collect layout — Specify table layout for the current collection

Description

`collect layout` builds a table from the current collection. With `collect layout`, you specify which of the values that were collected from other Stata commands are to appear in the table, and you specify how the table is to be arranged.

The values in a collection are categorized into dimensions. These dimensions may represent types of statistics and covariate names. To specify a table layout, you specify which of these dimensions go on the rows and which of these dimensions go on the columns of your table.

As you specify the table layout, you can also determine which levels of a dimension are to be included in the table. For instance, if a collection includes three types of statistics—means, standard deviations, and frequencies—you may specify that only means are to appear in the table.

Table layouts can go beyond a single table with rows and columns. You can also specify dimensions that identify multiple tables.

`collect layout`, typed without arguments, reports the current layout.

Quick start

Table with dimension `dim1` on the rows and `dim2` on the columns

```
collect layout (dim1) (dim2)
```

Same as above, but include levels `lev1` and `lev2` of `dim1` instead of the levels automatically determined by `collect`

```
collect layout (dim1[lev1 lev2]) (dim2)
```

Table with interacted levels of dimension `dim1` and `dim2` on the rows and `dim3` on the columns

```
collect layout (dim1#dim2) (dim3)
```

Table with levels of dimension `dim1` and then the levels of `dim2` on the rows and `dim3` on the columns

```
collect layout (dim1 dim2) (dim3)
```

Separate tables for each level of dimension `dim3`, each with `dim1` on the rows and `dim2` on the columns

```
collect layout (dim1) (dim2) (dim3)
```

Menu

Statistics > Summaries, tables, and tests > Tables and collections > Build and style table

Syntax

Basic syntax for specifying the table layout

Single column layout with specified rows

```
collect layout (rows)
```

Single row layout with specified columns

```
collect layout () (cols)
```

Single table layout with specified rows and columns

```
collect layout (rows) (cols)
```

Multiple tables layout with specified rows and columns

```
collect layout (rows) (cols) (tabs)
```

Full syntax for specifying the table layout

```
collect layout ([rows]) [ ([cols]) [(tabs)]] [, warn|nowarn]
```

Report the current layout

```
collect layout
```

Clear the layout information

```
collect layout, clear
```

rows, *cols*, and *tabs* are each composed of a *taglist* that selects dimensions and possibly levels within a dimension. Within the *taglist*, if tags are joined by **#**, their levels are interacted to identify rows, columns, or separate tables; if tags are separated by a space, their levels are appended to identify rows, columns or separate tables.

taglist contains

tagspec

tagspec taglist

tagspec contains

tag

tag#*tag*[#*tag*[...]]

tag contains

dimension

dimension[*levels*]

dimension is a dimension in the collection.

levels are levels of the corresponding dimension.

Distinguish between `[]`, which are to be typed, and [ ], which indicate optional arguments.

Options

clear resets the collection's layout information.

warn and nowarn control the display of notes when collect encounters a tag it does not recognize.

warn, the default, specifies that collect display notes when it encounters a tag it does not recognize.

nowarn specifies that collect not show the notes.

These options override the collect_warn setting; see [TABLES] **set collect_warn**.

Remarks and examples

After collecting results from Stata commands using collect get or the collect prefix, we can arrange results into a table using collect layout.

The values in a collection are organized by their associated tags. These tags allow us to lay out a table by specifying which tags we wish to put on the rows and columns. More specifically, tags have two parts, the dimension and the level within the dimension. For example, the dimensions may represent types of statistics and covariate names. The levels within those dimensions may be coefficients, standard errors, and test statistics and x1, x2, and x3. The dimension for our statistics is called result, and the dimension for our covariates is called colname, so we can type

```
. collect layout (colname) (result)
```

to lay out a table with the covariates on the rows and the statistics on the columns. In this case, we specified only the dimension name, so the levels of each dimension that appear in the table are those that are selected as automatic levels. These automatic levels may be decided by the default collection style by specifying the levels you are interested in at the time you collect the results or by specifying automatic levels using collect style autolevels. If you want to see levels other than the automatic levels in your table, you can specify both the dimensions and their levels in your layout. For instance, we might type

```
. collect layout (colname[x1 x2]) (result[_r_b _r_se])
```

to lay out a table with variables x1 and x2 appearing on the rows and with statistics _r_b and _r_se, the coefficients and standard errors, appearing on the columns.

