From estimation output to document tables:
A long way made short

Ben Jann, ETH Zürich, jann@soz.gess.ethz.ch

North American Stata Users Group Meeting
August 14, 2007, Boston

Outline
• Introduction
• A primer on estout
• Easy-peasy: eststo and esttab
• Advanced examples
• Discussion

Introduction

• Output from statistical estimation routines contains all sorts of details and it is often sensible to display only selected and rearranged results in form of "regression tables".

Regression tables are useful
  - to get an overview of results when analyzing data,
  - for results presentation in reports and publications.

• Various user commands have been around to compile regression tables (and export them to other formats). Examples are:
  - outreg (John Luke Gallup)
  - outtex (Antoine Terracol)
  - est2tex (Marc Muendler)
  - mktab (Nicholas Winter)
  - parmest (Roger Newson)
Introduction

- Stata Corp then came out with their own tool called estimates table in Stata 8.

estimates table was a relief because it was so much more straight forward to use than the other packages. Especially the concept of first storing a bunch of models and not worry about tabulation until later, appealed to me.

However, estimates table does have some serious limitations:

- It is only intended for displaying the models in the Results window or the log.

- It has only very limited functionality in terms of formatting. For example, is not possible to place standard errors or t-statistics in parentheses, which makes the tables very unreadable.

Introduction

- I liked estimates table, but it was not flexible enough, so I started working on a wrapper that runs estimates table, grabs the results (an R-matrix containing point estimates and variances), puts together an improved table, and exports it to a tab-delimited or LaTeX-formatted file (that's what I needed at that time; this was around June 2004).

This first estout version is still available from ssc as estout1.

- estout was easy to use but it had the limitation that only point estimates and one of t-stats, standard errors, p-values or CIs could be printed. So I came up with the cells option, allowing to freely specify and arrange the contents of the table.

Furthermore, this guy came along, Ian Watson, and pushed me to implement all sorts of LaTeX related things. The number of options grew and grew ...

(July-August 2004)
Introduction

- **estout** then remained pretty much as it was until today, with some new options every now and then. One notable event was the addition of **estadd**, a tool to compute an add extra statistics to a model's e()-results, in November 2004.

- Many people were using **estout** and I had lots of positive responses. However, people kept complaining that **estout** was too complicated to use.

The motivational orientation of **estout** towards functionality rather than ease-of-use and towards exporting tables for use with LaTeX brought with it some limitations:

- **estout** tables usually do not display well in Stata's results window.

- **estout**'s syntax is not as intuitive and user-friendly as it could be.

- The amount of typing required to compile even a simple table can be considerable. (There are "defaults files" to pre-specify options, but I think hardly anyone uses them.)

Introduction

- An additional issue with **estout** is that the estimation sets have to be stored using official Stata's **estimates store** before they can be tabulated.

Drawbacks of **estimates store** are:

- The user is required to specify names under which to store the estimation sets. This can be distracting.

- The stored estimates consume a considerable amount of memory. (In order to preserve functionality of postestimation commands, an estimation sample indicator variable is stored for each estimation set. These indicators may greatly enlarge the dataset if it contains a many of observations or if many estimation sets are stored. Additionally, storing the estimation samples has the side effect of slowing down cycling through the stored sets, which also slows down tabulation programs such as **estout** or official Stata's **estimates table**.)
Introduction

- To summarize, there seemed to be a need for
  
  (1) an easy-to-use version of `estout`,
  
  (2) a simplified procedure to hold on to estimates for tabulation.

- So I started working on wrappers for `estimates store` and `estout`, which first appeared on SSC in September 2006 as `esto` and `esta`. It turned out that names were not chosen very well. They were changed to `eststo` and `esttab` (May 2007).

Package overview

`estout`: Generic program to compile a table of coefficients, significance stars, summary statistics, standard errors, t- or z-statistics, p-values, confidence intervals, or other statistics for one or more models previously fitted and stored.

`esttab`: User-friendly command to produce publication-style regression tables that display nicely in Stata’s results window or, optionally, are exported to various formats such as CSV, RTF, HTML, or LaTeX.

`eststo`: Utility to store estimation results for later tabulation, as an alternative to official `estimates store`. The main advantages of `eststo` is that no name has to be provided for the stored estimation set and that it can be used as a prefix command.

`estadd`: Program to add extra results (such as e.g., beta coefficients) to the returns of an estimation command. This is useful to make the the results available for tabulation by `esttab` or `estout`. 
Basic usage of estout

Syntax:

estout [ name(list) ] [ using filename ] [ , cells(all) stats(scalar(list)) style(style) more_options ]

where the amount of more_options is considerable and many options also have suboptions (similar to graph’s syntax).

The basic procedure is to first estimate and store a bunch of models and then apply estout to tabulate them:

sysuse auto

regress price weight mpg
estimates store m1

regress price weight mpg foreign
estimates store m2

estout m1 m2 using example.txt
<run>

|       | Coef.  | Std. Err. | t     | P>|t|   | [95% Conf.] |
|-------|--------|-----------|-------|-------|-------------|
| weight| 3.464706 | .630749  | 5.49  | 0.000 | 2.206717    |
| mpg   | 21.8536  | 74.22114 | 0.29  | 0.769 | -126.1758   |
| foreign| 3673.06  | 683.9783 | 5.37  | 0.000 | 2308.909    |
| _cons | -5853.696 | 3376.987 | -1.73 | 0.087 | -12588.88    |

estimates store m2

estout m1 m2 using example.txt

m1     m2
b      b
weight 1.746559  3.464706
mpg   -49.51222  21.8536
foreign |3673.06   |1946.069   | -5853.696 |

end of do-file

more
The `style()` option

The table looks messy in Stata’s results window or the Stata log because the columns are tab-separated (note that tab characters are not preserved in the results window or the log). However, the stored example.txt would look tidy if it were opened in a spreadsheet program.

To compile a table that looks good in Stata’s results window, the `style()` option can be used. It provides a style called smcl (Stata Markup and Control Language):

```stata
estout m1 m2, style(smcl)
<run>
```

Other predefined styles are `tab` (the default), `fixed`, `tex`, and `html`, and it is also possible to define one’s own styles. The `tex` style, for example, modifies the output table for use with LaTeX’s `tabular` environment:

```stata
estout m1 m2, style(tex)
<run>
```

```
foreign  3673.06
_cons    1946.066  -5853.696

. end of do-file
. do_example do/0b.ihlp
   . estout m1 m2, style(smcl)

       m1          m2
      b          b

     weight   1.746559   3.464706
      mpg   -49.51222   21.8536
    foreign   3673.06
    _cons    1946.066  -5853.696

. end of do-file
```

```stata
---more---
```
The `cells()` option

The `cells()` option is used to determine the primary contents of the table and its arrangement. For example, to report point estimates and standard errors, type:

```
estout m1 m2, cells(b se) style(smcl)
```

Other examples:

```
estout m2, cells("b se t p") style(smcl)
```
```
estout m1 m2, cells("b p" se) style(smcl)
```

