11

Survival and Event-Count Models

This chapter presents methods for analyzing event data. *Survival analysis* encompasses several related techniques that focus on times until the event of interest occurs. Although the event could be good or bad, by convention we refer to that event as a "failure." The time until failure is "survival time." Survival analysis is important in biomedical research, but it can be applied equally well to other fields from engineering to social science — for example, in modeling the time until an unemployed person gets a job, or a single person gets married. Stata offers a full range of survival analysis procedures, only a few of which are illustrated in this chapter.

We also look briefly at Poisson regression and its relatives. These methods focus not on survival times but, rather, on the rates or counts of events over a specified interval of time. Event-count methods include Poisson regression and negative binomial regression. Such models can be fit either through specialized commands, or through the broader approach of generalized linear modeling (GLM).

Consult the *Survival Analsysis and Epidemiological Tables Reference Manual* for more information about Stata's capabilities. Type **help st** to see an online overview. Selvin (1995) provides well-illustrated introductions to survival analysis and Poisson regression. I have borrowed (with permission) several of his examples. Other good introductions to survival analysis include the Stata-oriented volume by Cleves, Gould and Gutierrez (2004), a chapter in Rosner (1995), and comprehensive treatments by Hosmer and Lemeshow (1999) and Lee (1992). McCullagh and Nelder (1989) describe generalized linear models. Long (1997) has a chapter on regression models for count data (including Poisson and negative binomial), and also has some material on generalized linear models. An extensive and current treatment of generalized linear models is found in Hardin and Hilbe (2001).

Stata menu groups most relevant to this chapter include:

Statistics – Survival analysis

Graphics – Survival analysis graphs

Statistics – Count outcomes

Statistics – Generalized linear models (GLM)

Regarding epidemiological tables, not covered in this chapter, further information can be found by typing **help epitab** or exploring the menus for

Statistics – Observational/Epi. analysis.

Example Commands

Most of Stata's survival-analysis (**st***) commands require that the data have previously been identified as survival-time by issuing an **stset** command (see following). **stset** need only be run once, and the data subsequently saved.

. **stset *timevar*, failure(*failvar*)**

Identifies single-record survival-time data. Variable *timevar* indicates the time elapsed before either a particular event (called a "failure") occurred, or the period of observation ended ("censoring"). Variable *failvar* indicates whether a failure (*failvar* = 1) or censoring (*failvar* = 0) occurred at *timevar*. The dataset contains only one record per individual. The dataset must be **stset** before any further **st*** commands will work. If we subsequently **save** the dataset, however, the **stset** definitions are saved as well. **stset** creates new variables named *_st*, *_d*, *_t*, and *_t0* that encode information necessary for subsequent **st*** commands.

. **stset *timevar*, failure(*failvar*) id(*patient*) enter(time *start*)**

Identifies multiple-record survival-time data. In this example, the variable *timevar* indicates elapsed time before failure or censoring; *failvar* indicates whether failure (1) or censoring (0) occurred at this time. *patient* is an identification number. The same individual might contribute more than one record to the data, but always has the same identification number. *start* records the time when each individual came under observation.

. **stdes**

Describes survival-time data, listing definitions set by **stset** and other characteristics of the data.

. **stsum**

Obtains summary statistics: the total time at risk, incidence rate, number of subjects, and percentiles of survival time.

. **ctset *time nfail ncensor nenter*, by(*ethnic sex*)**

Identifies count-time data. In this example, the variable *time* is a measure of time; *nfail* is the number of failures occurring at *time*. We also specified *ncensor* (number of censored observations at *time*) and *nenter* (number entering at *time*), although these can be optional. *ethnic* and *sex* are other categorical variables defining observations in these data.

. **cttost**

Converts count-time data, previously identified by a **ctset** command, into survival-time form that can be analyzed by **st*** commands.

. **sts graph**

Graphs the Kaplan–Meier survivor function. To visually compare two or more survivor functions, such as one for each value of the categorical variable *sex*, use the **by()** option,

```
. sts graph, by(sex)
```

To adjust, through Cox regression, for the effects of a continuous independent variable such as *age*, use the **adjustfor()** option,

```
. sts graph, by(sex) adjustfor(age)
```

Note: the **by()** and **adjustfor()** options work similarly with the other **sts** commands **sts list**, **sts generate**, and **sts test**.

. **sts list**
Lists the estimated Kaplan–Meier survivor (failure) function.

. **sts test *sex***
Tests the equality of the Kaplan–Meier survivor function across categories of *sex*.

. **sts generate *survfunc* = S**
Creates a new variable arbitrarily named *survfunc*, containing the estimated Kaplan–Meier survivor function.

. **stcox *x1 x2 x3***
Fits a Cox proportional hazard model, regressing time to failure on continuous or dummy variable predictors *x1–x3*.

. **stcox *x1 x2 x3*, strata(*x4*) basechazard(*hazard*) robust**
Fits a Cox proportional hazard model, stratified by *x4*. Stores the group-specific baseline cumulative hazard function as a new variable named *hazard*. (Baseline survivor function estimates could be obtained through a **basesur(*survive*)** option.) Obtains robust standard error estimates. See Chapter 9 or, for a more complete explanation of robust standard errors, consult the *User's Guide*.

. **stphplot, by(*sex*)**
Plots –ln(–ln(survival)) versus ln(analysis time) for each level of the categorical variable *sex*, from the previous **stcox** model. Roughly parallel curves support the Cox model assumption that the hazard ratio does not change with time. Other checks on the Cox assumptions are performed by the commands **stcoxkm** (compares Cox predicted curves with Kaplan–Meier observed survival curves) and **stphtest** (performs test based on Schoenfeld residuals). See **help stcox** for syntax and options.

. **streg *x1 x2*, dist(weibull)**
Fits Weibull-distribution model regression of time-to-failure on continuous or dummy variable predictors *x1* and *x2*.

. **streg *x1 x2 x3 x4*, dist(exponential) robust**
Fits exponential-distribution model regression of time-to-failure on continuous or dummy predictors *x1–x4*. Obtains heteroskedasticity-robust standard error estimates. In addition to Weibull and exponential, other **dist()** specifications for **streg** include lognormal, log-logistic, Gompertz, or generalized gamma distributions. Type **help streg** for more information.

. **stcurve, survival**
After **streg**, plots the survival function from this model at mean values of all the *x* variables.

. **stcurve, cumhaz at(*x3*=50, *x4*=0)**
After **streg**, plots the cumulative hazard function from this model at mean values of *x1* and *x2*, *x3* set at 50, and *x4* set at 0.

. **poisson *count x1 x2 x3*, irr exposure(*x4*)**
Performs Poisson regression of event-count variable *count* (assumed to follow a Poisson distribution) on continuous or dummy independent variables *x1–x3*. Independent-variable effects will be reported as incidence rate ratios (**irr**). The **exposure()** option identifies a variable indicating the amount of exposure, if this is not the same for all observations.

Note: A Poisson model assumes that the event probability remains constant, regardless of how many times an event occurs for each observation. If the probability does not remain constant, we should consider using **nbreg** (negative binomial regression) or **gnbreg** (generalized negative binomial regression) instead.

