

The Stata Journal (2012)
12, Number 2, pp. 257–283

Threshold regression for time-to-event analysis: The `stthreg` package

Tao Xiao
The Ohio State University
Columbus, OH
xiao.51@osu.edu

G. A. Whitmore
McGill University
Montreal, Canada
george.whitmore@mcgill.ca

Xin He
University of Maryland
College Park, MD
xinhe@umd.edu

Mei-Ling Ting Lee
University of Maryland
College Park, MD
mltlee@umd.edu

Abstract. In this article, we introduce the `stthreg` package of Stata commands to fit the threshold regression model, which is based on the first hitting time of a boundary by the sample path of a Wiener diffusion process and is well suited to applications involving time-to-event and survival data. The threshold regression model serves as an important alternative to the Cox proportional hazards model. The four commands that comprise this package for the threshold regression model are the model-fitting command `stthreg`, the postestimation command `trhr` for hazard-ratio calculation, the postestimation command `trpredict` for prediction, and the model diagnostics command `sttrkm`. These commands can also be used to implement an extended threshold regression model that accommodates applications where a cure rate exists.

Keywords: `st0257`, `stthreg`, `trhr`, `trpredict`, `sttrkm`, bootstrap, Cox proportional hazards regression, cure rate, first hitting time, hazard ratios, model diagnostics, survival analysis, threshold regression, time-to-event data, Wiener diffusion process

1 Introduction to threshold regression

Threshold regression is a methodology to analyze time-to-event data. For a review of this methodology, see [Lee and Whitmore \(2006\)](#) and [Lee and Whitmore \(2010\)](#). A unique feature of threshold regression is that the event time is the first time an underlying stochastic process hits a boundary threshold. In the context of survival data, for example, the event can be death and the time of death is the moment when his/her latent health status first decreases to a boundary at zero.

With the `stthreg` command, a Wiener process $Y(t)$ is used to model the latent health status process. An event is observed when $Y(t)$ reaches 0 for the first time. Three parameters of the Wiener process are involved: μ , y_0 , and σ^2 . Parameter μ , called the drift of the Wiener process, is the mean change per unit of time in the level of the sample path. The sample path approaches the threshold if $\mu < 0$. Parameter y_0 is the initial value of the process and is taken as positive. Parameter σ^2 represents

the variance per unit of time of the process (Lee and Whitmore 2006). The first hitting time (FHT) of a Wiener process with μ , y_0 , and σ^2 has an inverse Gaussian distribution with the probability density function

$$f(t|\mu, \sigma^2, y_0) = \frac{y_0}{\sqrt{2\pi\sigma^2 t^3}} \exp\left\{-\frac{(y_0 + \mu t)^2}{2\sigma^2 t}\right\} \quad (1)$$

where $-\infty < \mu < \infty$, $\sigma^2 > 0$, and $y_0 > 0$. The probability density function is proper if $\mu \leq 0$. The cumulative distribution function of the FHT is

$$F(t|\mu, \sigma^2, y_0) = \Phi\left\{-\frac{(y_0 + \mu t)}{\sqrt{\sigma^2 t}}\right\} + \exp\left(-\frac{2y_0\mu}{\sigma^2}\right) \Phi\left(\frac{\mu t - y_0}{\sqrt{\sigma^2 t}}\right) \quad (2)$$

where $\Phi(\cdot)$ is the cumulative distribution function of the standard normal distribution. Note that if $\mu > 0$, the Wiener process may never hit the boundary at zero, and hence there is a probability that the FHT is ∞ ; specifically, $P(\text{FHT} = \infty) = 1 - \exp(-2y_0\mu/\sigma^2)$ (Cox and Miller 1965).

Because the health status process is usually latent (that is, unobserved), an arbitrary unit can be used to measure such a process. Hence the variance parameter σ^2 of the process is set to 1 in the `stthreg` command to fix the measurement unit of the process. Then we can regress the other two process parameters, y_0 and μ , on the covariate data. We assume that μ and $\ln(y_0)$ are linear in regression coefficients.

Suppose that the covariate vector is $\mathbf{Z}' = (1, Z_1, \dots, Z_k)$, where Z_1, \dots, Z_k are covariates and the leading 1 in \mathbf{Z}' allows for a constant term in the regression relationship. Then $\ln(y_0)$ and μ can be linked to the covariates with the following regression forms:

$$\ln(y_0) = \gamma_0 + \gamma_1 Z_1 + \dots + \gamma_k Z_k = \mathbf{Z}'\boldsymbol{\gamma} \quad (3)$$

$$\mu = \beta_0 + \beta_1 Z_1 + \dots + \beta_k Z_k = \mathbf{Z}'\boldsymbol{\beta} \quad (4)$$

Vectors $\boldsymbol{\gamma}$ in (3) and $\boldsymbol{\beta}$ in (4) represent regression coefficients for $\ln(y_0)$ and μ , respectively, with $\boldsymbol{\gamma}' = (\gamma_0, \dots, \gamma_k)$ and $\boldsymbol{\beta}' = (\beta_0, \dots, \beta_k)$. Note that researchers can set some elements in $\boldsymbol{\gamma}$ or $\boldsymbol{\beta}$ to zero if they feel the corresponding covariates are not important in predicting $\ln(y_0)$ or μ . For example, if covariate Z_1 in the vector \mathbf{Z}' is considered not important to predict $\ln(y_0)$, we can remove the Z_1 term in (3) by setting γ_1 to zero.

In the remaining sections of this article, we detail how to use our software package to implement threshold regression in Stata. In section 2, we compare threshold regression with Cox proportional hazards regression by running three of the four commands (that is, `sttrkm`, `stthreg`, and `trhr`), together with some existing Stata commands, on `leukemia.dta`. In sections 3 through 6, we introduce the uses of these four commands, respectively. We use `melanoma.dta` in these four sections to illustrate how to use these four commands. In section 7, we introduce these four commands for an extended threshold regression model that we refer to as the threshold regression cure-rate model, and we demonstrate their application to `kidney.dta`.

2 Comparison of threshold regression with Cox proportional hazards regression

The Cox regression model, also called proportional hazards regression, has played a key role in the area of time-to-event data analysis for many years (Cox 1972). The Cox model assumes that covariates alter hazard functions in a proportional manner. Threshold regression does not assume proportional hazards (PH) and can be used as an alternative to the Cox model, especially when the PH assumption of the Cox model is violated. The connections between threshold regression and Cox regression have been studied in Lee and Whitmore (2010), where it is shown that Cox regression is, for most purposes, a special case of threshold regression.

In this section, we use a leukemia remission study dataset (Garrett 1997) from the Stata website to compare these two models when the PH assumption of the Cox model is violated. Below we obtain this leukemia remission dataset from the Stata website and declare it to be survival-time data.

```
. webuse leukemia
(Leukemia Remission Study)
. describe
Contains data from http://www.stata-press.com/data/r12/leukemia.dta
obs:          42          Leukemia Remission
                    Study
vars:          8          23 Mar 2011 10:39
size:         336
```

variable name	storage type	display format	value label	variable label
weeks	byte	%8.0g		Weeks in Remission
relapse	byte	%8.0g	yesno	Relapse
treatment1	byte	%8.0g	trt1lbl	Treatment I
treatment2	byte	%8.0g	trt2lbl	Treatment II
wbc3cat	byte	%9.0g	wbc1bl	White Blood Cell Count
wbc1	byte	%8.0g		wbc3cat==Normal
wbc2	byte	%8.0g		wbc3cat==Moderate
wbc3	byte	%8.0g		wbc3cat==High

```
Sorted by: weeks
. stset weeks, failure(relapse)
      failure event: relapse != 0 & relapse < .
obs. time interval: (0, weeks]
exit on or before: failure
```

```
42 total obs.
0 exclusions
```

```
42 obs. remaining, representing
30 failures in single record/single failure data
541 total analysis time at risk, at risk from t = 0
      earliest observed entry t = 0
      last observed exit t = 35
```

The dataset consists of 42 patients who were monitored to see if they relapsed (`relapse`: 1 = yes, 0 = no) and how long (in weeks) they remained in remission (`weeks`). These 42 patients received two different treatments. For the first treatment, 21 patients received a new experimental drug (drug A), and the other 21 received a standard drug (`treatment1`: 1 = drug A, 0 = standard). For the second treatment, 20 patients received a different drug (drug B), and the other 22 received a standard drug (`treatment2`: 1 = drug B, 0 = standard). White blood cell count, a strong indicator of the presence of leukemia, was recorded in three categories (`wbc3cat`: 1 = normal, 2 = moderate, 3 = high).

