# MA397 – Applied Regression Modeling
# Regression Involving Interaction and Quadratic Terms, the Nested-F-test and Variable Coding in Stata

## Goals

We will see how to fit a broad range of models by generating appropriate independent variables. We also see how to utilize character-coded qualitative variables, and how to test nested models.

## Data

For the first part of this exercise we will be using the *lung.dta* dataset found on the course webpage at http://www.colby.edu/personal/l/lobrien/ma397.html. Once you open the data you should see that there are two variables. The variable "volume" gives lung capacity in liters, while "method" gives the method used to measure lung function. Method A indicates lung function was measured while the subject was laying down, method B while sitting upright, and method C while standing. Note that the "method" variable in coded using the characters "A", "B", and "C".

## Running a Regression with Qualitative Independent Variables

Stata has a built-in command for automatically generating the appropriate number of indicator (or binary) variables. The command **xi** when used in conjunction with the **regress** command will generate dummy variables for the *k – 1* levels of the independent variable that the "*i.*" notation is used in front of. This is best done using the command line. In this example, type "*xi: regress volume i.method*". You will see that Stata created two new variables:

- *_Imethod_2* is an indicator variable that equals when if the observation is from method 2.
- *_Imethod_3* is an indicator variable that equals when if the observation is from method 3.

```
. xi:  regress volume i.method
i.method          _Imethod_1-3        (_Imethod_1 for method==A omitted)

      Source |       SS       df       MS              Number of obs =      18
-------------+------------------------------           F(  2,    15) =    2.69
       Model |   1.0811112     2  .540555601           Prob > F      =  0.1004
    Residual |       3.015    15        .201           R-squared     =  0.2639
-------------+------------------------------           Adj R-squared =  0.1658
       Total |  4.09611121    17  .240947718           Root MSE      =  .44833

------------------------------------------------------------------------------
      volume |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
  _Imethod_2 |   .2833333   .2588436     1.09   0.291    -.2683787    .8350453
  _Imethod_3 |         .6   .2588436     2.32   0.035      .048288    1.151712
       _cons |   2.933333   .1830301    16.03   0.000     2.543214    3.323453
------------------------------------------------------------------------------
```

In this case, the F-test for the regression is equivalent to the ANOVA F-test (the means do not significantly differ at the 0.05 level).

**Encoding Character Data into Numeric Data**

Stata handles character data fairly well. However, many procedures and programs do not handle character data at all. If you have a character or a string that uniquely identifies groups, you can use the Stata command, *encode oldvar, generate(newvar)* to generate the variable "newvar" that contains numeric data that has been labeled to match the character data in "oldvar."

- In this case, encode the "method" variable by generating a new variable called "method2". Open the data browser and verify what you have done is correct.
- Also give labels to the variable names themselves to make the data more user friendly.

**Generating Coding Variables in Stata**

There are many other coding schemes you can use to define your groups when using regression to look for mean differences. You need to generate *k – 1* variables to define *k* groups. In this case, you should define two variables, *x1* and *x2* that serve to compare lying and sitting measures with standing measures, and to compare lying and standing measures together (ignoring sitting), respectively.

To o this, you first must use the *generate* command in Stata to define each variable. Then you may use the *recode* or *replace* commands to alter them.

In this case, define *x1* and *x2* by issuing the following commands:

- generate x1=2
- generate x2= -1

Now you can issue the following to finish the coding. Note that there are other possibilities as well and that this is just one example:

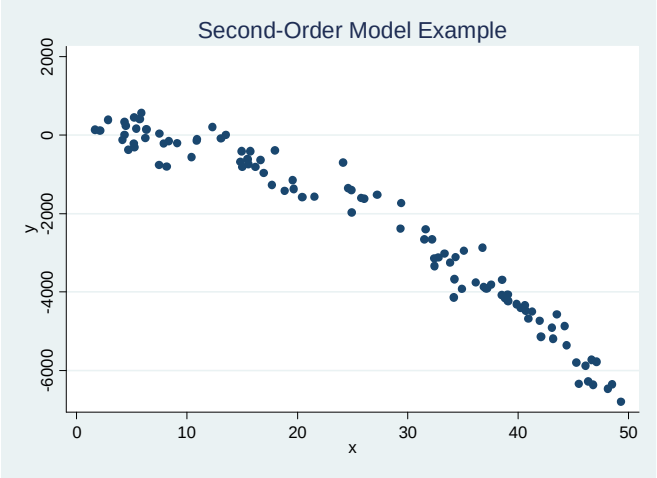
- replace x1 = -1 if method2==1 | method2==2
- replace x2 = 1 if method2==3
- recode x2 –1 = 0 if method2==2