To demonstrate, we use data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). Below, we fit a model for systolic blood pressure as a function of age and weight. We use the collect prefix to collect the results, and we specify the quietly prefix to suppress the output.

```
. use https://www.stata-press.com/data/r19/nhanes2
. quietly: collect: regress bpsystol age weight
. collect dims
Collection dimensions
Collection: default
```

Dimension	No. levels
Layout, style, header, label	
cmdset	1
coleq	1
colname	3
program_class	1
result	33
result_type	3
rowname	1
Style only	
border_block	4
cell_type	4

After collecting the results, we used `collect dims` to list the dimensions in our collection. These can be used to specify the rows and columns of our table. Let's put `colname`, the dimension containing covariates, on the rows and `result`, the dimension containing types of statistics, on the columns. If we type

```
. collect layout (colname) (result)
```

all covariates will be placed on the rows and all covariate-specific statistics will be reported on the columns. These statistics include the coefficient, confidence interval, test statistic, and p-value and more. This creates a wide table, and we likely want only a subset of these statistics reported.

Say that we instead wanted to include both the coefficient and its standard error. We could use `collect label list` to find that the levels of `result` that represent the reported coefficients and standard errors are `_r_b` and `_r_se`. We request these levels by typing

```
. collect layout (colname) (result[_r_b _r_se])
Collection: default
      Rows: colname
   Columns: result[_r_b _r_se]
   Table 1: 3 x 2
```

	Coefficient	Std. error
Age (years)	.6379892	.0111315
Weight (kg)	.4069041	.0124786
Intercept	71.27096	1.041742

Often we will want more than one dimension on the rows or columns. To demonstrate, we first collect results from another regression that includes the interaction between age and weight.

```
. collect: quietly: regress bpsystol age weight c.age#c.weight
```

Now we can place both the `colname` and `result` dimensions on the rows. We separate the dimension names by `#` to specify that the interacted levels should form the rows. If they were separated by a space, this would mean that we want to first see the levels of `colname` followed by the levels of `result`, but this is not what we want and would not uniquely identify the values corresponding to coefficients and

standard errors in our collection. After collecting the results from the second regression, we have two levels of the dimension cmdset that identify the two commands we ran. We will put this dimension on the columns.

```
. collect layout (colname#result[_r_b _r_se]) (cmdset)
Collection: default
      Rows: colname#result[_r_b _r_se]
   Columns: cmdset
   Table 1: 12 x 2
```

	1	2
Age (years)		
Coefficient	.6379892	.8898576
Std. error	.0111315	.0536198
Weight (kg)		
Coefficient	.4069041	.5733109
Std. error	.0124786	.0368295
Age (years) # Weight (kg)		
Coefficient		-.003581
Std. error		.0007458
Intercept		
Coefficient	71.27096	59.60983
Std. error	1.041742	2.64211

collect layout also allows you to build multiple tables. With this collection, we could, for instance, create a separate table for each of the models and again put variable names on the rows and statistics on the columns.

```
. collect layout (colname) (result[_r_b _r_se]) (cmdset)
Collection: default
      Rows: colname
   Columns: result[_r_b _r_se]
    Tables: cmdset
   Table 1: 3 x 2
   Table 2: 4 x 2
1
```

	Coefficient	Std. error
Age (years)	.6379892	.0111315
Weight (kg)	.4069041	.0124786
Intercept	71.27096	1.041742

```
2
```

	Coefficient	Std. error
Age (years)	.8898576	.0536198
Weight (kg)	.5733109	.0368295
Age (years) # Weight (kg)	-.003581	.0007458
Intercept	59.60983	2.64211

If you have a layout that you prefer to use for many of the tables you create, you can save the layout along with any preferred styles with collect style save. Then, after collecting new results, you can use collect style use to apply the same layout to the new collection.

Stored results

`collect layout` stores the following in `s()`:

Macros

`s(collection)`	name of collection
`s(rows)`	rows specification
`s(columns)`	columns specification
`s(tables)`	tables specification
`s(k_tables)`	number of tables
`s(table#)`	layout for the #th table

References

Huber, C. 2021. Customizable tables in Stata 17, part 3: The classic table 1. *The Stata Blog: Not Elsewhere Classified.* https://blog.stata.com/2021/06/24/customizable-tables-in-stata-17-part-3-the-classic-table-1/.

McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. "Plan and operation of the Second National Health and Nutrition Examination Survey, 1976–1980". In *Vital and Health Statistics*, ser. 1, no. 15. Hyattsville, MD: National Center for Health Statistics.

Also see

[TABLES] **collect get** — Collect results from a Stata command

[TABLES] **collect preview** — Preview the table in a collection

[TABLES] **collect style save** — Save collection styles to disk

[TABLES] **collect style use** — Use collection styles from disk