. **glm *count x1 x2 x3*, link(log) family(poisson) lnoffset(*x4*) eform**
Performs the same regression specified in the **poisson** example above, but as a generalized linear model (GLM). **glm** can fit Poisson, negative binomial, logit, and many other types of models, depending on what **link()** (link function) and **family()** (distribution family) options we employ.

Survival-Time Data

Survival-time data contain, at a minimum, one variable measuring how much time elapsed before a certain event occurred to each observation. The literature often terms this event of interest a "failure," regardless of its substantive meaning. When failure has not occurred to an observation by the time data collection ends, that observation is said to be "censored." The **stset** command sets up a dataset for survival-time analysis by identifying which variable measures time and (if necessary) which variable is a dummy indicating whether the observation failed or was censored. The dataset can also contain any number of other measurement or categorical variables, and individuals (for example, medical patients) can be represented by more than one observation.

To illustrate the use of **stset**, we will begin with an example from Selvin (1995:453) concerning 51 individuals diagnosed with HIV. The data initially reside in a raw-data file (*aids.raw*) that looks like this:

```
 1       1       1      34
 2      17       1      42
 3      37       0      47
    (rows 4–50 omitted)
51      81       0      29
```

The first column values are case numbers (1, 2, 3, . . . , 51). The second column tells how many months elapsed after the diagnosis, before that person either developed symptoms of AIDS or the study ended (1, 17, 37, . . .). The third column holds a 1 if the individual developed AIDS symptoms (failure), or a 0 if no symptoms had appeared by the end of the study (censoring). The last column reports the individual's age at the time of diagnosis.

We can read the raw data into memory using **infile**, then label the variables and data and save in Stata format as file *aids1.dta*:

```
. infile case time aids age using aids.raw, clear
(51 observations read)
. label variable case "Case ID number"
. label variable time "Months since HIV diagnosis"
. label variable aids "Developed AIDS symptoms"
. label variable age "Age in years"
```

```
. label data "AIDS (Selvin 1995:453)"

. compress
case was float now byte
time was float now byte
aids was float now byte
age was float now byte

. save aids1
file c:\data\aids1.dta saved
```

The next step is to identify which variable measures time and which indicates failure/censoring. Although not necessary with these single-record data, we can also note which variable holds individual case identification numbers. In an **stset** command, the first-named variable measures time. Subsequently, we identify with **failure()** the dummy representing whether an observation failed (1) or was censored (0). After using **stset**, we save the data again to preserve this information.

```
. stset time, failure(aids) id(case)

                id:  case
     failure event:  aids != 0 & aids < .
obs. time interval:  (time[_n-1], time]
 exit on or before:  failure

--------------------------------------------------------------------------------
       51  total obs.
        0  exclusions
--------------------------------------------------------------------------------
       51  obs. remaining, representing
       51  subjects
       25  failures in single failure-per-subject data
     3164  total analysis time at risk, at risk from t =         0
                             earliest observed entry t =         0
                                  last observed exit t =        97

. save, replace
file c:\data\aids1.dta saved
```

stdes yields a brief description of how our survival-time data are structured. In this simple example we have only one record per subject, so some of this information is unneeded.


```
. stdes
        failure _d:  aids
  analysis time _t:  time
                id:  case

                                |-------------- per subject --------------|
Category                  total        mean         min      median         max
--------------------------------------------------------------------------------
no. of subjects              51
no. of records               51           1           1           1           1

(first) entry time                        0           0           0           0
(final) exit time                  62.03922           1          67          97

subjects with gap             0
time on gap if gap            0           .           .           .           .
time at risk               3164    62.03922           1          67          97

failures                     25   .4901961           0           0           1
--------------------------------------------------------------------------------
```

The **stsum** command obtains summary statistics. We have 25 failures out of 3,164 person-months, giving an incidence rate of 25/3164 = .0079014. The percentiles of survival time derive from a Kaplan–Meier survivor function (next section). This function estimates about a 25% chance of developing AIDS within 41 months after diagnosis, and 50% within 81 months. Over the observed range of the data (up to 97 months) the probability of AIDS does not reach 75%, so there is no 75th percentile given.

```
. stsum

        failure _d:  aids
  analysis time _t:  time
                id:  case

                |                  incidence      no. of   |------ Survival time -----|
                | time at risk        rate       subjects       25%        50%        75%
----------------+---------------------------------------------------------------------
          total |         3164    .0079014            51         41         81          .
```

If the data happen to include a grouping or categorical variable such as *sex* (0 = male, 1 = female), we could obtain summary statistics on survival time separately for each group by a command of the following form:

```
. stsum, by(sex)
```

Later sections describe more formal methods for comparing survival times from two or more groups.

Count-Time Data

Survival-time (**st**) datasets like *aids1.dta* contain information on individual people or things, with variables indicating the time at which failure or censoring occurred for each individual. A different type of dataset called count-time (**ct**) contains aggregate data, with variables counting the number of individuals that failed or were censored at time *t*. For example, *diskdriv.dta* contains hypothetical test information on 25 disk drives. All but 5 drives failed before testing ended at 1,200 hours.

```
Contains data from C:\data\diskdriv.dta
  obs:             6                          Count-time data on disk drives
 vars:             3                          21 Jul 2003 09:34
 size:            48 (99.9% of memory free)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
hours           int    %8.0g                  Hours of continuous operation
failures        byte   %8.0g                  Number of failures observed
censored        byte   %8.0g                  Number still working
-------------------------------------------------------------------------------
Sorted by:
```

```
. list

     +-----------------------------+
     | hours   failures   censored |
     |-----------------------------|
  1. |   200          2          0 |
  2. |   400          3          0 |
  3. |   600          4          0 |
  4. |   800          8          0 |
  5. |  1000          3          0 |
     |-----------------------------|
  6. |  1200          0          5 |
     +-----------------------------+
```

To set up a count-time dataset, we specify the time variable, the number-of-failures variable, and the number-censored variable, in that order. After **ctset**, the **cttost** command automatically converts our count-time data to survival-time format.

```
. ctset hours failures censored

   dataset name:  C:\data\diskdriv.dta
           time:  hours
      no. fail:  failures
      no. lost:  censored
     no. enter:  --                        (meaning all enter at time 0)
```

```
. cttost

(data are now st)

     failure event:  failures != 0 & failures < .
obs. time interval:  (0, hours]
 exit on or before:  failure
           weight:  [fweight=w]

-------------------------------------------------------------------------------
        6  total obs.
        0  exclusions
-------------------------------------------------------------------------------
        6  physical obs. remaining, equal to
       25  weighted obs., representing
       20  failures in single record/single failure data
    19400  total analysis time at risk, at risk from t =         0
                                earliest observed entry t =         0
                                     last observed exit t =      1200
```



```
. list

     +-----------------------------------------------+
     | hours   failures   w   _st   _d     _t   _t0 |
     |-----------------------------------------------|
  1. |  1200          0   5    1    0   1200    0 |
  2. |   200          1   2    1    1    200    0 |
  3. |   400          1   3    1    1    400    0 |
  4. |   600          1   4    1    1    600    0 |
  5. |   800          1   8    1    1    800    0 |
     |-----------------------------------------------|
  6. |  1000          1   3    1    1   1000    0 |
     +-----------------------------------------------+
```

```
. stdes

        failure _d:  failures
  analysis time _t:  hours
            weight:  [fweight=w]

                                |-------------- per subject --------------|
                      unweighted  unweighted                     unweighted
Category                 total        mean         min      median        max
-------------------------------------------------------------------------------
no. of subjects              6
no. of records               6           1           1           1           1

(first) entry time                       0           0           0           0
(final) exit time                      700         200         700        1200

subjects with gap            0
time on gap if gap           0
time at risk              4200         700         200         700        1200

failures                     5   .8333333           0           1           1
-------------------------------------------------------------------------------
```

