SAS/STAT® 14.1 User’s Guide

The ICLIFETEST Procedure

This document is an individual chapter from *SAS/STAT® 14.1 User's Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2015. *SAS/STAT® 14.1 User's Guide*. Cary, NC: SAS Institute Inc.

SAS/STAT® 14.1 User's Guide

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

July 2015

Chapter 62

The ICLIFETEST Procedure

Contents

Overview: ICLIFETEST Procedure

The ICLIFETEST procedure performs nonparametric survival analysis for interval-censored data. You can use the ICLIFETEST procedure to compute nonparametric estimates of the survival functions and to examine the equality of the survival functions through statistical tests.

The ICLIFETEST procedure compares most closely to the LIFETEST procedure. The two procedures share the same analytic objectives: estimating and summarizing subjects' survival experiences and comparing them systematically. The distinction between these procedures lies in the types of data that they are designed to handle. The ICLIFETEST procedure is intended primarily for handling interval-censored data, whereas the LIFETEST procedure deals exclusively with right-censored data. You can use the ICLIFETEST procedure to analyze data that are left-censored, interval-censored, or right-censored. However, if the data to be analyzed contain only exact or right-censored observations, it is recommended that you use the LIFETEST procedure because it provides specialized methods for dealing with right-censored data.

Features

The main features of the ICLIFETEST procedure are as follows:

- **Nonparametric estimation**
  - uses the efficient EMICM algorithm (Wellner and Zhan 1997) to estimate survival functions by default
  - supports Turnbull's algorithm (Turnbull 1976) and the iterative convex minorant (ICM) algorithm (Groeneboom and Wellner 1992)
  - computes standard errors of the survival estimates by using a multiple imputation method or a bootstrap method
  - supports several transformation-based confidence intervals
  - produces survival plots
- **Survival comparisons**
  - provides the weighted generalized log-rank test
  - supports a variety of weight functions for testing early or late differences
  - supports a stratified test for survival differences within predefined populations
  - supports a trend test for ordered alternatives
  - supports multiple-comparison functionalities

The ICLIFETEST procedure uses ODS Graphics to create graphs as part of its output. For general information about ODS Graphics, see Chapter 21, "Statistical Graphics Using ODS."

Getting Started: ICLIFETEST Procedure

This example illustrates the use of the ICLIFETEST procedure to estimate the survival function and test for equality of survival functions by using interval-censored data from a breast cancer study.

The data consist of observations on 94 subjects from a retrospective study that compares the risks of breast cosmetic deterioration after tumorectomy (Finkelstein 1986). There are two treatment groups: patients who receive radiation alone (TRT=RT) and patients who receive radiation plus chemotherapy (TRT=RCT). Patients are followed up every four to six months, leading to interval-censored observations of deterioration times. Of the 94 observation times, 38 are right-censored and the remaining 56 are censored into intervals of finite length.

The following statements create a SAS data set named RT for the group that receives radiation alone. The variable lTime provides the last follow-up time at which cosmetic deterioration has not occurred for the patient, and the variable rTime provides the last follow-up time immediately after the event. Note that for the ICLIFETEST procedure to recognize the observations as right-censored, their right bounds must be represented by missing values in the input data set.

```
data RT;
   input lTime rTime @@;
   trt = ' RT';
   datalines;
45  .  25 37  37  .
 6 10  46  .   0  5
 0  7  26 40  18  .
46  .  46  .  24  .
46  .  27 34  36  .
 7 16  36 44   5 11
17  .  46  .  19 35
 7 14  36 48  17 25
37 44  37  .  24  .
 0  8  40  .  32  .
 4 11  17 25  33  .
15  .  46  .  19 26
11 15  11 18  37  .
22  .  38  .  34  .
46  .   5 12  36  .
46  .
;
```

The following statements create a SAS data set named RCT for patients who receive both radiation and chemotherapy:

```
data RCT;
   input lTime rTime @@;
   trt = 'RCT';
   datalines;
 8 12   0  5  30 34
 0 22   5  8  13  .
24 31  12 20  10 17
```

```
   17 27  11  .   8 21
   17 23  33 40   4  9
   24 30  31  .  11  .
   16 24  13 39  14 19
   13  .  19 32   4  8
   11 13  34  .  34  .
   16 20  13  .  30 36
   18 25  16 24  18 24
   17 26  35  .  16 60
   32  .  15 22  35 39
   23  .  11 17  21  .
   44 48  22 32  11 20
   14 17  10 35  48  .
   ;
```

The following statements combine the data sets RT and RCT into a single data set named BCS that is to be analyzed by the ICLIFETEST procedure:

```
   data BCS;
      set RT RCT;
   run;
```

Suppose you want to gain insights into the incidence rate of cosmetic deterioration over time after the two treatments. From the perspective of survival analysis, the tasks are to estimate the survival probabilities for the two treatment groups and to test for a systematic difference between the groups.

The following statements invoke the ICLIFETEST procedure to estimate the survival functions for both treatment groups:

```
   ods graphics on;
   proc iclifetest plots=(survival logsurv) data=BCS impute(seed=1234);
      strata trt;
      time (lTime, rTime);
   run;
```

In the TIME statement, the variables that represent the interval boundaries, lTime and rTime, are enclosed in parentheses and separated by a comma. Because the treatment indicator variable, Trt, is specified in the STRATA statement, PROC ICLIFETEST conducts the analysis separately for each treatment group. When you specify the keywords SURVIVAL and LOGSURV in the PLOTS= option, the procedure plots the estimated survival functions and the negative of the log transformations of the estimates. You can specify an integer seed for the random number generator that is used in creating imputed data sets for calculating standard errors of the survival estimates. If the SEED= option is not specified, a random seed is obtained from the computer's clock. For more information about the IMPUTE option, see the section "IMPUTE < (*options*) >" on page 4671.

Figure 62.1 displays the nonparametric survival estimates for the radiation group (TRT=RT).

Figure 62.1 Nonparametric Survival Estimates

The ICLIFETEST Procedure

Stratum 1: trt = RT

Nonparametric Survival Estimates

Time Interval		Probability Estimate		Imputation Standard Error
		Failure	Survival	
0	4	0.0000	1.0000	0.0000
5	6	0.0463	0.9537	0.0354
7	7	0.0797	0.9203	0.0458
8	11	0.1684	0.8316	0.0580
12	24	0.2391	0.7609	0.0629
25	33	0.3318	0.6682	0.0706
34	38	0.4136	0.5864	0.0739
40	46	0.5344	0.4656	0.0758
48	Inf	1.0000	0.0000	0.0000

Note that because interval censoring is present, the nonparametric estimates of failure probability and survival probability are computed for a set of nonoverlapping intervals. These estimates are constant within each interval. For example, consider the time interval $(8, 11]$. The estimated failure probability is 0.1684 between 8 and 11. The interval $(11, 12]$, which is not displayed, is an interval for which the survival estimate cannot be uniquely determined; such intervals are referred to as Turnbull intervals. The failure probability increases to 0.2391 at 12 and remains constant until 24. The table in Figure 62.1 also contains standard errors for the estimates. For a plot of the survival probabilities, see Figure 62.4.

Figure 62.2 displays estimated quartiles of the deterioration time and corresponding 95% confidence limits for the group that receives radiation only.

Figure 62.2 Quartile Estimates

Quartile Estimates

Percentile	Point Estimate	Transform	95% Confidence Interval	
			[Lower	Upper)
75	.	LOGLOG	.	.
50	40	LOGLOG	34	48
25	25	LOGLOG	8	34

Because the estimated failure distribution is undefined inside the Turnbull intervals, the quartiles of deterioration time and their confidence intervals are obtained by imputing probabilities within the intervals. This approach is ad hoc in nature, so the results should be interpreted with caution. For more information, see the section "Quartile Estimation" on page 4687.

