Getting Centered and Standardized Coefficients Right

Doug Hemken
Social Science Computing Cooperative
University of Wisconsin – Madison
Madison, Wisconsin
dehemken@wisc.edu

Abstract. The Stata command **regress, beta** works for only additive models with no factor variables. For models with interaction effects it fails to center lower-order terms it uses the wrong standard deviation for higher-order terms. With factor variables, it treats slopes the same as intercepts. The **stdBeta** program makes it simple to get correctly centered or standardized coefficients.

1 INTRODUCTION

Centering and standardizing data are fundamental data manipulations, widely used in the derivation of statistical theory, the improvement of numerical computation, and at times useful in understanding and reporting statistical models. Most statistical software includes routines for centering and standardizing your data, and for calculating and reporting standardized coefficients. However, these routines have limitations that are not necessarily obvious in use – you can produce nonsense results with no warnings. If you are going to report and use standardized coefficients, you need to calculate them correctly. And simply typing **regress, beta** will only get you there some of the time.

The fundamental idea of a standardized coefficient is that it tells you about the relationships in your data as if your data were scaled in standardized (z-score) form. While you can always estimate these coefficients by first standardizing your data and then estimating your model, most students learn that you can calculate standardized coefficients directly from the unstandardized coefficients along with the standard deviations of the variables involved, without having to actually transform the data.

$$\beta = \frac{\sigma_x}{\sigma_y} b$$

The limitation, though, is that this classic formula only works within additive models, not models that include interaction terms (see Friedrich 1982 or Aiken & West 1991). And yet most software, Stata included, will blithely report incorrect betas for models with interaction terms.

Centering your data helps, in that the classic formula works for all first order terms (main effects) after the data are centered, but it only works for higher order terms when the correlation between the independent variables involved in the interaction is precisely zero. The reported “standardized” coefficient from **–beta--** will sometimes be close to the correct value, yet not quite right.

Which leaves the analyst to go back to square one: first transform the data, then estimate the model. This is not necessarily a bad thing, because it allows us to consider another major short-coming of most (all?) coefficient standardizing software: no distinction is made between indicator coefficients (intercepts) and regression coefficients (slopes). Many folks would argue that we only want to standardize regression coefficients, and not indicators.

Fortunately, with modern software like Stata it is pretty easy both to automate the distinction between indicators and continuous variables, and to rescale and re-run models. What follows is a review of what works, where it goes wrong, and how to get it right. We introduce a program, **stdBeta**, that makes it easy to get the correctly centered or standardized coefficients after the analysis in the original units.

2 THE STDBETA COMMAND

2.1 SYNTAX

stdBeta [, nodepvar store replace *estimates_table_options*]

2.2 DESCRIPTION

stdBeta is a post-estimation command that works by separating the independent variables in a linear model into intercepts and slopes (factors and covariates), centering and rescaling the slopes, and refitting the model to the transformed data. **stdBeta** restores the data to its original form, and leaves current the estimation results from the original model, with an option to store the centered and standardized estimation results. Reported results include any option available with **estimates table**.

2.3 OPTIONS

nodepvar prevents the dependent variable in a model from being centered and standardized.

store saves the estimation results of the original, centered, and standardized models, using **estimates store**. The results are named Original, Centered, and Standardized.

replace overwrites previously stored results. To both overwrite and save results, specify both **replace store**.

estimates_table_options any estimation reporting options available with **estimates table** may be used, including a variety of fit statistics, and exponentiation of the resulting coefficients.

2.4 SAVED RESULTS

None per se. Estimation results from the original model are restored.

3 GETTING TO THE BOTTOM OF BETA

It helps to think of standardizing a variable as a two-stage process: first you center a variable, then you rescale it with its standard deviation.