The **cttost** command defines a set of frequency weights, *w*, in the resulting **st**-format dataset. **st*** commands automatically recognize and use these weights in any survival-time analysis, so the data now are viewed as containing 25 observations (25 disk drives) instead of the previous 6 (six time periods).

```
. stsum

    failure time:  hours
  failure/censor:  failures
          weight:  [fweight=w]

       |                incidence       no. of   |---- Survival time ----|
       | time at risk       rate     subjects      25%       50%       75%
-------+-------------------------------------------------------------------
 total |        19400   .0010309           25      600       800      1000
```

Kaplan–Meier Survivor Functions

Let n_t represent the number of observations that have not failed, and are not censored, at the beginning of time period t. d_t represents the number of failures that occur to these observations during time period t. The Kaplan–Meier estimator of surviving beyond time t is the product of survival probabilities in t and the preceding periods:

$$S(t) = \prod_{j=1}^{t} \{(n_j - d_j)/n_j\} \qquad [11.1]$$

For example, in the AIDS data seen earlier, one of the 51 individuals developed symptoms only one month after diagnosis. No observations were censored this early, so the probability of "surviving" (meaning, not developing AIDS) beyond *time* = 1 is

$$S(1) = (51 - 1)/51 = .9804$$

A second patient developed symptoms at *time* = 2, and a third at *time* = 9:

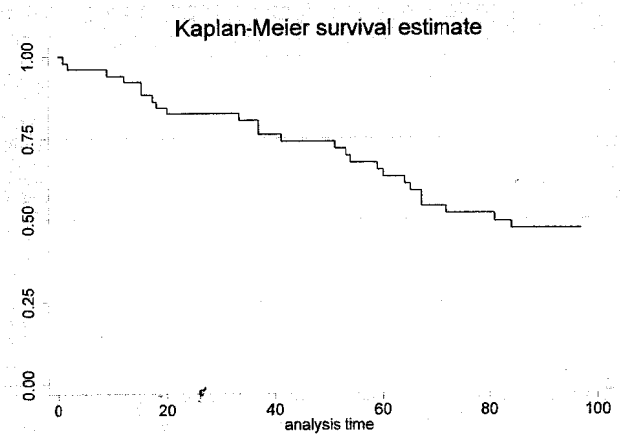
$$S(2) = .9804 \times (50 - 1)/50 = .9608$$

$$S(9) = .9608 \times (49 - 1)/49 = .9412$$

Graphing $S(t)$ against t produces a Kaplan–Meier survivor curve, like the one seen in Figure 11.1. Stata draws such graphs automatically with the **sts graph** command. For example,

```
. use aids, clear
(AIDS (Selvin 1995:453))

. sts graph

        failure _d:  aids
  analysis time _t:  time
                id:  case
```

Figure 11.1



For a second example of survivor functions, we turn to data in *smoking1.dta*, adapted from Rosner (1995). The observations are 234 former smokers, attempting to quit. Most did not succeed. Variable *days* records how many days elapsed between quitting and starting up again. The study lasted one year, and variable *smoking* indicates whether an individual resumed

smoking before the end of this study (*smoking* = 1, "failure") or not (*smoking* = 0, "censored"). With new data, we should begin by using **stset** to set the data up for survival-time analysis:

```
Contains data from C:\data\smoking1.dta
  obs:           234                          Smoking (Rosner 1995:607)
 vars:             8                          21 Jul 2003 09:35
 size:         3,744 (99.9% of memory free)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
id              int    %9.0g                  Case ID number
days            int    %9.0g                  Days abstinent
smoking         byte   %9.0g                  Resumed smoking
age             byte   %9.0g                  Age in years
sex             byte   %9.0g       sex        Sex (female)
cigs            byte   %9.0g                  Cigarettes per day
co              int    %9.0g                  Carbon monoxide x 10
minutes         int    %9.0g                  Minutes elapsed since last cig
-------------------------------------------------------------------------------
Sorted by:

. stset days, failure(smoking)

     failure event:  smoking != 0 & smoking < .
obs. time interval:  (0, days]
 exit on or before:  failure

-------------------------------------------------------------------------------
      234  total obs.
        0  exclusions
-------------------------------------------------------------------------------
      234  obs. remaining, representing
      201  failures in single record/single failure data
    18946  total analysis time at risk, at risk from t =         0
                              earliest observed entry t =         0
                                   last observed exit t =       366
```

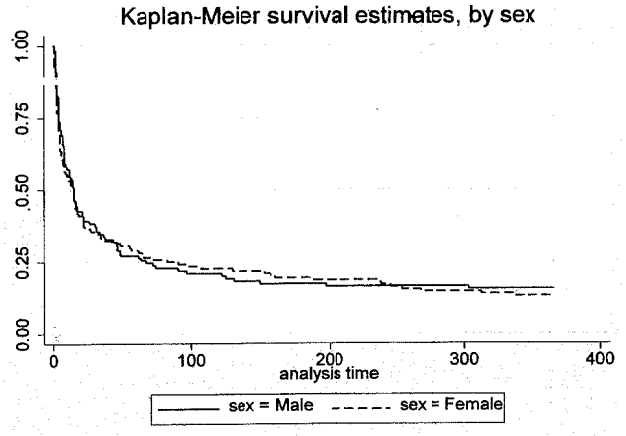
The study involved 110 men and 124 women. Incidence rates for both sexes appear to be similar:

```
. stsum, by(sex)

        failure _d:  smoking
  analysis time _t:  days

          |                  incidence       no. of   |------ Survival time -----|
sex       | time at risk        rate        subjects        25%        50%        75%
----------+--------------------------------------------------------------------------
     Male |         8813   .0105526             110          4         15         68
   Female |        10133   .0106582             124          4         15         91
----------+--------------------------------------------------------------------------
    total |        18946   .0106091             234          4         15         73
```

Figure 11.2 confirms this similarity, showing little difference between the survivor functions of men and women. That is, both sexes returned to smoking at about the same rate. The survival probabilities of nonsmokers decline very steeply during the first 30 days after quitting. For either sex, there is less than a 15% chance of surviving beyond a full year.

```
. sts graph, by(sex)

        failure _d:  smoking
  analysis time _t:  days
```

