
i.black           _Iblack_1-2         (naturally coded; _Iblack_1 omitted)

      Source |       SS       df       MS              Number of obs =    1738
-------------+------------------------------           F(  4,  1733) =  170.03
       Model |  93151.8494     4  23287.9624           Prob > F      =  0.0000
    Residual |  237360.857  1733  136.965296           R-squared     =  0.2818
-------------+------------------------------           Adj R-squared =  0.2802
       Total |  330512.707  1737  190.277897           Root MSE      =  11.703

------------------------------------------------------------------------------
    prestg80 |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        educ |   2.458468   .0971547    25.30   0.000     2.267915     2.64902
         age |   .0995691   .0166093     5.99   0.000     .0669926    .1321455
  _Ifemale_2 |   .3035981    .563288     0.54   0.590    -.8011976    1.408394
   _Iblack_2 |  -1.443287   .7970215    -1.81   0.070    -3.006512    .1199385
       _cons |   5.707446   1.677661     3.40   0.001     2.416993      8.9979
------------------------------------------------------------------------------

. keep if e(sample)
(102 observations deleted)
```

The variables in this model collectively explain approximately 28% of the variation in occupational prestige. All of the independent measures are statistically significant (p<0.05) except for respondent gender and race.

What does the `keep` command in Figure 1 do? For the rest of my analysis I only want to analyze observations that were used in my model, i.e. all cases with complete data for all variables. There are many ways to do this but I will use the `keep` command with `e(sample)`. All estimation commands like `regress` save something called `e(sample)` that indicate with 0's and 1's which observations were used in the last estimation. A values of 1 means the observation was used in the last estimation (i.e. no missing data) and a value of 0 means the observation was excluded from the model due to missing data (i.e. missing data on at least one variable). This can then be used with almost any Stata command after estimation to restrict that command to the estimation sample (see `help postest`).

Combining this with `keep` I can isolate the two classes of observations—those included in the model or excluded from the model. The following Stata commands are equivalent ways to keep (or drop) cases used (or not used) in the estimated model: `keep if e(sample)` or `drop if !e(sample)`

We will use `adjust` to look at the expected values or means of the predicted values so we need to create this variable. There are many ways to create predicted values and I will show you three just so you learn a little more about Stata.

Method one generates $\hat{Y}$ by using the parameter estimates from the regression model. Method two uses the same values that are stored automatically by Stata after estimating the model (see `help _variables`). The third method, my preferred method, used the `predict` command. The predict command can also be used to calculate residuals, the error term (see `help regress postestimation##predict`).

Method 1:

```
gen yhat=5.707446   + ( 2.458468 *educ) +           ///
                      (  .0995691*age) +            ///
                      (  .3035981*_Ifemale_2) +     ///
                      (-1.443287 *_Iblack_2)
```

Method 2:

```
gen yhat1=_b[_cons] + (_b[educ]*educ) +             ///
                      (_b[age]*age) +               ///
                      (_b[_Ifemale_2]*_Ifemale_2) + ///
                      (_b[_Iblack_2]*_Iblack_2)
```

Method 3:

```
Predict yhat, xb
```

After creating $\hat{Y}$ we can examine summary statistics using normal summary commands and `adjust`.

In this example you can see that the results are the same! When you use the `adjust` command without specifying any variables, it simply summarizes the linear predictions, the expected values, of the regression as does `adjust`.

Figure 2.

```
. tabstat yhat

    variable |      mean
-------------+----------
        yhat |  43.82163
------------------------

. adjust

------------------------------------------------------------------
     Dependent variable: prestg80     Command: regress
  Variables left as is: age, educ, _Ifemale_2, _Iblack_2
------------------------------------------------------------------

----------------------
      All |         xb
----------+-----------
          |    43.8216
----------------------
     Key:  xb  =  Linear Prediction
```

In this example I use the `table` command (though `tabstat` would have produced the same results). This command is very powerful and worth some time looking at the documentation (`help table`).

Again, the results are the same because without specifying any variables `adjust` simply summarizes the linear predictions of the regression by *marital*.

Figure 3.

```
. table marital if !missing(marital), contents(mean  yhat) format(%6.4f)

----------------------------
marital       |
status        | mean(yhat)
--------------+-------------
      married |    44.5551
      widowed |    44.1365
     divorced |    43.6481
    separated |    39.9461
never married |    42.8221
----------------------------

. adjust, by(marital)

----------------------------------------------------------------------
     Dependent variable: prestg80     Command: regress
   Variables left as is: age, educ, _Ifemale_2, _Iblack_2
----------------------------------------------------------------------

----------------------------
marital       |
status        |         xb
--------------+-------------
      married |    44.5551
      widowed |    44.1365
     divorced |    43.6481
    separated |    39.9461
never married |    42.8221
----------------------------
     Key:  xb  =  Linear Prediction
```

This example demonstrates how `adjust` (and `table`) produce the average predicted values for two discrete measures, in this case the intersection of marital status and gender.

As you might suspect, the results are the same. So, you see that `adjust` easily provides average expected values for categorical variables. Up to seven variables can be used with the `by()` option.

Figure 4.