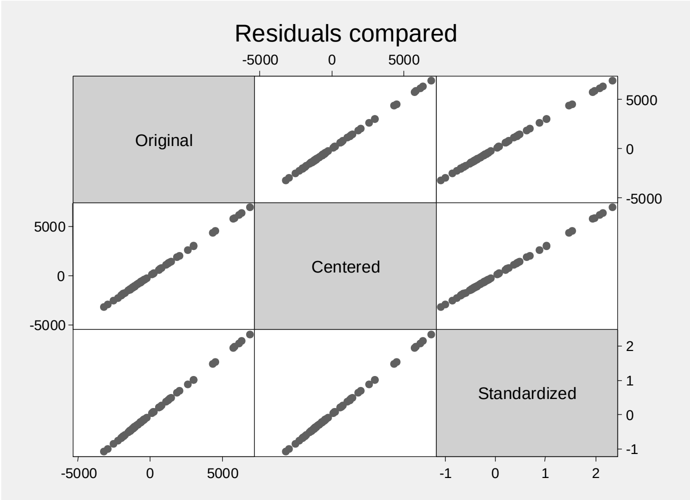
The command **regress, beta** works perfectly for additive regression models, models with a single intercept and no interaction terms. We use the **stdBeta** command to see the coefficients at each stage of standardization.

```
. sysuse auto

. regress price weight displacement, beta

------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|                     Beta
-------------+----------------------------------------------------------------
      weight |   1.823366   .8498204     2.15   0.035                 .4804578
displacement |   2.087054     7.1918     0.29   0.773                 .0649837
       _cons |    247.907   1472.021     0.17   0.867                        .
------------------------------------------------------------------------------

. stdBeta

-----------------------------------------------------------
    Variable |   Original       Centered     Standardized
-------------+---------------------------------------------
      weight |     1.823366       1.823366       .4804578
displacement |    2.0870541      2.0870538      .06498373
       _cons |    247.90702      4.579e-06     -4.725e-10
-----------------------------------------------------------
```

We see that in an additive model, when we center the data the regression slopes remain unchanged and the intercept moves to the new zero. By adding the **se** option to **stdBeta** we also see that the standard errors of the regression coefficients are also unchanged by centering (although the standard error of the constant changes), and that statistics related to the overall model fit such as F or R^2 are unchanged by both centering and rescaling.

```
. stdBeta, se stats(r2 F)

-----------------------------------------------------------
    Variable |   Original       Centered     Standardized
-------------+---------------------------------------------
      weight |     1.823366       1.823366       .4804578
             |    .84982037      .84982036      .22392806
displacement |    2.0870541      2.0870538      .06498373
             |    7.1918002      7.1918002      .22392807
       _cons |    247.90702      4.579e-06     -4.725e-10
             |    1472.0213      292.75523      .09925602
-------------+---------------------------------------------
          r2 |    .29094334      .29094334      .29094334
           F |    14.566521      14.566521      14.566521
-----------------------------------------------------------
                                               legend: b/se
```

Related to the equal R^2, it is also helpful to understand that these models are equivalent in that their residuals are the same, except for a scaling constant. This is especially obvious in a scatter plot of the residuals.



However, if our model includes an interaction term, then the results of **beta** versus actually transforming the data and refitting the model disagree. The **beta** routine, the simple application of our old classic formula, breaks in three different ways.

```
. regress price c.weight##c.displacement, beta

------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|                     Beta
-------------+----------------------------------------------------------------
      weight |  -.6695052   1.009044    -0.66   0.509                -.1764149
displacement |   -47.9457   14.50171    -3.31   0.001                -1.492865
             |
   c.weight#|
          c. |
displacement |   .0143162   .0036987     3.87   0.000                 2.194285
             |
       _cons |   8215.684   2459.235     3.34   0.001                        .
------------------------------------------------------------------------------

. stdBeta

------------------------------------------------------------
    Variable |   Original       Centered     Standardized
-------------+----------------------------------------------
      weight |   -.66950518      2.1550417      .56785452
displacement |   -47.945695     -4.7185194     -.14691856
             |
   c.weight#|
          c. |
displacement |     .0143162       .0143162      .34643981
             |
       _cons |    8215.6839     -902.06777     -.30583796
------------------------------------------------------------
```