As demonstrated in [ST] **stcox PH-assumption tests**, the variable `treatment2` violates the PH assumption. We focus on this variable below to demonstrate the advantages of the threshold regression model over the Cox model when the PH assumption is violated.

First, we use our new model diagnostic command `sttrkm` to demonstrate the better fit of the threshold regression model over the Cox model for the leukemia remission data. To address this comparison, we also use three existing diagnostic commands provided by Stata: `stphplot`, `sts`, and `stcoxkm`. Figures 1 through 4 are generated by the following four commands sequentially:

```
. stphplot, by(treatment2) noshow title("Log-Log Plot") plot2opts(lpattern(-))
. sts, by(treatment2) noshow
. stcoxkm, by(treatment2) noshow title("Cox Predicted vs. Observed")
> pred1opts(lpattern(dash)) pred2opts(lpattern(longdash))
. sttrkm, lny0(treatment2) mu(treatment2) noshow
> title("TR Predicted vs. Observed")
```

In figure 1, the curves of $-\ln\{-\ln(\text{survival})\}$ versus $\ln(\text{analysis time})$ for both the drug B group and the standard group are plotted in a log-log plot. If the plotted lines are reasonably parallel, the PH assumption has not been violated. Because the two curves corresponding to the two groups cross each other in figure 1, the PH assumption is clearly violated for `treatment2`. The PH violation is also suggested by figure 2 where the Kaplan–Meier survival curves for the two groups also cross each other. In figure 3, we use the diagnostic command `stcoxkm` for the Cox model to overlay the Kaplan–Meier survival curves and the Cox predicted curves for `treatment2`. The Kaplan–Meier curves and Cox predicted curves are not close to each other, suggesting a poor fit of the Cox model. Obviously, this poor fit results from a violation of the PH assumption for `treatment2`. The threshold regression model based on a Wiener process, however, does not assume proportional hazards. In figure 4, we use our diagnostic command `sttrkm` for the threshold regression model to overlay the Kaplan–Meier survival curves and the threshold regression predicted curves for `treatment2`. The Kaplan–Meier curves and the threshold regression predicted curves match very well.

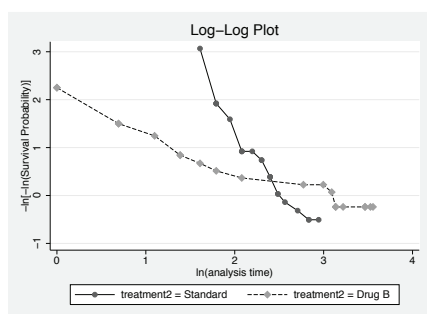


Figure 1. Log-log plot by the `treatment2` variable for the leukemia data

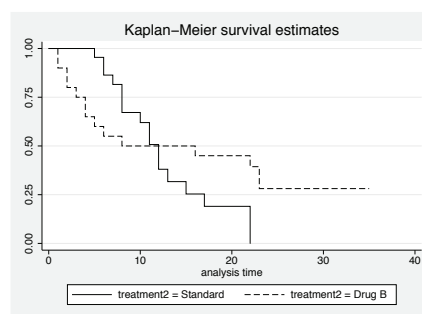


Figure 2. Kaplan-Meier plot by the `treatment2` variable for the leukemia data

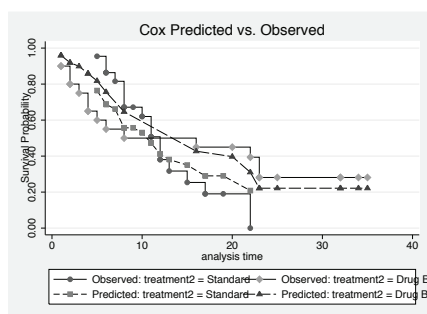


Figure 3. Cox predicted plot versus Kaplan-Meier plot by the `treatment2` variable for the leukemia data

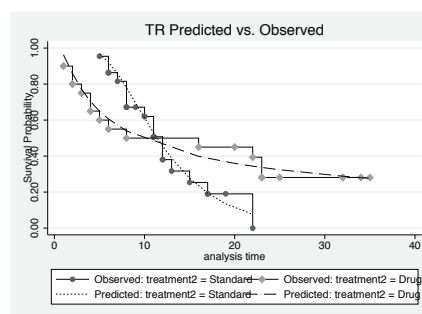


Figure 4. Threshold regression predicted plot versus Kaplan-Meier plot by the `treatment2` variable for the leukemia data

Further, let us use another two new commands, `stthreg` and `trhr`, to fit the threshold regression model to the leukemia remission data and calculate the hazard ratios at week 4, week 9, and week 14. In addition, the `trhr` command calculates pointwise bootstrap confidence intervals for the hazard ratios with its `ci` option (before the `trhr` command, we set the random seed to 1 for the bootstrap procedure). It also plots the estimated hazard function of the drug B group against that of the standard drug group by its `graph(hz)` option. This plot is shown in figure 5.

```

. xi: stthreg, lny0(i.treatment2) mu(i.treatment2)
i.treatment2      _Itreatment_0-1      (naturally coded; _Itreatment_0 omitted)
      failure _d: relapse
      analysis time _t: weeks
initial:      log likelihood = -140.16289
alternative:  log likelihood = -173.94938
rescale:      log likelihood = -136.69022
rescale eq:   log likelihood = -116.95156
Iteration 0:  log likelihood = -116.95156
Iteration 1:  log likelihood = -110.07546
Iteration 2:  log likelihood = -104.66173
Iteration 3:  log likelihood = -104.64227
Iteration 4:  log likelihood = -104.64227
ml model estimated; type -ml display- to display results
Threshold Regression Estimates
                                Number of obs   =          42
                                Wald chi2(2)     =          27.39
Log likelihood = -104.64227      Prob > chi2   =          0.0000

```

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
lny0						
_Itreatmen-1	-1.273925	.2442516	-5.22	0.000	-1.75265	-.7952011
_cons	2.009787	.170641	11.78	0.000	1.675336	2.344237
mu						
_Itreatmen-1	.588838	.1535616	3.83	0.000	.2878628	.8898132
_cons	-.5886176	.1340773	-4.39	0.000	-.8514042	-.3258309

```

. set seed 1
. trhr, var(treatment2) timevalue(4 9 14) ci graph(hz)
(running stthreg on estimation sample)
Bootstrap replications (2000)
(output omitted)
Hazard Ratio for Scenario , at Time = 4

```

var	Hazard Ratio	[95% Percentile C.I.]	
_Itreatment_1	5.9588129935	1.8366861	147.06873

```

(running stthreg on estimation sample)
Bootstrap replications (2000)
(output omitted)
Hazard Ratio for Scenario , at Time = 9

```

var	Hazard Ratio	[95% Percentile C.I.]	
_Itreatment_1	.38571350099	.17518535	.84617594

```
(running stthreg on estimation sample)
Bootstrap replications (2000)
(output omitted)
Hazard Ratio for Scenario , at Time = 14
```

var	Hazard Ratio	[95% Percentile C.I.]
_Itreatment_1	.19011874598	.07150129 .44462474

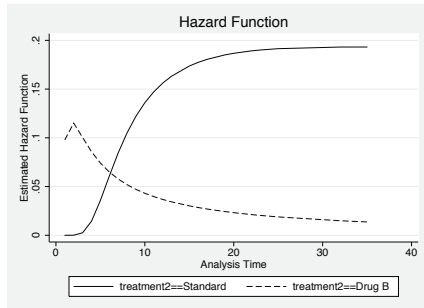


Figure 5. Estimated hazard functions by the `treatment2` variable for the leukemia data by the threshold regression model

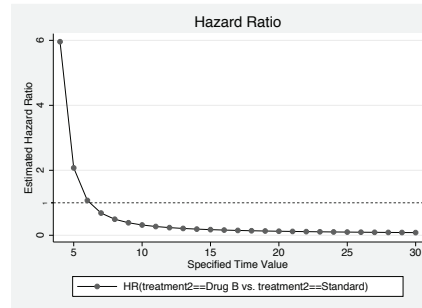


Figure 6. Estimated hazard ratio values over time by the `treatment2` variable for the leukemia data by the threshold regression model

It is clear that the hazard ratios calculated at weeks 4, 9, and 14 are quite different. At week 4, the estimated hazard ratio of the drug B group versus the standard drug group is 5.96 (the 95% bootstrap CI of the hazard ratio excludes 1 on the left); while at week 9, the estimated hazard ratio drops to 0.38 (the 95% bootstrap CI of hazard ratio excludes 1 on the right); the hazard ratio keeps decreasing to 0.19 at week 14 (the 95% bootstrap CI of the hazard ratio excludes 1 on the right).

