Example 25 — Creating summary statistics data from raw data

Description Remarks and examples Also see

Description

Below we show how to create summary statistics data (SSD) from raw data. We use `auto2.dta`:

```
. use https://www.stata-press.com/data/r19/auto2
(1978 automobile data)
. describe
  (output omitted)
. summarize
  (output omitted)
```

Remarks and examples

Remarks are presented under the following headings:

We are going to create SSD containing the variables `price`, `mpg`, `weight`, `displacement`, and `foreign`.

Preparing data for conversion

Before building the SSD, prepare the data to be converted:

1. Drop variables that you do not intend to include in the SSD. Dropping variables is not a requirement, but it will be easier to spot problems if you begin by eliminating the irrelevant variables.
2. Verify that you have no string variables in the resulting data. Summary statistics datasets cannot contain string values.
3. Verify that there are no missing values. If there are, be aware that observations containing one or more variables with missing values will be omitted from the SSD.
4. Verify that all variables are on a reasonable scale. We recommend that the means of variables be only 3 or 4 orders of magnitude different from each other. This will help to preserve numerical accuracy when the SSD are used.
5. Create any new variables containing transformations of existing variables that might be useful later. Once the data are converted to summary statistics form, you will not be able to create such variables.
6. Place the variables in a logical order. That will help the user of the SSD understand the data.
7. Save the resulting prepared data. Probably you will never need the prepared data, but one never knows for sure.

We take our own advice below:

```
. * ------------------------------------------------------------------------------
. * Suggestion 1: Keep relevant variables:
. *
. keep price mpg weight displacement foreign

. 
. * ------------------------------------------------------------------------------
. * Suggestion 2: Check for string variables
. * Suggestion 3: Verify no missing values
. * Suggestion 4: Verify variables on a reasonable scale:
. *
. summarize
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
       price |         74    6165.257    2949.496       3291      15906
         mpg |         74     21.2973    5.785503         12         41
      weight |         74    3019.459    777.1936       1760       4840
displacement |         74    197.2973    91.83722         79        425
     foreign |         74    .2972973    .4601885          0          1

. 
. * We will rescale weight and price:
. replace   weight = weight/1000
variable weight was int now float
(74 real changes made)
. replace   price  =  price/1000
variable price was int now float
(74 real changes made)
. label var weight "Weight (1000s lbs.)"
. label var price  "Price  ($1,000s)"
. * and now we check our work:
. *
. summarize
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
       price |         74    6.165257    2.949496       3.291     15.906
         mpg |         74     21.2973    5.785503         12         41
      weight |         74    3.019459    .7771936        1.76       4.84
displacement |         74    197.2973    91.83722         79        425
     foreign |         74    .2972973    .4601885          0          1

. 
. * ------------------------------------------------------------------------------
. * Suggestion 5: Create useful transformations:
. *
. generate gpm    = 1/mpg
. label var gpm    "Gallons per mile"
. 
```

```
. * ------------------------------------------------------------------------
. * Suggestion 6: Place variables in logical order:
. *
. order price mpg gpm
. summarize
```

Variable	Obs	Mean	Std. dev.	Min	Max
price	74	6.165257	2.949496	3.291	15.906
mpg	74	21.2973	5.785503	12	41
gpm	74	.0501928	.0127986	.0243902	.0833333
weight	74	3.019459	.7771936	1.76	4.84
displacement	74	197.2973	91.83722	79	425
foreign	74	.2972973	.4601885	0	1

```
.
. * ------------------------------------------------------------------------
. * Suggestion 7: save prepared data
. *
. save auto_raw
file auto_raw.dta saved
. * ------------------------------------------------------------------------
```

Converting to summary statistics form

To create the summary statistics dataset, you just need to type `sud build` and the names of the variables to be included. If you have previously kept the relevant variables, you can type `ssd build _all`.

