

Creating final versions of your tables: outreg2 and outsum

The final version of your paper should not have STATA output, it should have nicely formatted tables (e.g., in excel) that look like a real published paper! No, you do not have to do a bunch of cutting and pasting to get this look: instead, in STATA there are commands that allow you to construct regression tables (outreg2) and tables of means (outsum) directly.

Outreg2

Outreg2 is an installable STATA command which allows you to make regression tables like you see in papers. The first step is to install the command. With a computer connected to the internet, type on the STATA command line (you only ever have to do this once):

```
ssc install outreg2
```

After that, you can use outreg2 **after running any regression**. So first you run a regression, then you type the syntax:

```
outreg2 [list of variables] using [filename], excel [replace/append] [other options]
```

...where (brackets not actually part of the command, by the way):

- [list of variables] - is a list of any subset of variables that were used in the regression. These coefficients on these variables will be shown. By default, all of the independent variables in your regression are shown. But if you, say, have a lot of dummy variables you might want to suppress these coefficients.
 - Note that in some versions of outreg2 listing the variables which you want to show does not work. So instead to hide coefficients you must use the “drop” option, described below.
- [filename] - can be any name you want to give it -- although if it has a space in the name it should be in quotes. The regression table will appear in this file.
- excel -- output the table as an excel file
 - By the way, you don’t have to output it as an excel file. The default is a text file, which you can also read into excel. Open the file from excel directly (Open... and choose the “All Files” option), and read in a “delimited” file.
 - There is also a **latex** option for the hardcore. Like the excel option, it does not create a final version of the table, just some latex starter code that gives the outlines of a possible table.
- [replace/append] -- To create the first column of the table use “replace.” To add columns to the table, type “append.” Put differently, the first time you use outreg2 when you are creating a given table you type “replace.” Every time thereafter, you type “append”

instead.

- Other options include:
 - **ctitle(text)** - creates a column header containing the text of *title*. This is a good way to keep track of what was in each regression. For example you might have column headers “OLS” and “Fixed effects.” You can continue a column header to a second line by have two sets of text separated by a comma, as in, `ctitle(OLS,Females Only)`
 - **noaster** -- removes the “*”s that indicate statistical significance levels.
 - **drop(list of vars)** – do not show the coefficients on the specified *list of vars*. This is especially with long lists of controls – such as fixed effects – where we are not really interested in the coefficients, just the knowledge that that list of variables is controlled for. It is especially useful in combination with the “addtext” option.
 - **keep(list of vars)** – similar to previous.
 - **addtext** – allows you do add that bottom row with things like “Fixed effects? Yes.” The syntax is simple: `addtext(description1,value1, description2,value2,...)`. The “descriptions” are the things that go in the leftmost column (like “Fixed effects?”), while the “values” are the things that vary from column to column (“Yes” “No” etc.). Also:
 1. Notice that you could have several of these, for example, “`addtext(Year Effects, Yes, Fixed Effects, No)`”
 2. Notice also that the value of this command is being able to repeat it across *different* regressions. So if the above was used after a regression without fixed effects, then you might run the regression again with fixed effects, but instead have “`addtext(Year Effects, Yes, Fixed Effects, Yes)`”
 - **addstat** -- if you want to add an F-test to the table, this is how you’d do it:
 1. After your regression, do your “test” command.
 2. Then do your “outreg2” command, and after the comma have “`addstat(description1,r(F),description2,r(p))`” Note that “description1” and “description 2” are just labels that will appear in the leftmost column of your table that you want to give to the F-stat (which is stored in the variable `r(F)`) and the p-value (`r(p)`). So, generically it might be “F-stat” and “p-value.” It adds it to the bottom of the table. We can talk in more detail about how to do that if you want. You could also add the root MSE if you wanted, with “`addstat(RMSE, e(rmse))`”.
 - **addnote(text)** -- add table footnotes.
 - Known “bugs”:
 1. I have noticed that on some machines the “addstat” command only works properly with you put the statistics inside special single quotes, as in “`addstat(description1,`r(F)',description2,`r(p)')`”. These special single quotes are left quote (` -- the key below the “~” in the upper left-hand side of the

keyboard) and right quote (‘ – regular apostrophe). On other computers adding this will crash the program.

2. The command outputs a special excel “executable” format of excel file, which can theoretically open directly from STATA. However, this differs on Macs vs. PCs:
 - On PCs, you can just click on the output window to pull up the excel table.
 - On macs the “excel” format seems not to work by default as an executable. Instead, you have to open the file directly from the excel File menu (“Open...”) after which you can save it in regular excel format.

More generally, typing “help outreg2” on the command line will give you a complete list of options. Example on blackboard.

Finally, you are not done once you have created an outreg table. You’ll need edit the output of the outreg2 table even further to give it nice titles, formatting, etc. (I would then save it under a new file name so you don’t accidentally overwrite it if you run your STATA program again.) The final version of the tables should be self-contained and readable by someone not very familiar with the details of what’s in the paper.

Outsum

There is a similar STATA command for making fancy tables of means like the one above: outsum. Like with outreg2, you must first install the command, which does not come standard in STATA packages:

```
ssc install outsum
```

The outsum command creates a table with means and standard deviations (in parentheses underneath means). For an examples, see data exercise 4 – the optional part at the end created the table of means by whether or not you had health insurance.

The syntax for using this is shown on the next page. It is similar to outreg2 except that:

- You don't need to run a regression!
- There is, unfortunately, no “excel” option. Instead, it creates a text file which you can read into excel (by opening “all files”).
 - I think the best option is to use the “comma” option and use a file name with a “.csv” at the end, which make the file directly openable in excel. For example:

```
use alcohol.dta, clear
gen emp = empst == 1
gen female = sex==2
outsum emp days momhgc dadhgc using means.csv, comma replace bracket ///
ctitle("Overall") nol

outsum emp days momhgc dadhc if fem==1 using means.csv, comma append ///
bracket ctitle("Women") nol

outsum emp days momhgc dadhgc if fem==0 using means.csv, comma append bracket
ctitle("Men") nol
```

This creates a table of means with columns for overall, women, and men in a file called “means.csv” which should be able to open in excel, and should look something like the table of means above in this handout. The optional “bracket” option uses square, rather than round parenthesis for the standard errors, as was in the table above. You could follow up by putting in nice title formatting, etc. (and then actually saving it in excel format.) “nol” is an option which hides the long variable labels; I have found the tables format better if you put in this option. Of course, to actually make the table look like it does above, you would have to edit means.csv and give it nice labels and save it as an excel file.

I can provide additional help on this if you need it, but see if you can't figure it out yourself (use "help outsum.")

The detailed syntax is:

```
outsum [list of variables] if sample==1 using filename.csv, replace ///
      nol bracket comma ctitle("column header")
```

[list of variables] is not actually a thing you type in, you replace it with the list of variables you want to take the means of. Similarly, *filename.txt* is any filename you want to give it. The other options just make the output easier to deal with in excel

replace -- use this to create the table of means. If you want to add columns to it, change it instead to "append." See example below.

nol – optional – it says show the variable names rather than the variable labels.

bracket – puts the standard deviations in brackets "[" rather than parentheses. Good for distinguishing for standard errors.

comma – saves it as a comma delimited file. You can open such a file in excel and "save as..." an excel file.

ctitle – Put custom "column header" at the top of the column.