The *recode* command allows you to systematically change one value of a variable with another given a condition specified after the *if* statement. This is particularly useful when you have coded a variable, say 1 and 2 when you wanted it to be 0 and 1. Note that in Stata OR is denoted by | and AND is denoted by &. The double equals (==) tests for equality and != tests for something NOT equal to the specified value.

We need to be especially careful of missing values as well. Stata codes missing values as "." But actually assigns them a really really small value. So if we were missing a value for the independent variable, we would need to issue command such as "replace x1=. if method==.".

**Regression Using Second-Order Models**

In general, you will not have higher-order terms in your data set. Consider the file *quadex.dta* on the course webpage. It contains artificial data for a response, Y, and independent variable, X. The plot of the data is below.



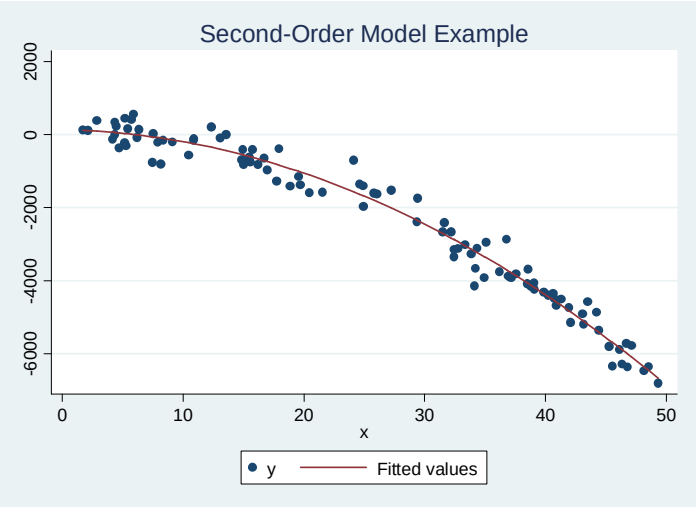

It is clear that the best fit line will contain a quadratic term for X. Since we do not have one, we need to generate one. Use the *generate* command for this by typing, "*generate xsq = x^2*". Use the *regress* command as you normally would (*regress y x xsq)*to obtain the following output:

```
. regress y x xsq

      Source |       SS       df       MS              Number of obs =     100
-------------+------------------------------           F(  2,    97) = 1962.15
       Model |   445397130     2   222698565           Prob > F      =  0.0000
    Residual |  11009247.9    97  113497.401           R-squared     =  0.9759
-------------+------------------------------           Adj R-squared =  0.9754
       Total |   456406378    99  4610165.44           Root MSE      =  336.89

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           x |  -5.540437   10.46796    -0.53   0.598    -26.31643    15.23556
         xsq |  -2.684278   .2044015   -13.13   0.000    -3.089958   -2.278597
       _cons |   130.0978   107.0433     1.22   0.227    -82.35355    342.5491
------------------------------------------------------------------------------
```

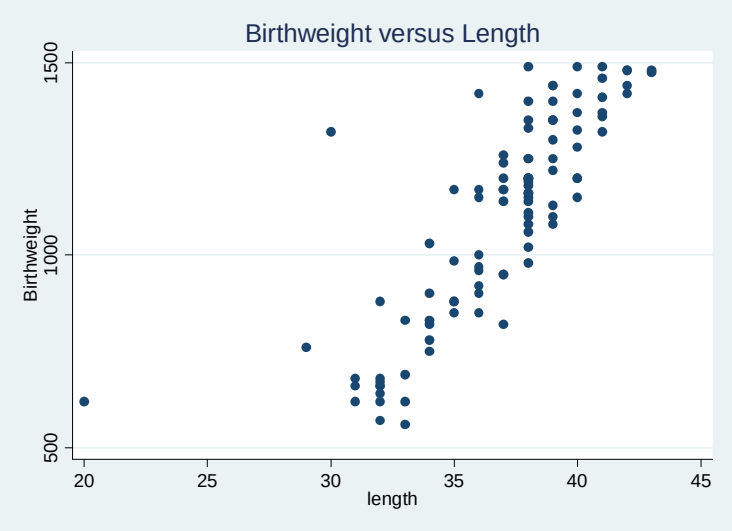
We can see that the quadratic term is significantly different from 0 (at the 0.05 level), and the fitted line is overlaid in the plot below.