The ICLIFETEST procedure also produces a summary of the frequencies of various types of censoring (Figure 62.3).

Figure 62.3 Summary of Various Censoring Types

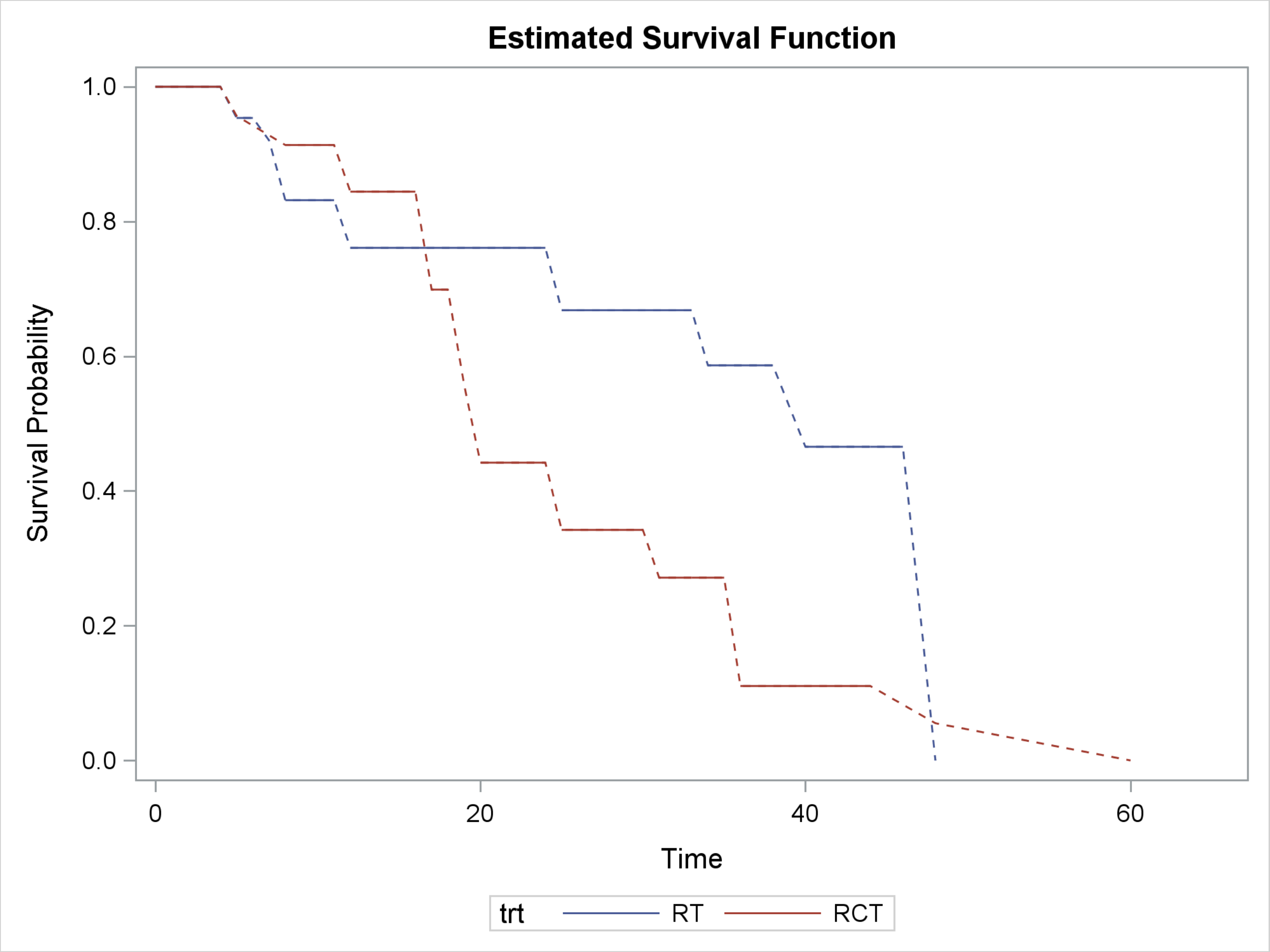
Number of Censored and Uncensored Values

Stratum			Type of Censoring			
ID	**trt**	**Total**	**Left**	**Interval**	**Right**	**Uncensored**
1	**RT**	46	3 (6.5%)	18 (39.1%)	25 (54.3%)	0 (0.0%)
2	**RCT**	48	2 (4.2%)	33 (68.8%)	13 (27.1%)	0 (0.0%)
Total		94	5 (5.3%)	51 (54.3%)	38 (40.4%)	0 (0.0%)

The categories are exact times, left-censoring, right-censoring, and interval-censoring. The percentage of each category within a stratum is also calculated.

Figure 62.4 displays the estimated survival functions for the two treatments.

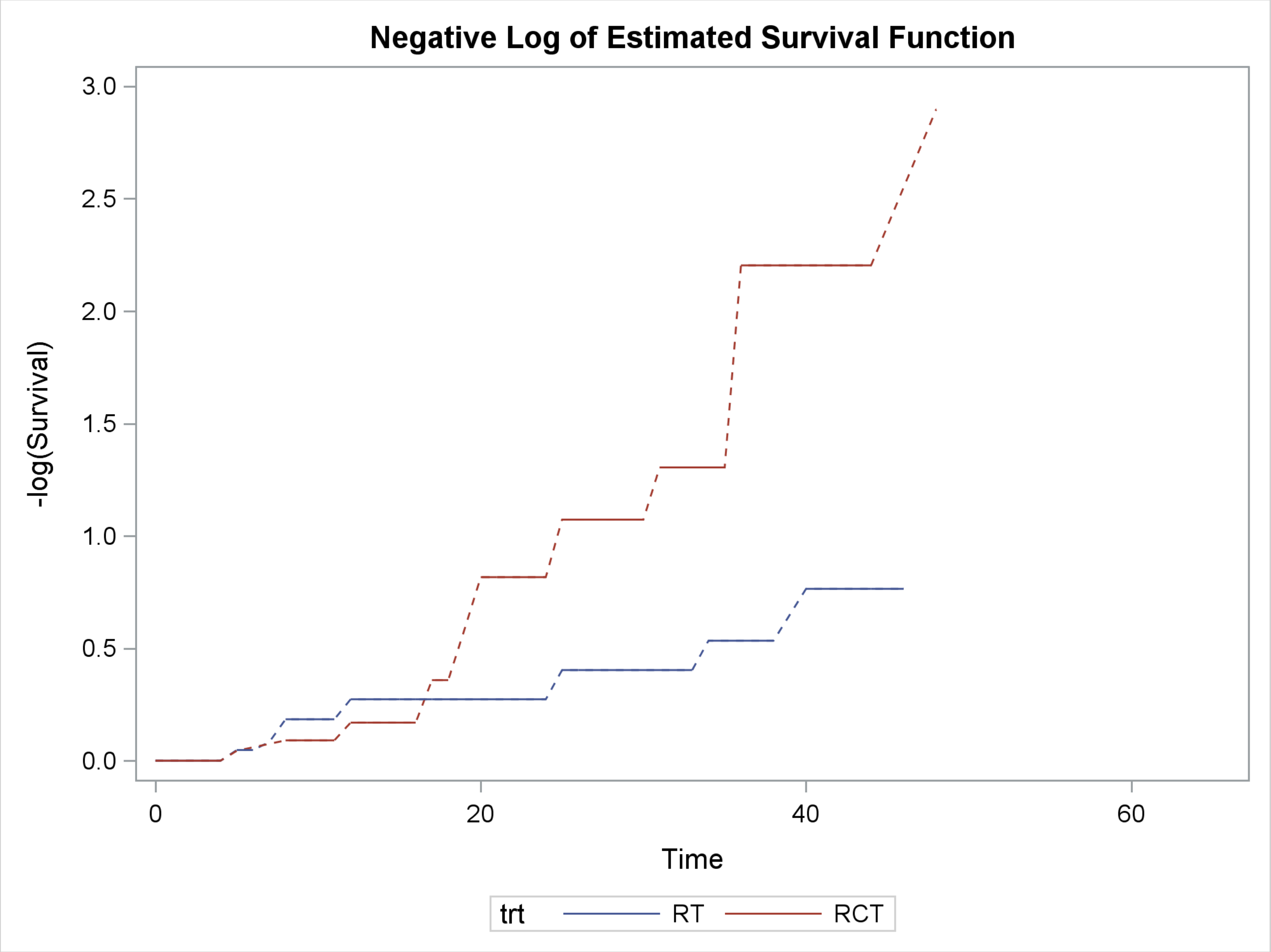
Figure 62.4 Nonparametric Survival Estimates by Treatment