```
. table marital female if !missing(marital), contents(mean yhat) format(%6.4f)

--------------------------------
marital      | respondents sex
status       |    male   female
-------------+------------------
     married | 44.4602  44.6455
     widowed | 45.0683  43.8953
    divorced | 42.9887  44.1403
   separated | 40.3372  39.7289
never married| 42.5428  43.1378
--------------------------------

. adjust, by(marital female)

-------------------------------------------------------------------------------
     Dependent variable: prestg80     Command: regress
  Variables left as is: age, educ, _Ifemale_2, _Iblack_2
-------------------------------------------------------------------------------

--------------------------------
marital      | respondents sex
status       |    male   female
-------------+------------------
     married | 44.4602  44.6455
     widowed | 45.0683  43.8953
    divorced | 42.9887  44.1403
   separated | 40.3372  39.7289
never married| 42.5428  43.1378
--------------------------------
     Key:  Linear Prediction
```

This example is silly but shows some of the power of the `adjust` command. In the first example we see the expected values for the intersection of marital status and gender. The second example shows the same thing except that the race dummy variable, specified as part of the `adjust` command, not as a `by()` variable, is set to the mean of race, 0.15025907.

The silliness here is that the race dummy is either 0 or 1 and unless we can somehow justify viewing people as 15% black and 85% white, this doesn't make sense, though there are analysis situations where examining "mixtures" would be interesting.

In the second example you see that *age*, *educ*, and *_Ifemale_2* are left "as is" meaning the value of each observation. The race dummy, *_Iblack_2*, on the other hand, is set to the mean.

Furthermore, it is possible to set values that are theoretically interesting.

Figure 5.

```
. adjust, by(marital female) format(%6.2f)

-------------------------------------------------------------------------------
     Dependent variable: prestg80     Command: regress
  Variables left as is: age, educ, _Ifemale_2, _Iblack_2
-------------------------------------------------------------------------------

-----------------------------
marital      | respondents sex
status       |    male female
-------------+---------------
     married |   44.46  44.65
     widowed |   45.07  43.90
    divorced |   42.99  44.14
   separated |   40.34  39.73
never married|   42.54  43.14
-----------------------------
     Key:  Linear Prediction

. adjust _Iblack_2, by(marital female) format(%6.2f)

-------------------------------------------------------------------------------
     Dependent variable: prestg80     Command: regress
  Variables left as is: age, educ, _Ifemale_2
 Covariate set to mean: _Iblack_2 = .15025907
-------------------------------------------------------------------------------

-----------------------------
marital      | respondents sex
status       |    male female
-------------+---------------
     married |   44.38  44.56
     widowed |   45.20  43.79
    divorced |   42.92  44.15
   separated |   40.77  40.11
never married|   42.61  43.37
-----------------------------
     Key:  Linear Prediction
```

This table shows the predicted values for respondents with average education (13.547496) and age (48.75475) by marital status and gender.

Figure 6.

```
. adjust educ age, by(marital female) format(%6.2f)

-------------------------------------------------------------------------------
     Dependent variable: prestg80     Command: regress
  Variables left as is: _Ifemale_2, _Iblack_2
Covariates set to mean: educ = 13.547496, age = 48.75475
-------------------------------------------------------------------------------

-----------------------------
marital      | respondents sex
status       |    male female
-------------+---------------
     married |   43.73  44.04
     widowed |   43.52  44.06
    divorced |   43.72  43.94
   separated |   43.22  43.57
never married|   43.58  43.72
-----------------------------
     Key:  Linear Prediction
```

Even crazier, you can evaluate the expected values under situations like if "all of observations are white males." (*_ifemale_2* and *_Iblack_2* are both equal to 0)

Figure 7.

```
. adjust _Ifemale_2=0 _Iblack_2=0 , by(marital female) format(%6.2f)

     Dependent variable: prestg80     Command: regress
   Variables left as is: age, educ
Covariates set to value: _Ifemale_2 = 0, _Iblack_2 = 0
```

marital status	respondents sex male	female
married	44.60	44.47
widowed	45.42	43.71
divorced	43.13	44.06
separated	40.99	40.03
never married	42.83	43.28

Key: Linear Prediction

The `adjust` command has other useful options including calculating confidence and prediction (forecast) intervals around the predicted values. The `ci` option and `stdf` options are used to produce these results.

Here we see the 95% confidence intervals for $\hat{Y}$ for different marital categories.

Figure 8.