Obviously, the change of hazard ratio over time is huge, and the change is captured by the threshold regression model. In figure 5, it is clear that the hazard function curves of the two groups cross over time, and that is why the hazard ratio of the drug B group versus the standard drug group changes from 5.96 (a value greater than 1) at week 4 to 0.19 (a value less than 1) at week 14. We can use the `graph(hr)` option of the `trhr` command to depict the hazard ratios for different time points to illustrate the changing pattern of the hazard ratio over time. For example, the following command can be used to plot the estimated hazard-ratio values in a specified time span from week 4 to week 30 with a reference line for a hazard ratio of 1. The generated plot is shown in figure 6.

```
. trhr, var(treatment2) timevalue(4(1)30) graph(hr) graphopt(title(Hazard Ratio))
> ytitle(Estimated Hazard Ratio) legend(on) yline(1,pattern(shortdash))
> ymtick(1) ylabel(1))
```

On the other hand, because of the proportional-hazards assumption, the Cox model can only be used to estimate a constant hazard ratio across the whole time span for the drug B group versus the standard drug group. The following command fits the Cox model to the leukemia remission data. The resulting hazard ratio is a constant 0.75 with the 95% CI including 1. This is obviously a misleading outcome for these data where the PH assumption is violated.

```
. xi: stcox i.treatment2
i.treatment2      _Itreatment_0-1      (naturally coded; _Itreatment_0 omitted)
      failure _d: relapse
      analysis time _t: weeks
Iteration 0:  log likelihood = -93.98505
Iteration 1:  log likelihood = -93.71683
Iteration 2:  log likelihood = -93.716786
Refining estimates:
Iteration 0:  log likelihood = -93.716786
Cox regression -- Breslow method for ties
No. of subjects =          42                Number of obs   =          42
No. of failures =          30
Time at risk    =          541
Log likelihood   = -93.716786                LR chi2(1)       =          0.54
                                                Prob > chi2      =          0.4639
```

_t	Haz. Ratio	Std. Err.	z	P> z	[95% Conf. Interval]
_Itreatmen-1	.7462828	.3001652	-0.73	0.467	.3392646 1.641604

In the following sections, we detail the use of the four new threshold regression commands one by one with examples.

3 stthreg: The command to fit the threshold regression model

3.1 Introduction to stthreg

`stthreg` fits the threshold regression model introduced in section 1. Maximum likelihood estimation is used to estimate the regression coefficients in vectors γ and β . A subject i in the sample dataset who is observed to die contributes the FHT probability density $f(t^{(i)}|\mu^{(i)}, y_0^{(i)})$ to the sample likelihood function, where $t^{(i)}$ is the observed time of death of the subject. A subject j in the sample dataset who lives to the end of the study contributes the survival probability $1 - F(t^{(j)}|\mu^{(j)}, y_0^{(j)})$ to the sample likelihood function, where $t^{(j)}$ is the right-censored survival time of the subject. Among the n subjects in the sample, subjects with observed death times are indexed 1 to n_1 and subjects with right-censored observations are indexed $n_1 + 1$ to n . Then the log-likelihood function is written as

$$\ln L(\beta, \gamma) = \sum_{i=1}^{n_1} \ln f\left(t^{(i)} | \mu^{(i)}, y_0^{(i)}\right) + \sum_{j=n_1+1}^n \ln \left\{1 - F\left(t^{(j)} | \mu^{(j)}, y_0^{(j)}\right)\right\} \quad (5)$$

The maximum-likelihood estimation routine with the `lf` method (Gould et al. 2010) is incorporated in the `stthreg` command to find the maximum likelihood estimates of the regression coefficients in the threshold regression model. The convergence speed is fairly fast.

3.2 Syntax of `stthreg`

```
stthreg [if] [in], lny0(varlist) mu(varlist) [noconstant cure lgtp(varlist)
    _level(#) init(string) nolog maximize_options]
```

3.3 Options

`lny0(varlist)` specifies independent variables that will be used in the linear regression function for $\ln y_0$ in the threshold regression model, as in (3). For example, if Z_1, \dots, Z_k in (3) are used, then this `lny0(varlist)` option should be written as `lny0($Z_1 \dots Z_k$)`. Note that the intercept coefficient γ_0 in (3) will be automatically added in the linear regression function for $\ln y_0$, and this intercept coefficient will correspond to the `_cons` term in the `lny0` section of the output regression coefficient table. The estimation results for the other regression coefficients in the linear regression function for $\ln y_0$ correspond to the independent variables in the `lny0` section of the output regression coefficient table. If no independent variable is listed in `lny0()`, then only the intercept coefficient will be used in the regression function for $\ln y_0$. In this case, the estimated value of $\ln y_0$ is equal to the estimated value of this intercept coefficient, as can be easily seen in (3). `lny0()` is required.

`mu(varlist)` specifies independent variables that will be used in the linear regression function for μ in the threshold regression model, as in (4). For example, if Z_1, \dots, Z_l in (4) are used, then this `mu(varlist)` option should be written as `mu($Z_1 \dots Z_l$)`. Note that the intercept coefficient β_0 in (4) will be automatically added in the linear regression function for μ , and this intercept coefficient will correspond to the `_cons` term in the `mu` section of the output regression coefficient table. The estimation results for the other regression coefficients in the linear regression function for μ will appear after the names of the corresponding independent variables in the `mu` section of the output regression coefficient table. If no independent variable is listed in `mu()`, then only the intercept coefficient will be used in the regression function for μ . In this case, the estimated value of μ is merely equal to the estimated value of this intercept coefficient, as can be easily seen in (4). `mu()` is required.

`noconstant` specifies that no intercept is included in the linear regression functions for $\ln y_0$ and μ .

`cure` specifies that the model to be fit is a threshold regression cure-rate model. See section 7 for details about this model.

`lgtp(varlist)` specifies independent variables that will be used in the linear regression function for `lgtp()` in the threshold regression cure-rate model. This `lgtp()` option can be used only when the `cure` option is used.

`level(#)` specifies the confidence level, as a percentage, for confidence intervals. The default is `level(95)` or as set by `set level`.

`init(string)` specifies initialization values of the regression coefficients in maximum likelihood iterations. The syntax is the same as `ml init`, which is the command to set initial values in the maximum likelihood estimation routine of Stata.

`nolog` specifies that the iteration log of the log likelihood not be displayed.

maximize_options: `iterate(#)`, `[no]log`, `trace`, `tolerance(#)`, `ltolerance(#)`, `nrtolerance(#)`, and `nonrtolerance`; see [R] `maximize`. These options are seldom used.