Clearly, the group that receives radiation alone tends to survive longer before experiencing cosmetic deterioration than the group that receives both radiation and chemotherapy. As shown in Figure 62.1, the estimated survival probabilities are undetermined within the Turnbull intervals. For ease of visualization, dashed lines are plotted across the Turnbull intervals for which the estimates are not defined. You can suppress plotting of the dashed lines by specifying the NODASH option. For more information about options for customizing the survival plots, see the section "SURVIVAL <(CL | FAILURE | NODASH | STRATA | TEST)>" on page 4675.

Figure 62.5 displays log transformations of the survival estimates over time.

Figure 62.5 Negative Log of Nonparametric Survival Estimates



The following statements formally test whether patients in the two treatment groups have the same survival rate. By default, the generalized log-rank test with constant weights over time is used. For more information about alternative weight functions, see the section "Generalized Log-Rank Statistic" on page 4689. You can specify an integer seed for the random number generator that is used in creating imputed data sets for obtaining standard errors of the survival estimates and for calculating the generalized log-rank test. If the SEED= option is not specified, a random seed is obtained from the computer's clock.

```
proc iclifetest data=BCS impute(seed=1234);
   time (lTime, rTime);
   test trt;
run;
```

Figure 62.6 displays results of the generalized log-rank test.

Figure 62.6 Generalized Log-Rank Statistics

The ICLIFETEST Procedure

Test of Equality over Group

Weight	Chi-Square	DF	Pr > Chi-Square
SUN	7.3349	1	0.0068

The chi-square statistic for the generalized log-rank test is 7.3349. The *p*-value of 0.0068 is computed using the chi-square distribution with one degree of freedom.

Syntax: ICLIFETEST Procedure

The following statements are available in the ICLIFETEST procedure:

PROC ICLIFETEST < *options* > ;
  BY *variables* ;
  FREQ *variable* ;
  STRATA *variables* ;
  TEST *variable* < /*options* > ;
  TIME (*variable,variable*) ;

At a minimum, you must provide the PROC and TIME statements, and you must specify the left and right boundaries of the intervals in the TIME statements. For instance, the following statements compute a nonparametric estimate of the survival function:

```
proc iclifetest;
   time (Left, Right);
run;
```

You use the STRATA statement to obtain separate survival estimates for different groups of the input data. The ICLIFETEST procedure produces an individual survival estimate for each group that is formed by one or more variables that you specify in the STRATA statement.

You use the TEST statement to test whether the underlying survival functions are the same between the groups. When you specify both the TEST and STRATA statements, PROC ICLIFETEST produces a stratified test in which the comparisons are conditional on the strata. The variables that you specify in the STRATA statement must be different from the variables that you specify in the TEST statement. Note that this setup is different from that of the LIFETEST procedure, in which survival comparisons are handled implicitly by the STRATA statement.

The rest of this section provides detailed syntax information for each statement, beginning with the PROC ICLIFETEST statement. The remaining statements are covered in alphabetical order.

PROC ICLIFETEST Statement

PROC ICLIFETEST < *options* > **;**

The PROC ICLIFETEST statement invokes the ICLIFETEST procedure. Table 62.1 summarizes the options available in the PROC ICLIFETEST statement.

Table 62.1 Options Available in the PROC ICLIFETEST Statement

Option	Description
Input and Output Data Sets	
DATA=	Specifies the input SAS data set
OUTSURV=	Names an output data set to contain survival estimates and confidence limits
Nonparametric Estimation	
BOOTSTRAP	Requests that the simple bootstrap method be used for computing standard errors of the survival estimates
IMPUTE	Specifies details for multiple imputations used for computing standard errors and estimating covariance matrix of the generalized log-rank statistic
ITHISTORY	Displays the iteration history of nonparametric estimation
MAXITER=	Set the maximum number of iterations allowed for the estimation method
METHOD=	Specifies the method to estimate the survival function
TOLLIKE=	Sets the log-likelihood convergence criterion
Confidence Limits for Survival Function	
ALPHA=	Sets the level for confidence interval estimation
CONFTYPE=	Specifies the transformation applied to the survival function for computing confidence limits
ODS Graphics	
MAXTIME=	Specifies the maximum time value for plotting
PLOTS=	Specifies plots to display
Control Output	
NOPRINT	Suppresses the display of printed output
NOSUMMARY	Suppresses the display of summary of censored and uncensored values
SHOWTI	Displays the Turnbull intervals
Miscellaneous	
ALPHAQT=	Sets the level for confidence intervals for survival time quartiles
MISSING	Treats missing values as a valid stratum or group for the variables specified in the STRATA and TEST statements
SINGULAR=	Sets the tolerance for testing singularity of the covariance matrix of test statistics

If no options are specified, PROC ICLIFETEST computes and displays a nonparametric estimate of the survival function. If ODS Graphics is enabled, a plot of the estimated survival function is also displayed.

You can specify the following options.

ALPHA=α

specifies the level α for the $100(1-\alpha)\%$ confidence intervals for the survival functions. For example, ALPHA=0.05 requests 95% confidence limits. By default, ALPHA=0.05.

ALPHAQT=α

specifies the level α for the $100(1-\alpha)\%$ confidence intervals for the quartiles of the survival time. For example, ALPHAQT=0.05 requests a 95% confidence interval. By default, ALPHAQT=0.05.

BOOTSTRAP < (*options*) >

BOOT < (*options*) >

performs simple bootstrap sampling and computes bootstrap standard errors of nonparametric survival estimates. You can specify the following suboptions to control how the bootstrap samples are generated:

NBOOT

specifies the number of bootstrap samples to be generated. The default value is 1,000.

SEED

specifies the seed to generate bootstrap samples. The default seed is selected randomly.

CONFTYPE=ASINSQRT | LINEAR | LOG | LOGLOG

specifies the transformation to be applied to $S(t)$ to obtain pointwise confidence intervals for the survival function in addition to the confidence intervals for the quartiles of the survival times. You can specify the following keywords:

ASINSQRT specifies the arcsine–square root transformation,

$$g(x) = \sin^{-1}(\sqrt{x})$$

LOGLOG specifies the log–log transformation,

$$g(x) = \log(-\log(x))$$

This is also referred to as the log cumulative hazard transformation, because it applies the log transformation to the cumulative hazard function. Collett (1994) and Lachin (2000) call it the complementary log-log transformation.