The first problem is that **regress, beta** applies the classic coefficient standardization formula to the uncentered coefficients. A "hand" calculation using the coefficients from **stdBeta** easily verifies that we get the correct standardized coefficient for lower-order terms if we apply the standardization formula to the centered coefficient. That is

$$\beta = \frac{\sigma_x}{\sigma_y} b^c \neq \frac{\sigma_x}{\sigma_y} b$$

```
. tabstat price weight, statistics(sd)

   stats |     price    weight
---------+--------------------
      sd |  2949.496  777.1936
------------------------------
```

So $\beta_{weight} \neq \frac{\sigma_{weight}}{\sigma_{price}} b_{weight}$

```
// not centered, not correct
. display -.6695052 *777.1936/2949.496
-.17641494
```

while $\beta_{weight} = \frac{\sigma_{weight}}{\sigma_{price}} b^c_{weight}$

```
// centered, correct
. display 2.1550417 *777.1936/2949.496
.56785451
```

Our software uses the classic formula on the wrong coefficient! A part of the magical simplicity of the classic standardization formula is that in additive models, uncentered and centered regression coefficients are the same.

A second problem is that the intercept is no longer zero, but **beta** always ignores the constant. If we are willing to ignore the constant, it appears the first problem might be solved by first centering our data and then applying **regress, beta**.

```
. preserve
. foreach var of varlist price weight displacement {
  2.         quietly summarize `var'
  3.         replace `var' = `var' - r(mean)
  4.         }

. regress price c.weight##c.displacement, beta

------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|                     Beta
-------------+----------------------------------------------------------------
      weight |   2.155042   .7814836     2.76   0.007                 .5678545
displacement |  -4.718519   6.804687    -0.69   0.490                -.1469186
             |
    c.weight#|
          c. |
displacement |   .0143162   .0036987     3.87   0.000                 .3799967
             |
       _cons |  -902.0678   354.8505    -2.54   0.013                        .
------------------------------------------------------------------------------
```

Almost but not quite! First centering the data gives us the correct first order betas (the standardized main effects), but does not get the standardized interaction coefficient quite correct. This third problem is that the **beta** algorithm uses the standard deviation of the product formed by the components of the interaction, σ_{x1x2}, rather than using the product of the standard deviations, $\sigma_{x1}\sigma_{x2}$. That is, for a second order term

$$\beta = \frac{\sigma_{x1}\sigma_{x2}}{\sigma_y} b^c \neq \frac{\sigma_{x1x2}}{\sigma_y} b$$

Again, a few "hand" calculations using the coefficients from **stdBeta** can verify this.

```
. generate wgtxdisp = weight*displacement

. tabstat price wgtxdisp weight displacement, statistics(sd) save
   stats |     price  wgtxdisp    weight  displa~t
---------+----------------------------------------
      sd |  2949.496  78288.85  777.1936  91.83722
```

Not $\beta_{price} \neq \frac{\sigma_{wgtxdisp}}{\sigma_{price}} b^c_{wgtxdisp}$

```
. // sd of product, not correct
. display .0143162 *78288.85/2949.496
.37999673
```

But instead $\beta = \frac{\sigma_{weight}\sigma_{displacement}}{\sigma_{price}} b^c_{wgtxdisp}$

```
. // product of sds, correct
. display .0143162 *777.1936*91.83722/2949.496
.34643989

. restore
```

Here it is worth noting that the centered coefficient is the same as the original coefficient for the second order term, so the problem with the standard deviation is the sticking point.

Simply centering the data first does not quite get us what we want from **regress, beta**. Which leaves us back at the beginning: in order to correctly calculate the standardized coefficients of a model containing an interaction (without resorting to piecemeal computations), we standardize our data first.