Formatting is done via suboptions within `cells()` (this is the part where most people get lost):

```
estout m1 m2, cells(b(star fmt(3)) t(par fmt(2))) style(smcl)
```
end of do-file

. do_example do/0e.ihlp

    estout m1 m2, cells(b se) style(smcl)

    +-----------------+-----------------+
    |      m1         |      m2         |
    | b    se         | b    se         |
    +-----------------+-----------------+
    | weight          | 1.746559 3.464706 |
    |                 | .6413538 .630749  |
    | mpg             | -49.51222 21.8536 |
    |                 | 86.15604 74.22114 |
    | foreign         | 3673.06 683.9783 |
    | _cons           | 1946.069 -5853.696 |
    |                 | 3597.05 3376.987 |
    +-----------------+-----------------+

end of do-file

---more---

. do_example do/0f.ihlp

    estout m2, cells("b se t p") style(smcl)

    +-----------------+-----------------+-----------------+-----------------+
    |      m2         |        b        |         se       |         t        |         p        |
    +-----------------+-----------------+-----------------+-----------------+-----------------+
    | weight          | 3.464706       | .630749         | 5.493003       | 5.99e-07       |
    | mpg             | 21.8536        | 74.22114        | .2944391       | .7692938       |
    | foreign         | 3673.06        | 683.9783        | 5.370142       | 9.72e-07       |
    | _cons           | -5853.696      | 3376.987        | -1.733408      | .0874262       |
    +-----------------+-----------------+-----------------+-----------------+-----------------+
end of do-file

. do_example do/0g.ihlp
   . estout m1 m2, cells("b p" se) style(smcl)

<table>
<thead>
<tr>
<th></th>
<th>m1</th>
<th></th>
<th></th>
<th>m2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>b</td>
<td>se</td>
<td>p</td>
<td>b</td>
<td>se</td>
</tr>
<tr>
<td>weight</td>
<td>1.746559</td>
<td>.0081298</td>
<td>3.464706</td>
<td>5.99e-07</td>
<td></td>
</tr>
<tr>
<td></td>
<td>.6413538</td>
<td>.630749</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mpg</td>
<td>-49.51222</td>
<td>.5673237</td>
<td>21.8536</td>
<td>.7692938</td>
<td></td>
</tr>
<tr>
<td></td>
<td>86.15604</td>
<td>74.22114</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>foreign</td>
<td></td>
<td></td>
<td>3673.06</td>
<td>9.72e-07</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>.683.9783</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_cons</td>
<td>1946.069</td>
<td>.5901886</td>
<td>-5853.696</td>
<td>.0874262</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3597.05</td>
<td>3376.987</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

end of do-file

---more---

end of do-file

. do_example do/0h.ihlp
   . estout m1 m2, cells(b(star fmt(3)) t(par fmt(2))) style(smcl)

<table>
<thead>
<tr>
<th></th>
<th>m1</th>
<th></th>
<th></th>
<th>m2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>b/t</td>
<td></td>
<td></td>
<td>b/t</td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>1.747**</td>
<td>3.465***</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(2.72)</td>
<td>(5.49)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mpg</td>
<td>-49.512</td>
<td>21.854</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(-0.57)</td>
<td>(0.29)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>foreign</td>
<td></td>
<td></td>
<td>3673.060***</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(5.37)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_cons</td>
<td>1946.069</td>
<td>-5853.696</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(0.54)</td>
<td>(-1.73)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

. end of do-file

---more---
The `stats()` option

The `stats()` option specifies the scalar statistics to be displayed for each model in the table footer. Example:

```
estout m1 m2, stats(r2 bic N) style(smcl)
<run>
```

Actually, the newest addition to `estout` is the ability to tabulate string `e()-` macros in the table footer:

```
regress price weight
estimates store m1
regress price weight, robust
estimates store m2
regress price weight, vce(bootstrap)
estimates store m3
estout m1 m2 m3, cells(b se(par)) stats(N vcetype) ///
    style(smcl)
estimates clear
<run>
```

do_example.do/ihlp

```
end of do-file
.do_example do/0i.ihlp
```

```
estout m1 m2, stats(r2 bic N) style(smcl)
```

<table>
<thead>
<tr>
<th></th>
<th>m1</th>
<th>m2</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>1.746559</td>
<td>3.464706</td>
</tr>
<tr>
<td>mpg</td>
<td>-49.51222</td>
<td>21.8536</td>
</tr>
<tr>
<td>foreign</td>
<td>3673.06</td>
<td></td>
</tr>
<tr>
<td>_cons</td>
<td>1946.069</td>
<td>-5853.696</td>
</tr>
<tr>
<td>r2</td>
<td>.2933891</td>
<td>.4995594</td>
</tr>
<tr>
<td>bic</td>
<td>1378.64</td>
<td>1357.414</td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td>74</td>
</tr>
</tbody>
</table>

end of do-file
estimates store m3

estout m1 m2 m3, cells(b se(par)) stats(N vcetype) ///
style(smcl)

<table>
<thead>
<tr>
<th></th>
<th>m1</th>
<th>m2</th>
<th>m3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>b/se</td>
<td>b/se</td>
<td>b/se</td>
</tr>
<tr>
<td>weight</td>
<td>2.044063</td>
<td>2.044063</td>
<td>2.044063</td>
</tr>
<tr>
<td></td>
<td>(.3768341)</td>
<td>(.3897465)</td>
<td>(.4378883)</td>
</tr>
<tr>
<td>_cons</td>
<td>-6.707353</td>
<td>-6.707353</td>
<td>-6.707353</td>
</tr>
<tr>
<td></td>
<td>(1174.43)</td>
<td>(1032.394)</td>
<td>(1195.078)</td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td>74</td>
<td>74</td>
</tr>
<tr>
<td>vcetype</td>
<td>Robust</td>
<td>Bootstrap</td>
<td></td>
</tr>
</tbody>
</table>

estimates clear

end of do-file

---

Technical note

This point was bothering me since long. The problem was that Stata has no string matrices and so I could not come up with a good approach to collect the eO-macros.

But then, last week, it occurred to me that it is real easy. Just set up a virtual matrix of string scalars and then fill it up. Here’s a snippet of the code that initializes the "string matrix":

```stata
local strscalars
forv m=1/nmodels' {
    local temp
    forv i=1/`list size of emptystats' {
        tempname m`m's`i'
        local temp `temp`m`m's`i'
    }
    local strscalars `''strscalars' `temp'''
}
```
--- Other estout options ---

`estout` has tons of other options to achieve all sorts of effects. For example,

- `label` uses variable labels instead of names,
- `eform` reports exponentiated results (odds ratio, hazard ratio)
- `keep()`, `drop()`, and `order()` select and order the coefficients to be included in the table,
- and `margin` can be used to report marginal effects after `mfx`.

Syntax can get complicated, especially when it comes to suboptions, and that’s what many users have trouble with.

I will therefore not go into detail here and now present `eststo` and `esttab`, which are supposed make things easier.