**Generating Interactive Models**

When considering interactive terms, the same general procedure as above is often used: manually generate a new term using the *generate* command.

Let us consider the *birthwt.dta* data again. Rather than relate birthweight to head circumference, let us relate birthweight to length. The data are displayed below:



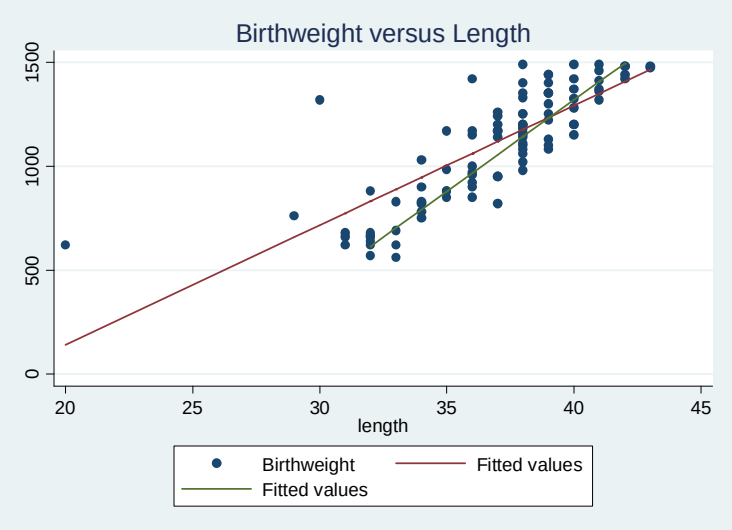

We will consider whether or not toxemia modifies the relationship between birthweight and length (which is significantly positive). We first need to generate this interaction term, "*generate lentox = length * toxemia*". We then run the regression, "*regress birthwt length toxemia lentox*". The output is below.

```
. regress birthwt length toxemia lentox

      Source |       SS       df       MS              Number of obs =     100
-------------+------------------------------           F(  3,    96) =   72.08
       Model |  4997922.76     3  1665974.25           Prob > F      =  0.0000
    Residual |  2218819.99    96  23112.7083           R-squared     =  0.6925
-------------+------------------------------           Adj R-squared =  0.6829
       Total |  7216742.75    99  72896.3914           Root MSE      =  152.03

------------------------------------------------------------------------------
     birthwt |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      length |   57.47765   4.690349    12.25   0.000     48.16738    66.78792
     toxemia |  -1192.974   443.1596    -2.69   0.008    -2072.639   -313.3094
      lentox |   30.50423    11.7999     2.59   0.011     7.081609    53.92686
       _cons |  -1007.631   172.6113    -5.84   0.000    -1350.262   -665.0001
------------------------------------------------------------------------------
```

We see that toxemia significantly modifies the association between length and birthweight. We can generate separate regression lines by first generating the predicted values (*predict yhat*). Then generate the observed data plot as before (using *length* as the independent variable this time). Generate a line plot with the predicted (*yhat*) values but click on the *if/in* tab and enter *toxemia==0* to limit the first line to infants whose mothers do not have toxemia. Generate a second line plot with the *yhat* values, but now on the *if/in* tab enter toxemia == 1. You should get the following:




This smaller line is for infants whose mothers have toxemia (this could be verified by checking the minimum value for length in each subgroup).

Note that when using the *xi* command, you can automatically generate interaction variables with the dummy variables. For example, "*xi: regress y x1 i.x2 i.x2*x1*" will generate dummy variables for the levels of *x2* as well as all interactions between those dummy variables and *x1*.