Figure 11.2



We can also formally test for the equality of survivor functions using a log-rank test. Unsurprisingly, this test finds no significant difference (P = .6772) between the smoking recidivism of men and women.

```
. sts test sex

        failure _d:  smoking
  analysis time _t:  days

Log-rank test for equality of survivor functions

       |  Events       Events
sex    |  observed     expected
-------+-------------------------
Male   |        93        95.88
Female |       108       105.12
-------+-------------------------
Total  |       201       201.00

           chi2(1) =       0.17
           Pr>chi2 =     0.6772
```


Cox Proportional Hazard Models

Regression methods allow us to take survival analysis further and examine the effects of multiple continuous or categorical predictors. One widely-used method known as Cox regression employs a proportional hazard model. The hazard rate for failure at time t is defined as

$$h(t) = \frac{\text{probability of failing between times } t \text{ and } t+\Delta t}{(\Delta t)\,(\text{probability of failing after time } t)} \qquad [11.2]$$

We model this hazard rate as a function of the baseline hazard (h_0) at time t, and the effects of one or more x variables,

$$h(t) = h_0(t)\exp(\beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k) \qquad [11.3a]$$

or, equivalently,

$$\ln[h(t)] = \ln[h_0(t)] + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k \qquad [11.3b]$$

"Baseline hazard" means the hazard for an observation with all x variables equal to 0. Cox regression estimates this hazard nonparametrically and obtains maximum-likelihood estimates of the β parameters in [11.3]. Stata's **stcox** procedure ordinarily reports hazard ratios, which are estimates of exp(β). These indicate proportional changes relative to the baseline hazard rate.

Does age affect the onset of AIDS symptoms? Dataset *aids.dta* contains information that helps answer this question. Note that with **stcox**, unlike most other Stata model-fitting commands, we list only the independent variable(s). The survival-analysis dependent variables, timevariables, and censoring variables are understood automatically with **stset** data.

```
. use aids
(AIDS (Selvin 1995:453))

. stcox age, nolog

        failure _d:  aids
  analysis time _t:  time
             id:  case

Cox regression -- Breslow method for ties

No. of subjects =           51                     Number of obs   =        51
No. of failures =           25
Time at risk    =         3164
                                                   LR chi2(1)      =      5.00
Log likelihood  =   -86.576295                     Prob > chi2     =    0.0254

------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |   1.084557   .0378623     2.33   0.020      1.01283    1.161363
------------------------------------------------------------------------------
```

We might interpret the estimated hazard ratio, 1.084557, with reference to two HIV-positive individuals whose ages are a and a + 1. The older person is 8.5% more likely to develop AIDS symptoms over a short period of time (that is, the ratio of their respective hazards

is 1.084557). This ratio differs significantly ($P = .020$) from 1. If we wanted to state our findings for a five-year difference in age, we could raise the hazard ratio to the fifth power:

```
. display exp(_b[age])^5
1.5005865
```

Thus, the hazard of AIDS onset is about 50% higher when the second person is five years older than the first. Alternatively, we could learn the same thing (and obtain the new confidence interval) by repeating the regression after creating a new version of *age* measured in five-year units. The **nolog noshow** options below suppress display of the iteration log and the **st**-dataset description.

```
. generate age5 = age/5
. label variable age5 "age in 5-year units"
. stcox age5, nolog noshow
```

```
Cox regression -- Breslow method for ties

No. of subjects =           51                     Number of obs    =        51
No. of failures =           25
Time at risk    =         3164
                                                   LR chi2(1)       =      5.00
Log likelihood  =   -86.576295                     Prob > chi2      =    0.0254

------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        age5 |   1.500587   .2619305     2.33   0.020     1.065815    2.112711
------------------------------------------------------------------------------
```

Like ordinary regression, Cox models can have more than one independent variable. Dataset *heart.dta* contains survival-time data from Selvin (1995) on 35 patients with very high cholesterol levels. Variable *time* gives the number of days each patient was under observation. *coronary* indicates whether a coronary event occurred during this time (*coronary* = 1) or not (*coronary* = 0). The data also include cholesterol levels and other factors thought to affect heart disease. File *heart.dta* was previously set up for survival-time analysis by an **stset *time*, failure(*coronary*)** command, so we can go directly to **st** analysis.

```
. describe patient - ab
```

```
              storage  display     value
variable name   type   format      label      variable label
------------------------------------------------------------------------------
patient         byte   %9.0g                  Patient ID number
time            int    %9.0g                  Time in days
coronary        byte   %9.0g                  Coronary event (1) or none (0)
weight          int    %9.0g                  Weight in pounds
sbp             int    %9.0g                  Systolic blood pressure
chol            int    %9.0g                  Cholesterol level
cigs            byte   %9.0g                  Cigarettes smoked per day
ab              byte   %9.0g                  Type A (1) or B (0) personality
```



```
. stdes
```

```
        failure _d:  coronary
  analysis time _t:  time

                              |-------------- per subject --------------|
Category                total         mean         min      median         max
---------------------------------------------------------------------------
no. of subjects            35
no. of records             35            1           1           1           1

(first) entry time                       0           0           0           0
(final) exit time                2580.629         773        2875        3141

subjects with gap           0
time on gap if gap          0
time at risk            90322     2580.629         773        2875        3141

failures                    8    .2285714           0           0           1
---------------------------------------------------------------------------
```

Cox regression finds that cholesterol level and cigarettes both significantly increase the hazard of a coronary event. Counterintuitively, weight appears to decrease the hazard. Systolic blood pressure and A/B personality do not have significant net effects.

```
. stcox weight sbp chol cigs ab, noshow nolog
```

```
Cox regression -- no ties

No. of subjects =           35                     Number of obs    =        35
No. of failures =            8
Time at risk    =        90322
                                                   LR chi2(5)       =     13.97
Log likelihood  =   -17.263231                     Prob > chi2      =    0.0158

------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      weight |   .9349336   .0305184    -2.06   0.039     .8769919    .9967034
         sbp |   1.012947   .0338061     0.39   0.700     .9488087    1.081421
        chol |   1.032142   .0139984     2.33   0.020     1.005067    1.059947
        cigs |   1.203335   .1071031     2.08   0.038     1.010707    1.432676
          ab |    3.04969   2.985616     1.14   0.255     .4476492    20.77655
------------------------------------------------------------------------------
```

After estimating the model, **stcox** can also generate new variables holding the estimated baseline cumulative hazard and survivor functions. Since "baseline" refers to a situation with all x variables equal to zero, however, we first need to recenter some variables so that 0 values make sense. A patient who weighs 0 pounds, or has 0 blood pressure, does not provide a useful comparison. Guided by the minimum values actually in our data, we might shift *weight* so that 0 indicates 120 pounds, *sbp* so that 0 indicates 100, and *chol* so that 0 indicates 340:

```
. summarize patient - ab
```

```
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
     patient |        35          18    10.24695         1         35
        time |        35    2580.629    616.0796       773       3141
    coronary |        35   .2285714     .426043         0          1
      weight |        35    170.0857    23.55516       120        225
         sbp |        35    129.7143    14.28403       104        154
-------------+--------------------------------------------------------
```

```
       chol |        35    369.2857    51.32284         343         645
       cigs |        35     17.4286    13.07702           0          40
         ab |        35    .5142857    .5070926           0           1

. replace weight = weight - 120
(35 real changes made)

. replace sbp = sbp - 100
(35 real changes made)

. replace chol = chol - 340
(35 real changes made)

. summarize patient - ab

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
     patient |         35          18    10.24695         1         35
        time |         35    2580.629    616.0796       773       3141
    coronary |         35    .2285714     .426043         0          1
      weight |         35    50.08571    23.55516         0        105
         sbp |         35    29.71429    14.28403         4         54
-------------+--------------------------------------------------------
        chol |         35    29.28571    51.32284         3        305
        cigs |         35     17.4286    13.07702         0         40
          ab |         35    .5142857    .5070926         0          1
```