--- Storing estimates simplified ---

Syntax:

```
eststo [ name ] [, options ] [: command ]
eststo drop {#|name} [...]  
eststo clear
```

`options` description

- `[no]sample` do not/do store `e(sample)`
- `title(string)` specify a title for the stored set
- `addscalars(...)` add scalar statistics

... ... ...

`_eststo` is short for `eststo`, `noexample`

`by` is allowed with `eststo` if `eststo` is used as a prefix command:

```
by ... : eststo ... : estimation_command
```
Storing estimates simplified

Basic example:

    sysuse auto, clear
    regress price weight mpg
    eststo
    regress price weight mpg foreign
    eststo
    estout, style(smcl)
      <run>
    macro dir
      <run>
    eststo clear
      <run>

<table>
<thead>
<tr>
<th></th>
<th>est2 stored</th>
</tr>
</thead>
<tbody>
<tr>
<td>foreign</td>
<td>3673.06</td>
</tr>
<tr>
<td></td>
<td>683.9783</td>
</tr>
<tr>
<td>_cons</td>
<td>-5853.696</td>
</tr>
<tr>
<td></td>
<td>3376.987</td>
</tr>
<tr>
<td></td>
<td>-1.73</td>
</tr>
<tr>
<td></td>
<td>0.087</td>
</tr>
<tr>
<td></td>
<td>-12588.88</td>
</tr>
</tbody>
</table>

    . eststo
    (est2 stored)

    . estout, style(smcl)

<table>
<thead>
<tr>
<th></th>
<th>est1</th>
<th>est2</th>
</tr>
</thead>
<tbody>
<tr>
<td>b</td>
<td>1.746559</td>
<td>3.464706</td>
</tr>
<tr>
<td>b</td>
<td>-49.51222</td>
<td>21.8536</td>
</tr>
<tr>
<td>weight</td>
<td>3673.06</td>
<td></td>
</tr>
<tr>
<td>mpg</td>
<td>1946.069</td>
<td>-5853.696</td>
</tr>
<tr>
<td>foreign</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_cons</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

end of do-file

more
eststo: regress price weight mpg
eststo: regress price weight mpg foreign
estout, style(smcl)
<run>

Drop the e(sample):
eststo, noesample: reg price weight mpg
_eststo: reg price weight mpg
estimates dir
describe _est*
eststo clear
<run>
```
model     command     depvar     npar     title
  est1     regress     price     3
  est2     regress     price     4
  est3     regress     price     3
  est4     regress     price     3

.    describe _est*

  storage     display     value
  variable   name      type      format     label     variable label

  _est_est1     byte     %8.0g
  _est_est2     byte     %8.0g

.    eststo clear
```

```
end of do-file
```

```
---

storing estimates simplified

Add additional results while storing:

  regress price weight mpg
test weight = mpg
eststo, add(p_diff r(p))
estout, style(smcl) stats(p_diff)
eststo clear

<run>

use with by:

  by foreign: eststo: quietly reg price weight mpg
estout, style(smcl)
eststo clear
<run>
```
. test weight = mpg
   ( 1) weight - mpg = 0

   F(  1,    71) =  0.36
   Prob > F =  0.5514

. eststo, add(p_diff r(p))
(e(p_diff) = .55138216 added)
(est1 stored)

. estout, style(smcl) stats(p_diff)

. est1
   b

weight     1.746559
  mpg    -49.51222
     _cons  1946.069

  p_diff     .5513822

. by foreign: eststo: quietly reg price weight mpg

-> Domestic
(est1 stored)

-> Foreign
(est2 stored)

. estout, style(smcl)

<table>
<thead>
<tr>
<th></th>
<th>est1</th>
<th>est2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>b</td>
<td>b</td>
</tr>
<tr>
<td>weight</td>
<td>4.415037</td>
<td>5.155842</td>
</tr>
<tr>
<td>mpg</td>
<td>237.691</td>
<td>-19.77737</td>
</tr>
<tr>
<td>_cons</td>
<td>-13285.44</td>
<td>-5065.841</td>
</tr>
</tbody>
</table>

. eststo clear
Tabulating estimates

My recommendation is to use esttab in most situations.

- It’s syntax is much simpler than that of estout.
- It has better defaults (publication-style table that displays nicely in Stata’s results window).
- It provides full estout functionality (all estout options are allowed).
- And, to some degree, it extends functionality (e.g. Word RTF and Excel CSV output modes, improved LaTeX support).

Basic syntax:

```
esttab [ name[=list] ] [ using filename ] [, options ]
```

If name[=list] is omitted, esttab tabulates the estimation sets stored by eststo.

Overview of esttab options

```
b(fmt) beta[(fmt)] main(name [fmt]) t(fmt) abs not
se(fmt) p[(fmt)] ci[(fmt)] aux(name [fmt]) [no]constant
[no]star[list] staraux
r2[(fmt)] ar2[(fmt)] pr2[(fmt)] aic[(fmt)] bic[(fmt)]
scalars(list) sfmt(fmt [,...]) noobs obslast
wide [no]parentheses brackets [no]gaps [no]lines noeqlines
compress plain
label title(string) mtitles(list) nomtitles [no]depvars
[no]numbers coeflabels(list) [no]notes addnotes(list)
smcl | fixed | tab | csv | shtml | rtf | html | tex | booktabs
fragment page[(packages)] alignment(string) width(string)
replace append type nosil
```

drop(list) keep(list) order(list) equations(list)
eform margin unstack other_estout_options
Tabulating estimates using esttab

Default table:

sysuse auto, clear
eststo: regress price weight mpg
eststo: regress price weight mpg foreign
esttab
<run>

Display standard errors and add some summary statistics:

esttab, se ar2 nostar
<run>

Display beta coefficients:

esttab, beta not
<run>

```
. esttab

              |       (1)     |       (2)     |
-------------|---------------|---------------|
              |     price     |     price     |
-------------|---------------|---------------|
weight       | 1.747**      | 3.465***     |
              |   (2.72)      |   (5.49)      |
mpg          | -49.51        |  21.85        |
              |  -0.57        |   (0.29)      |
foreign      | 3673.1***     |              |
              |   (5.37)      |               |
_cons        |  1946.1       | -5853.7       |
              |   (0.54)      |   (-1.73)     |
-------------|---------------|---------------|
N            |      74       |      74       |
-------------|---------------|---------------|

* p<0.05, ** p<0.01, *** p<0.001
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
Tabulating estimates using esttab

Wide format:

esttab, wide compress
<run>

Labels and titles:

esttab, se ar2 nostar brackets label ///
  title(This is a regression table) ///
  nonumbers mtitles("Model A" "Model B") ///
  addnote("source: auto.dta")
<run>

Plain table:

esttab, plain
<run>

do_example do/10.ihlp

. esttab, wide compress

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>price</td>
<td>price</td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td>-----------</td>
</tr>
<tr>
<td>weight</td>
<td>1.747**</td>
<td>3.465***</td>
</tr>
<tr>
<td>mpg</td>
<td>(2.72)</td>
<td>(5.49)</td>
</tr>
<tr>
<td>foreign</td>
<td>-49.51</td>
<td>21.85</td>
</tr>
<tr>
<td>_cons</td>
<td>1946.1</td>
<td>3673.1***</td>
</tr>
<tr>
<td></td>
<td>(0.54)</td>
<td>(5.37)</td>
</tr>
<tr>
<td></td>
<td>-5853.7</td>
<td>(-1.73)</td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td>74</td>
</tr>
</tbody>
</table>

t statistics in parentheses
* p<0.05, ** p<0.01, *** p<0.001
.
end of do-file

more
addnote("source: auto.dta")

This is a regression table

<table>
<thead>
<tr>
<th></th>
<th>Model A</th>
<th>Model B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weight (lbs.)</td>
<td>1.747</td>
<td>3.465</td>
</tr>
<tr>
<td></td>
<td>[0.641]</td>
<td>[0.631]</td>
</tr>
<tr>
<td>Mileage (mpg)</td>
<td>-49.51</td>
<td>21.85</td>
</tr>
<tr>
<td></td>
<td>[86.16]</td>
<td>[74.22]</td>
</tr>
<tr>
<td>Car type</td>
<td>3673.1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>[684.0]</td>
<td></td>
</tr>
<tr>
<td>Constant</td>
<td>1946.1</td>
<td>-5853.7</td>
</tr>
<tr>
<td></td>
<td>[3597.0]</td>
<td>[3377.0]</td>
</tr>
<tr>
<td>Observations</td>
<td>74</td>
<td>74</td>
</tr>
<tr>
<td>Adjusted R-squared</td>
<td>0.273</td>
<td>0.478</td>
</tr>
</tbody>
</table>

Standard errors in brackets
Source: auto.dta

end of do-file

.do_example do/12.ihlp

esttab, plain

<table>
<thead>
<tr>
<th></th>
<th>est1</th>
<th>est2</th>
</tr>
</thead>
<tbody>
<tr>
<td>b/t</td>
<td>weight</td>
<td>1.746559</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3.464706</td>
</tr>
<tr>
<td></td>
<td>mpg</td>
<td>-49.51222</td>
</tr>
<tr>
<td></td>
<td></td>
<td>21.8536</td>
</tr>
<tr>
<td></td>
<td>foreign</td>
<td>-0.5746808</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.2944391</td>
</tr>
<tr>
<td></td>
<td>_cons</td>
<td>3673.06</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5.370142</td>
</tr>
<tr>
<td></td>
<td>N</td>
<td>74</td>
</tr>
<tr>
<td></td>
<td></td>
<td>74</td>
</tr>
</tbody>
</table>

end of do-file
Display formats

Numerical display formats may be specified as:

- official Stata's display formats such as \%9.0g or \%8.2f
- integer values such as 0, 1, 2, etc. for fixed formats
- a1, a2, ..., or a9 to cause \texttt{esttab} choose a reasonable display format depending on the scale of the displayed number (the \# in a\# is the minimum number of significant digits)

The default display format depends on type of displayed statistic (e.g. a3 for point estimates and fixed format 3 for p-values and the R-squared):

\texttt{esttab, p r2 nostar wide}

\texttt{<run>}

Specifying alternative formats:

\texttt{esttab, b(\%9.0g) p(4) r2(4) nostar wide}

\texttt{<run>}

\texttt{end of do-file}

.\ do_example do/13.ihlp

.\ esttab, p r2 nostar wide

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th></th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>price</td>
<td></td>
<td>price</td>
</tr>
<tr>
<td>weight</td>
<td>1.747</td>
<td>(0.008)</td>
<td>3.465</td>
</tr>
<tr>
<td>mpg</td>
<td>-49.51</td>
<td>(0.567)</td>
<td>21.85</td>
</tr>
<tr>
<td>foreign</td>
<td>1946.1</td>
<td>(0.590)</td>
<td>-5835.7</td>
</tr>
<tr>
<td>_cons</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td>74</td>
<td></td>
</tr>
<tr>
<td>R-sq</td>
<td>0.293</td>
<td>0.500</td>
<td></td>
</tr>
</tbody>
</table>

p-values in parentheses

.\ end of do-file

\texttt{more}
end of do-file

. do_example do/14.ihlp

.   esttab, b(%9.0g) p(4) r2(4) nostar wide

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight</td>
<td>1.746559</td>
<td>3.464706</td>
</tr>
<tr>
<td>mpg</td>
<td>-49.51222</td>
<td>21.8536</td>
</tr>
<tr>
<td>foreign</td>
<td>1946.069</td>
<td>-5853.696</td>
</tr>
<tr>
<td>_cons</td>
<td>1946.069</td>
<td>-5853.696</td>
</tr>
</tbody>
</table>

N     |  74    |  74    |
R-sq  |  0.2934|  0.4996|

p-values in parentheses

end of do-file

--- Use with Word, Excel, LaTeX, etc. ---

esttab features a variety of output formats:

- **smcl**: SMCL formatted (default unless using is specified)
- **fixed**: fixed-format ASCII (default if using is specified)
- **tab**: tab-delimited ASCII
- **csv**: CSV (Comma Separated Value format) for use with Excel
- **scsv**: "German" version of CSV (semicolon instead of comma)
- **rtf**: Rich Text Format for use with word processors
- **html**: HTML-formatted
- **tex**: LaTeX-formatted
- **booktabs**: LaTeX-formatted for use with booktabs
Use with Word, Excel, LaTeX, etc.

Excel: csv or scsv

esttab using example.csv
<run>

esttab using example.csv, scsv replace
<run>

Use the plain option if you intend to do additional computations in Excel:
esttab using example.csv, replace wide plain
<run>

(No Excel XML support. Sorry.)

+-----------------+-----------------+-----------------+-----------------+
|                | price           | price           |                |
|-----------------+-----------------+-----------------+-----------------|
| weight          | 1.746559        | (0.0081)        | 3.464706        | (0.0000)        |
| mpg             | -49.51222       | (0.5673)        | 21.8536         | (0.7693)        |
| foreign         | 1946.069        | (0.5902)        | -5853.696       | (0.0874)        |
| _cons           |                 |                 |                 |
| N               | 74              | 74              |                 |
| R-sq            | 0.2934          | 0.4996          |                 |

p-values in parentheses

.end of do-file

.do_example do/15.ihlp

.esttab using example.csv
(output written to example.csv)

.end of do-file

---more---
<table>
<thead>
<tr>
<th></th>
<th>est1</th>
<th>est2</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight</td>
<td>1.746559</td>
<td>3.465706</td>
</tr>
<tr>
<td>mpg</td>
<td>-49.512222</td>
<td>21.853478</td>
</tr>
<tr>
<td>foreign</td>
<td>3673.06</td>
<td>5.370142</td>
</tr>
<tr>
<td>cons</td>
<td>1946.069</td>
<td>-5853.896</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>b</th>
<th>t</th>
<th>b</th>
<th>t</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight</td>
<td>1.746559</td>
<td>2.723238</td>
<td>3.465706</td>
<td>5.483003</td>
</tr>
<tr>
<td>mpg</td>
<td>-49.512222</td>
<td>-0.5746808</td>
<td>21.853478</td>
<td>0.2944391</td>
</tr>
<tr>
<td>foreign</td>
<td>3673.06</td>
<td>5.370142</td>
<td>-5853.896</td>
<td>-1.733408</td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td>74</td>
<td>74</td>
<td>74</td>
</tr>
</tbody>
</table>

*p<0.05, **p<0.01, ***p<0.001*
Use with Word, Excel, LaTeX, etc.

Word: rtf

