

Customizable tables

Build tables of

- Summary statistics
- Hypothesis test results
- Regression results
- Postestimation tests
- Marginal means, marginal effects, adjusted predictions
- Results returned by any Stata command

Customize table look

- Table layout
- Row and column header styles
- Numeric formats, font, shading, color, and alignment
- Labels

Export tables to

- Word
- Excel
- L^AT_EX
- PDF
- HTML
- Markdown
- More

Incorporate tables into reports

- Save table styles and apply customizations to future tables

	Male		Female		
Diabetes status					
Not diabetic	4898	95.6%	5152	94.8%	
Diabetic	217	4.4%	282	5.2%	
Age, mean (sd)	47.4	(17.2)	47.7	(17.3)	
BMI, mean (sd)	25.5	(4.0)	25.6	(5.6)	
Health status					
Excellent	1252	25.5%	1155	21.3%	
Very good	1213	24.7%	1378	25.4%	
Good	1340	27.3%	1508	28.5%	
Fair	722	14.7%	948	17.5%	
Poor	382	7.8%	347	6.4%	
Systolic BP, mean (sd)	132.9	(21.0)	129.1	(25.1)	

With the **table** command, you can create tabulations, tables of summary statistics, tables of regression results, and more.

Cross-tabulations

table can create one-way, two-way, and multiway tables, reporting frequencies, percentages, and proportions. For instance, you can create a two-way tabulation.

	Hypertension		Total
	No	Yes	
Sex			
Male	2,611	2,304	4,915
Female	3,364	2,072	5,436
Total	5,975	4,376	10,351

And you can add percentages and format the results.

	Hypertension		Total
	No	Yes	
Sex			
Male			
Frequency	2,611	2,304	4,915
Percent	25.2%	22.3%	47.5%
Female			
Frequency	3,364	2,072	5,436
Percent	32.5%	20.6%	52.5%
Total			
Frequency	5,975	4,376	10,351
Percent	57.7%	42.3%	100.0%

Summary statistics

With **table's** **statistic()** option, you can create tables of means, standard deviations, percentiles, counts, and much more. For example, you can create a table of means and standard deviations.

	Hypertension		Total
	No	Yes	
Age (years)			
Mean	42.165	54.973	47.580
Standard deviation	16.772	14.909	17.215
Weight (kg)			
Mean	68.266	76.856	71.898
Standard deviation	13.572	16.241	15.356
Systolic blood pressure			
Mean	116.485	150.539	130.882
Standard deviation	11.799	20.695	23.333

Regression results

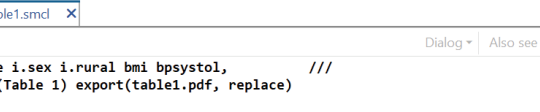
The **command()** option reports results from any Stata command in your table. You may want to compare regression results across groups.

	Hypertension		Total
	No	Yes	
Age (years)			
	0.22	0.53	0.63
	(0.01)	(0.02)	(0.01)
Weight (kg)			
	0.20	0.17	0.41
	(0.01)	(0.02)	(0.01)
Sex			
Female	-2.32	3.58	0.71
	(0.31)	(0.60)	(0.41)
Diabetes status			
Diabetic	-0.45	6.52	5.87
	(0.82)	(1.11)	(0.91)
Intercept			
	94.63	105.99	70.99
	(0.94)	(2.05)	(1.19)

With the **dtable** command, you can create and export a table of descriptive statistics, commonly known as a Table 1.

Create and export your Table 1

You can use **dtable** to easily create a Table 1 with means and standard deviations for continuous variables and with counts and percentages for categorical variables. And with the same command, you can export the table to, say, a PDF.



The screenshot shows a RStudio viewer window titled "view dtable1.smcl". The window contains a PDF document titled "Table 1". The PDF content is as follows:

```

+
. dtable i.sex i.rural bmi bpsystol, ///
> title(Table 1) export(table1.pdf, replace)

```

Table 1

	Summary
N	10,351
Sex	
Male	4,915 (47.5%)
Female	5,436 (52.5%)
Rural	
Urban	6,548 (63.3%)
Rural	3,803 (36.7%)
Body mass index (BMI)	25.538 (4.915)
Systolic blood pressure	130.882 (23.333)

(collection DTable exported to file [table1.pdf](#))

You could have instead exported the table to Word, Excel, L^AT_EX, HTML, or Markdown by specifying the appropriate file extension.