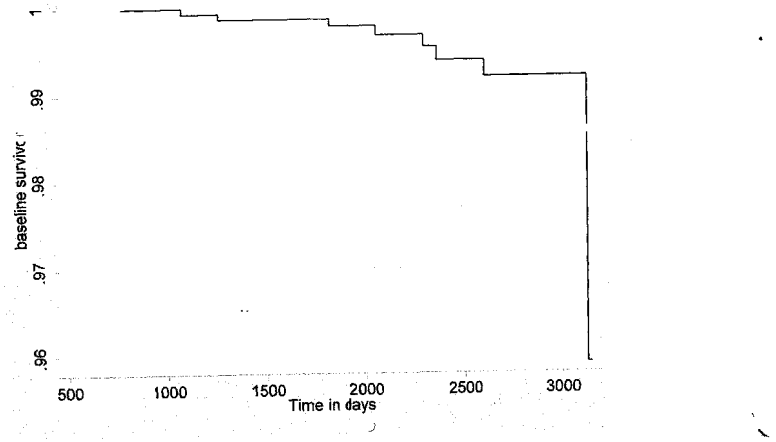
Zero values for all the *x* variables now make more substantive sense. To create new variables holding the baseline survivor and cumulative hazard function estimates, we repeat the regression with **basesurv()** and **basechaz()** options:

```
. stcox weight sbp chol cigs ab, noshow nolog basesurv(survivor)
      basechaz(hazard)

Cox regression -- no ties

No. of subjects =           35                     Number of obs   =        35
No. of failures =            8
Time at risk    =        90322

                                                   LR chi2(5)      =     13.97
Log likelihood  =   -17.263231                     Prob > chi2     =    0.0158

------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      weight |   .9349336   .0305184    -2.06   0.039     .8769919    .9967034
         sbp |   1.012947   .0338061     0.39   0.700     .9488087    1.081421
        chol |   1.032142   .0139984     2.33   0.020     1.005067    1.059947
        cigs |   1.203335   .1071031     2.08   0.038     1.010707    1.432676
          ab |    3.04969   2.985616     1.14   0.255     .4476492    20.77655
------------------------------------------------------------------------------
```

Note that recentering three *x* variables had no effect on the hazard ratios, standard errors, and so forth. The command created two new variables, arbitrarily named *survivor* and *hazard*. To graph the baseline survivor function, we plot *survivor* against *time* and connect data points with in a stairstep fashion, as seen in Figure 11.3.


```
. graph twoway line survivor time, connect(stairstep) sort
```

Figure 11.3



The baseline survivor function — which depicts survival probabilities for patients having "0" weight (120 pounds), "0" blood pressure (100), "0" cholesterol (340), 0 cigarettes per day, and a type B personality — declines with time. Although this decline looks precipitous at the right, notice that the probability really only falls from 1 to about .96. Given less favorable values of the predictor variables, the survival probabilities would fall much faster.

The same baseline survivor-function graph could have been obtained another way, without **stcox**. The alternative, shown in Figure 11.4, employs an **sts graph** command with **adjustfor()** option listing the predictor variables:

```
. sts graph, adjustfor(weight sbp chol cigs ab)

        failure _d:  coronary
  analysis time _t:  time
```

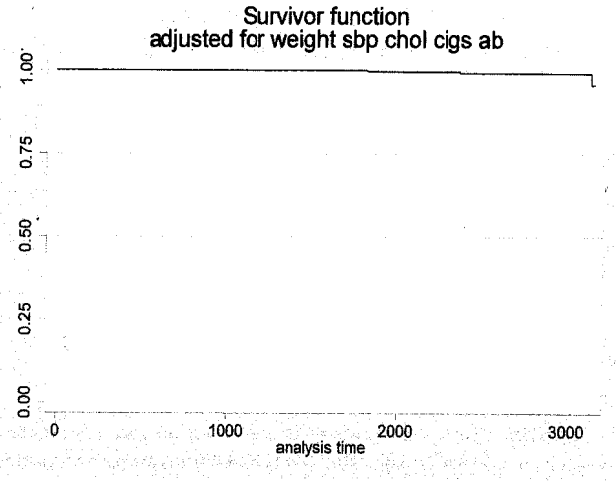


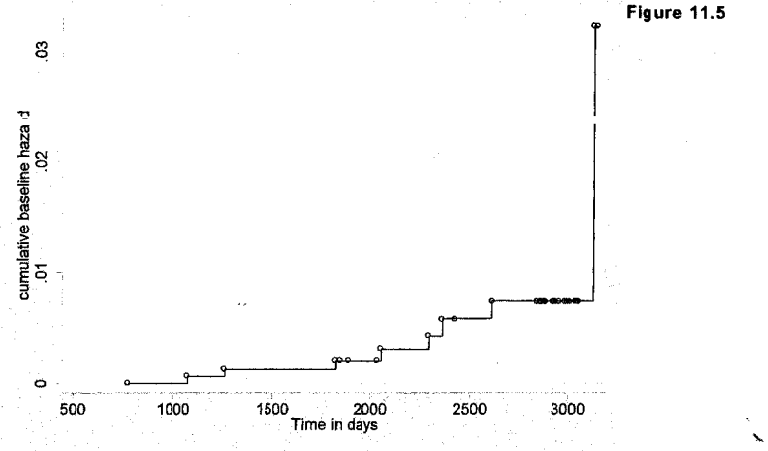
Figure 11.4

Figure 11.4, unlike Figure 11.3, follows the usual survivor-function convention of scaling the vertical axis from 0 to 1. Apart from this difference in scaling, Figures 11.3 and 11.4 depict the same curve.

Figure 11.5 graphs the estimated baseline cumulative hazard against time, using the variable (*hazard*) generated by our **stcox** command. This graph shows the baseline cumulative hazard increasing in 8 steps (because 8 patients "failed" or had coronary events), from near 0 to .033.


```
. graph twoway connected hazard time, connect(stairstep) sort
      msymbol(Oh)
```

Figure 11.5



Exponential and Weibull Regression

Cox regression estimates the baseline survivor function empirically without reference to any theoretical distribution. Several alternative "parametric" approaches begin instead from assumptions that survival times do follow a known theoretical distribution. Possible distribution families include the exponential, Weibull, lognormal, log-logistic, Gompertz, or generalized gamma. Models based on any of these can be fit through the **streg** command. Such models have the same general form as Cox regression (equations [11.2] and [11.3]), but define the baseline hazard $h_0(t)$ differently. Two examples appear in this section.

