How can I use -estout- to make regression tables that look like those in journal articles?

Authored by

stats writer

July 1, 2024

To use the -estout- package in Stata to create regression tables resembling those found in journal articles, you can follow these steps. First, install the -estout- package using the "ssc install estout" command. Then, run your regression model and save the results using the -estimates- command. Next, use the -estout- command to create the table, specifying the desired results and formatting options. You can further customize the table and save it in a desired format. For more detailed instructions and examples, refer to the -estout- package help file or online resources.

How can I use -estout- to make regression tables that look like those in journal articles? | Stata FAQ

This FAQ illustrates the estout command that makes regression tables in a format that is commonly used in journal articles. The estout command was written by Ben Jann of ETH Zurich. You can download estout from within Stata by typing search estout (see How can I use the search command to search for programs and get additional help? for more information about using search).

Let's illustrate use of the estout command using the high school and beyond data file.

```
use https://stats.idre.ucla.edu/stat/stata/notes/hsb2, clear
```

(highschool and beyond (200 cases))

**We will run 3 regression models predicting
the variable read. The first model will predict from the variables
female and write; the second model will predict from female, write
and math;
and the third model will predict from female, write, math, science
and socst. After each regress we will run an estimates store
command.
We will then use estout to create a single table that will summarize
these models side by side.**

```
regress read female write

      Source |       SS       df       MS      Number of obs =     200
-------------+------------------------------   F(  2,   197) =   66.11
       Model |  8401.94189     2  4200.97094   Prob > F      =  0.0000
    Residual |  12517.4781   197  63.540498   R-squared     =  0.4016
-------------+------------------------------   Adj R-squared =  0.3956
       Total |   20919.42   199  105.122714   Root MSE      =  7.9712
```

---

read | Coef. Std. Err. t P>|t|

---

female | -4.532084 1.171072 -3.87 0.000 -6.84153 -2.222637

write | .7067537 .0616783 11.46 0.000 .5851192 .8283882

_cons | 17.40106 3.202315 5.43 0.000 11.08584 23.71628

---

estimates store m1, title(Model 1)

regress read female write math

Source | SS df MS Number of obs = 200

---+--- F(3, 196) = 68.34

Model | 10695.1896 3 3565.06321 Prob > F = 0.0000

Residual | 10224.2304 196 52.1644406 R-squared = 0.5113

---+--- Adj R-squared = 0.5038

Total | 20919.42 199 105.122714 Root MSE = 7.2225

---

read | Coef. Std. Err. t P>|t|

---

female | -2.739657 1.09497 -2.50 0.013 -4.899092 -.5802214

write | .3924361 .0732832 5.36 0.000 .2479114 .5369609
math | .4753659 .0716952 6.63 0.000 .3339729 .6167589
_cons | 7.986659 3.230313 2.47 0.014 1.616025 14.35729
----

estimates store m2, title(Model 2)

regress read female write math science socst

Source | SS df MS Number of obs = 200
-------------+---------------------------------- F(5, 194) = 56.29
Model | 12383.6535 5 2476.7307 Prob > F = 0.0000
Residual | 8535.76652 194 43.9987965 R-squared = 0.5920
-------------+---------------------------------- Adj R-squared = 0.5815
Total | 20919.42 199 105.122714 Root MSE = 6.6332

----
read | Coef. Std. Err. t P>|t|
-------------+--
female | -1.328513 1.046793 -1.27 0.206 -3.393068 .7360424
write | .1503085 .0778554 1.93 0.055 -.0032431 .3038601
math | .2934723 .0728471 4.03 0.000 .1497983 .4371463
science | .2508791 .0667312 3.76 0.000 .1192673 .382491

socst | .2694578 .0574134 4.69 0.000 .1562232 .3826923
_cons | 2.44264 3.107255 0.79 0.433 -3.685698 8.570977
---

estimates store m3, title(Model 3)

estout m1 m2 m3

---
m1 m2 m3
b b b
---
female -4.532084 -2.739657 -1.328513
write .7067537 .3924361 .1503085
math .4753659 .2934723
science .2508791
socst .2694578
_cons 17.40106 7.986659 2.44264