3.4 Saved results of `stthreg`

`stthreg` saves the following in `e()`:

Scalars

<code>e(N)</code>	number of observations	<code>e(chi2)</code>	χ^2
<code>e(df_m)</code>	model degrees of freedom	<code>e(p)</code>	significance
<code>e(ll)</code>	log likelihood		

Macros

<code>e(cmd)</code>	<code>stthreg</code>	<code>e(ml_method)</code>	type of ml method
<code>e(depvar)</code>	name of dependent variable	<code>e(technique)</code>	maximization technique
<code>e(chi2type)</code>	Wald or LR; type of model chi-squared test	<code>e(crittype)</code>	optimization criterion
<code>e(vce)</code>	<code>oim</code>	<code>e(properties)</code>	<code>b V</code>
<code>e(opt)</code>	type of optimization	<code>e(predict)</code>	program used to implement <code>predict</code>

Matrices

<code>e(b)</code>	coefficient vector	<code>e(V)</code>	variance-covariance matrix of the estimators
-------------------	--------------------	-------------------	---

Functions

<code>e(sample)</code>	marks estimation sample
------------------------	-------------------------

3.5 Melanoma dataset

We use `melanoma.dta` to illustrate how to use the `stthreg` package of commands. `melanoma.dta` is originally from Drzewiecki and Andersen (1982). It contains observations retrieved from case records for 205 melanoma patients. These 205 patients, who were all in clinical stage I, were treated at the Plastic Surgery Unit in Odense from 1964 to 1973. Follow-up was terminated on January 1, 1978. Some patients died during the study period, but others were still alive at the end of the study and thus were considered to be right-censored observations in our analysis. Eight variables are included in

this dataset to illustrate the use of the `stthreg` command: `survtime` (on-study time in days), `status` (0 = censored, 1 = death), `sex` (0 = female, 1 = male), `ici` (degree of inflammatory cell infiltrate), `ecells` (epithelioid cell type), `ulcerat` (ulceration), `thick` (tumor thickness), and `age` (in years).

```
. use melanoma, clear
. describe
Contains data from melanoma.dta
  obs:          205
  vars:          8                27 Apr 2008 14:21
  size:         6,560
```

variable name	storage type	display format	value label	variable label
sex	float	%9.0g	sex	
survtime	float	%9.0g		survival time
status	float	%9.0g	status	status at end of follow-up
ici	float	%9.0g	ici	degree of ici
ecells	float	%9.0g	ecells	epitheloid cells
ulcerat	float	%9.0g	ulcerat	ulceration
thick	float	%9.0g		tumor thickness (1/100 mm)
age	float	%9.0g		age in years

```
Sorted by:
. stset survtime, failure(status)
      failure event:  status != 0 & status < .
obs. time interval:  (0, survtime]
exit on or before:  failure
```

```
      205 total obs.
       0 exclusions
```

```
      205 obs. remaining, representing
       57 failures in single record/single failure data
441324 total analysis time at risk, at risk from t =      0
              earliest observed entry t =      0
              last observed exit t =      5565
```

3.6 Example of `stthreg`

Before using `stthreg`, you must `stset` the data as we have already done in section 3.5. Below we will run the `stthreg` command on `melanoma.dta`. Each of the two parameters of the underlying Wiener process, $\ln y_0$ and μ , are linked to the covariates `thick`, `age`, and `sex` by the `lny0()` and `mu()` options. Note that both $\ln y_0$ and μ are linked to the same covariates in this example, while they can be linked to different covariates in other contexts.

Deciding which covariates influence $\ln y_0$ and μ requires subject matter knowledge and does not adhere to a simple general rule. Covariates linked to $\ln y_0$ are those that influence the initial health state, such as baseline characteristics. Covariates linked to μ often include baseline characteristics but also covariates (such as treatment effects) that affect the drift in health over time. See [Lee and Whitmore \(2010\)](#) for relevant discussion of this issue.

The `stthreg` command to fit this threshold regression model is given below:

```
. xi: stthreg, lny0(sex ecells thick age i.ulcerat)
> mu(sex ecells thick age i.ulcerat)
i.ulcerat      _Iulcerat_0-1      (naturally coded; _Iulcerat_0 omitted)
      failure _d: status
      analysis time _t: survtime
initial:      log likelihood = -1243.6028
alternative:  log likelihood = -9621.3647
rescale:      log likelihood = -1081.5328
rescale eq:   log likelihood = -573.47172
Iteration 0:  log likelihood = -573.47172
Iteration 1:  log likelihood = -538.3778
Iteration 2:  log likelihood = -531.08833
Iteration 3:  log likelihood = -530.83209
Iteration 4:  log likelihood = -530.83114
Iteration 5:  log likelihood = -530.83114
ml model estimated; type -ml display- to display results
Threshold Regression Estimates      Number of obs =      205
      Wald chi2(10) =      64.94
Log likelihood = -530.83114      Prob > chi2 =      0.0000
```

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
lny0						
sex	.1070979	.1724429	0.62	0.535	-.2308839	.4450797
ecells	.3300398	.194412	1.70	0.090	-.0510008	.7110804
thick	-.0737792	.0333807	-2.21	0.027	-.1392043	-.0083542
age	.0077415	.0042245	1.83	0.067	-.0005384	.0160215
_Iulcerat_1	-.5143228	.2165787	-2.37	0.018	-.9388093	-.0898363
_cons	3.803575	.3145534	12.09	0.000	3.187061	4.420088
mu						
sex	-.0120426	.0073686	-1.63	0.102	-.0264849	.0023996
ecells	-.0224773	.0083094	-2.71	0.007	-.0387633	-.0061912
thick	.0005888	.0012388	0.48	0.635	-.0018392	.0030167
age	-.000579	.0002101	-2.76	0.006	-.0009908	-.0001672
_Iulcerat_1	-.0000191	.0083654	-0.00	0.998	-.016415	.0163768
_cons	.0462283	.0143098	3.23	0.001	.0181816	.0742751

4 trhr: The postestimation command to calculate hazard ratios

4.1 Introduction to trhr

trhr is a postestimation command to calculate hazard ratios based on the threshold regression model fit by **stthreg** in advance. That is, suppose that a set of predictor variables $\{Z_1, \dots, Z_{k-1}, G\}$ has been used to predict $\ln y_0$ and μ in the threshold regression model by using **stthreg**, where G is a categorical variable. We can then use **trhr** to estimate the hazard ratio of level $G = g$ over the reference level $G = 0$ at given values of other predictors, $Z_1 = z_1, \dots, Z_{k-1} = z_{k-1}$, and a given value of time $t = t_0$. We state as follows how such a hazard ratio is calculated. First, we need to estimate the hazard value for level $G = g$ at $Z_1 = z_1, \dots, Z_{k-1} = z_{k-1}$ and time $t = t_0$. Set

$$(\mathbf{z}^g)' = (1, z_1, \dots, z_{k-1}, g) \quad (6)$$

Then $\ln y_0$ and μ can be estimated for the given $(\mathbf{z}^g)'$ in (6) as

$$\ln \hat{y}_0^g = (\mathbf{z}^g)' \hat{\boldsymbol{\gamma}} \quad (7)$$

$$\hat{\mu}^g = (\mathbf{z}^g)' \hat{\boldsymbol{\beta}} \quad (8)$$

where $\hat{\boldsymbol{\gamma}}$ and $\hat{\boldsymbol{\beta}}$ are vectors of regression coefficients already estimated by **stthreg** (see section 3.1). Furthermore, the estimates of the density, survival, and hazard functions at time t_0 are given as

$$f(t_0 | \hat{\mu}^g, \hat{y}_0^g) \quad (9)$$

$$S(t_0 | \hat{\mu}^g, \hat{y}_0^g) = 1 - F(t_0 | \hat{\mu}^g, \hat{y}_0^g) \quad (10)$$

$$h(t_0 | \hat{\mu}^g, \hat{y}_0^g) = \frac{f(t_0 | \hat{\mu}^g, \hat{y}_0^g)}{S(t_0 | \hat{\mu}^g, \hat{y}_0^g)} = \frac{f(t_0 | \hat{\mu}^g, \hat{y}_0^g)}{1 - F(t_0 | \hat{\mu}^g, \hat{y}_0^g)} \quad (11)$$

where $f(\cdot)$ and $F(\cdot)$ are given in (1) and (2), with σ^2 set to 1, and y_0 and μ replaced by their estimates in level g . If we change the nonreference level g to the reference level 0 in (6), we can obtain $f(t_0 | \hat{\mu}^0, \hat{y}_0^0)$, $S(t_0 | \hat{\mu}^0, \hat{y}_0^0)$, and $h(t_0 | \hat{\mu}^0, \hat{y}_0^0)$ by the same means. The hazard ratio of level $G = g$ over level $G = 0$ at $Z_1 = z_1, \dots, Z_{k-1} = z_{k-1}$ and time $t = t_0$ is therefore

$$\text{Hazard Ratio} = \frac{h(t_0 | \hat{\mu}^g, \hat{y}_0^g)}{h(t_0 | \hat{\mu}^0, \hat{y}_0^0)} \quad (12)$$

Using the formulas above, **trhr** estimates the hazard ratios for a categorical variable with three options: **var()**, **timevalue()**, and **scenario()**. The **var()** option specifies

the name of the categorical variable for which the hazard ratios are to be calculated. Note that a prerequisite for the execution of `trhr` is that the categorical variable specified in the `var()` option needs to have been expanded into a dummy-variable set by the `xi` command (see [R] `xi`), which can be placed ahead of the previous `stthreg` command. The `timevalue()` option specifies the desired time values at which the hazard ratios are to be calculated. And the `scenario()` option specifies the values of all predictors except G . A setting of these values is referred to as a *scenario*. The calculated hazard ratios are with reference to the specified scenario. We do not need to specify a level value for the categorical G in the `scenario()` option because all nonreference levels g of G are enumerated in calculating hazard ratios relative to the reference level 0. The reference level or the nonreference levels are decided by the `xi` command, and that is why `xi` is needed before `trhr` is run. By default, the lowest level is used as the reference level by `xi`.