If failures occur randomly, with a constant hazard, then survival times follow an exponential distribution and could be analyzed by *exponential regression*. Constant hazard means that the individuals studied do not "age," in the sense that they are no more or less likely to fail late in the period of observation than they were at its start. Over the long term, this assumption seems unjustified for machines or living organisms, but it might approximately hold if the period of observation covers a relatively small fraction of their life spans. An exponential model implies that logarithms of the survivor function, $\ln(S(t))$, are linearly related to t.

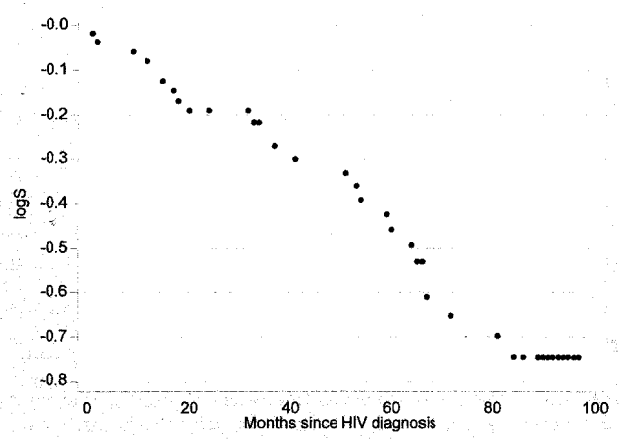
A second common parametric approach, *Weibull regression*, is based on the more general Weibull distribution. This does not require failure rates to remain constant, but allows them to increase or decrease smoothly over time. The Weibull model implies that $\ln(-\ln(S(t)))$ is a linear function of $\ln(t)$.

Graphs provide a useful diagnostic for the appropriateness of exponential or Weibull models. For example, returning to *aids.dta*, we construct a graph (Figure 11.6) of $\ln(S(t))$ versus time, after first generating Kaplan–Meier estimates of the survivor function $S(t)$. The

y-axis labels in Figure 11.6 are given a fixed two-digit, one-decimal display format (%2.1f) and oriented horizontally, to improve their readability.

```
. use aids, clear
(AIDS (Selvin 1995:453))
. sts gen S = S
. generate logS = ln(S)
. graph twoway scatter logS time,
      ylabel(-.8(.1)0, format(%2.1f) angle(horizontal))
```

Figure 11.6



The pattern in Figure 11.6 appears somewhat linear, encouraging us to try an exponential regression:

```
. streg age, dist(exponential) nolog noshow

Exponential regression -- log relative-hazard form

No. of subjects =           51                     Number of obs   =        51
No. of failures =           25
Time at risk    =         3164
                                                   LR chi2(1)      =      4.34
Log likelihood  =   -59.996976                     Prob > chi2     =    0.0372

------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |   1.074414   .0349626     2.21   0.027     1.008028    1.145172
------------------------------------------------------------------------------
```

The hazard ratio (1.074) and standard error (.035) estimated by this exponential regression do not greatly differ from their counterparts (1.085 and .038) in our earlier Cox regression. The similarity reflects the degree of correspondence between empirical and exponential hazard

functions. According to this exponential model, the hazard of an HIV-positive individual developing AIDS increases about 7.4% with each year of age.

After **streg**, the **stcurve** command draws a graph of the models' cumulative hazard, survival, or hazard functions. By default, **stcurve** draws these curves holding all *x* variables in the model at their means. We can specify other *x* values by using the **at()** option. The individuals in *aids.dta* ranged from 26 to 50 years old. We could graph the survival function at *age* = 26 by issuing a command such as

```
. stcurve, survival at(age=26)
```

A more informative graph uses the **at1()** and **at2()** options to show the survival curve at two different sets of *x* values, such as the low and high extremes of *age*:

```
. stcurve, survival at1(age=26) at2(age=50) connect(direct direct)
```

Figure 11.7

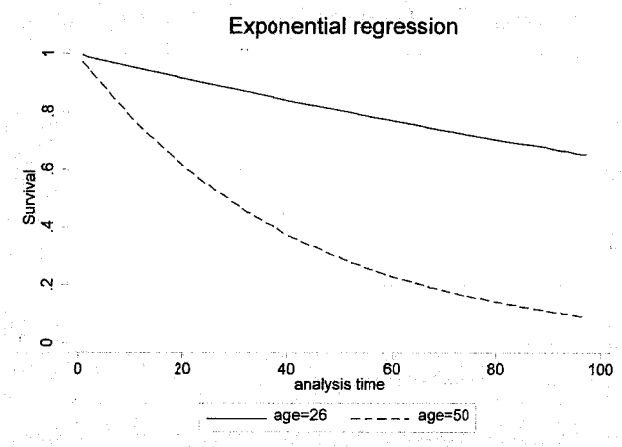


Figure 11.7 shows the predicted survival curve (for transition from HIV diagnosis to AIDS) falling more steeply among older patients. The significant *age* hazard ratio greater than 1 in our exponential regression table implied the same thing, but using **stcurve** with **at1()** and **at2()** values gives a strong visual interpretation of this effect. These options work in a similar manner with all three types of **stcurve** graphs:

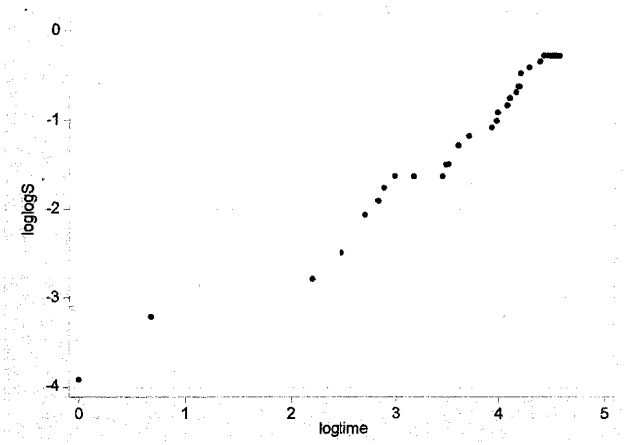
stcurve, survival	Survival function.
stcurve, hazard	Hazard function.
stcurve, cumhaz	Cumulative hazard function.