We recommend the following steps:

1. Convert data to summary statistics form:

```
. ssd build _all
```

2. Review the result:

```
. ssd describe
. notes
. ssd list
```

3. Digitally sign the data:

```
. datasignature set
```

4. Save the data:

```
. save auto_ss
```

We follow our advice below. After that, we will show you the advantages of digitally signing the data.

```
. * ------------------------------------------------------------------------
. * Convert data:
. *
. ssd build _all
 (data in memory now summary statistics data; you can use ssd describe and
  ssd list to describe and list results.)
```

```
. * ------------------------------------------------------------------------
. * Review results:
. *
. ssd describe
  Summary statistics data
  Observations:              74
     Variables:               6
                                        (_dta has notes)
  ------------------------------------------------------------------------
  Variable name                  Variable label
  ------------------------------------------------------------------------
  price                          Price  ($1,000s)
  mpg                            Mileage (mpg)
  gpm                            Gallons per mile
  weight                         Weight (1000s lbs.)
  displacement                   Displacement (cu. in.)
  foreign                        Car origin
  ------------------------------------------------------------------------

. notes
_dta:
  1.  summary statistics data built from 'auto_raw.dta' on 27 Mar 2025 21:01:44
      using -ssd build _all-

. ssd list
  Observations = 74

  Means:
         price           mpg           gpm        weight  displacement
     6.1652567     21.297297      .0501928     3.0194595      197.2973

       foreign
      .2972973

  Variances implicitly defined; they are the diagonal of the covariance
  matrix.

  Covariances:
         price           mpg           gpm        weight  displacement
     8.6995258
    -7.9962828     33.472047
     .02178417    -.06991586      .0001638
     1.2346748    -3.6294262     .00849897     .60402985
     134.06705    -374.92521     .90648519      63.87345     8434.0748
     .06612809     1.0473899    -.00212897    -.21202888    -25.938912

       foreign
     .21177342

. * ------------------------------------------------------------------------
. * Digitally sign:
. *
. datasignature set
  8:8(102846):1914186416:2867097560       (data signature set)

. * ------------------------------------------------------------------------
. * Save:
. *
. save auto_ss
file auto_ss.dta saved

. * ------------------------------------------------------------------------
```

We recommend digitally signing the data so that anyone can verify later that the data are unchanged:

```
. datasignature confirm
  (data unchanged since 29may2024 15:18)
```

Let us show you what would happen if the data had changed:

```
. replace mpg = mpg+.0001 in 5
(1 real change made)
. datasignature confirm
  data have changed since 29may2024 15:18
r(9);
```

There is no reason for you or anyone else to change the SSD after it has been created, so we recommend that you digitally sign the data. With regular datasets, users do make changes, if only by adding variables.

Be aware that the data signature is a function of the variable names, so if you rename a variable—something you are allowed to do—the signature will change and `datasignature` will report, for example, "data have changed since 29may2024 15:18". Solutions to that problem are discussed in [SEM] **ssd**.

Publishing SSD

The summary statistics dataset you have just created can obviously be sent to and used by any Stata user. If you wish to publish your data in printed form, use `ssd describe` and `ssd list` to describe and list the data.

Creating SSD with multiple groups

The process for creating SSD containing multiple groups is nearly the same as for creating single-group data. The only differences are that you do not drop the group variable during preparation and that rather than typing

```
. ssd build _all
```

you type

```
. ssd build _all, group(varname)
```

Below we build the automobile SSD again, but this time, we specify `group(rep78)`:

```
. ssd build _all, group(rep78)
```

If you think carefully about this, you may be worried that `_all` includes `rep78` and thus we will be including the grouping variable among the summary statistics. `ssd build` knows to omit the group variable:

```
. * ------------------------------------------------------------------------
. * Suggestion 1: Keep relevant variables:
. *
. webuse auto2, clear
(1978 automobile data)
. keep price mpg weight displacement foreign rep78
```

```
. * ------------------------------------------------------------------------
. * Suggestion 2: Check for string variables
. * Suggestion 3: Verify no missing values
. * Suggestion 4: Verify variables on a reasonable scale:
. *
. summarize
```