4 But Do We Want to Standardize Everything?

Do we want all of the independent variables standardized? Very often the answer is: no! At its simplest this is related to the calculation of the standardized intercept that we have so far ignored. In our previous example, the standardized value of the centered intercept from **stdBeta** was calculated as

$$\beta_0 = \frac{1}{\sigma_y} b_0$$

When we have categorical variables in our models, we fundamentally have multiple intercepts, with different intercepts parameterized as coefficients on indicator variables. To treat these consistently, we want our standardization routine to ignore indicator variables, and to center and rescale only continuous variables. This is the final problem with **regress, beta**: it comes a from a time before factor variables were built into Stata, and happily treats indicator variables as if they were regression slopes. An example in an additive model is

```
. regress price weight i.foreign, beta

------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|                     Beta
-------------+----------------------------------------------------------------
      weight |   3.320737   .3958784     8.39   0.000                 .8750157
             |
     foreign |
    Foreign  |   3637.001    668.583     5.44   0.000                 .5674549
       _cons |  -4942.844   1345.591    -3.67   0.000                        .
------------------------------------------------------------------------------

. stdBeta

-----------------------------------------------------------
    Variable |   Original       Centered     Standardized
-------------+---------------------------------------------
      weight |    3.3207368      3.3207367       .8750157
             |
     foreign |
    Foreign  |    3637.0013      3637.0013      1.2330925
             |
       _cons |    -4942.844     -1081.2706     -.36659505
-----------------------------------------------------------
```

(It is worth noting that Long & Freese's **listcoef** has for many years provided Stata users with partially- (semi-) standardized coefficients for additive models, of which standardized indicators are one variety, if you know what you are looking for in the output. However, **listcoef** does not work with Stata's factor variable notation or with interaction terms, and its real point is to provide a variety of interpretive options for categorical dependent variables. See also Long 1997.)

```
. listcoef, cons

-------------------------------------------------------------------------------
       price |      b         t     P>|t|    bStdX    bStdY   bStdXY      SDofX
-------------+-----------------------------------------------------------------
      weight |   3.32074    8.388   0.0002580.8553   0.0011   0.8750   777.1936
   1.foreign |3637.00130    5.440   0.0001673.7060   1.2331   0.5675     0.4602
       _cons |-4.943e+03   -3.673   0.000
-------------------------------------------------------------------------------
```

If you go a step farther, and add a regression interaction to a model with multiple intercepts, then **regress, beta** breaks in all the previously examined ways, plus the interpretation of the coefficient of the indicator becomes more convoluted. By excluding indicator variables from the standardization, their interpretation remains as clear as in the original model: intercepts are intercepts, and regression coefficients are in fully standardized units.

```
. regress price c.weight##c.displacement i.foreign, beta
```

```
------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|                     Beta
-------------+----------------------------------------------------------------
      weight |   .7915753   .9411297     0.84   0.403                 .2085805
displacement |  -20.28166   14.07028    -1.44   0.154                -.6315016
             |
    c.weight#|
          c. |
displacement |   .0083661   .0034946     2.39   0.019                 1.282296
             |
     foreign |
    Foreign  |   3280.128   706.0526     4.65   0.000                 .5117745
       _cons |   1290.376   2625.975     0.49   0.625                        .
------------------------------------------------------------------------------
. stdBeta

-----------------------------------------------------------
    Variable |   Original       Centered     Standardized
-------------+---------------------------------------------
      weight |    .79157535      2.4421837      .64351653
displacement |    -20.28166      4.9794301      .15504244
             |
    c.weight#|
          c. |
displacement |     .0083661       .0083661      .20245245
             |
     foreign |
    Foreign  |    3280.1277      3280.1276      1.1120977
             |
       _cons |    1290.3756     -1502.3233     -.50934917
-----------------------------------------------------------

. capture noisily listcoef  // does not support factor variables
weight#c:  operator invalid
```

Note that when indicators and continuous variables are combined in interaction terms, the order of the term is defined by the number of continuous variables in the term, which is the same as the number of different standard deviations in the numerator of the standardization formula. (Interactions among indicators just produce more indicators, i.e. intercepts.)