LINEAR specifies the identity transformation,

$$g(x) = x$$

LOG specifies the log transformation,

$$g(x) = \log(x)$$

LOGIT specifies the logit transformation,

$$g(x) = \log\left(\frac{x}{1-x}\right)$$

By default, CONFTYPE=LOGLOG.

DATA=*SAS-data-set*

names the SAS data set that PROC ICLIFETEST reads. By default, the most recently created SAS data set is used.

IMPUTE < **(***options***)** **>**

IM < **(***options***)** **>**

specifies details for multiple imputations. You can specify the following two suboptions to control how the samples are generated:

NIMSE

specifies the number of imputation samples to be generated for computing standard errors of the survival estimates. The default value is 1,000.

NIMTEST

specifies the number of imputation samples to be generated for estimating the covariance matrix of the generalized log-rank statistic. The default value is 1,000.

SEED

specifies the seed to generate imputation data sets. The default seed is selected randomly.

ITHISTORY

prints the iteration history of the nonparametric estimation, including the log likelihood, the probability estimate that is associated with each Turnbull interval, and whether the EM or ICM method is used for each iteration.

MAXITER=*n*

MAXIT=*n*

specifies the maximum number of iterations for estimating the survival function. The default value depends on the estimation method as follows:

- EMICM method: 200
- ICM method: 200
- Turnbull method: 500

Note that you specify the method with the METHOD= option.

MAXTIME=*value*

specifies the maximum value of the time variable that is allowed in plots so that outlying points do not determine the scale of the time axis of the plots. This option affects only the way in which plots are displayed and has no effect on any calculations.

METHOD=TURNBULL | ICM | EMICM

specifies the method to be used for computing survival function estimates. You can specify the following:

TURNBULL

EM

requests that the Turnbull method be used to estimate the survival function.

EMICM

requests that the EMICM algorithm be used to estimate the survival function.

ICM

requests that the iterative convex minorant algorithm be used to estimate the survival function.

By default, METHOD=EMICM.

MISSING

allows missing values to be a stratum level or a valid group for the variables that are specified in the STRATA and TEST statements.

NOPRINT

suppresses the display of output. This option is useful when only an output data set is needed. It temporarily disables the Output Delivery System (ODS); for more information about ODS, see Chapter 20, "Using the Output Delivery System."

NOSUMMARY

suppresses the summary table of the number of censored and uncensored values.

OUTSURV=*SAS-data-set*

OUTS=*SAS-data-set*

creates an output SAS data set to contain the estimates of the survival function and corresponding confidence limits for all the strata. For more information about the contents of the OUTSURV= data set, see the section "OUTSURV= Data Set" on page 4695.

PLOTS< (*global-plot-options***) >=***plot-request* **< (***options***) >**

PLOTS< (*global-plot-options***) >=(***plot-request* **< (***options***) > < . . .** *plot-request* **< (***options***) > >)**

controls the plots that are produced by using ODS Graphics. When you specify only one *plot-request*, you can omit the parentheses around it. Here are some examples:

```
plots=none
plots=(survival(cl) logsurv)
plots(only)=hazard
```

ODS Graphics must be enabled before plots can be requested. For example:

```
ods graphics on;

proc iclifetest plots=survival;
   time (Left,Right);
run;

ods graphics off;
```

For more information about enabling and disabling ODS Graphics, see the section "Enabling and Disabling ODS Graphics" on page 609 in Chapter 21, "Statistical Graphics Using ODS."

If ODS Graphics is enabled but you do not specify the PLOTS= option, then by default PROC ICLIFETEST produces a plot of the estimated survival functions.

You can specify the following *global-plot-option*:

ONLY

specifies that only the specified plots in the list be produced; otherwise, the default survival function plot is also displayed.

You can specify the following *plot-requests* and their options:

ALL

produces all appropriate plots. Specifying PLOTS=ALL is equivalent to specifying PLOTS=(SURVIVAL LOGSURV LOGLOGS HAZARD).

HAZARD < **(***hazard-options***)** >

H < *hazard-options* >

plots the kernel-smoothed hazard functions. You can specify the following *hazard-options*.

BANDWIDTH=*value* | *numeric-list* | **RANGE(***lower,upper***)**

BW=

specifies the bandwidth for kernel-smoothed estimation of the hazard function. You can specify one of the following:

value

sets the bandwidth to the given *value*.

numeric-list

selects the bandwidth from the specified *numeric-list* that minimizes the cross validation pseudo-likelihood.

RANGE(*lower,upper***)**

selects from the interval (*lower, upper*) the bandwidth that minimizes a cross validation pseudo-likelihood function. PROC ICLIFETEST uses the golden section search algorithm to find the minimum. If the interval contains more than one local minimum, there is no guarantee that the local minimum that the algorithm finds is also the global minimum.

For more information about the cross validation pseudo-likelihood function, see the section "Kernel-Smoothed Estimation" on page 4687. By default, BANDWIDTH= RANGE(0.1b,0.25b), where $b = g_u - g_l$; g_l is the left boundary of the first Turnbull interval; and g_u is the right boundary of the last Turnbull interval if it is finite or the left boundary of that interval multiplied by *1.1* otherwise.

SAMPLING=LEAVEONE | RANDOM

specifies how to partition the data set to form cross validation groups. You can specify the following values:

LEAVEONE

partitions the data set into leave-one-out subsets.

RANDOM

randomly assigns observations to cross validation groups with equal probabilities.

By default, SAMPLING=RANDOM.

GRIDL=*number*

specifies the lower grid limit for the kernel-smoothed estimate. The default value is the time origin.

GRIDU=*number*

specifies the upper grid limit for the kernel-smoothed estimate. The default value is the maximum input boundary value.

KERNEL=*kernel-option*

specifies the kernel to be used. You can specify the following values:

BIWEIGHT

BW

uses the following kernel: $K_{BW}(x) = \frac{15}{16}(1 - x^2)^2, \qquad -1 \le x \le 1$

EPANECHNIKOV

E

uses the following kernel: $K_E(x) = \frac{3}{4}(1 - x^2), \qquad -1 \le x \le 1$

UNIFORM

U

uses the following kernel: $K_U(x) = \frac{1}{2}, \qquad -1 \le x \le 1$

By default, KERNEL=EPANECHNIKOV.

CVFOLD=*number*

specifies the number of cross validation groups. This option is applicable only when SAMPLING=RANDOM. The default *number* is either 5 or the sample size, whichever value is less.

CVGRID=*number*

specifies the number of grid points to use in determining the cross validation pseudo-likelihood. By default, CVGRID=51.

HGRID=*number*

specifies the number of grid points for discretizing the smoothed hazard function. By default, HGRID=101.

LOGLOGS

LLS

plots the log of the negative log of the estimated survival function versus the log of time.

LOGSURV

LS

plots the negative log of the estimated survival function versus time.

NONE

suppresses all plots.

SURVIVAL < (CL | FAILURE | NODASH | STRATA | TEST) >

S < (CL | FAILURE | STRATA | TEST) >

plots the estimated survival function. You can customize the display by specifying the following values:

CL

displays pointwise confidence limits for the survival function.

FAILURE

F

changes all the displays for survival function to those for the failure function. For example, if you specify both the FAILURE and CL options, the plot displays the failure curves in addition to pointwise confidence limits for the failure function.

NODASH

suppresses the plotting of dashed lines as linear interpolations for the Turnbull intervals.

STRATA=INDIVIDUAL | OVERLAY | PANEL

specifies how to display the survival or failure curves for multiple strata. This option has no effect if there is only one stratum. You can specify one of the following values:

INDIVIDUAL

UNPACK

requests that a separate plot be displayed for each stratum.