Variable	Obs	Mean	Std. dev.	Min	Max
price	74	6165.257	2949.496	3291	15906
mpg	74	21.2973	5.785503	12	41
rep78	69	3.405797	.9899323	1	5
weight	74	3019.459	777.1936	1760	4840
displacement	74	197.2973	91.83722	79	425
foreign	74	.2972973	.4601885	0	1

```
. drop if rep78 >= .
(5 observations deleted)

. * We will rescale weight and price:
. replace   weight = weight/1000
variable weight was int now float
(69 real changes made)

. replace   price  =  price/1000
variable price was int now float
(69 real changes made)

. label var weight "Weight (1000s lbs.)"

. label var price  "Price  ($1,000s)"

. * and now we check our work:
. summarize
```

Variable	Obs	Mean	Std. dev.	Min	Max
price	69	6.146043	2.91244	3.291	15.906
mpg	69	21.28986	5.866408	12	41
rep78	69	3.405797	.9899323	1	5
weight	69	3.032029	.7928515	1.76	4.84
displacement	69	198	93.14789	79	425
foreign	69	.3043478	.4635016	0	1

```
. * ------------------------------------------------------------------------
. * Suggestion 5: Create useful transformations:
. *
. generate gpm    = 1/mpg

. label var gpm    "Gallons per mile"

. * ------------------------------------------------------------------------
. * Suggestion 6: Place variables in logical order:
. *
. order price mpg gpm

. summarize
```

Variable	Obs	Mean	Std. dev.	Min	Max
price	69	6.146043	2.91244	3.291	15.906
mpg	69	21.28986	5.866408	12	41
gpm	69	.0502584	.0128353	.0243902	.0833333
rep78	69	3.405797	.9899323	1	5
weight	69	3.032029	.7928515	1.76	4.84
displacement	69	198	93.14789	79	425
foreign	69	.3043478	.4635016	0	1

```
. * ------------------------------------------------------------------------
. * Suggestion 7: save prepared data
. *
. save auto_group_raw
file auto_group_raw.dta saved
. * ------------------------------------------------------------------------
. * Convert data:
. *
. ssd build _all, group(rep78)
 (data in memory now summary statistics data; you can use ssd describe and
  ssd list to describe and list results.)
. * ------------------------------------------------------------------------
. * Review results:
. *
. ssd describe
  Summary statistics data
  Observations:              69
     Variables:               6
                                          (_dta has notes)
  ------------------------------------------------------------------------------
  Variable name                 Variable label
  ------------------------------------------------------------------------------
  price                         Price  ($1,000s)
  mpg                           Mileage (mpg)
  gpm                           Gallons per mile
  weight                        Weight (1000s lbs.)
  displacement                  Displacement (cu. in.)
  foreign                       Car origin
  ------------------------------------------------------------------------------
  Group variable: rep78 (5 groups)
    Obs by group: 2, 8, 30, 18, 11
. notes
_dta:
  1.  summary statistics data built from 'auto_group_raw.dta' on 27 Mar 2025
      21:01:44 using -ssd build _all, group(rep78)-
. ssd list
-------------------------------------------------------------------------------
Group rep78==Poor:
  (output omitted)
-------------------------------------------------------------------------------
Group rep78==Fair:
  (output omitted)
-------------------------------------------------------------------------------
Group rep78==Average:
  (output omitted)
-------------------------------------------------------------------------------
Group rep78==Good:
  (output omitted)
-------------------------------------------------------------------------------
Group rep78==Excellent:
  Observations = 11
  Means:
         price           mpg           gpm        weight  displacement
         5.913     27.363636     .04048131     2.3227273     111.09091
       foreign
     .81818182
```

```
  Variances implicitly defined; they are the diagonal of the covariance
  matrix.
  Covariances:
         price           mpg           gpm        weight  displacement
     6.8422143
    -15.608899     76.254545
     .02750797     -.1184875     .00019114
       .956802    -3.0610912     .00510833     .16856184
     55.493298    -201.03636     .33150758     9.9577283     648.09091
     .34169998    -.92727273     .00182175     .07254547     3.8181818
       foreign
     .16363636
```

```
. * ------------------------------------------------------------------------
. * Digitally sign:
. *
. datasignature set
  40:8(34334):2679920516:3596756814       (data signature set)
. * ------------------------------------------------------------------------
. * Save:
. *
. save auto_group_ss
file auto_group_ss.dta saved
. * ------------------------------------------------------------------------
```

Also see

[SEM] **ssd** — Making summary statistics data (sem only)