**st_dropvar() — Drop variables or observations**

**Description**

`st_dropvar(vars)` drops the variables specified. `vars` is a row vector that may contain either variable names or variable indices. `st_dropvar(.)` drops all variables and observations.

`st_dropobsin()` and `st_dropobsif()` have to do with dropping observations.

`st_dropobsin(range)` specifies the observations to be dropped:

- `st_dropobsin(5)` drops observation 5.
- `st_dropobsin((5,9))` drops observations 5 through 9.
- `st_dropobsin((5\8\12))` drops observations 5 and 8 and 12.
- `st_dropobsin((5,7\8,11\13,13))` drops observations 5 through 7, 8 through 11, and 13.
- `st_dropobsin(.)` drops all observations (but not the variables).
- `st_dropobsin(J(0,1,.))` drops no observations (or variables).

`st_dropobsif(select)` specifies a `st_nobs()` × 1 vector. Observations `i` for which `select_i ≠ 0` are dropped.

`st_keepvar()`, `st_keepobsin()`, and `st_keepobsif()` do the same thing, except that the variables and observations to be kept are specified.

**Syntax**

```plaintext
void st_dropvar(transmorphic rowvector vars)
void st_dropobsin(real matrix range)
void st_dropobsif(real colvector select)
void st_keepvar(transmorphic rowvector vars)
void st_keepobsin(real matrix range)
void st_keepobsif(real colvector select)
```
Remarks and examples

To drop all variables and observations, code any of the following:

\[
\text{st\_dropvar(\text{.})} \\
\text{st\_keepvar(J(1,0,.))} \\
\text{st\_keepvar(J(1,0,\text{\"\}}))}
\]

All do the same thing. Dropping all the variables clears the dataset.
Dropping all the observations, however, leaves the variables in place.

Conformability

\[
\text{st\_dropvar(vars), st\_keepvar(vars):} \\
\text{vars:} \quad 1 \times k \\
\text{result:} \quad \text{void}
\]

\[
\text{st\_dropobsin(range), st\_keepobsin(range):} \\
\text{range:} \quad k \times 1 \quad \text{or} \quad k \times 2 \\
\text{result:} \quad \text{void}
\]

\[
\text{st\_dropobsif(select), st\_keepobsif(select):} \\
\text{select:} \quad \text{st\_nobs()} \times 1 \\
\text{result:} \quad \text{void}
\]

Diagnostics

\text{st\_dropvar(vars) and st\_keepvar(vars)} abort with error if any element of \text{vars} is missing unless \text{vars} is \(1 \times 1\), in which case they drop or keep all the variables.

\text{st\_dropvar(vars) and st\_keepvar(vars)} abort with error if any element of \text{vars} is not a valid variable index or name, or if \text{vars} is a view. If \text{vars} is specified as names, abbreviations are not allowed.

\text{st\_dropvar()} and \text{st\_keepvar()} set \text{st\_update()} (see \text{[M-5] st\_update()}) unless all variables dropped are temporary; see \text{[M-5] st\_tempname()}.

\text{st\_dropobsin(range) and st\_keepobsin(range)} abort with error if any element of \text{range} is missing unless \text{range} is \(1 \times 1\), in which case they drop or keep all the observations.

\text{st\_dropobsin(range) and st\_keepobsin(range)} abort with error if any element of \text{range} is not a valid observation number (is not between 1 and \text{st\_nobs()}) [see \text{[M-5] st\_nvar()}] inclusive) or if \text{range} is a view.

\text{st\_dropobsif(select) and st\_keepobsif(select)} abort with error if \text{select} is a view.

\text{st\_dropobsin(), st\_dropobsif(), st\_keepobsin(), and st\_keepobsif()} set \text{st\_update()} if any observations are removed from the data.

Be aware that, after dropping any variables or observations, any previously constructed views (see \text{[M-5] st\_view()}) are probably invalid because views are internally stored in terms of variable and observation numbers. Subsequent use of an invalid view may lead to unexpected results or an abort with error.
Also see

[M-4] Stata — Stata interface functions