### Performing the Nested Model F-test in Stata

We learned that we can check "chunks" of independent variables all at once by using the nested model F-test (provided the "reduced" model is a simplification of the "complete" model). To see how this is done in Stata, consider a model with length, toxemia, length*toxemia, length^2 and mother's age as the "full" model. We will need to generate the length^2 term (*generate sqlength=length^2*). The output is below:

```
. regress birthwt length toxemia lentox sqlength   momage

      Source |       SS       df       MS              Number of obs =     100
-------------+------------------------------           F(  5,    94) =   52.18
       Model |  5305224.82     5  1061044.96           Prob > F      =  0.0000
    Residual |  1911517.93    94  20335.2971           R-squared     =  0.7351
-------------+------------------------------           Adj R-squared =  0.7210
       Total |  7216742.75    99  72896.3914           Root MSE      =   142.6

------------------------------------------------------------------------------
     birthwt |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      length |  -98.76791   40.76145    -2.42   0.017    -179.7007   -17.83509
     toxemia |  -705.9804   434.9478    -1.62   0.108     -1569.58    157.6187
      lentox |   17.47173   11.59203     1.51   0.135    -5.544519    40.48797
    sqlength |   2.290847   .5927709     3.86   0.000     1.113887    3.467808
      momage |  -.8577658    2.46751    -0.35   0.729    -5.757064    4.041533
       _cons |   1635.016   699.6966     2.34   0.022     245.7524     3024.28
------------------------------------------------------------------------------
```

If we want to test whether the coefficients for length^2, momage, and lentox are simultaneously equal to zero, we can use a nested F-test. The Stata command *nestreg* will perform this test automatically. You group the variables you want tested together inside parentheses in the *regress* command after typing *nestreg:*. For this example type, "*nestreg: regress birthwt (length toxemia) (lentox sqlength momage)*". The output is given below.

```
. nestreg: regress birthwt (length toxemia) (lentox sqlength momage)

Block  1: length toxemia

      Source |       SS       df       MS              Number of obs =     100
-------------+------------------------------           F(  2,    97) =   98.98
       Model |  4843463.45     2  2421731.72           Prob > F      =  0.0000
    Residual |   2373279.3    97   24466.797           R-squared     =  0.6711
-------------+------------------------------           Adj R-squared =  0.6644
       Total |  7216742.75    99  72896.3914           Root MSE      =  156.42

------------------------------------------------------------------------------
     birthwt |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      length |   62.29728   4.428174    14.07   0.000     53.50858    71.08598
     toxemia |  -51.47083   38.63331    -1.33   0.186    -128.1473     25.2056
       _cons |  -1184.127    163.113    -7.26   0.000    -1507.861   -860.3927
------------------------------------------------------------------------------
```

```
Block  2: lentox sqlength momage

      Source |       SS       df       MS              Number of obs =     100
-------------+------------------------------           F(  5,    94) =   52.18
       Model |  5305224.82     5  1061044.96           Prob > F      =  0.0000
    Residual |  1911517.93    94  20335.2971           R-squared     =  0.7351
-------------+------------------------------           Adj R-squared =  0.7210
       Total |  7216742.75    99  72896.3914           Root MSE      =   142.6

------------------------------------------------------------------------------
     birthwt |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      length |  -98.76791   40.76145    -2.42   0.017    -179.7007   -17.83509
     toxemia |  -705.9804   434.9478    -1.62   0.108     -1569.58    157.6187
      lentox |   17.47173   11.59203     1.51   0.135    -5.544519    40.48797
    sqlength |   2.290847   .5927709     3.86   0.000     1.113887    3.467808
      momage |  -.8577658    2.46751    -0.35   0.729    -5.757064    4.041533
       _cons |   1635.016   699.6966     2.34   0.022     245.7524     3024.28
------------------------------------------------------------------------------
```

| Block | Block F | Block df | Residual df | Pr > F | R2 | Change in R2 |
|---|---|---|---|---|---|---|
| 1 | 98.98 | 2 | 97 | 0.0000 | 0.6711 | |
| 2 | 7.57 | 3 | 94 | 0.0001 | 0.7351 | 0.0640 |

The first block is the overall ANOVA F-test for the reduced model. The second block tests out three added variables together and leads us to reject the null hypothesis at the 0.05 level. This indicates that the complete model is necessary and that those three independent variables add significantly to the model.