OVERLAY

requests that the survival or failure curves for the strata be overlaid on the same plot.

PANEL

requests that separate plots for the strata be organized into panels of two or four plots, depending on the number of strata.

By default, STRATA=OVERLAY.

TEST

displays the *p*-value for a *K*-sample test that is specified in the TEST statement.

SHOWTI

presents the survival estimates in terms of Turnbull intervals.

SINGULAR=*value*

sets the tolerance for testing the singularity of the covariance matrix of test statistics.

TOLLIKE=*value*

sets the log-likelihood convergence criterion for the estimation algorithm. The default *value* depends on the estimation method as follows:

- EMICM method: 10^{-10}
- ICM method: 10^{-10}
- Turnbull method: 10^{-8}

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC ICLIFETEST to obtain separate analyses of observations in groups that are defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in order of the BY variables. If you specify more than one BY statement, only the last one specified is used.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the NOTSORTED or DESCENDING option in the BY statement for the ICLIFETEST procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

For more information about BY-group processing, see the discussion in *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the discussion in the *Base SAS Procedures Guide*.

FREQ Statement

FREQ *variable* **;**

The FREQ statement names a variable that provides frequencies for each observation in the DATA= data set. Specifically, if n is the value of the FREQ variable for a given observation, then that observation is used n times.

The analysis that is produced by using a FREQ statement reflects the expanded number of observations. You can produce the same analysis (without the FREQ statement) by first creating a new data set that contains the expanded number of observations. For example, if the value of the FREQ variable is 5 for the first observation, the first five observations in the new data set are identical. Each observation in the old data set is replicated n_i times in the new data set, where n_i is the value of the FREQ variable for that observation.

If the value of the FREQ variable is missing or is less than 1, the observation is not used in the analysis. If the value is not an integer, only the integer portion is used.

STRATA Statement

STRATA *variables* **;**

The STRATA statement identifies the variables that determine the stratum levels. You can specify more than one variable in the STRATA statement. By default, strata are formed according to the nonmissing values of these variables. However, missing values are treated as a valid stratum level if you specify the MISSING option in the PROC ICLIFETEST statement.

The specification of several STRATA *variables*, such as

```
   strata A B C;
```

is equivalent to the A*B*C syntax of the TABLES statement in the FREQ procedure. The number of stratum levels usually grows very rapidly along with the number of STRATA variables, so you must be cautious when specifying the list of STRATA variables.

If you specify the STRATA statement without specifying a TEST statement, the ICLIFETEST procedure produces an individual survival estimate for each stratum. On the other hand, if you specify both the STRATA and TEST statements, the ICLIFETEST procedure produces a stratified test in which the comparisons are made among the groups that are formed by the levels of the TEST variable within strata. Also, individual survival estimates are computed for all the groups that are formed by the levels of the TEST variable within a stratum.

In contrast with the STRATA statement in the LIFETEST procedure, the STRATA statement in PROC ICLIFETEST does not support the options for specifying various weight functions, stratified tests, or trend tests and to make multiple-comparison adjustments for paired differences. Instead, PROC ICLIFETEST provides these options in the TEST statement.

TEST Statement

TEST *variable < /options >* **;**

In the preceding syntax, variable is a variable whose values determine the groups to be tested. The values for variable can be formatted or unformatted. If variable is a character or numeric variable, then the groups are defined by the unique values of the TEST variable. You can specify only one variable in the TEST statement.

When you are comparing more than two survival curves, a generalized log-rank test tells you whether the curves are significantly different from each other, but it does not identify which pairs of curves are different. A multiple-comparison adjustment of the p-values for the paired comparisons retains the same overall probability of a Type I error as the K-sample test. Two types of paired comparisons can be made: comparisons between all pairs of curves and comparisons between a control curve and all other curves. You use the DIFF= option to specify the comparison type, and you use the ADJUST= option to select a method of multiple-comparison adjustments.

Compared with the TEST statement in the LIFETEST procedure, the TEST statement in PROC ICLIFETEST is designed for comparing survival between predefined groups. Unlike the LIFETEST procedure, PROC ICLIFETEST does not support a similar test for detecting association with multiple covariates.

Table 62.2 summarizes the *options* available in the TEST statement.

Table 62.2 Options Available in the TEST Statement

Option	Description
Homogeneity Tests	
NODETAIL	Suppresses printing the test statistic and covariance matrix
NOTEST	Suppresses all tests
OUTSCORE=	Names an output data set to contain the scores derived from the permutation form of the generalized log-rank test

Table 62.2 *continued*

Option	Description
TREND	Requests a trend test
WEIGHT=	Specifies tests that correspond to various weight functions
Multiple Comparisons	
ADJUST=	Requests a multiple-comparison adjustment
DIFF=	Specifies the type of differences to consider

You can specify the following *options* in the TEST statement after a slash (/).

ADJUST=*method*

specifies the multiple-comparison method to use for adjusting the p-values of the paired tests. For mathematical details, see the section "Multiple-Comparison Adjustments" on page 4692; also see Westfall et al. (1999). You can specify the following adjustment *methods*:

BONFERRONI

BON

applies the Bonferroni correction to the raw p-values.

DUNNETT

performs Dunnett's two-tailed comparisons of the control group to all other groups. PROC ICLIFETEST uses the factor-analytic covariance approximation that is described in Hsu (1992) and identifies the adjustment in the results as "Dunnett-Hsu." ADJUST=DUNNETT is incompatible with DIFF=ALL.

SCHEFFE

performs Scheffé's multiple-comparison adjustment.

SIDAK

applies the Šidák correction to the raw p-values.

SMM

GTE

performs the paired comparisons based on the studentized maximum modulus test.

TUKEY

performs the paired comparisons based on Tukey's studentized range test. PROC ICLIFETEST uses the approximation that is described in Kramer (1956) and identifies the adjustment as "Tukey-Kramer" in the results. ADJUST=TUKEY is incompatible with DIFF=CONTROL.

SIMULATE < (*simulate-options***) >**

computes the adjusted p-values from the simulated distribution of the maximum or maximum absolute value of a multivariate normal random vector. The simulation estimates q, the true $(1-\alpha)$ quantile, where α is the value of the ALPHA= *simulate-option.*

The number of samples for the simulation adjustment is set so that the tail area for the simulated q is within a certain accuracy radius γ of $1-\alpha$, with an accuracy confidence of $100(1-\epsilon)\%$. In equation form,

$$\Pr(|F(\hat{q}) - (1-\alpha)| \le \gamma) \quad = \quad 1-\epsilon$$

where $\hat{q}$ is the simulated q and F is the true distribution function of the maximum; for more information, see Edwards and Berry (1987). By default, $\gamma = 0.005$ and $\epsilon = 0.01$, so the tail area of $\hat{q}$ is within 0.005 of 0.95 with 99% confidence.

You can specify the following *simulate-options*:

ACC=*value*

specifies the target accuracy radius γ of a $100(1-\epsilon)\%$ confidence interval for the true probability content of the estimated $(1-\alpha)$ quantile. By default, ACC=0.005.

ALPHA=*value*

specifies the value α for estimating the $(1-\alpha)$ quantile. The default *value* is the ALPHA= value in the PROC ICLIFETEST statement, or 0.05 if that option is not specified.

EPS=*value*

specifies the value ϵ for a $100(1-\epsilon)\%$ confidence interval for the true probability content of the estimated $(1-\alpha)$ quantile. The default *value* for the accuracy confidence is 99%, corresponding to EPS=0.01.