Following the notation from (6) to (12), `trhr` calculates the hazard ratio of level $G = g$ over level $G = 0$ with the following command: `trhr, var(G) scenario($Z_1 = z_1 \cdots Z_{k-1} = z_{k-1}$) timevalue(t_0)`. If the threshold regression model has categorical predictors other than G being fit by `stthreg` and if the user has also expanded some of these categorical predictors to sets of dummy variables by using “i.”, then the user needs to provide the values of all the created dummy variables instead of the values of the corresponding original categorical variables in the `scenario()` option when calculating the hazard ratios for G . That is because `trhr` will use these dummy variables, instead of the original categorical variables, as predictors in calculations. In the `scenario()` option, the order of presentation of the predictors does not matter, and the terms in this option are separated by blanks. If G is the only predictor in the model specified by `stthreg` (G also needs to be expanded by `xi` in this case), then the `scenario()` option is not necessary because there is no predictor that needs to be specified for the scenario value (the levels of G will still be enumerated to calculate the hazard ratios for G in this case).

Note that if `trhr` is used after a threshold regression cure-rate model is fit by `stthreg`, all the calculations by `trhr` then automatically correspond to the threshold regression cure-rate model. See section 7 for details about the threshold regression cure-rate model.

4.2 Syntax of `trhr`

```
trhr, var(varname) timevalue(numlist) [scenario(string)
  graph(hz|sv|ds|hr) graphopt(string) ci bootstrap(#) level keep
  prefix(string) ]
```

4.3 Options

var(*varname*) specifies the name of the categorical variable for the calculation of hazard ratios. **var**() is required.

timevalue(*numlist*) specifies the desired values of time at which the hazard ratios be calculated (for the use of *numlist*, see [U] 11.1.8 **numlist**). **timevalue**() is required.

scenario(*string*) must be used if any covariates other than the categorical variable specified in **var**() are also included in the threshold regression model fit by the **stthreg** command. The **scenario**() option is to assign values to those covariates by equations, one by one and separated by a blank space. The **scenario**() option is to set up a scenario at which hazard ratios are to be calculated.

graph(*hz|sv|ds|hr*) specifies the type of curves to be generated. The **graph**(*hz*) option is to plot hazard function curves, the **graph**(*sv*) option is to plot survival function curves, the **graph**(*ds*) option is to plot density function curves, and the **graph**(*hr*) option is to plot a hazard-ratio tendency plot over the specified time points.

graphopt(*string*) specifies the graph settings (titles, legends, etc.). If this option is used, the default graph settings in the **trhr** command will be ignored, and the styles (such as titles, legends, and labels) of the output graphs completely depend on the options provided in **graphopt**(). Because the **line** command in Stata is used in the **trhr** command to plot curves, any options suitable for the **line** command can be used in the **graphopt**() option to produce desired graphic styles.

ci specifies to output the bootstrap percentile confidence interval of the estimated hazard-ratio value. By default, 2,000 bootstrap replications will be performed and the confidence level is 95%.

bootstrap(*#*) specifies the number of bootstrap replications. The default is **bootstrap**(2000).

level(*#*) sets the confidence level of the bootstrap percentile confidence interval of the hazard-ratio value; the default is **level**(95).

keep saves the estimated hazard-ratio values of specified time points in the current dataset. The specified time values are saved under a new variable with the name **hr_t**; the hazard-ratio values are saved under a new variable name that begins with **hr** and ends with the name of the corresponding dummy variable. If the **ci** option is used, then the lower and upper limit values of the confidence intervals will also be saved under two new variable names that begin with **hrll** and **hrul**, respectively, and end with the name of the corresponding dummy variable.

prefix(*string*) specifies a string that will be attached to the head of the names of the new variables generated by the **keep** option.

4.4 Example of trhr

The following command estimates the hazard-ratio value for the two levels of the `ulcerat` variable at time 4,000 for a subject with these features: `sex=0`, `ecells=0`, `thick=12`, and `age=40`.

```
. set seed 1
. trhr, var(ulcerat) timevalue(4000) scenario(sex=0 ecells=0 thick=12 age=40) ci
(running stthreg on estimation sample)
Bootstrap replications (2000)
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 1 | 2 | 3 | 4 | 5
..... 50
..... 100
(output omitted)
..... 1950
..... 2000
Hazard Ratio for Scenario sex=0 ecells=0 thick=12 age=40, at Time = 4000
```

var	Hazard Ratio	[95% Percentile C.I.]
_Iulcerat_1	1.1206067911	.05079897 46.630562

5 trpredict: The postestimation command for predictions

5.1 Introduction to trpredict

After a threshold regression model is fit by `stthreg`, we can use the command `trpredict` to predict the initial health status value y_0 , the drift value of the health process μ , the probability density function of the survival time $f(t|\mu, y_0)$, the survival function $S(t|\mu, y_0)$, and the hazard function $h(t|\mu, y_0)$ for a specified scenario, as we did in calculating the hazard ratios in section 4.1 where hazard function values at a specified time point are predicted and are used to calculate the corresponding hazard-ratio values. To specify the scenario values, use the `scenario()` option. Note that in the `scenario()` option, we need to provide the scenario values of all predictors (including dummy-variable predictors) specified in the previous `stthreg` command. This is a little bit different from the `scenario()` option in the `trhr` command, where we do not need to provide the scenario values for the dummy variables expanded from the categorical variable G for which the hazard ratios are calculated. Again the reason is that the program will automatically enumerate all levels of G .

Also we can predict those quantities for the scenario values and on-study time values corresponding to each subject in the dataset (this use is similar to the `predict` command in Stata). Following equations from (7) to (11), we can see that when the elements of \mathbf{z} in (7) and (8) are set to be the corresponding covariate values of a subject, and when t in (9), (10), and (11) is set to the on-study time of this subject, we can calculate

the predicted values $\ln \hat{y}_0$ (and hence \hat{y}_0), $\hat{\mu}$, $f(t|\hat{\mu}, \hat{y}_0)$, $S(t|\hat{\mu}, \hat{y}_0)$, and $h(t|\hat{\mu}, \hat{y}_0)$ for this subject by using the regression coefficient estimates $\hat{\gamma}$ and $\hat{\beta}$. Then by applying this procedure to each subject in the dataset we obtain those predicted values for each subject.

Note that after running the `trpredict` command, the following six variables will be added to the current dataset: `lny0`, `y0`, `mu`, `f`, `S`, and `h`. These six variables, respectively, correspond to $\ln \hat{y}_0$, \hat{y}_0 , $\hat{\mu}$, $f(t|\hat{\mu}, \hat{y}_0)$, $S(t|\hat{\mu}, \hat{y}_0)$, and $h(t|\hat{\mu}, \hat{y}_0)$. The option `prefix(string)` is used to add a specified prefix string to the default names of those six variables. Also note that if the `trpredict` command is run after fitting a threshold regression cure-rate model by using the `cure` option in `stthreg`, then two more variables, `lgtp` and `p`, in addition to the six variables above, will be added to the current dataset. These two variables correspond to the estimate of the logit of p and the estimate of p , where p is the susceptibility rate. See section 7 for details of the threshold regression cure-rate model.