Customize the table contents

You can change formats and select which statistics you want for each variable. For instance, you can report quartiles for **bmi** and **bpsystol**, formatted to display one digit after the decimal.

```
Viewer - view dtable2.smc1
view dtable2.smc1 X
+
. dtable i.sex i.rural age, ///
> continuous(bmi bpsystol, stat(q1 q2 q3)) ///
> nformat(%6.1f q1 q2 q3)

Summary

N 10,351
Sex
  Male 4,915 (47.5%)
  Female 5,436 (52.5%)
Rural
  Urban 6,548 (63.3%)
  Rural 3,803 (36.7%)
Age (years) 47.580 (17.215)
Body mass index (BMI) 22.1 24.8 28.0
Systolic blood pressure 114.0 128.0 142.0
```

Report statistics by group

You can compute statistics separately for each group and test for equality across groups. For instance, you can test for equality of means in rural and urban populations.

```
Viewer - view dtable3.smcld
view dtable3.smcld

+ Dialog ▾ Also see ▾ Jump to ▾

. dtable age bpsystol tcresult, by(rural, tests) ///
> sample(, stat(freq) place(seplabels)) ///
> sformat("(N=%s)" frequency) column(by(hide))

note: using test regress across levels of rural for age, bpsystol, and tcresult.
```

	Urban (N=6,548)	Rural (N=3,803)	Total (N=10,351)	Test
Age (years)	46.835 (17.484)	48.862 (16.666)	47.580 (17.215)	<0.001
Systolic blood pressure	130.460 (23.526)	131.607 (22.980)	130.882 (23.333)	0.016
Serum cholesterol (mg/dL)	217.742 (49.874)	217.545 (48.544)	217.670 (49.387)	0.845

Survey data

Do you have data from a complex survey design? You can add the **svy** option to compute summary statistics and perform tests that account for the survey design.

```

Viewer - view dtable4.smcl
view dtable4.smcl x
+
. dtable, by(heartatk, tests) svy ///
> factor(highbp diabetes, test(svylr))
note: using test svylr across levels of heartatk for highbp and diabetes.

Prior heart attack
No heart attack      Had heart attack      Total      Test
N      113,647,835 (97.0%)  3,483,276 (3.0%)  117,131,111 (100.0%)
High blood pressure
  No      72,451,166 (63.8%)  1,502,443 (43.1%)  73,953,609 (63.1%) <0.001
  Yes     41,196,669 (36.2%)  1,980,833 (56.9%)  43,177,502 (36.9%)
Diabetes status
  Not diabetic      110,028,641 (96.8%)  3,091,189 (88.7%)  113,119,830 (96.6%) <0.001
  Diabetic          3,619,194 (3.2%)    392,087 (11.3%)   4,011,281 (3.4%)

```

Further customization

dtable allows you to customize how each statistic is reported. You can even build your own composite results made up of multiple statistics so that you can display them combined and with any format you like. And because **dtable** creates a collection, you can customize results even further with the **collect** suite of commands. (See page 4.)

With the **etable** command, you can create and export a table of estimation results in one step.

Create and export a table

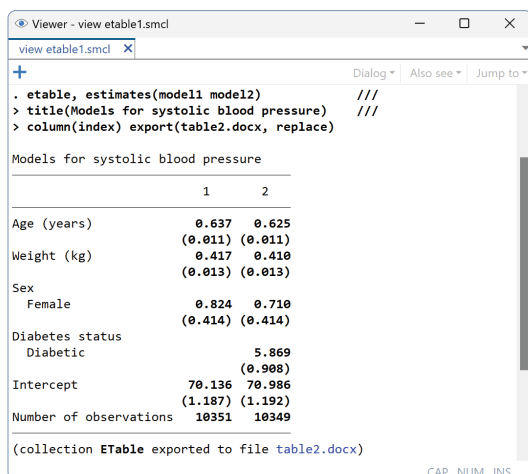
You can create a table immediately after fitting a model with any estimation command.

```
. regress bpsystol age weight i.sex  
. etable
```

Or you can fit multiple models and store the results. For instance,

```
. regress bpsystol age weight i.sex  
. estimates store model1  
. regress bpsystol age weight i.sex i.diabetes  
. estimates store model2
```

And then you can use **etable** to create, customize, and export a table with results from the stored models.



