Title

> **stfill** — Fill in by carrying forward values of covariates

Syntax

> stfill *varlist* [*if*] [*in*] , { <u>base</u>line | <u>f</u>orward } [*options*]

options	Description
Main	
* <u>base</u>line	replace with values at baseline
* <u>f</u>orward	carry forward values
<u>nosh</u>ow	do not show st setting information

* Either baseline or forward is required.

You must stset your data before using stfill; see [ST] **stset**.

fweights, iweights, and pweights may be specified using stset; see [ST] **stset**.

Menu

Statistics > Survival analysis > Setup and utilities > Fill forward with values of covariates

Description

stfill is intended for use with multiple-record st data for which id() has been stset. stfill may be used with single-record data, but it does nothing. That is, stfill can be used with multiple-record or single- or multiple-failure st data.

stfill, baseline changes variables to contain the value at the earliest time each subject was observed, making the variable constant over time. stfill, baseline changes all subsequent values of the specified variables to equal the first value, whether they originally contained missing or not.

stfill, forward fills in missing values of each variable with that of the most recent time at which the variable was last observed. stfill, forward changes only missing values.

You must specify either the baseline or the forward option.

if *exp* and in *range* operate slightly differently from their usual definitions to work as you would expect. if and in restrict where changes can be made to the data, but no matter what, all stset observations are used to provide the values to be carried forward.

Options

> [Main]

baseline specifies that values be replaced with the values at baseline, the earliest time at which the subject was observed. All values of the specified variables are replaced, missing and nonmissing.

forward specifies that values be carried forward and that previously observed, nonmissing values be used to fill in later values that are missing in the specified variables.

noshow prevents stfill from showing the key st variables. This option is seldom used because most people type stset, show or stset, noshow to set whether they want to see these variables mentioned at the top of the output of every st command; see [ST] stset.

Remarks and examples

stata.com

stfill assists in fixing data errors and makes baseline analyses easier.

▷ Example 1

Let's begin by repairing broken data.

You have a multiple-record st dataset that, because of how it was constructed, has a problem with the gender variable:

```
. use http://www.stata-press.com/data/r13/mrecord

. stvary sex

          failure _d:  myopic
    analysis time _t:  t
                  id:  id

            subjects for whom the variable is
                                            never    always   sometimes
     variable │  constant    varying        missing  missing   missing
    ──────────┼──────────────────────────────────────────────────────
          sex │       131          1             22        0        110
```

For 110 subjects, sex is sometimes missing, and for one more subject, the value of sex changes over time! The sex change is an error, but the missing values occurred because sometimes the subject's sex was not filled in on the revisit forms. We will assume that you have checked the changing-sex subject and determined that the baseline record is correct in that case, too.

```
. stfill sex, baseline

          failure _d:  myopic
    analysis time _t:  t
                  id:  id

replace all values with value at earliest observed:
            sex:  221 real changes made

. stvary sex

          failure _d:  myopic
    analysis time _t:  t
                  id:  id

            subjects for whom the variable is
                                            never    always   sometimes
     variable │  constant    varying        missing  missing   missing
    ──────────┼──────────────────────────────────────────────────────
          sex │       132          0            132        0          0
```

The sex variable is now completely filled in.

In this same dataset, there is another variable—bp, blood pressure—that is not always filled in because readings were not always taken.

```
. stvary bp

        failure _d:  myopic
  analysis time _t:  t
                id:  id

            subjects for whom the variable is
                                       never     always   sometimes
    variable │  constant    varying    missing   missing   missing
    ─────────┼──────────────────────────────────────────────────────
          bp │        18        114          9         0        123
```

(bp is constant for 18 patients because it was taken only once—at baseline.) Anyway, you decide that it will be good enough when bp is missing to use the previous value of bp:

```
. stfill bp, forward noshow

replace missing values with previously observed values:
            bp:   263 real changes made

. stvary bp, noshow

            subjects for whom the variable is
                                       never     always   sometimes
    variable │  constant    varying    missing   missing   missing
    ─────────┼──────────────────────────────────────────────────────
          bp │        18        114        132         0          0
```

So much for data repair and fabrication.

◁

▷ Example 2

Much later, deep in analysis, you are concerned about the bp variable and decide to compare results with a model that simply includes blood pressure at baseline. You are undecided on the issue and want to have both variables in your data:

```
. stset, noshow

. gen bp0 = bp

. stfill bp0, baseline

replace all values with value at earliest observed:
           bp0:   406 real changes made

. stvary bp bp0

            subjects for whom the variable is
                                       never     always   sometimes
    variable │  constant    varying    missing   missing   missing
    ─────────┼──────────────────────────────────────────────────────
          bp │        18        114        132         0          0
         bp0 │       132          0        132         0          0
```

◁

Also see

[ST] **stbase** — Form baseline dataset

[ST] **stgen** — Generate variables reflecting entire histories

[ST] **stset** — Declare data to be survival-time data

[ST] **stvary** — Report variables that vary over time