Now we have a perfectly fine table that just includes the regression coefficients. We will
modify the estout command to add standard errors and stars for statistical significance.
We will also format the output so that coefficients will

have three decimal places and the
standard errors to two decimal places. Note, the par option for "se" places
parentheses around the standard error.

estout m1 m2 m3, cells(b(star fmt(3)) se(par fmt(2)))

--
m1 m2 m3
b/se b/se b/se
--
female -4.532* -2.740* -1.329**
(1.17) (1.09) (1.05)
write 0.707* 0.392*** 0.150**
(0.06) (0.07) (0.08)
math 0.475* 0.293*****
(0.07) (0.07)
science 0.251***
(0.07)
socst 0.269***
(0.06)
_cons 17.401* 7.987* 2.443**
(3.20) (3.23) (3.11)
--

The table is better now, but it can be improved further by putting the model names above the columns, adding a legend and by changing the label for "_cons" to "constant."

```
estout m1 m2 m3, cells(b(star fmt(3)) se(par fmt(2))) ///
legend label varlabels(_cons Constant)

------------------------------------------------------------
Model 1 Model 2 Model 3
b/se b/se b/se
------------------------------------------------------------
female -4.532*** -2.740* -1.329
(1.17) (1.09) (1.05)
writing score 0.707*** 0.392*** 0.150
(0.06) (0.07) (0.08)
math score 0.475*** 0.293***
(0.07) (0.07)
science score 0.251***
(0.07)
social studies score 0.269***
(0.06)
constant 17.401*** 7.987* 2.443
(3.20) (3.23) (3.11)
```

---

* p

Next, we want to add some things to the table, like R-squared, residual degrees of
freedom and BIC. Stata has special names for each of these ancillary statistics,
"r2" is the name for R-squared, "df_r" for residual degrees of freedom and "bic"
for the BIC. You can get the names of these items from the ereturn list and
from the help file.

```
estout m1 m2 m3, cells(b(star fmt(3)) se(par fmt(2))) ///
legend label varlabels(_cons constant) ///
stats(r2 df_r bic)
```

Model 1 Model 2 Model 3
b/se b/se b/se

---

female -4.532*** -2.740* -1.329
(1.17) (1.09) (1.05)
writing score 0.707*** 0.392*** 0.150

(0.06) (0.07) (0.08)
math score 0.475*** 0.293***
(0.07) (0.07)
science score 0.251***
(0.07)
social studies score 0.269***
(0.06)
constant 17.401*** 7.987* 2.443
(3.20) (3.23) (3.11)
---
r2 0.402 0.511 0.592
df_r 197.000 196.000 194.000
bic 1410.783 1375.608 1350.106
---
* p

Okay, we're almost done. We just need to clean up the lower part of the table giving
each of the items a better label and adjusting the number of decimal places for each of
the items.

```
estout m1 m2 m3, cells(b(star fmt(3)) se(par fmt(2))) ///
legend label varlabels(_cons constant) ///
```

```
stats(r2 df_r bic, fmt(3 0 1) label(R-sqr dfres BIC))

------------------------------------------------------------
Model 1 Model 2 Model 3
b/se b/se b/se
------------------------------------------------------------
female -4.532*** -2.740* -1.329
(1.17) (1.09) (1.05)
writing score 0.707*** 0.392*** 0.150
(0.06) (0.07) (0.08)
math score 0.475*** 0.293***
(0.07) (0.07)
science score 0.251***
(0.07)
social studies score 0.269***
(0.06)
constant 17.401*** 7.987* 2.443
(3.20) (3.23) (3.11)
------------------------------------------------------------
R-sqr 0.402 0.511 0.592
dfres 197 196 194
BIC 1410.8 1375.6 1350.1
------------------------------------------------------------
* p
```

**We now have a table that's acceptable for publication in many journals. Of course,
each periodical defines its own formats. Fortunately, estout is very flexible
and has many options that will adapt to almost any periodical's requirements.**