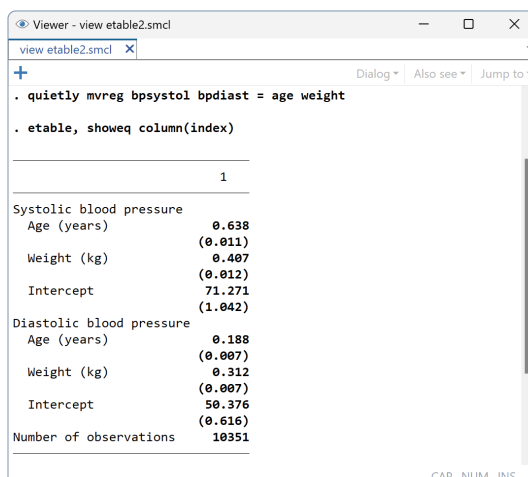
	1	2
Age (years)	0.637 (0.011)	0.625 (0.011)
Weight (kg)	0.417 (0.013)	0.410 (0.013)
Sex		
Female	0.824 (0.414)	0.710 (0.414)
Diabetes status		
Diabetic		5.869 (0.908)
Intercept	70.136 (1.187)	70.986 (1.192)
R-squared	0.30	0.31
AIC	90848.72	90791.95
Number of observations	10351	10349

(collection ETable exported to file table2.docx)

This table is exported in Word format, but you can also export to Excel, L^AT_EX, PDF, HTML, or SMCL.

Multiple-equation models

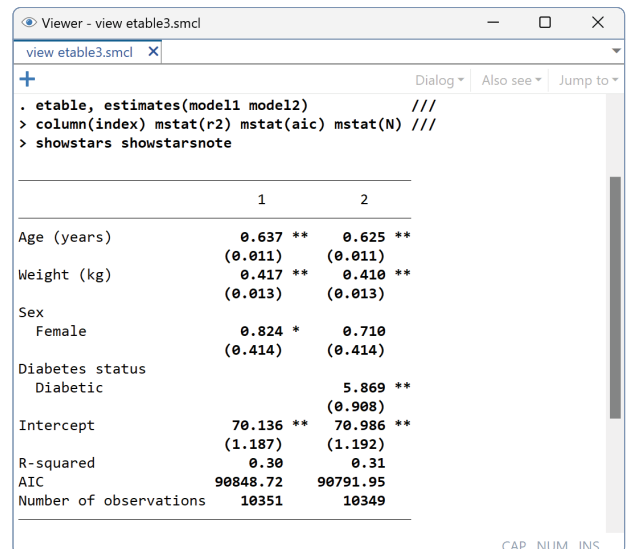
etable can easily report results of a multivariate regression or any other model that includes multiple equations. And you can customize the table to show equation descriptions.



	1	2
Systolic blood pressure		
Age (years)	0.638 (0.011)	0.625 (0.011)
Weight (kg)	0.407 (0.012)	0.410 (0.013)
Intercept	71.271 (1.042)	70.986 (1.192)
Diastolic blood pressure		
Age (years)	0.188 (0.007)	0.188 (0.007)
Weight (kg)	0.312 (0.007)	0.312 (0.007)
Intercept	50.376 (0.616)	50.376 (0.616)
Number of observations	10351	10349

Customize the table contents

With **etable**, you can specify which statistics should be displayed. Select from coefficients, standard errors, *p*-values, confidence intervals, *R*², AIC, BIC, stars for significance levels, and more. And you can specify how you would like these statistics to be displayed.