5.2 Syntax of `trpredict`

```
trpredict [, prefix(string) scenario(string) replace]
```

5.3 Options

`prefix(string)` specifies a prefix string that will be attached to the names of the six new variables generated by the `trpredict` command. For example, if `prefix(abc_)` is used, then the names of the six variables will be `abc.lny0`, `abc.y0`, `abc.mu`, `abc.f`, `abc.S`, and `abc.h`.

`scenario(string)` specifies a scenario for which the predictions are calculated by using the standard threshold regression model or the threshold regression cure-rate model fit previously. Note that *specifying a scenario* means specifying the values of the predictors in the standard threshold regression model or the threshold regression cure-rate model fit previously.

`replace` is used to force replacement of the variables (in the current dataset) with the same names as the six (or eight) new prediction variables without warning.

5.4 Example of `trpredict`

Below we first use two `trpredict` commands to predict the six quantities $\{(\ln \hat{y}_0, \hat{y}_0, \hat{\mu}, f(t|\hat{\mu}, \hat{y}_0), S(t|\hat{\mu}, \hat{y}_0), \text{ and } h(t|\hat{\mu}, \hat{y}_0))\}$ for two different scenarios (scenario 1: `_Iulcerat_1=0`, `ecells=0`, `thick=12`, `age=40`, and `sex=0`; scenario 2: `_Iulcerat_1=1`, `ecells=0`, `thick=12`, `age=30`, and `sex=1`). Then we use the `line` commands to overlay the predicted survival curves to compare these two scenarios (see figure 7).

```

. trpredict, scenario(_lulcerat_1=0 ecells=0 thick=12 age=40 sex=0)
> prefix(Scenario1_) replace
. trpredict, scenario(_lulcerat_1=1 ecells=0 thick=12 age=30 sex=1) prefix(Scen
> ario2_) replace
. twoway line Scenario1_S _t || line Scenario2_S _t,
> title("Scenario Comparison") ytitle("Predicted Survival Function")
> xtitle("Analysis Time (Days)")
> text(.85 3500 "_lulcerat_1=0, ecells=0, thick=12, age=40, sex=0")
> text(.6 3500 "_lulcerat_1=1, ecells=0, thick=12, age=30, sex=1")

```

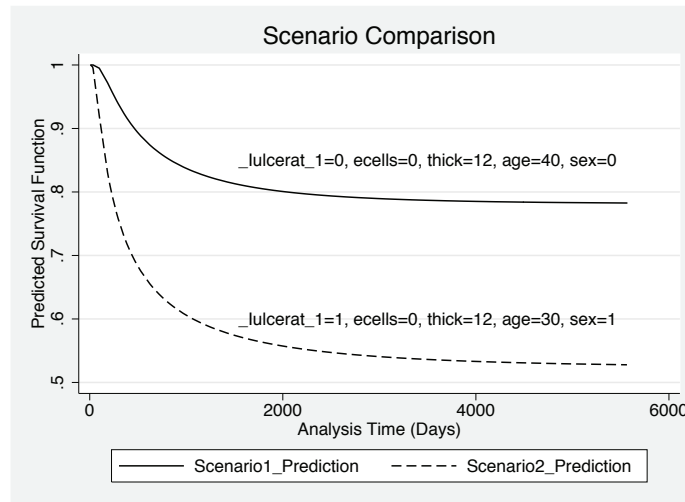


Figure 7. Predicted survival functions for two different scenarios by threshold regression

6 sttrkm: The model diagnostic command

6.1 Introduction to sttrkm

For a categorical independent variable, the Kaplan–Meier nonparametric survival curve can be plotted for each categorical level of this variable. If we include such a categorical independent variable as the only predictor in the threshold regression, survival curves for each level can also be predicted parametrically by the threshold regression model (see section 5.1). The command `sttrkm` overlays these two types of curves and provides a graphic goodness-of-fit diagnosis of the threshold regression model. The closer the Kaplan–Meier nonparametric survival curves are to the predicted curves, the better the threshold regression model fits the data with this variable. A counterpart command for the Cox model is `stcoxkm`. Note that unlike `trhr` and `trpredict`, the `sttrkm` command is not a postestimation command, and hence the estimation command `stthreg` is not required before the `sttrkm` command. However, you must `stset` your data before using `sttrkm`.

6.2 Syntax of `sttrkm`

```
sttrkm [if] [, lny0(varname) mu(varname) cure lgtp(varname) noshow
  separate obsopts(sttrkm_plot_options) obs#opts(sttrkm_plot_options)
  predopts(sttrkm_plot_options) pred#opts(sttrkm_plot_options) addplot(plot)
  twoway_options byopts(byopts) ]
```

6.3 Options

`lny0(varname)` specifies the categorical predictor that will be used in the linear regression function for $\ln y_0$ in the threshold regression model. Note that either `lny0()` or `mu()` (see below) can be omitted if you do not want to use this categorical predictor for either $\ln y_0$ or μ in the threshold regression model. However, if both `lny0()` and `mu()` are used, the categorical predictor specified in these two options must be the same.

`mu(varname)` specifies the categorical predictor that will be used in the linear regression function for μ in the threshold regression model.

`cure` is used to diagnose the goodness of fit of a threshold regression cure-rate model. See section 7 for details of this model.

`lgtp(varname)` specifies the categorical predictor that will be used in the linear regression function for `lgtp()` in the threshold regression cure-rate model. This `lgtp()` option can be used only when the `cure` option is used. Note that when `cure` is used, `lny0()`, `mu()`, or `lgtp()` can be omitted if you do not want to use this categorical predictor for $\ln y_0$, μ , or `lgtp()` in the threshold regression cure-rate model. However, if any of these three options is used, the categorical predictor specified in these options must be the same.

`noshow` specifies to not show `st` setting information.

`separate` specifies to draw separate plots for predicted and observed curves.

`obsopts(sttrkm_plot_options)` affects rendition of the observed curve.

`obs#opts(sttrkm_plot_options)` affects rendition of the `#`th observed curve; not allowed with `separate`.

`predopts(sttrkm_plot_options)` affects rendition of the predicted curve.

`pred#opts(sttrkm_plot_options)` affects rendition of the `#`th predicted curve; not allowed with `separate`.

`addplot(plot)` specifies other plots to add to the generated graph.

`twoway_options` are any options documented in [G-3] *twoway_options*.

`byopts(byopts)` specifies how subgraphs are combined, labeled, etc.

6.4 Example of sttrkm

Below we use the `sttrkm` command to overlay the threshold regression predicted survival curves with the Kaplan–Meier observed survival curves for each level of the `ici` variable. We used the `separate` option of the `sttrkm` command to separate the plots for different levels of `ici`. The generated graph is shown in figure 8.

```
. sttrkm, lny0(ici) mu(ici) noshow separate
```

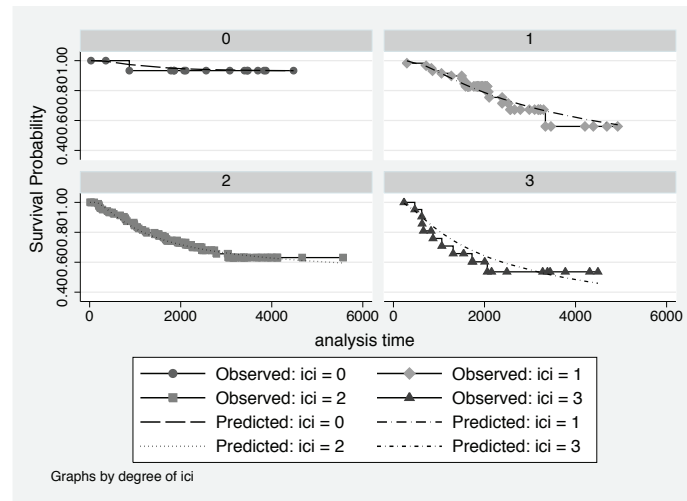


Figure 8. Threshold regression predicted plot versus Kaplan–Meier plot by the `ici` variable for the melanoma data

7 Threshold regression cure-rate model

7.1 Introduction to threshold regression cure-rate model

As mentioned in section 1, the Wiener process has probability $P(\text{FHT} = \infty) = 1 - \exp(-2y_0\mu/\sigma^2)$ that it will never hit the boundary at zero if it is drifting away from the boundary, that is, if $\mu > 0$. We now denote this probability by $1 - p_0$. When hitting the boundary represents a medical failure, p_0 is sometimes called the *susceptibility rate* and $1 - p_0$ the *nonsusceptibility rate* or *cure rate*. The former is the proportion of the population that would eventually experience the medical failure if given enough time, while the latter is the proportion that is cured and will never experience the medical failure.