NSAMP=*n*

specifies the sample size for the simulation. By default, *n* is set based on the values of the target accuracy radius γ and accuracy confidence $100(1-\epsilon)\%$ for an interval for the true probability content of the estimated $(1-\alpha)$ quantile. With the default values for γ, ϵ, and α (0.005, 0.01, and 0.05, respectively), by default NSAMP=12604.

REPORT

specifies that a report of the simulation be displayed, including a listing of the parameters, such as γ, ϵ, and α, in addition to an analysis of various methods of estimating or approximating the quantile.

SEED=*number*

specifies an integer used to start the pseudorandom number generator for the simulation. If you do not specify a seed, or if you specify a value less than or equal to 0, the seed is generated by default from reading the time of day from the computer's clock.

DIFF=ALL | CONTROL< (*'string'* < . . . , *'string'*>**) >**

specifies which pairs of survival curves to consider for the multiple comparisons. You can specify the following values:

ALL

requests all paired comparisons.

CONTROL < (*'string'* < . . . *'string'*>**) >**

requests comparisons of the control survival curve with all other survival curves. To specify the control curve, you specify the quoted strings of formatted values that represent the curve in parentheses. For example, if CELL=LARGE identifies the control group, you specify

```
DIFF=CONTROL('large')
```

If more than one variable is used to identify the curves (for example, if CELL=LARGE and SEX=F represent the control), you specify

```
DIFF=CONTROL('large' 'F')
```

The order of the quoted strings should correspond to the order of the TEST variable. If no *string* is specified as the control, the first group value is used.

By default, DIFF=ALL unless you specify ADJUST= DUNNETT, in which case DIFF=CONTROL.

NODETAIL

suppresses the display of the generalized log-rank statistics and the corresponding covariance matrices. If you specify the TREND option, the display of the scores for computing the trend test is suppressed.

NOTEST

suppresses the K-sample tests, stratified tests, and trend tests.

OUTSCORE=*SAS-data-set*

OUTSC=*SAS-data-set*

creates an output SAS data set to contain subject scores that are derived from a permutation form of the generalized log-rank statistic. For more information about the contents of the OUTSCORE= data set, see the section "OUTSCORE= Data Set" on page 4696.

TREND

computes the trend test for the null hypothesis that the survival rates are the same for the groups versus an ordered alternative. If the TEST variable is numeric, the unformatted values of the variable are used as the scores; otherwise, the scores are $1, 2, \ldots,$ in the order of the strata. For more information, see the section "Trend Tests" on page 4694.

WEIGHT=*test-request* **| (***test-request* < . . . *test-request*>**)**

requests weights to be applied to the generalized log-rank statistics to perform various tests. For more information about the various weight functions that the ICLIFETEST procedure supports, see the section "Generalized Log-Rank Statistic" on page 4689. You can specify the following *test-requests*:

FAY specifies the test that uses Fay's weights (Fay 1999).

FINKELSTEIN specifies the test that uses Finkelstein's weights (Finkelstein 1986).

FLEMING($\rho 1$, $\rho 2$) specifies the family of tests in Harrington and Fleming (1982), where $\rho 1$ and $\rho 2$ are nonnegative numbers. FLEMING($\rho 1$,$\rho 2$) corresponds to the Fleming-Harrington G^{ρ} family (Fleming and Harrington 1981) when $\rho 2$=0. Alternatively, you can specify a test in this family as FLEMING(ρ) with one argument. When ρ=0, the test becomes the log-rank test. When ρ=1, the test should be very close to the Peto-Peto test.

NONE suppresses all comparison tests. Specifying WEIGHT=NONE is equivalent to specifying NOTEST.

SUN specifies the test that uses Sun's weights (Sun 1996).

TIME Statement

TIME **(***variable, variable* **)** **;**

The TIME statement names the variables that represent the interval-censored observations. The variables should be numeric. For example, the following statement identifies Left as the lower bound and Right as the upper bound of an observed interval:

```
time (Left, Right);
```

Observations for which Left < Right and both values are nonnegative are used. If the value for Left is missing or 0 and the value for Right is not missing, then the observation is treated as left-censored. If the Right value is missing and the value for Left is not missing, then the observation is treated as right-censored. Observations for which both variables are missing are not analyzed.

Details: ICLIFETEST Procedure

Statistical Methods

Nonparametric Estimation of the Survival Function

Suppose the event times for a total of n subjects, T_1, T_2, ..., T_n, are independent random variables with an underlying cumulative distribution function $F(t)$. Denote the corresponding survival function as $S(t)$. Interval-censoring occurs when some or all T_i's cannot be observed directly but are known to be within the interval $(L_i, R_i]$, $L_i \leq R_i$.

The observed intervals might or might not overlap. It they do not overlap, then you can usually use conventional methods for right-censored data, with minor modifications. On the other hand, if some intervals overlap, you need special algorithms to compute an unbiased estimate of the underlying survival function.

To characterize the nonparametric estimate of the survival function, Peto (1973) and Turnbull (1976) show that the estimate can jump only at the right endpoint of a set of nonoverlapping intervals (also known as Turnbull intervals), $\{I_j = (q_j, p_j], j = 1, \ldots, m\}$. A simple algorithm for finding these intervals is to order all the boundary values $\{L_i, R_i, i = 1, \ldots, n\}$ with labels of L and R attached and then pick up the intervals that have L as the left boundary and R as the right boundary. For example, suppose that the data set contains only three intervals, $(1,3]$, $(2,4]$, and $(5,6]$. The ordered values are $1(L), 2(L), 3(R), 4(R), 5(L), 6(R)$. Then the Turnbull intervals are $(2,3]$ and $(5,6]$.

For the exact observation $L_i = R_i = t$, Ng (2002) suggests that it be represented by the interval $(t - \epsilon, t)$ for a positive small value ϵ. If $R_j = t$ for an observation $(L_j, R_j]$ $(L_j < R_j)$, then the observation is represented by $(L_j + \epsilon, R_j - \epsilon)$.

Define $\theta_j = P(T \in I_j)$, $j = 1, \ldots, m$. Given the data, the survival function, $S(t)$, can be determined only up to equivalence classes t, which are complements of the Turnbull intervals. $S(t)$ is undefined if t is within some I_j. The likelihood function for $\boldsymbol{\theta} = \{\theta_j, j = 1, \ldots, m\}$ is then

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} \Big(\sum_{j=1}^{m} \alpha_{ij}\theta_j \Big)$$

where α_{ij} is 1 if I_j is contained in $(L_i, R_i]$ and 0 otherwise.

Denote the maximum likelihood estimate for $\hat{\boldsymbol{\theta}}$ as $\hat{\boldsymbol{\theta}} = \{\hat{\theta}_j, j = 1, \ldots, m\}$. The survival function can then be estimated as

$$\hat{S}(t) = \sum_{k:p_k>t} \hat{\theta}_k, \quad t \notin \text{any } I_j, j = 1, \ldots, m$$

Estimation Algorithms

Peto (1973) suggests maximizing this likelihood function by using a Newton-Raphson algorithm subject to the constraint $\sum_{i=j}^{m} \theta_j = 1$. This approach has been implemented in the %ICE macro. Although feasible, the optimization becomes less stable as the dimension of $\boldsymbol{\theta}$ increases.

Treating interval-censored data as missing data, Turnbull (1976) derives a self-consistent equation for estimating the θ_j's:

$$\theta_j = \frac{1}{n}\sum_{i=1}^{n} \mu_{ij}(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \frac{\alpha_{ij}\theta_j}{\sum_{j=1}^{m} \alpha_{ij}\theta_j}$$

where $\mu_{ij}(\boldsymbol{\theta})$ is the expected probability that the event T_i occurs within I_j for the ith subject, given the observed data.