Instead of the exponential distribution, **streg** can also fit survival models based on the Weibull distribution. A Weibull distribution might appear curvilinear in a plot of $\ln(S(t))$ versus t, but it should be linear in a plot of $\ln(-\ln(S(t)))$ versus $\ln(t)$, such as Figure 11.8. An exponential distribution, on the other hand, will appear linear in both plots and have a slope

equal to 1 in the ln(–ln(*S*(*t*))) versus ln(*t*) plot. In fact, the data points in Figure 11.8 are not far from a line with slope 1, suggesting that our previous exponential model is adequate.

```
. generate loglogS = ln(-ln(S))
. generate logtime = ln(time)
. graph twoway scatter loglogS logtime, ylabel(,angle(horizontal))
```

Figure 11.8



Although we do not need the additional complexity of a Weibull model with these data, results are given below for illustration.

```
. streg age, dist(weibull) noshow nolog

Weibull regression -- log relative-hazard form

No. of subjects =           51                 Number of obs   =         51
No. of failures =           25
Time at risk    =         3164
                                               LR chi2(1)      =       4.68
Log likelihood  =   -59.778257                 Prob > chi2     =     0.0306

------------------------------------------------------------------------------
          _t | Haz. Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |   1.079477   .0363509     2.27   0.023     1.010531    1.153127
-------------+----------------------------------------------------------------
       /ln_p |   .1232638   .1820858     0.68   0.498    -.2336179    .4801454
-------------+----------------------------------------------------------------
           p |   1.131183   .2059723                      .7916643    1.616309
         1/p |   .8840305   .1609694                      .6186934    1.263162
------------------------------------------------------------------------------
```

The Weibull regression obtains a hazard ratio estimate (1.079) intermediate between our previous Cox and exponential results. The most noticeable difference from those earlier models is the presence of three new lines at the bottom of the table. These refer to the Weibull distribution shape parameter *p*. A *p* value of 1 corresponds to an exponential model: the hazard

does not change with time. $p > 1$ indicates that the hazard increases with time; $p < 1$ indicates that the hazard decreases. A 95% confidence interval for *p* ranges from .79 to 1.62, so we have no reason to reject an exponential ($p = 1$) model here. Different, but mathematically equivalent, parameterizations of the Weibull model focus on ln(*p*), *p*, or 1/*p*, so Stata provides all three. **stcurve** draws survival, hazard, or cumulative hazard functions after **streg, dist(weibull)** just as it does after **streg, dist(exponential)** or other **streg** models.

Exponential or Weibull regression is preferable to Cox regression when survival times actually follow an exponential or Weibull distribution. When they do not, these models are misspecified and can yield misleading results. Cox regression, which makes no *a priori* assumptions about distribution shape, remains useful in a wider variety of situations.

In addition to exponential and Weibull models, **streg** can fit models based on the Gompertz, lognormal, log-logistic, or generalized gamma distributions. Type **help streg**, or consult the *Survival Analysis and Epidemiological Tables Reference Manual*, for syntax and a list of current options.

Poisson Regression

If events occur independently and with constant probability, then counts of events over a given period of time follow a Poisson distribution. Let r_j represent the incidence rate:

$$r_j = \frac{\text{count of events}}{\text{number of times event could have occurred}} \qquad [11.4]$$

The denominator in [11.4] is termed the "exposure" and is often measured in units such as person-years. We model the logarithm of incidence rate as a linear function of one or more predictor (x) variables:

$$\ln(r_j) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_k x_k \qquad [11.5a]$$

Equivalently, the model describes logs of expected event counts:

$$\ln(\textit{expected count}) = \ln(\textit{exposure}) + \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots \beta_k x_k \qquad [11.5b]$$

Assuming that a Poisson process underlies the events of interest, Poisson regression finds maximum-likelihood estimates of the β parameters.

Data on radiation exposure and cancer deaths among workers at Oak Ridge National Laboratory provide an example. The 56 observations in dataset *oakridge.dta* represent 56 age/radiation-exposure categories (7 categories of age × 8 categories of radiation). For each combination, we know the number of deaths and the number of person-years of exposure.

```
Contains data from C:\data\oakridge.dta
  obs:            56                          Radiation (Selvin 1995:474)
 vars:             4                          21 Jul 2003 09:34
 size:           616 (99.9% of memory free)
------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
------------------------------------------------------------------------------
age             byte   %9.0g       ageg       Age group
rad             byte   %9.0g                  Radiation exposure level
```

```
deaths          byte    %9.0g                 Number of deaths
pyears          float   %9.0g                 Person-years
------------------------------------------------------------------------------
Sorted by:
```

```
. summarize
```

```
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
         age |        56           4       2.0181         1          7
         rad |        56         4.5     2.312024         1          8
      deaths |        56    1.839286     3.178203         0         16
      pyears |        56    3807.679     10455.91        23      71382
```

```
. list in 1/6
```

```
     +-------------------------------+
     |   age   rad   deaths   pyears |
     |-------------------------------|
  1. |  < 45     1        0    29901 |
  2. | 45-49     1        1     6251 |
  3. | 50-54     1        4     5251 |
  4. | 55-59     1        3     4126 |
  5. | 60-64     1        3     2778 |
     |-------------------------------|
  6. | 65-69     1        1     1607 |
     +-------------------------------+
```

Does the death rate increase with exposure to radiation? Poisson regression finds a statistically significant effect:

```
. poisson deaths rad, nolog exposure(pyears) irr
```

```
Poisson regression                                Number of obs   =         56
                                                  LR chi2(1)      =      14.87
                                                  Prob > chi2     =     0.0001
Log likelihood =  -169.7364                       Pseudo R2       =     0.0420

------------------------------------------------------------------------------
      deaths |        IRR   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         rad |   1.236469   .0603551     4.35   0.000     1.123657    1.360606
      pyears |  (exposure)
------------------------------------------------------------------------------
```

For the regression above, we specified the event count (*deaths*) as the dependent variable and radiation (*rad*) as the independent variable. The Poisson "exposure" variable is *pyears*, or person-years in each category of *rad*. The **irr** option calls for incidence rate ratios rather than regression coefficients in the results table — that is, we get estimates of exp(β) instead of β, the default. According to this incidence rate ratio, the death rate becomes 1.236 times higher (increases by 23.6%) with each increase in radiation category. Although that ratio is statistically significant, the fit is not impressive; the pseudo R^2 (see equation [10.4]) is only .042.

To perform a goodness-of-fit test, comparing the Poisson model's predictions with the observed counts, use the follow-up command **poisgof**:


```
. poisgof
```

```
      Goodness-of-fit chi2  =  254.5475
      Prob > chi2(54)       =    0.0000
```

These goodness-of-fit test results ($\chi^2 = 254.5, P < .00005$) indicate that our model's predictions are significantly different from the actual counts — another sign that the model fits poorly.

We obtain better results when we include *age* as a second predictor. Pseudo R^2 then rises to .5966, and the goodness-of-fit test no longer leads us to reject our model.

```
. poisson deaths rad age, nolog exposure(pyears) irr
```

```
Poisson regression                                Number of obs   =         56
                                                  LR chi2(2)      =     211.41
                                                  Prob > chi2     =     0.0000
Log likelihood =  -71.4653                        Pseudo R2       =     0.5966

------------------------------------------------------------------------------
      deaths |        IRR   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         rad |   1.176673   .0593445     3.23   0.001     1.065924    1.298929
         age |   1.960034   .0997535    13.22   0.000     1.773955    2.165631
      pyears |  (exposure)
------------------------------------------------------------------------------
```

```
. poisgof
```

```
      Goodness-of-fit chi2  =  58.00534
      Prob > chi2(53)       =   0.2960
```