The preceding susceptibility rate p_0 is determined by the parameter values of the Wiener model when $\mu > 0$. Research populations, however, often have proportions of susceptible and nonsusceptible individuals that do not correspond to p_0 and $1 - p_0$.

In this common situation, it is better to let the susceptibility rate be a free parameter that is independently linked to covariates in the threshold regression model. We will use p to denote this mathematically independent susceptibility rate and use $1 - p$ to denote the corresponding cure rate. We therefore enrich the threshold regression model with an additional parameter p . Specifically, we replace the probability density function $f(t|\mu, \sigma^2, y_0)$ in (1) by $pf(t|\mu, \sigma^2, y_0)$ and the cumulative distribution function $F(t|\mu, \sigma^2, y_0)$ in (2) by $pF(t|\mu, \sigma^2, y_0)$. Note that the probability density function integrates to p in this case and therefore is improper. We link the log-odds ratio or *logit* of p to a linear combination of covariates, as follows:

$$\text{logit}(p) = \log\left(\frac{p}{1-p}\right) = \lambda_0 + \lambda_1 Z_1 + \cdots + \lambda_k Z_k = \mathbf{Z}'\boldsymbol{\lambda}$$

We continue to set the variance parameter σ^2 of the Wiener process to 1. Hence the log-likelihood function for this enriched threshold regression model is the following extension of (5):

$$\ln L(\boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\lambda}) = \sum_{i=1}^{n_1} \ln p^{(i)} f\left(t^{(i)}|\mu^{(i)}, y_0^{(i)}\right) + \sum_{j=n_1+1}^n \ln \left\{1 - p^{(j)} F\left(t^{(j)}|\mu^{(j)}, y_0^{(j)}\right)\right\}$$

We call this enriched threshold regression model the *threshold regression cure-rate model*. The `stthreg` package can also be used for the threshold regression cure-rate model, and the commands are summarized as follows:

- **stthreg**: The `cure` option is used to fit a threshold regression cure-rate model; when the `cure` option is used, the `lgtp()` option is enabled and can be used to specify independent variables that will be used in the linear regression function for `lgtp` in the threshold regression cure-rate model.
- **trhr**: When used after `stthreg` that fits a threshold regression cure-rate model, all the results produced by `trhr` will correspond to the threshold regression cure-rate model.
- **trpredict**: When used after `stthreg` that fits a threshold regression cure-rate model, all the results produced by `trpredict` will correspond to the threshold regression cure-rate model.
- **sttrkm**: The `cure` option is used to diagnose the goodness of fit of a threshold regression cure-rate model; when the `cure` option is used, the `lgtp()` option is enabled and can be used to specify the categorical predictor that will be used in the linear regression function for `lgtp` in the threshold regression cure-rate model.

7.2 Kidney dataset

In the next section, we use `kidney.dta` to illustrate how to use the four commands for the threshold regression cure-rate model. The kidney dialysis dataset (`kidney.dta`) is taken from [Nahman et al. \(1992\)](#) and analyzed by Klein and Moeschberger (2003). The dataset considers the time to first exit-site infection (in months) in patients with renal insufficiency. Two groups of patients are compared: patients who used a surgically placed catheter and patients who used a percutaneous placed catheter. There are three variables in the dataset—`time`, `infection`, and `group`. The variable `time` records the on-study time; `infection` indicates whether the time is an event time (`infection = 1`) or a right-censoring time (`infection = 0`) for each observation; and `group` indicates which group the observation is in (1 = surgical group, 2 = percutaneous group).

```
. use kidney, clear
. describe
Contains data from kidney.dta
  obs:          119
  vars:           3                27 Jan 2008 03:49
  size:          714
```

variable name	storage type	display format	value label	variable label
<code>time</code>	float	%9.0g		Time to infection (months)
<code>infection</code>	byte	%8.0g		Infection indicator (0=no, 1=yes)
<code>group</code>	byte	%8.0g	group	Catheter placement (1=surgically, 2=percutaneously)

```
Sorted by:
. stset time, failure(infection)
      failure event:  infection != 0 & infection < .
obs. time interval:  (0, time]
exit on or before:  failure
```

```
      119 total obs.
       0 exclusions
```

```
      119 obs. remaining, representing
       26 failures in single record/single failure data
1092.5 total analysis time at risk, at risk from t =          0
      earliest observed entry t =          0
      last observed exit t =          28.5
```

7.3 Examples for the threshold regression cure-rate model

The following four commands generate figures 9 to 11 for kidney.dta.

```
. sts, by(group) noshow
. stcoxkm, by(group) noshow separate title("Cox Predicted vs. Observed")
. sttrkm, lny0(group) mu(group) noshow separate
> title("TR Predicted vs. Observed")
. sttrkm, lny0(group) mu(group) lgtp(group) cure noshow separate
> title("TRC Predicted vs. Observed")
```

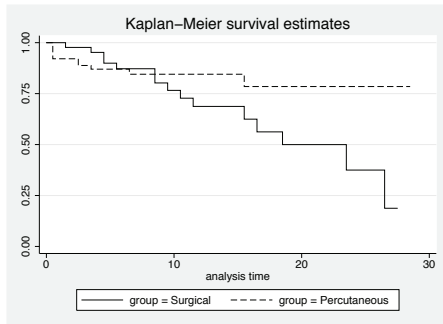


Figure 9. Kaplan-Meier plot by the group variable for the kidney data

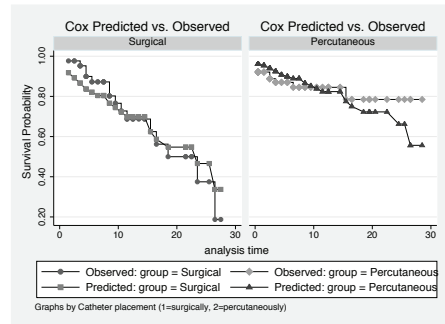


Figure 10. Cox predicted plot versus Kaplan-Meier plot by the group variable for the kidney data

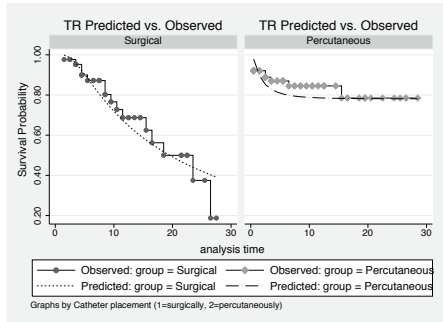


Figure 11. Threshold regression predicted plot versus Kaplan-Meier plot by the group variable for the kidney data

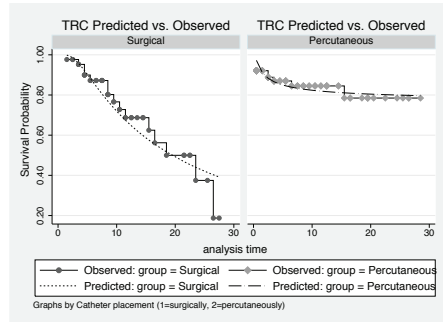


Figure 12. Threshold regression cure-rate model predicted plot versus Kaplan-Meier plot by the group variable for the kidney data

Figure 9 shows the Kaplan-Meier survival curves for the two groups. The two survival curves can be seen to cross each other. Before a time point around month 8, the estimated survival probability (infection-free probability) of the surgical group is higher

than that of the percutaneous group; after that time point, however, the estimated survival probability of the surgical group falls away relative to that of the percutaneous group. The crossing survival curves suggest that the proportional-hazards assumption does not hold. The violation of the PH assumption can also be diagnosed from a graph generated by the `stphplot` command (graph not shown).

In figure 10, we use the `stcoxkm` command to overlay the Cox predicted survival curves and the Kaplan–Meier survival curves. As expected, the curves do not match well. In figure 11, we use the `sttrkm` command to overlay the threshold regression predicted survival curves and the Kaplan–Meier survival curves. Clearly, compared with the Cox predicted curves (figure 10), the threshold regression predicted curves (figure 11) match the Kaplan–Meier curves better. This application of `sttrkm` produces a positive estimate for μ of 0.542 for the percutaneous group (the regression output is not shown). The susceptibility rate p_0 for this group is estimated to be 0.219, and hence the estimated cure rate is $1 - 0.219 = 0.781$. The threshold regression predicted curve is slightly miscalibrated for the percutaneous group in this application of standard threshold regression.