```
esttab using example.rtf
<run>
```

Appending is possible. Furthermore, use `varwidth(#)` and `modelwidth(#)` to change column widths:

```
esttab using example.rtf, append wide label modelwidth(8)
<run>
```

Including RTF literals:

```
esttab using example.rtf, replace
    title({\b Table 1: This is a bold title})
<run>
esttab using example.rtf, replace
    cells(b(fmt(a3)) t(par(\i( ))))
<run>
```

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>price</td>
<td>price</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>weight</td>
<td>1.747**</td>
<td>3.465***</td>
</tr>
<tr>
<td></td>
<td>(2.72)</td>
<td>(5.49)</td>
</tr>
<tr>
<td>mpg</td>
<td>-49.51</td>
<td>21.85</td>
</tr>
<tr>
<td></td>
<td>(-0.57)</td>
<td>(0.29)</td>
</tr>
<tr>
<td>foreign</td>
<td>3673.1***</td>
<td>(5.37)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>const</td>
<td>1946.1</td>
<td>-5853.7</td>
</tr>
<tr>
<td></td>
<td>(0.54)</td>
<td>(-1.73)</td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td>74</td>
</tr>
</tbody>
</table>

* statistics in parentheses
* * p < 0.05, ** p < 0.01, *** p < 0.001
Table 1: This is a bold title

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight</td>
<td>1.747**</td>
<td>3.465***</td>
</tr>
<tr>
<td></td>
<td>(2.72)</td>
<td>(5.49)</td>
</tr>
<tr>
<td>mpg</td>
<td>-49.51</td>
<td>21.85</td>
</tr>
<tr>
<td></td>
<td>(-0.57)</td>
<td>(0.29)</td>
</tr>
<tr>
<td>foreign</td>
<td>3673.1***</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(5.37)</td>
<td></td>
</tr>
<tr>
<td>_cons</td>
<td>1946.1</td>
<td>-5853.7</td>
</tr>
<tr>
<td></td>
<td>(0.54)</td>
<td>(-1.73)</td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td>74</td>
</tr>
</tbody>
</table>

If statistics in parentheses:
* p < 0.05, ** p < 0.01, *** p < 0.001
Use with Word, Excel, LaTeX, etc.

LaTeX: tex

esttab using example1.tex, label nostar
   title(Regression table\label{tab1}) page
<run>

!texify.exe --pdf example1.tex
winexec $AcroRd example1.pdf
<run>

LaTeX: booktabs

esttab using example2.tex, label nostar replace booktabs
   title(Regression table\label{tab1}) page

!texify.exe --pdf example2.tex
winexec $AcroRd example2.pdf
<run>
\begin{table}
\centering
\begin{tabular}{|l|c|c|}
\hline
& (1) & (2) \\
\hline
Weight (lbs.) & 1.747 & 3.465 \\
& (2.72) & (5.49) \\
\hline
Mileage (mpg) & -49.51 & 21.85 \\
& (-0.57) & (0.29) \\
\hline
Car type & & 3673.1 \\
& & (5.37) \\
\hline
Constant & 1946.1 & -5853.7 \\
& (0.54) & (-1.73) \\
\hline
Observations & 74 & 74 \\
\hline
\end{tabular}
\caption{Regression table}
\end{table}

\textit{t} statistics in parentheses
Table 1: Regression table

<table>
<thead>
<tr>
<th></th>
<th>(1) Price</th>
<th>(2) Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weight (lbs.)</td>
<td>1.747 (2.72)</td>
<td>3.465 (5.49)</td>
</tr>
<tr>
<td>Mileage (mpg)</td>
<td>-49.51 (-0.57)</td>
<td>21.85 (0.29)</td>
</tr>
<tr>
<td>Car type</td>
<td>3673.1 (5.37)</td>
<td></td>
</tr>
<tr>
<td>Constant</td>
<td>1946.1 (0.54)</td>
<td>-5853.7 (-1.73)</td>
</tr>
<tr>
<td>Observations</td>
<td>74</td>
<td>74</td>
</tr>
</tbody>
</table>

*t statistics in parentheses*

---

Use with Word, Excel, LaTeX, etc.

Improved LaTeX table using the \texttt{dcolumn} package:

\begin{verbatim}
\texttt{esttab using example3.tex, label replace booktabs}\\
\hspace{1cm}alignment(D{{}.}{-1})\\
\hspace{1cm}page(dcolumn)\\
\hspace{1cm}title(Regression table\label{tab1})
\end{verbatim}

\texttt{!texify.exe --pdf example3.tex}
\texttt{winexec $AcroRd example3.pdf}

<run>
Table 1: Regression table

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Price</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weight (lbs.)</td>
<td>1.747**</td>
<td>3.465***</td>
</tr>
<tr>
<td></td>
<td>(2.72)</td>
<td>(5.49)</td>
</tr>
<tr>
<td>Mileage (mpg)</td>
<td>-49.51</td>
<td>21.85</td>
</tr>
<tr>
<td></td>
<td>(-0.57)</td>
<td>(0.29)</td>
</tr>
<tr>
<td>Car type</td>
<td></td>
<td>3673.1***</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(5.37)</td>
</tr>
<tr>
<td>Constant</td>
<td>1946.1</td>
<td>-5853.7</td>
</tr>
<tr>
<td></td>
<td>(0.54)</td>
<td>(-1.73)</td>
</tr>
<tr>
<td>Observations</td>
<td>74</td>
<td>74</td>
</tr>
</tbody>
</table>

\* t statistics in parentheses
\* * p < 0.05, ** p < 0.01, *** p < 0.001

Get the estout code

*esttab's noisily option shows the issued estout command:

```
esttab, noisily
<run>
return list
<run>
`r(estout)`
eststo clear
<run>
```
esttab, noisily
estout
cells(b(fmt(a3) star) t(fmt(2)) par("{\textalign l2:{txt:}" {txt:}}"))
stats(N, fmt(%18.0g) labels("{N}"))
startlevels(0.05 ** 0.01 *** 0.001)
varwidth(12)
modelwidth(12)
abbrev
delimiter("")
smcltags
prehead("\{hline @width\}"")
posthead("\{hline @width\}")
prefoot("\{hline @width\}")
postfoot("\{hline @width\} "t statistics in parentheses" @starleq
varlabels(, end(""""""""""""""""""""""""""") nolast)
mlabels(, depvar)
numbers
collabels(, none)
eqlabels(, begin("\{hline @width\}" """" nofirst)
level(95)

--- Advanced examples ---

- Transformations
- Stacking models
- Some advanced LaTeX
- Rearranging statistics in the table footer
- Include results form LR-Tests
- Writing one's own estadd subcommands
- Table of descriptives
- Tabulating t-tests
- SPost and estout
Transformations

Example 1: Exponentiation (odds ratio, hazard ratio, incidence-rate ratio, relative risk ratio)

logistic foreign weight price
eststo
eststo
esttab, eform(0 1)
eststo clear
<run>

Example 2: transform and xtmixed

Syntax: transform(fx dfx)
transform(coef fx dfx [ ... [coef] fx dfx ])

use pig, clear
xtmixed weight week || _all: R.id || _all: R.week
esttab, transform(ln*: exp(@) exp(@))
<run>

(est2 stored)

. esttab, eform(0 1)

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>foreign</td>
<td>foreign</td>
</tr>
<tr>
<td>weight</td>
<td>-0.00588***</td>
<td>0.994***</td>
</tr>
<tr>
<td></td>
<td>(-3.46)</td>
<td>(-3.46)</td>
</tr>
<tr>
<td>price</td>
<td>0.000930**</td>
<td>1.001**</td>
</tr>
<tr>
<td></td>
<td>(3.10)</td>
<td>(3.10)</td>
</tr>
<tr>
<td>_cons</td>
<td>9.000***</td>
<td>8106.9***</td>
</tr>
<tr>
<td></td>
<td>(3.43)</td>
<td>(3.43)</td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td>74</td>
</tr>
</tbody>
</table>

t statistics in parentheses
* p<0.05, ** p<0.01, *** p<0.001
Log restricted-likelihood = -1015.4214

| weight  | Coef.  | Std. Err. | z      | P>|z| | [95% Conf.] |
|---------|--------|-----------|--------|------|----------------|
| week    | 6.209896 | .0578669  | 107.31 | 0.000 | 6.096479 |
| _cons   | 19.35561 | .6493996  | 29.81  | 0.000 | 18.08281 |

Random-effects Parameters

<table>
<thead>
<tr>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf.]</th>
</tr>
</thead>
<tbody>
<tr>
<td>_all: Identity sd(R.id)</td>
<td>3.892648</td>
<td>.4141707</td>
</tr>
<tr>
<td>_all: Identity sd(R.week)</td>
<td>.3337581</td>
<td>.1611824</td>
</tr>
<tr>
<td>sd(Residual)</td>
<td>2.072917</td>
<td>.0759515</td>
</tr>
</tbody>
</table>

LR test vs. linear regression:

\( \text{chi2}(2) = 476.10 \)  

\( \text{Prob > chi2} = \)
Stacking models

estout cannot stack models. The solution is to stack the models in advance and save in e().

Example: Include a table column containing the bivariate effects.

capt prog drop appendmodels
program appendmodels, eclass
    // using models’ first equations
    version 8
    syntax namelist
    tempname b v tmp
    foreach name of local namelist {
        qui est restore `name'
        mat `tmp' = e(b)
        local eq1: coeq `tmp'
        gettoken eq1 : eq1
        mat `tmp' = `tmp'[1,"eq1":,]
        local cons = colnumb(`tmp',"_cons")
        if `cons'<. & `cons'>1 {
            mat `tmp' = `tmp'[1,1..`cons'-1]
        }
    }
    mat `b' = nullmat(`b') , `tmp'
    mat `tmp' = e(v)
    mat `tmp' = `tmp'['eq1':,"eq1":]
    if `cons'<. & `cons'>1 {
        mat `tmp' = `tmp'[1..`cons'-1,1..`cons'-1]
    }
    capt confirm matrix `v'
    if _rc {
        mat `v' = `tmp'
    }
    else {
        mat `v' = ///
            ( `v' , j(rowsof(`v'),colsof(`tmp'),0) ) \ ///
            ( j(rowsof(`tmp'),colsof(`v'),0) , `tmp' )
    }
    local names: colfullnames `b'
    mat coln `v' = `names'
    mat rrown `v' = `names'
    eret post `b' `v'
    eret local cmd "whatever"
end
sysuse auto, clear
eststo b1: regress price weight
eststo b2: regress price mpg
eststo b3: regress price foreign
eststo b1: appendmodels b1 b2 b3
eststo multi: regress price weight mpg foreign
esttab multi bi, nodepvar
eststo clear

<run>

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>multi</td>
<td>bi</td>
</tr>
<tr>
<td>weight</td>
<td>3.465***</td>
<td>2.044***</td>
</tr>
<tr>
<td></td>
<td>(5.49)</td>
<td>(5.42)</td>
</tr>
<tr>
<td>mpg</td>
<td>21.85</td>
<td>-238.9***</td>
</tr>
<tr>
<td></td>
<td>(0.29)</td>
<td>(-4.50)</td>
</tr>
<tr>
<td>foreign</td>
<td>3673.1***</td>
<td>312.3</td>
</tr>
<tr>
<td></td>
<td>(5.37)</td>
<td>(0.41)</td>
</tr>
<tr>
<td>_cons</td>
<td>-5853.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(-1.73)</td>
<td></td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td></td>
</tr>
</tbody>
</table>

t statistics in parentheses
* p<0.05, ** p<0.01, *** p<0.001
Some advanced LaTeX

Example: Arrange models in groups.

sysuse auto
eststo: reg weight mpg
eststo: reg weight mpg foreign
eststo: reg price weight mpg
eststo: reg price weight mpg foreign
esttab using example4.tex, booktabs replace label
   \mgroups\{A B\, pattern\{1 0 1 0\}\}
   \prefix\{\multicolumn\{0\}\{c\}\{\}\}
   \suffix\{}
   \span \er\{\midrule\{r\}\{0\}\{span\}\}\}
   \alignment\{D\{.\}\{.\}\{.\}\{-1\}\}
   \page\{dcolumn\}
   nonumber
eststo clear
\textify.exe --pdf example4.tex
winexec $AcroRd example4.pdf

A

<table>
<thead>
<tr>
<th></th>
<th>Weight (lbs.)</th>
<th>Weight (lbs.)</th>
<th>Price</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mileage (mpg)</td>
<td>-108.4***</td>
<td>-91.22***</td>
<td>-49.51</td>
<td>21.85</td>
</tr>
<tr>
<td></td>
<td>(-11.60)</td>
<td>(-10.34)</td>
<td>(-0.57)</td>
<td>(0.29)</td>
</tr>
<tr>
<td>Car type</td>
<td>-550.1***</td>
<td>3673.1***</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(-4.96)</td>
<td>(5.37)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Weight (lbs.)</td>
<td>1.747***</td>
<td></td>
<td></td>
<td>3.465***</td>
</tr>
<tr>
<td></td>
<td>(2.72)</td>
<td></td>
<td></td>
<td>(5.49)</td>
</tr>
<tr>
<td>Constant</td>
<td>5328.8***</td>
<td>5125.7***</td>
<td>1946.1</td>
<td>-5853.7</td>
</tr>
<tr>
<td></td>
<td>(25.85)</td>
<td>(27.93)</td>
<td>(0.54)</td>
<td>(-1.73)</td>
</tr>
<tr>
<td>Observations</td>
<td>74</td>
<td>74</td>
<td>74</td>
<td>74</td>
</tr>
</tbody>
</table>

* t statistics in parentheses
  * p < 0.05, ** p < 0.01, *** p < 0.001
Rearranging statistics in the table footer

Example 1:

```
regress price weight foreign
estout, stats(F p N, layout("@ @" @) fmt(a3 3 a3) ///
         labels("F statistic" "Observations")) ///
         cells("b(fmt(a3)) p(fmt(3))") label style(smcl)
```

```
eststo clear
<run>
```

Example 2:

```
eststo: logit foreign weight mpg
eststo: logit foreign weight mpg turn disp1
esttab, stats(chi2 df m r2 p N, layout(`" @ (@)" @`))
eststo clear
<run>
```

```
estout, stats(F p N, layout("@ @" @) fmt(a3 3 a3) ///
         labels("F statistic" "Observations")) ///
         cells("b(fmt(a3)) p(fmt(3))") label style(smcl)
```

<table>
<thead>
<tr>
<th>Weight (lbs.)</th>
<th>b</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>Car type</td>
<td>3.321</td>
<td>0.000</td>
</tr>
<tr>
<td>_cons</td>
<td>3637.0</td>
<td>0.000</td>
</tr>
<tr>
<td>_cons</td>
<td>-4942.8</td>
<td>0.000</td>
</tr>
</tbody>
</table>

| F statistic   | 35.35 | 0.000  |
| Observations  | 74     |        |

```
eststo clear
```

end of do-file
<table>
<thead>
<tr>
<th></th>
<th>Estimate 1</th>
<th>Estimate 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight</td>
<td>-0.00391***</td>
<td>0.00239</td>
</tr>
<tr>
<td></td>
<td>(-3.86)</td>
<td>(0.99)</td>
</tr>
<tr>
<td>mpg</td>
<td>-0.169</td>
<td>-0.196*</td>
</tr>
<tr>
<td></td>
<td>(-1.83)</td>
<td>(-2.07)</td>
</tr>
<tr>
<td>turn</td>
<td>-0.502*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(-2.28)</td>
<td></td>
</tr>
<tr>
<td>disp</td>
<td>-0.0769*</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(-2.06)</td>
<td></td>
</tr>
<tr>
<td>_cons</td>
<td>13.71**</td>
<td>26.95**</td>
</tr>
<tr>
<td></td>
<td>(3.03)</td>
<td>(3.00)</td>
</tr>
</tbody>
</table>

Chi2 (df_m) 35.72 (2) 55.82 (4)
r2_p 0.397 0.620
N 74 74

* p<0.05, ** p<0.01, *** p<0.001

---

Include results form LR-Tests

**estadd** has a **lrtest** subcommand that can be used as follows:

eststo A: quietly logit foreign weight

eststo B: quietly logit foreign weight mpg price

estadd lrtest A

esttab, scalars(lrtest_chi2 lrtest_df lrtest_p)
eststo clear

<run>
<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>foreign</td>
<td>foreign</td>
</tr>
<tr>
<td>weight</td>
<td>-0.00259***</td>
<td>-0.00685***</td>
</tr>
<tr>
<td></td>
<td>(-4.25)</td>
<td>(-3.43)</td>
</tr>
<tr>
<td>mpg</td>
<td>-0.121</td>
<td>(-1.27)</td>
</tr>
<tr>
<td>price</td>
<td>0.000926**</td>
<td>(3.01)</td>
</tr>
<tr>
<td>_cons</td>
<td>6.283***</td>
<td>14.42**</td>
</tr>
<tr>
<td></td>
<td>(3.92)</td>
<td>(2.66)</td>
</tr>
<tr>
<td>N</td>
<td>74</td>
<td>74</td>
</tr>
<tr>
<td>lrtest_chi2</td>
<td>23.78</td>
<td></td>
</tr>
<tr>
<td>lrtest_df</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>lrtest_p</td>
<td>0.00000684</td>
<td></td>
</tr>
</tbody>
</table>

* t statistics in parentheses
* p<0.05, ** p<0.01, *** p<0.001

writing one’s own estadd subcommands

Example 1: Report the multiple correlation (square root of the R-squared).

- manual approach:
  
eststo: quietly regress price weight mpg
  estadd scalar R = sqrt(e(r2))
  
eststo: quietly regress price weight mpg foreign
  estadd scalar R = sqrt(e(r2))
  
estout, stats(r2 R) style(smcl)
eststo clear
<run>
Writing one's own estadd subcommands (continued)

- approach using a subroutine:

  capture program drop estadd_R

  program estadd_R, eclass
    ereturn scalar R = sqrt(e(r2))
  end

  eststo: quietly regress price weight mpg
  eststo: quietly regress price weight mpg foreign
  estadd R : *

  estout, stats(r2 R) style(smcl)
  eststo clear
  <run>

estadd R : *

estout, stats(r2 R) style(smcl)

<table>
<thead>
<tr>
<th></th>
<th>est1</th>
<th>est2</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight</td>
<td>1.746559</td>
<td>3.464706</td>
</tr>
<tr>
<td>mpg</td>
<td>-49.51222</td>
<td>21.8536</td>
</tr>
<tr>
<td>foreign</td>
<td></td>
<td>3673.06</td>
</tr>
<tr>
<td>_cons</td>
<td>1946.069</td>
<td>-5853.696</td>
</tr>
<tr>
<td>r2</td>
<td>0.2933891</td>
<td>0.4995594</td>
</tr>
<tr>
<td>R</td>
<td>0.5416541</td>
<td>0.7067952</td>
</tr>
</tbody>
</table>

eststo clear

end of do-file

---more---
Writing one’s own estadd subcommands (continued)

Example 2: Report y-standardized coefficients.

capture program drop estadd_bstdy

program estadd_bstdy, eclass
tempname bstdy
    matrix `bstdy' = e(b)
    quietly summarize `e(depvar)' if e(sample)
    matrix `bstdy' = `bstdy' / r(sd)
    ereturn matrix bstdy = `bstdy'
end

eststo: quietly regress price weight mpg
eststo: quietly regress price weight mpg foreign

estadd bstdy : *
estout, cells(b bstdy(par)) style(smcl)
eststo clear
<run>

estout, cells(b bstdy(par)) style(smcl)

<table>
<thead>
<tr>
<th></th>
<th>est1</th>
<th>est2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>b/bstdy</td>
<td>b/bstdy</td>
</tr>
<tr>
<td>weight</td>
<td>1.746559</td>
<td>3.464706</td>
</tr>
<tr>
<td></td>
<td>(.0005922)</td>
<td>(.0011747)</td>
</tr>
<tr>
<td>mpg</td>
<td>-49.51222</td>
<td>21.8536</td>
</tr>
<tr>
<td></td>
<td>(-.0167867)</td>
<td>(.0074093)</td>
</tr>
<tr>
<td>foreign</td>
<td>3673.06</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(1.245318)</td>
<td></td>
</tr>
<tr>
<td>_cons</td>
<td>1946.069</td>
<td>-5853.696</td>
</tr>
<tr>
<td></td>
<td>(.659797)</td>
<td>(-1.984643)</td>
</tr>
</tbody>
</table>

eststo clear

end of do-file

more
Table of descriptives

The trick is to regress a fake variable on all variables including the dependent variable.