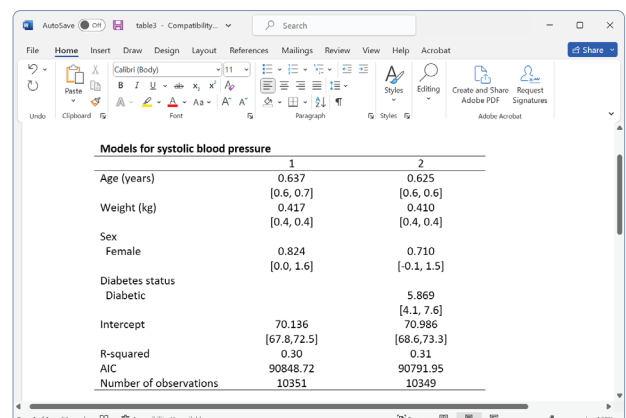
	1	2
Age (years)	0.637 ** (0.011)	0.625 ** (0.011)
Weight (kg)	0.417 ** (0.013)	0.410 ** (0.013)
Sex		
Female	0.824 * (0.414)	0.710 (0.414)
Diabetes status		
Diabetic		5.869 ** (0.908)
Intercept	70.136 ** (1.187)	70.986 ** (1.192)
R-squared	0.30	0.31
AIC	90848.72	90791.95
Number of observations	10351	10349

Further customization

etable allows you to customize your table in many ways, including column headers, labels, alignment, formats, fonts, shading, titles, notes, and stars. Beyond this, because **etable** creates a collection, you can use the **collect** suite of commands to further customize your table. (See page 4.)

For example, you can specify the width of a table in a Word document.

```
. collect style putdocx, width(75%)  
. collect export table3.docx
```



	1	2
Age (years)	0.637 [0.6, 0.7]	0.625 [0.6, 0.6]
Weight (kg)	0.417 [0.4, 0.4]	0.410 [0.4, 0.4]
Sex		
Female	0.824 [0.0, 1.6]	0.710 [-0.1, 1.5]
Diabetes status		
Diabetic		5.869 [4.1, 7.6]
Intercept	70.136 [67.8, 72.5]	70.986 [68.6, 73.3]
R-squared	0.30	0.31
AIC	90848.72	90791.95
Number of observations	10351	10349

With the **collect** suite of commands, you can build more complex tables, and you can modify the labels, layout, formatting, and more.

Collect results from multiple commands

You can store three sets of results in a collection.

```
. collect: regress bpsystol age i.sex
. collect: regress bpsystol age i.sex c.age#i.sex
. collect: regress bpsystol age i.sex c.age#i.sex
          i.diabetes
```

Lay out the table

Then you define rows and columns—place covariates (**colname**) and results (coefficients, standard errors, and R^2) on the rows, and place models (**cmdset**) on the columns.

```
. collect layout (colname#result[_r_b _r_se]
                 result[r2]) (cmdset)
```

Modify the labels

You may specify new labels for your column headers.

```
. collect label levels cmdset 1 "Model 1" 2 "Model 2"
3 "Model 3"
```

Customize the table style

You can modify formats, results for base levels, labels, delimiters for interactions, spacing, alignment, and borders.

```
. collect style cell, nformat(%6.2f)
. collect style cell result[_r_se], sformat((%s))

. collect style showbase off

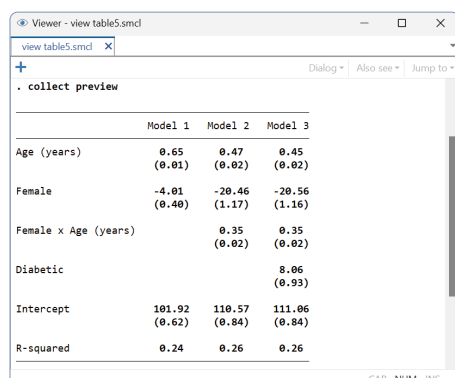
. collect style header result, level(hide)
. collect style header result[r2], level(label)

. collect style row stack, spacer delimiter(" x ")
. collect style column, extraspace(1)

. collect style cell cell_type[item column-header],
    halign(center)

. collect style cell border_block,
    border(right, pattern(none))
```

