

Dynamic Markdown documents in Stata 15: An initial look

First I open the example data nlsw88 and run the describe the command.

The tags around the code «**dd_do**» tells Stata to execute this block of code and include its output in the document. «**/dd_do**» indicates the end of a «do_do».

The four tilde (~) symbols indicate that this block should appear as a block of code in the HTML document (i.e. they will be inset from the text).

Here I have run the commands twice. The first time I add the tag «**dd_ignore**» which tells Stata not to run this command but print all the code in the document. This just lets you see what the whole block of code looks like in the plain text file.

The second time I do not include the «**dd_ignore**» tag so you see the output and code as you would see it in Stata.

```
<<dd_do>>
sysuse nlsw88, clear
describe
<</dd_do>>

. sysuse nlsw88, clear
(NLSW, 1988 extract)

. describe

Contains data from C:\Program Files (x86)\Stata15\ado\base\n\nls
> w88.dta
  obs:           2,246                NLSW, 1988
                                     extract
vars:             17                1 May 2016 22:52
size:            60,642              (_dta has notes)
-----
      storage   display   value
variable name  type     format   label      variable label
-----
idcode         int       %8.0g   NLS id
age            byte     %8.0g   age in current
                                     year
race           byte     %8.0g   racelbl   race
married        byte     %8.0g   marlbl   married
never_married  byte     %8.0g   never_married
grade          byte     %8.0g   current grade
                                     completed
collgrad       byte     %16.0g  gradlbl   college graduate
south          byte     %8.0g   lives in south
smsa           byte     %9.0g   smsalbl   lives in SMSA
c_city         byte     %8.0g   lives in central
                                     city
industry       byte     %23.0g  indlbl   industry
occupation     byte     %22.0g  occlbl   occupation
union          byte     %8.0g   unionlbl  union worker
wage           float    %9.0g   hourly wage
hours          byte     %8.0g   usual hours worked
ttl_exp        float    %9.0g   total work
                                     experience
tenure         float    %9.0g   job tenure (years)
-----
Sorted by: idcode
```

Now, I summarize the data.

```
<<dd_do>>
summ
<</dd_do>>

. summ

      Variable |           Obs       Mean   Std. Dev.   Min
>      Max
-----+-----
> -----
>      idcode |           2,246   2612.654   1480.864     1
>      5159
>      age |           2,246    39.15316    3.060002    34
>      46
>      race |           2,246    1.282725    .4754413     1
>      3
>      married |           2,246    .6420303    .4795099     0
>      1
never_marr~d |           2,246    .1041852    .3055687     0
>      1
-----+-----
> -----
>      grade |           2,244    13.09893    2.521246     0
>      18
>      collgrad |           2,246    .2368655    .4252538     0
>      1
>      south |           2,246    .4194123    .4935728     0
>      1
>      smsa |           2,246    .7039181    .4566292     0
>      1
>      c_city |           2,246    .2916296    .4546139     0
>      1
-----+-----
> -----
>      industry |           2,232     8.189516    3.010875     1
>      12
>      occupation |           2,237     4.642825    3.408897     1
>      13
>      union |           1,878    .2454739    .4304825     0
>      1
>      wage |           2,246     7.766949    5.755523    1.004952    4
> 0.74659
>      hours |           2,242    37.21811    10.50914     1
>      80
-----+-----
> -----
>      ttl_exp |           2,246    12.53498    4.610208    .1153846    2
> 8.88461
>      tenure |           2,231     5.97785    5.510331     0    2
> 5.91667
```

We can use numbers from the Stata output in our text in the HTML document. We always want to avoid cutting and pasting numbers. We can use the **dd_display** tag to retrieve stored results from the summ command and use them in our text.

First we run the summ command for our variable of interest. We have run this above so we don't need this to appear in the HTML

document. Therefore we can use the `<<dd_do:quietly>>` tag so Stata does the work but doesn't tell us about it.

```
<<dd_do: quietly>>
summ age
<</dd_do>>
```

Then we use the stored values for minimum and maximum age in the text.

```
> The variable age has minimum value <<dd_display: %4.2f `r(min)'\>>
and has maximum value <<dd_display: %4.2f `r(max)'\>>.

> The variable age has minimum value 34.00
and has maximum value 46.00.
```

Now lets see if this will work for something a little more complex. Here I have run a regression, the outcome variable is wage, I have run this quietly as we might not want to show all the output Stata gives in the standard regrssion output.

We store these estimates as "m1".

```
<<dd_do: quietly>>
regress wage age married i.race, allbaselevels
est sto m1
<</dd_do>>
```

No lets present the output of the regression in the HTML document.

```
<<dd_do>>
esttab m1, b se wide
<</dd_do>>

. esttab m1, b se wide

-----
                (1)
                wage
-----
age                -0.0839*      (0.0395)
married            -0.782**      (0.258)
1.race              0              (.)
2.race             -1.459***      (0.283)
3.race              0.463         (1.130)
_cons              11.93***      (1.575)
-----
N                  2246
-----

Standard errors in parentheses
* p<0.05, ** p<0.01, *** p<0.001
```

We can also use the stored results to say a little bit more about this regression model.

```
> The sample size was <<dd_display: %4.2f `e(N)'\>>
and the model has an adjusted R2 value of <<dd_display: %4.2f `e(r2_a)'\>>.

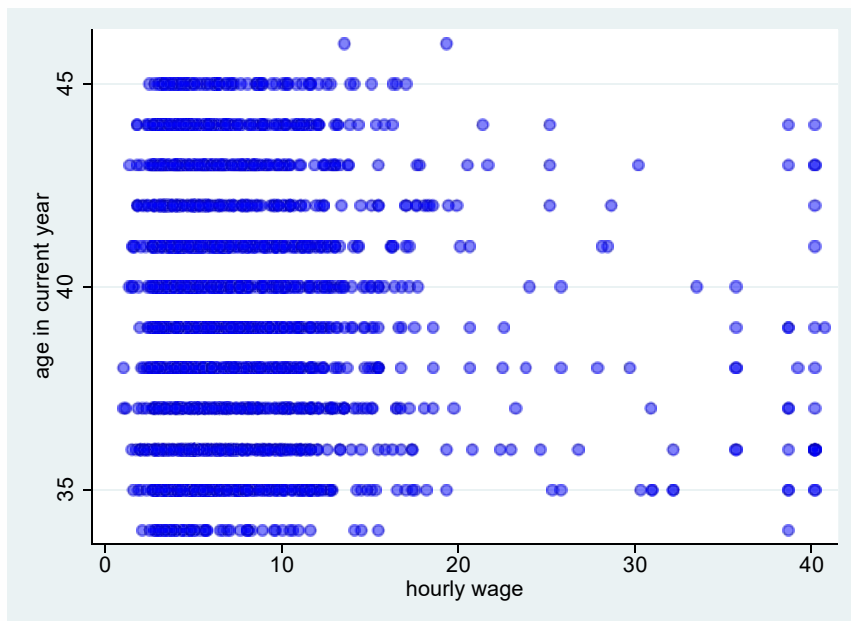
> The sample size was 2246.00
and the model has an adjusted R2 value of 0.01.
```

Now lets do a graph.

Here is a scatterplot of age and wage.

```
<<dd_do:nooutput>>  
scatter age wage, mcolor(blue%50)  
<</dd_do>>  
  
. scatter age wage, mcolor(blue%50)  
  
<<dd_graph: sav("graph1.svg") alt("scatter age wage") replace height(400)>>
```

Here is the graph:



end dyndoc blogexample.do