```
generate y = uniform()
regress y price weight mpg foreign, noconstant
estadd summ
estout, cells("mean sd min max") style(smcl)
<run>
```

Using by: eststo: and estadd to get descriptives by subgroups:

```
by foreign: eststo: regress y price weight mpg, nocons
estadd summ :
esttab, main(mean) aux(sd) label nodepvar nostar nonote eststo clear
<run>
```

```
<table>
<thead>
<tr>
<th></th>
<th>mean</th>
<th>sd</th>
<th>min</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>price</td>
<td>6165.257</td>
<td>2949.496</td>
<td>3291</td>
<td>15906</td>
</tr>
<tr>
<td>weight</td>
<td>3019.459</td>
<td>777.1936</td>
<td>1760</td>
<td>4840</td>
</tr>
<tr>
<td>mpg</td>
<td>21.2973</td>
<td>5.785503</td>
<td>12</td>
<td>41</td>
</tr>
<tr>
<td>foreign</td>
<td>.2972973</td>
<td>.4601885</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>
```

```
end of do-file
---more---
```
estadd summ:*
esttab, main(mean) aux(sd) label nodepvar nostar nonote

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domestic</td>
<td>6072.4</td>
<td>6384.7</td>
</tr>
<tr>
<td></td>
<td>(3097.1)</td>
<td>(2621.9)</td>
</tr>
<tr>
<td>Foreign</td>
<td>6384.7</td>
<td>6072.4</td>
</tr>
<tr>
<td></td>
<td>(2621.9)</td>
<td>(3097.1)</td>
</tr>
<tr>
<td>Price</td>
<td>3317.1</td>
<td>2315.9</td>
</tr>
<tr>
<td></td>
<td>(695.4)</td>
<td>(433.0)</td>
</tr>
<tr>
<td>Weight (lbs.)</td>
<td>19.83</td>
<td>24.77</td>
</tr>
<tr>
<td></td>
<td>(4.743)</td>
<td>(6.613)</td>
</tr>
<tr>
<td>Mileage (mpg)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Observations</td>
<td>52</td>
<td>22</td>
</tr>
<tr>
<td>eststo clear</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Tabulating t-Tests

Basically anything can be tabulated by **estout** or **esttab** once it is posted in **e**. Here is an example with t-tests:

capt prog drop myttests
program myttests, eclass
version 8
syntax varlist [if] [in], by(varname) [*]
marksample touse
markout "touse" `by`
tempname mu_1 mu_2 d d_se d_t d_p
foreach var of local varlist {
    qui ttest `var' if `touse', by(`by') `options'
    mat mu_1' = nullmat(mu_1'), r(mu_1)
    mat mu_2' = nullmat(mu_2'), r(mu_2)
    mat d' = nullmat(d'), r(mu_1')-r(mu_2')
    mat d_se' = nullmat(d_se'), r(se)
    mat d_t' = nullmat(d_t'), r(t)
    mat d_p' = nullmat(d_p'), r(p)
}
foreach mat in mu_1 mu_2 d d_se d_t d_p {
    mat coln `mat' = `varlist'
}
tempname b v
mat b' = `mu_1' * 0
mat v' = `b' .
ret post `b' `v'
ret local cmd "myttests"
foreach mat in mu_1 mu_2 d d_se d_t d_p {
    eret mat `mat' = `mat'
}
end

myttests price weight mpg, by(foreign)
estout, style(smcl) ///
    cells("mu_1(fmt(a3)) mu_2 d(star pvalue(d_p))")
</run>

(An alternative approach would be to save three sets of estimates, one for each group, and one for the differences.)
Example 1: fitstat

```
eststo: logit foreign weight mpg
'eststo: logit foreign weight mpg turn disp1
'estadd fitstat: *
esttab, scalars(r2_mf r2_ml r2_cu r2_ef)
<run>
```

Example 2: listcoef

```
estadd listcoef: *
estout, cell("b_fact b_facts") drop(_cons) style(smcl)
<run>
```

Example 3: prchange

```
estadd prchange: *
estout, cell("dc_mminmax dcspd") drop(_cons) style(smcl)
<run>
```

```
mpg
(-3.86)  (0.99)
-0.169   -0.196*
(-1.83)  (-2.07)

turn
-0.502*
(-2.28)

displacement
-0.0769*
(-2.06)

_cons  13.71**  26.95**
       (3.03)  (3.00)

N  74  74
r2_mf  0.397  0.620
r2_ml  0.383  0.530
r2_cu  0.544  0.752
r2_ef  0.411  0.636

* p<0.05, ** p<0.01, *** p<0.001
```
end of do-file

. do_example do/41.ihlp
. estadd listcoef: *
. estout, cell("b_fact b_facts") drop(_cons) style(smcl)

<table>
<thead>
<tr>
<th></th>
<th>b_fact</th>
<th>b_facts</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight</td>
<td>.9961009</td>
<td>.048014</td>
</tr>
<tr>
<td>mpg</td>
<td>.8448578</td>
<td>.3770566</td>
</tr>
<tr>
<td>turn</td>
<td>.6052757</td>
<td>.1098336</td>
</tr>
<tr>
<td>displacement</td>
<td>.9259888</td>
<td>.0008574</td>
</tr>
</tbody>
</table>

end of do-file

---more---

end of do-file

. do_example do/42.ihlp
. estadd prchange: *
. estout, cell("dcminmax dcsd") drop(_cons) style(smcl)

<table>
<thead>
<tr>
<th></th>
<th>dcminmax</th>
<th>dcsd</th>
<th>dcminmax</th>
<th>dcsd</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight</td>
<td>-.9622371</td>
<td>.4207634</td>
<td>.3229343</td>
<td>.0129412</td>
</tr>
<tr>
<td>mpg</td>
<td>-.4656343</td>
<td>-.1303092</td>
<td>-.0366136</td>
<td>-.0072707</td>
</tr>
<tr>
<td>turn</td>
<td>-.3217601</td>
<td>-.1062399</td>
<td></td>
<td></td>
</tr>
<tr>
<td>displacement</td>
<td>-.9821618</td>
<td>-.1738952</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

end of do-file

---more---
Thank you for listening!

References


http://repec.org/bocode/e/estout

<clean-up>