The algorithm is an expectation-maximization (EM) algorithm in the sense that it iteratively updates $\boldsymbol{\theta}$ and $\mu_{ij}(\boldsymbol{\theta})$. Convergence is declared if, for a chosen number $\epsilon > 0$,

$$\sum_{j=1}^{m} |\hat{\theta}_j^{(l)} - \hat{\theta}_j^{(l-1)}| < \epsilon$$

where $\hat{\theta}_j^{(l)}$ denotes the updated value for θ_j after the lth iteration.

An alternative criterion is to declare convergence when increments of the likelihood are small:

$$|L(\hat{\theta}_j^{(l)}; j = 1, \ldots, m) - L(\hat{\theta}_j^{(l-1)}; j = 1, \ldots, m)| < \epsilon$$

There is no guarantee that the converged values constitute a maximum likelihood estimate (MLE). Gentleman and Geyer (1994) introduced the Kuhn-Tucker conditions based on constrained programming as a check of whether the algorithm converges to a legitimate MLE. These conditions state that a sufficient and necessary condition for the estimate to be a MLE is that the Lagrange multipliers $\gamma_j = n - c_j$ are nonnegative for all the θ_j's that are estimated to be zero, where c_j is the derivative of the log-likelihood function with respect to θ_j:

$$c_j = \frac{\partial \log(L)}{\partial \theta_j} = \sum_{i=1}^{n} \frac{\alpha_{ij}}{\sum_{j=1}^{m} \alpha_{ij}\theta_j}$$

You can use Turnbull's method by specifying METHOD=TURNBULL in the ICLIFETEST statement. The Lagrange multipliers are displayed in the "Nonparametric Survival Estimates" table.

Groeneboom and Wellner (1992) propose using the iterative convex minorant (ICM) algorithm to estimate the underlying survival function as an alternative to Turnbull's method. Define $\beta_j = F(p_j)$, $j = 1, \ldots, m$ as the cumulative probability at the right boundary of the jth Turnbull interval: $\beta_j = \sum_{k=1}^{j} \theta_k$. It follows that $\beta_m = 1$. Denote $\beta_0 = 0$ and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_{m-1})'$. You can rewrite the likelihood function as

$$L(\boldsymbol{\beta}) = \prod_{i=1}^{n} \sum_{j=1}^{m} \alpha_{ij} (\beta_j - \beta_{j-1})$$

Maximizing the likelihood with respect to the θ_j's is equivalent to maximizing it with respect to the β_j's. Because the β_j's are naturally ordered, the optimization is subject to the following constraint:

$$C = \{\mathbf{x} = (\beta_1, \ldots, \beta_{m-1}) : 0 \le \beta_1 \le \cdots \le \beta_{m-1} \le 1\}$$

Denote the log-likelihood function as $l(\boldsymbol{\beta})$. Suppose its maximum occurs at $\hat{\boldsymbol{\beta}}$. Mathematically, it can be proved that $\hat{\boldsymbol{\beta}}$ equals the maximizer of the following quadratic function:

$$g^*(\mathbf{x}|\mathbf{y}, \mathbf{W}) = -\frac{1}{2}(\mathbf{x} - \mathbf{y})'\mathbf{W}(\mathbf{x} - \mathbf{y})$$

where $\mathbf{y} = \hat{\boldsymbol{\beta}} + \mathbf{W}^{-1}\nabla l(\hat{\boldsymbol{\beta}})$, $\nabla l(\cdot)$ denotes the derivatives of $l(\cdot)$ with respect to $\boldsymbol{\beta}$, and $\mathbf{W}$ is a positive definite matrix of size $(m-1) \times (m-1)$ (Groeneboom and Wellner 1992).

An iterative algorithm is needed to determine $\hat{\boldsymbol{\beta}}$. For the lth iteration, the algorithm updates the quantity

$$\mathbf{y}^{(l)} = \hat{\boldsymbol{\beta}}^{(l-1)} - \mathbf{W}^{-1}(\hat{\boldsymbol{\beta}}^{(l-1)})\nabla l(\hat{\boldsymbol{\beta}}^{(l-1)})$$

where $\hat{\boldsymbol{\beta}}^{(l-1)}$ is the parameter estimate from the previous iteration and $\mathbf{W}(\hat{\boldsymbol{\beta}}^{(l-1)}) = \mathrm{diag}(w_j, j = 1, \ldots, m-1)$ is a positive definite diagonal matrix that depends on $\hat{\boldsymbol{\beta}}^{(l-1)}$.

A convenient choice for $\mathbf{W}(\boldsymbol{\beta})$ is the negative of the second-order derivative of the log-likelihood function $l(\boldsymbol{\beta})$:

$$w_j = w_j(\boldsymbol{\beta}) = -\frac{\partial^2}{\partial \beta_j^2} l(\boldsymbol{\beta})$$

Given $\mathbf{y} = \mathbf{y}^{(l)} = (y_1^{(l)}, \ldots, y_{m-1}^{(l)})'$ and $\mathbf{W} = \mathbf{W}(\hat{\boldsymbol{\beta}}^{(l-1)})$, the parameter estimate for the lth iteration $\hat{\boldsymbol{\beta}}^{(l)}$ maximizes the quadratic function $g^*(\mathbf{x}|\mathbf{y}, \mathbf{W})$.

Define the cumulative sum diagram $\{P_k, k = 0, \ldots, m-1\}$ as a set of m points in the plane, where $P_0 = (0, 0)$ and

$$P_k = \Big(\sum_{i=1}^{k} w_i, \sum_{i=1}^{k} w_i y_i^{(l)} \Big)$$

Technically, $\hat{\boldsymbol{\beta}}^{(l)}$ equals the left derivative of the convex minorant, or in other words, the largest convex function below the diagram $\{P_k, k = 0, \ldots, m-1\}$. This optimization problem can be solved by the pool-adjacent-violators algorithm (Groeneboom and Wellner 1992).

Occasionally, the ICM step might not increase the likelihood. Jongbloed (1998) suggests conducting a line search to ensure that positive increments are always achieved. Alternatively, you can switch to the EM step, exploiting the fact that the EM iteration never decreases the likelihood, and then resume iterations of the ICM algorithm after the EM step. As with Turnbull's method, convergence can be determined based on the closeness of two consecutive sets of parameter values or likelihood values. You can use the ICM algorithm by specifying METHOD=ICM in the PROC ICLIFETEST statement.

As its name suggests, the EMICM algorithm combines the self-consistent EM algorithm and the ICM algorithm by alternating the two different steps in its iterations. Wellner and Zhan (1997) show that the converged values of the EMICM algorithm always constitute an MLE if it exists and is unique. The ICLIFETEST procedure uses the EMICM algorithm as the default.

Variance Estimation of the Survival Estimator

Peto (1973) and Turnbull (1976) suggest estimating the variances of the survival estimates by inverting the Hessian matrix, which is obtained by twice differentiating the log-likelihood function. This method can become less stable when the number of θ_j's increase as n increases. Simulations have shown that the confidence limits based on variances estimated with this method tend to have conservative coverage probabilities that are greater than the nominal level (Goodall, Dunn, and Babiker 2004).