```
. adjust, by(marital female) format(%6.2f) ci

     Dependent variable: prestg80     Command: regress
   Variables left as is: age, educ, _Ifemale_2, _Iblack_2
```

marital status	respondents sex male	female
married	44.46	44.65
	[43.64,45.28]	[43.89,45.40]
widowed	45.07	43.90
	[44.04,46.09]	[42.79,45.00]
divorced	42.99	44.14
	[42.17,43.81]	[43.38,44.90]
separated	40.34	39.73
	[39.38,41.29]	[38.84,40.62]
never married	42.54	43.14
	[41.65,43.44]	[42.23,44.05]

Key: Linear Prediction
[95% Confidence Interval]

This example shows how you can use `adjust` to provide confidence intervals around the predicted values for different ages (X_h).

The `if` statement uses the `mod()` function to calculate values only for people with ages divisible by 5 with no remainder. For example, it includes people who are 25 years old (25/5=5 so no remainder) and excludes people who are 26 years old (26/5 leaves a remainder of 1). Google "modulus" for more details.

Figure 9.

```
. adjust if mod(age,5)==0, by(age) format(%6.2f) ci

     Dependent variable: prestg80     Command: regress
   Variables left as is: educ, _Ifemale_2, _Iblack_2
```

age of responden t	xb	lb	ub
20	38.65	[37.54	39.76]
25	43.15	[42.18	44.13]
30	43.55	[42.69	44.40]
35	43.87	[43.15	44.58]
40	44.53	[43.88	45.19]
45	44.01	[43.43	44.59]
50	44.34	[43.78	44.90]
55	43.82	[43.21	44.43]
60	44.41	[43.71	45.11]
65	46.18	[45.39	46.97]
70	45.66	[44.77	46.54]
75	47.27	[46.23	48.31]
80	45.43	[44.26	46.59]
85	44.69	[43.39	46.00]

Key: xb = Linear Prediction
[lb , ub] = [95% Confidence Interval]

Of course you can use the `if` statement to determine these values for any age (e.g. `adjust if age==23`).

Finally, we can extending this syntax and calculate the prediction or forecast standard errors and intervals for the age groups defined by the `mod()` function by specifying the `stdf` option.

Figure 10.

```
. adjust if mod(age,5)==0, by(age) stdf ci

------------------------------------------------------------------
     Dependent variable: prestg80     Command: regress
  Variables left as is: educ, _Ifemale_2, _Iblack_2
------------------------------------------------------------------

----------------------------------------------------------------
age of    |
responden |
t         |        xb        stdf          lb              ub
----------+-----------------------------------------------------
       20 |   38.6481   (11.7169)     [15.6672        61.6289]
       25 |   43.1531   (11.7138)     [20.1785        66.1277]
       30 |   43.5463   (11.7113)     [20.5765        66.5161]
       35 |   43.8665   (11.7089)     [20.9014        66.8315]
       40 |   44.5349   (11.7079)     [21.5717        67.4981]
       45 |   44.0119   (11.7069)     [21.0507        66.9731]
       50 |   44.3416   (11.7067)      [21.381        67.3023]
       55 |   43.8195   (11.7074)     [20.8574        66.7816]
       60 |   44.4124   (11.7086)     [21.4478        67.3769]
       65 |   46.1796   (11.7101)     [23.2122         69.147]
       70 |   45.6589   (11.7119)     [22.6879        68.6299]
       75 |   47.2702   (11.7152)     [24.2927        70.2477]
       80 |   45.4257   (11.7182)     [22.4424         68.409]
       85 |   44.6943   (11.7222)     [21.7032        67.6855]
----------------------------------------------------------------
     Key:  xb         =  Linear Prediction
           stdf       =  Standard Error (forecast)
           [lb , ub]  =  [95% Prediction Interval]
```

Problems

Use the General Social Survey for 1988 and create a dataset with the following variables: *year sexfreq, sex, race, educ, marital, age, childs, reliten,* and *attend*. Recode *sexfreq* and *attend* to reflect yearly numbers and recode *reliten* to fix the problem with the order of the responses. Drop all observations where race is equal to "other".

sexfreq		*reliten*		*attend*	
Original	New	Original	New	Original	New
0	0	1	4	0	0.0
1	2	2	2	1	0.5
2	12	3	3	2	1.0
3	36	4	1	3	6.0
4	52			4	12.0
5	156			5	30.0
6	208			6	45.0
				7	52.0
				8	104.0

1. Regress the sexual frequency measure on *sex, race, educ, marital status,* and *childs.* The variables sex and race are dummy variables. Code them so that 1= female and 1=black, respectively. The variable marital requires four separate dummy variables since there are five categories (married, widowed, divorced, separated, and never married). Exclude the married people from the regression model. *Nota bene*: you can (maybe should) use the xi: prefix for your regressions to make life easier. (see `help xi`).
2. Use adjust to calculate the following for cases that were used in the regression model:
   a. What is the average predicted value for the entire sample?
   b. What are the average predicted values for the intersection of religious intensity and sex. Interpret this table.
   c. Hold the variables age and education constant at their means and calculate the average predicted values for the intersection of religious intensity and sex.
   d. Calculate the average predicted value of yearly sexual frequency for the intersection of age and sex for ages between 18 and 89 that end in 0 or 5 (e.g. 20, 25,…,85).