To improve the fit, we switch to the threshold regression cure-rate model. We use the `cure` and `lgtp(group)` options inside the `sttrkm` command to overlay the threshold regression cure-rate model predicted curves and the Kaplan–Meier curves in figure 12. There is a noticeable improvement in fit for the percutaneous group when compared with figure 11. The output for this last threshold regression analysis appears momentarily. The susceptibility rate p , when estimated as a free parameter, is 0.244 for the percutaneous group. Twice the difference of the maximum log-likelihood values for the standard and cure-rate models gives $2[-114.4747 - (-116.4909)] = 4.032$. Comparison of this test statistic with a χ_1^2 distribution gives a p -value of 0.045, which suggests that p_0 and p differ at the 0.05 significance level.

Next we illustrate how to use the `stthreg` and `trhr` commands for the threshold regression cure-rate model. In `stthreg`, we specify the `cure` option to fit a threshold regression cure-rate model. In addition to the `lny0()` and `mu()` options, we also use the `lgtp()` option to incorporate the `group` independent variable in the logit link function for the `lgtp()` parameter in this threshold regression cure-rate model. The `trhr` command is used after the `stthreg` command to calculate the hazard-ratio values at month 2 and month 20, based on the threshold regression cure-rate model fit by `stthreg`.

```
. xi: stthreg, lny0(i.group) mu(i.group) lgtp(i.group) cure
i.group      _Igroup_1-2      (naturally coded; _Igroup_1 omitted)
      failure_d: infection
      analysis time _t: time
initial:      log likelihood = -141.26094
alternative:  log likelihood =  -154.434
rescale:      log likelihood = -137.01023
rescale eq:   log likelihood = -130.91466
Iteration 0:  log likelihood = -130.91466 (not concave)
Iteration 1:  log likelihood = -124.57115
Iteration 2:  log likelihood = -116.61429
```

```

Iteration 3: log likelihood = -114.87662
Iteration 4: log likelihood = -114.6138
Iteration 5: log likelihood = -114.51506
Iteration 6: log likelihood = -114.48821
Iteration 7: log likelihood = -114.47805
Iteration 8: log likelihood = -114.47546
Iteration 9: log likelihood = -114.47485
Iteration 10: log likelihood = -114.47474
Iteration 11: log likelihood = -114.47471
Iteration 12: log likelihood = -114.4747
ml model estimated; type -ml display- to display results

```

```

Threshold Regression Cure Rate Model Estimates      Number of obs   =      119
                                                    Wald chi2(3)    =      30.34
Log likelihood = -114.4747                        Prob > chi2     =      0.0000

```

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
lny0						
_Igroup_2	-1.297609	.2357168	-5.50	0.000	-1.759606	-.835613
_cons	1.411293	.1434588	9.84	0.000	1.130119	1.692467
mu						
_Igroup_2	.0958833	.3969795	0.24	0.809	-.6821821	.8739488
_cons	-.095869	.0764876	-1.25	0.210	-.2457819	.054044
lgtp						
_Igroup_2	-14.90171	958.0936	-0.02	0.988	-1892.731	1862.927
_cons	13.76995	958.0933	0.01	0.989	-1864.058	1891.598

```

. trhr, var(group) timevalue(2 20)
Hazard Ratio for Scenario , at Time = 2

```

var	Hazard Ratio
_Igroup_2	2.4673714479

```

Hazard Ratio for Scenario , at Time = 20

```

var	Hazard Ratio
_Igroup_2	.04444226172

Finally, we use two `trpredict` commands to predict the eight quantities $[\ln \hat{y}_0, \hat{y}_0, \hat{\mu}, \widehat{lgtp}, \hat{p}, f(t|\hat{\mu}, \hat{y}_0), S(t|\hat{\mu}, \hat{y}_0), \text{ and } h(t|\hat{\mu}, \hat{y}_0)]$ by this threshold regression cure-rate model for the two scenarios: `group=surgical` and `group=percutaneous`. For example, the estimates for y_0 for the surgical group and the percutaneous group are 4.101 and 1.120, respectively. The estimates indicate that patients in the surgical group have a much higher initial health status than those in the percutaneous group. The estimates for the susceptibility rate p for the surgical group and the percutaneous group in the research population are 1 and 0.244, respectively. And hence the estimates for the cure rate $1 - p$ for these two groups in the research population are 0 and 0.756, respectively.

```

. trpredict, scenario(_Igroup_2=0) prefix(Surgical_) replace
. trpredict, scenario(_Igroup_2=1) prefix(Percutaneous_) replace

```

8 Conclusion

In this article, we introduced a package to implement threshold regression in Stata. Threshold regression is a newly developed methodology in the area of survival-data analysis. Applications of threshold regression can be carried out easily with the help of our package.

9 Acknowledgment

This project is supported in part by CDC/NIOSH Grant RO1 OH008649.

10 References

- Cox, D. R. 1972. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society, Series B* 34: 187–220.
- Cox, D. R., and H. D. Miller. 1965. *The Theory of Stochastic Processes*. Boca Raton, FL: Chapman & Hall/CRC.
- Drzewiecki, K. T., and P. K. Andersen. 1982. Survival with malignant melanoma: A regression analysis of prognostic factors. *Cancer* 49: 2414–2419.
- Garrett, J. M. 1997. sbe14: Odds ratios and confidence intervals for logistic regression models with effect modification. *Stata Technical Bulletin* 36: 15–22. Reprinted in *Stata Technical Bulletin Reprints*, vol. 6, pp. 104–114. College Station, TX: Stata Press.
- Gould, W., J. Pitblado, and B. Poi. 2010. *Maximum Likelihood Estimation with Stata*. 4th ed. College Station, TX: Stata Press.
- Klein, J. P., and M. L. Moeschberger. 2003. *Survival Analysis: Techniques for Censored and Truncated Data*. 2nd ed. New York: Springer.
- Lee, M.-L. T., and G. A. Whitmore. 2006. Threshold regression for survival analysis: Modeling event times by a stochastic process reaching a boundary. *Statistical Science* 21: 501–513.
- . 2010. Proportional hazards and threshold regression: Their theoretical and practical connections. *Lifetime Data Analysis* 16: 196–214.
- Nahman, N. S., Jr., D. F. Middendorf, W. H. Bay, R. McElligott, S. Powell, and J. Anderson. 1992. Modification of the percutaneous approach to peritoneal dialysis catheter placement under peritoneoscopic visualization: Clinical results in 78 patients. *Journal of the American Society of Nephrology* 3: 103–107.

About the authors

Tao Xiao is a PhD candidate in biostatistics in the College of Public Health at The Ohio State University. His current research interests include the first-hitting-time models for survival analysis, the Bayesian methodology related to such models, and statistical computing.

G. A. Whitmore is an emeritus professor at McGill University in Montreal and an associate investigator at the Ottawa Hospital Research Institute. His current research interests include statistical modeling and analysis of event histories, survival data, and risk processes, with an emphasis on applications in health care and medicine. Recent publications include studies of birth weight and chronic obstructive pulmonary disease.

Xin He is an assistant professor in the Department of Epidemiology and Biostatistics at the University of Maryland in College Park. His current research interests include longitudinal data analysis, survival analysis, nonparametric and semiparametric methods, as well as applications in clinical trials, epidemiology, and other studies related to public health.

Dr. Mei-Ling Ting Lee is a professor in and chair of the Department of Epidemiology and Biostatistics and is director of the Biostatistics and Risk Assessment Center at the University of Maryland. He has published in a broad spectrum of research areas, including statistical methods for genomic studies; threshold regression models for risk assessments, with applications in cancer, environmental research, and occupational exposure; and rank-based nonparametric tests for clustered data. Dr. Lee has published a book entitled *Analysis of Microarray Gene Expression Data* and has coedited two other books. He is the founding editor and editor-in-chief of the international journal *Lifetime Data Analysis*, which specializes in modeling time-to-event data. The journal is currently publishing its eighteenth volume.