For simplicity, to this point we have treated *rad* and *age* as if both were continuous variables, and we expect their effects on the log death rate to be linear. In fact, however, both independent variables are measured as ordered categories. *rad* = 1, for example, means 0 radiation exposure; *rad* = 2 means 0 to 19 milliseiverts; *rad* = 3 means 20 to 39 milliseiverts; and so forth. An alternative way to include radiation exposure categories in the regression, while watching for nonlinear effects, is as a set of dummy variables. Below we use the **gen()** option of **tabulate** to create 8 dummy variables, *r1* to *r8*, representing each of the 8 values of *rad*.

```
. tabulate rad, gen(r)
```

```
  Radiation |
   exposure |
      level |      Freq.     Percent        Cum.
------------+-----------------------------------
          1 |          7       12.50       12.50
          2 |          7       12.50       25.00
          3 |          7       12.50       37.50
          4 |          7       12.50       50.00
          5 |          7       12.50       62.50
          6 |          7       12.50       75.00
          7 |          7       12.50       87.50
          8 |          7       12.50      100.00
------------+-----------------------------------
      Total |         56      100.00
```

```
. describe
```

We might represent a "generic" **glm** command as follows:

```
. glm y x1 x2 x3, family(familyname) link(linkname)
    lnoffset(exposure) eform jknife
```

where **family()** specifies the y distribution family, **link()** the link function, and **lnoffset()** an "exposure" variable such as that needed for Poisson regression. The **eform** option asks for regression coefficients in exponentiated form, exp(β) rather than β. Standard errors are estimated through jackknife (**jknife**) calculations.

Possible distribution families are

family(gaussian)	Gaussian or normal (default)
family(igaussian)	Inverse Gaussian
family(binomial)	Bernoulli binomial
family(poisson)	Poisson
family(nbinomial)	Negative binomial
family(gamma)	Gamma

We can also specify a number or variable indicating the binomial denominator N (number of trials), or a number indicating the negative binomial variance and deviance functions, by declaring them in the **family()** option:

```
family(binomial #)
family(binomial varname)
family(nbinomial #)
```

Possible link functions are

link(identity)	Identity (default)
link(log)	Log
link(logit)	Logit
link(probit)	Probit
link(cloglog)	Complementary log-log
link(opower #)	Odds power
link(power #)	Power
link(nbinomial)	Negative binomial
link(loglog)	Log-log
link(logc)	Log-complement

Coefficient variances or standard errors can be estimated in a variety of ways. A partial list of **glm** variance-estimating options is given below:

opg	Berndt, Hall, Hall, and Hausman "B-H-cubed" variance estimator.
oim	Observed information matrix variance estimator.
robust	Huber/White/sandwich estimator of variance.
unbiased	Unbiased sandwich estimator of variance

nwest	Heteroskedasticity and autocorrelation-consistent variance estimator.
jknife	Jackknife estimate of variance.
jknife1	One-step jackknife estimate of variance.
bstrap	Bootstrap estimate of variance. The default is 199 repetitions; specify some other number by adding the **bsrep(#)** option.

For a full list of options with some technical details, look up **glm** in the *Reference Manuals*. A more in-depth treatment of GLM topics can be found in Hardin and Hilbe (2001).

Chapter 6 began with the simple regression of mean composite SAT scores (*csat*) on per-pupil expenditures (*expense*) of the 50 U.S. states and District of Columbia (*states.dta*):

```
. regress csat expense
```

We could fit the same model and obtain exactly the same estimates with the following command:

```
. glm csat expense, link(identity) family(gaussian)

Iteration 0:   log likelihood = -279.99869

Generalized linear models                         No. of obs      =         51
Optimization     : ML: Newton-Raphson             Residual df     =         49
                                                  Scale param     =   3577.678
Deviance         =  175306.2097                   (1/df) Deviance =   3577.678
Pearson          =  175306.2097                   (1/df) Pearson  =   3577.678

Variance function: V(u) = 1                       [Gaussian]
Link function    : g(u) = u                       [Identity]
Standard errors  : OIM

Log likelihood   = -279.9986936                   AIC             =  11.05877
BIC              =   175298.346

------------------------------------------------------------------------------
        csat |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     expense |  -.0222755   .0060371    -3.69   0.000    -.0341082   -.0104431
       _cons |   1060.732    32.7009    32.44   0.000     996.6399    1124.825
------------------------------------------------------------------------------
```

Because **link(identity)** and **family(gaussian)** are default options, we could actually have left them out of the previous **glm** command.

The **glm** command can do more than just duplicate our **regress** results, however. For example, we could fit the same OLS model but obtain bootstrap standard errors:

```
. glm csat expense, link(identity) family(gaussian) bstrap

Iteration 0:   log likelihood = -279.99869

Bootstrap iterations (199)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
.................................................. 50
.................................................. 100
.................................................. 150
.................................................

Generalized linear models                         No. of obs      =         51
Optimization     : ML: Newton-Raphson             Residual df     =         49
                                                  Scale param     =    4124.656
Deviance         =  175306.2097                   (1/df) Deviance =    3577.678
Pearson          =  175306.2097                   (1/df) Pearson  =    3577.678

Variance function: V(u) = 1                       [Gaussian]
Link function    : g(u) = u                       [Identity]
Standard errors  : Bootstrap

Log likelihood   = -279.9986936                   AIC             =   11.05877
BIC              =   175298.346

------------------------------------------------------------------------------
             |               Bootstrap
        csat |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     expense |  -.0222756   .0039284    -5.67   0.000    -.0299751   -.0145762
       _cons |   1060.732   25.36566    41.82   0.000     1011.017    1110.448
------------------------------------------------------------------------------
```

The bootstrap standard errors reflect observed variation among coefficients estimated from 199 samples of $n = 51$ cases each, drawn by random sampling with replacement from the original $n = 51$ dataset. In this example, the bootstrap standard errors are less than the corresponding theoretical standard errors, and the resulting confidence intervals are narrower.

Similarly, we could use **glm** to repeat the first **logistic** regression of Chapter 10. In the following example, we ask for jackknife standard errors and odds ratio or exponential-form (**eform**) coefficients:


```
. glm any date, link(logit) family(bernoulli) eform jknife

Iteration 0:   log likelihood = -12.995268
Iteration 1:   log likelihood = -12.991098
Iteration 2:   log likelihood = -12.991096

Jackknife iterations (23)
----+--- 1 ---+--- 2 ---+--- 3 ---+--- 4 ---+--- 5
.......................

Generalized linear models                         No. of obs      =         23
Optimization     : ML: Newton-Raphson             Residual df     =         21
                                                  Scale param     =          1
Deviance         =  25.98219269                   (1/df) Deviance =   1.237247
Pearson          =   22.8885488                   (1/df) Pearson  =   1.089931

Variance function: V(u) = u*(1-u)                 [Bernoulli]
Link function    : g(u) = ln(u/(1-u))             [Logit]
Standard errors  : Jackknife

Log likelihood   = -12.99109634                   AIC             =   1.303574
BIC              =  19.71120426

------------------------------------------------------------------------------
             |               Jackknife
         any | Odds Ratio   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        date |   1.002093   .0015486     1.35   0.176     .9990623    1.005133
------------------------------------------------------------------------------
```

The final **poisson** regression of the present chapter corresponds to this **glm** model:

```
. glm deaths r2-r6 r78 age, link(log) family(poisson)
      lnoffset(pyears) eform
```

Although **glm** can replicate the models fit by many specialized commands, and adds some new capabilities, the specialized commands have their own advantages including speed and customized options. A particular attraction of **glm** is its ability to fit models for which Stata has no specialized command.