5 CHANGE THE BASE CATEGORY

Stata’s factor variable notation makes it very easy to change the base category in a categorical variable. Since **stdBeta** is leaves categorical variables alone, rerunning estimation commands on transformed continuous variables is no problem. In the following example it is worth noticing that changing the base category leaves the regression coefficients unchanged.

```
. regress price c.weight##c.displacement ib1.foreign
. stdBeta
```

6 Force Indicators to be Standardized

If you really prefer your indictors to be standardized too, but would like to use the correct centering and rescaling for interaction terms (**beta** will be wrong), it is simply a matter of specifying the indicator variable as a continuous variable, making use of Stata's factor variable notation. Note that in the case of a categorical variable with more than two categories, you will need to generate your own set of indicators, just like in the good old days!

```
. regress price c.weight##c.displacement foreign
. stdBeta
```

7 Polynomials

Polynomial terms may be given the same treatment: first center, then rescale, then form the product terms. This is done by specifying higher order terms as though they were interactions, using Stata's model notation.

```
. regress price c.weight##c.weight
. stdBeta
```

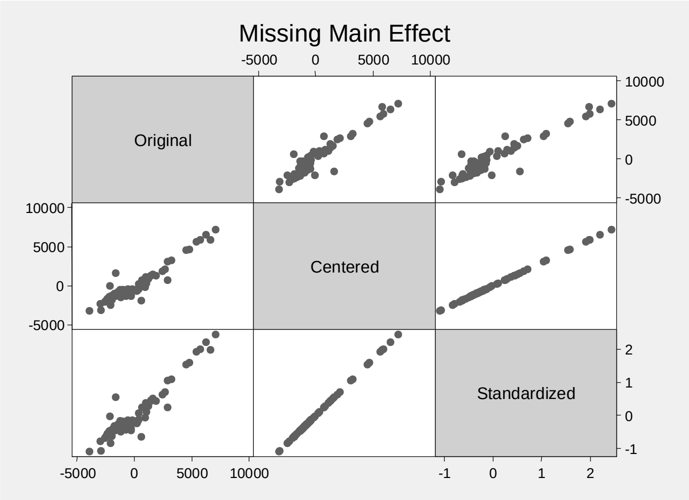
8 Skipping Lower Order Terms

While the approach of transforming the data and then re-estimating a model is very robust, it is in fact possible to produce nonsense results with no warning.

If you were to specify a model that included higher-order terms without the related lower-order terms, the refit models would not be equivalent. That is, if you specify the main effect to be zero while allowing the related interaction to be freely estimated, the similar-looking centered and standardized models would not be equivalent to the original model. A little algebra shows that these models are only equivalent when either the higher order effects are all zero or the data are already centered. **stdBeta** gives no warning that this is a problem, which is why we point it out here.

```
. regress price c.weight c.weight#c.displacement
. stdBeta // this produces nonsensical results
```

One diagnostic that there is a problem would be to look at the residuals from all three models and notice that a plot of the residuals from the original model versus the residuals from either transformed model do not form a line (as they should).



9 DON'T STANDARDIZE Y

Sometimes analysts prefer to standardize just the independent variables, leaving the dependent variable in its original units. **stdBeta** does this with the **nodepvar** option.

```
. regress price c.weight##c.displacement i.foreign
. stdBeta, nodepvar
```

10GENERALIZED LINEAR MODELS

In generalized linear models, the independent variables may be treated in a manner similar to the general linear model. However, the link function and transformations of the coefficients in their linear form pose hurdles that **stdBeta** is not designed to address. **stdBeta** can give you exponentiated coefficients (by passing the **exp** option to **estimates table**), and can leave the dependent variable unstandardized. "Fully standardized" coefficients, e.g. standardizing the latent dependent variable in a logistic or poisson model, is beyond the scope of this little program (see Long 1997). **stdBeta** can also store the results of the centered and standardized models via the **store** option (using **estimates store**).

11REFERENCES

Aiken, L.S., and S.G. West. 1991. *Multiple Regression: Testing and Interpreting Interactions*. Newbury Park: SAGE Publications.

Friedrich, R.I. 1982. In defense of multiplicative terms in multiple regression equations. *American Journal of Political Science*, 26: 797-833.

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