Preview the table

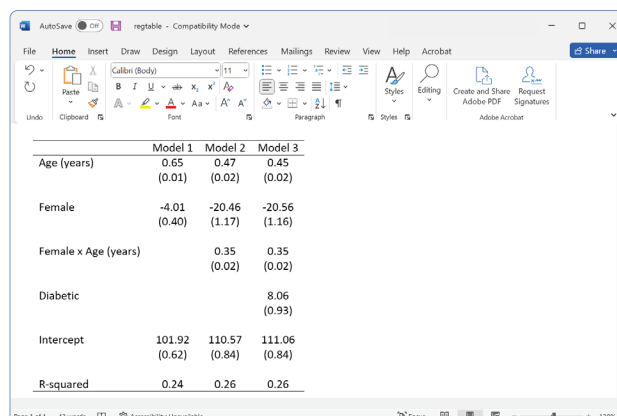


	Model 1	Model 2	Model 3
Age (years)	0.65 (0.01)	0.47 (0.02)	0.45 (0.02)
Female	-4.01 (0.40)	-20.46 (1.17)	-20.56 (1.16)
Female x Age (years)		0.35 (0.02)	0.35 (0.02)
Diabetic			8.06 (0.93)
Intercept	101.92 (0.62)	110.57 (0.84)	111.06 (0.84)
R-squared	0.24	0.26	0.26

Export the table

Once you have customized your table, you can export it to Word, Excel, \LaTeX , PDF, HTML, Markdown, and more.

```
. collect export regtable.docx
```



	Model 1	Model 2	Model 3
Age (years)	0.65 (0.01)	0.47 (0.02)	0.45 (0.02)
Female	-4.01 (0.40)	-20.46 (1.17)	-20.56 (1.16)
Female x Age (years)		0.35 (0.02)	0.35 (0.02)
Diabetic			8.06 (0.93)
Intercept	101.92 (0.62)	110.57 (0.84)	111.06 (0.84)
R-squared	0.24	0.26	0.26

Or include your table in a complete report.

```
. putdocx begin
...
. putdocx collect
...
. putdocx save myrpt
```

Easily create similar tables

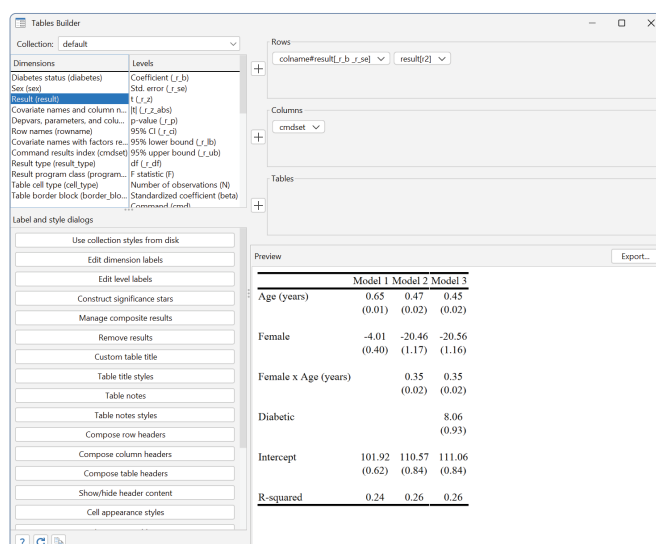
Want to use the same style for future tables? Save the style.

```
. collect style save mystyle
```

You can then use it again.

```
. collect style use mystyle
```

Type commands or use the GUI



	Model 1	Model 2	Model 3
Age (years)	0.65 (0.01)	0.47 (0.02)	0.45 (0.02)
Female	-4.01 (0.40)	-20.46 (1.17)	-20.56 (1.16)
Female x Age (years)		0.35 (0.02)	0.35 (0.02)
Diabetic			8.06 (0.93)
Intercept	101.92 (0.62)	110.57 (0.84)	111.06 (0.84)
R-squared	0.24	0.26	0.26