Sun (2001) proposes using two resampling techniques, simple bootstrap and multiple imputation, to estimate the variance of the survival estimator. The undefined regions that the Turnbull intervals represent create a special challenge using the bootstrap method. Because each bootstrap sample could have a different set of Turnbull intervals, some time points to evaluate the variances based on the original Turnbull intervals might be located within the intervals in a bootstrap sample, with the result that their survival probabilities become unknown. A simple ad hoc solution is to shrink the Turnbull interval to its right boundary and modify the survival estimates into a right continuous function:

$$\hat{S}_m(t) = \sum_{j:p_j>t} \hat{\theta}_j$$

Let M denote the number of resampling data sets. Let $A_1^k, \ldots, A_n^k$ denote the n independent samples from the original data with replacement, $k = 1, \ldots, M$. Let $\hat{S}_m^k(t)$ be the modified estimate of the survival function computed from the kth resampling data set. Then you can estimate the variance of $\hat{S}(t)$ by the sample variance as

$$\hat{\sigma}_b^2(t) = \frac{1}{M-1} \sum_{k=1}^{M} [\hat{S}_m^k(t) - \bar{S}_m(t)]^2$$

where

$$\bar{S}_m(t) = \frac{\sum_{k=1}^{M} \hat{S}_m^k(t)}{M}$$

The method of multiple imputations exploits the fact that interval-censored data reduce to right-censored data when all interval observations of finite length shrink to single points. Suppose that each finite interval has been converted to one of the p_j values it contains. For this right-censored data set, you can estimate the variance of the survival estimates via the well-known Greenwood formula as

$$\hat{\sigma}_G^2(t) = \hat{S}_{KM}^2(t) \sum_{q_j<t} \frac{d_j}{n_j(n_j - d_j)}$$

where d_j is the number of events at time p_j and n_j is the number of subjects at risk just prior to p_j, and $\hat{S}_{KM}(t)$ is the Kaplan-Meier estimator of the survival function,

$$\hat{S}_{KM}(t) = \prod_{q_j < t} \frac{n_j - d_j}{n_j}$$

Essentially, multiple imputation is used to account for the uncertainty of ranking overlapping intervals. The kth imputed data set is obtained by substituting every interval-censored observation of finite length with an exact event time randomly drawn from the conditional survival function:

$$\hat{S}_i(t) = \frac{\hat{S}_m(t) - \hat{S}_m(R_i+)}{\hat{S}_m(L_i) - \hat{S}_m(R_i+)}, \quad t \in (L_i, R_i]$$

Because $\hat{S}_m(t)$ only jumps at the p_j, this is a discrete function.

Denote the Kaplan-Meier estimate of each imputed data set as $\hat{S}^k_{KM}(t)$. The variance of $\hat{S}(t)$ is estimated by

$$\hat{\sigma}^2_I(t) = \hat{S}^2(t) \sum_{q_j < t} \frac{d'_j}{n'_j(n'_j - d'_j)} + \frac{1}{M-1} \sum_{k=1}^{M} [\hat{S}^k_{KM}(t) - \bar{S}_{KM}(t)]$$

where

$$\bar{S}_{KM}(t) = \frac{1}{M} \sum_{k=1}^{M} \hat{S}^k_{KM}(t)$$

and

$$d'_j = \sum_{i=1}^{n} \frac{\alpha_{ij}[\hat{S}(p_{j-1}) - \hat{S}(p_j)]}{\sum_{j=1}^{m} \alpha_{ij}[\hat{S}(p_{j-1}) - \hat{S}(p_j)]}$$

and

$$n'_j = \sum_{k=j}^{m} d'_j$$

Note that the first term in the formula for $\hat{\sigma}^2_I(t)$ mimics the Greenwood formula but uses expected numbers of deaths and subjects. The second term is the sample variance of the Kaplan-Meier estimates of imputed data sets, which accounts for between-imputation contributions.

Pointwise Confidence Limits of the Survival Function

Pointwise confidence limits can be computed for the survival function given the estimated standard errors. Let α be specified by the ALPHA= option. Let $z_{\alpha/2}$ be the critical value for the standard normal distribution. That is, $\Phi(-z_{\alpha/2}) = \alpha/2$, where Φ is the cumulative distribution function of the standard normal random variable.

Constructing the confidence limits for the survival function $S(t)$ as $\hat{S}(t) \pm z_{\alpha/2}\hat{\sigma}[\hat{S}(t)]$ might result in an estimate that exceeds the range [0,1] at extreme values of t. This problem can be avoided by applying a transformation to $S(t)$ so that the range is unrestricted. In addition, certain transformed confidence intervals for $S(t)$ perform better than the usual linear confidence intervals (Borgan and Liestøl 1990). You can use the CONFTYPE= option to set one of the following transformations: the log-log function (Kalbfleisch and Prentice 1980), the arcsine–square root function (Nair 1984), the logit function (Meeker and Escobar 1998), the log function, and the linear function.

Let g denote the transformation that is being applied to the survival function $S(t)$. Using the delta method, you estimate the standard error of $g(\hat{S}(t))$ by

$$\tau(t) = \hat{\sigma}\left[g(\hat{S}(t))\right] = g'\left(\hat{S}(t)\right)\hat{\sigma}[\hat{S}(t)]$$

where g' is the first derivative of the function g. The 100(1 – α)% confidence interval for $S(t)$ is given by

$$g^{-1}\left\{g[\hat{S}(t)] \pm z_{\frac{\alpha}{2}} g'[\hat{S}(t)]\hat{\sigma}[\hat{S}(t)]\right\}$$

where g^{-1} is the inverse function of g. The choices for the transformation g are as follows:

- arcsine–square root transformation: The estimated variance of $\sin^{-1}\left(\sqrt{\hat{S}(t)}\right)$ is $\hat{\tau}^2(t) = \frac{\hat{\sigma}^2[\hat{S}(t)]}{4\hat{S}(t)[1-\hat{S}(t)]}$. The 100(1 – α)% confidence interval for $S(t)$ is given by

$$\sin^2\left\{\max\left[0, \sin^{-1}(\sqrt{\hat{S}(t)}) - z_{\frac{\alpha}{2}}\hat{\tau}(t)\right]\right\} \le S(t) \le \sin^2\left\{\min\left[\frac{\pi}{2}, \sin^{-1}(\sqrt{\hat{S}(t)}) + z_{\frac{\alpha}{2}}\hat{\tau}(t)\right]\right\}$$

- linear transformation: This is the same as the identity transformation. The 100(1 – α)% confidence interval for $S(t)$ is given by

$$\hat{S}(t) - z_{\frac{\alpha}{2}}\hat{\sigma}\left[\hat{S}(t)\right] \le S(t) \le \hat{S}(t) + z_{\frac{\alpha}{2}}\hat{\sigma}\left[\hat{S}(t)\right]$$

- log transformation: The estimated variance of $\log(\hat{S}(t))$ is $\hat{\tau}^2(t) = \frac{\hat{\sigma}^2(\hat{S}(t))}{\hat{S}^2(t)}$. The 100(1 – α)% confidence interval for $S(t)$ is given by

$$\hat{S}(t)\exp\left(-z_{\frac{\alpha}{2}}\hat{\tau}(t)\right) \le S(t) \le \hat{S}(t)\exp\left(z_{\frac{\alpha}{2}}\hat{\tau}(t)\right)$$

- log-log transformation: The estimated variance of $\log(-\log(\hat{S}(t))$ is $\hat{\tau}^2(t) = \frac{\hat{\sigma}^2[\hat{S}(t)]}{[\hat{S}(t)\log(\hat{S}(t))]^2}$. The 100(1 – α)% confidence interval for $S(t)$ is given by

$$\left[\hat{S}(t)\right]^{\exp\left(z_{\frac{\alpha}{2}}\hat{\tau}(t)\right)} \le S(t) \le \left[\hat{S}(t)\right]^{\exp\left(-z_{\frac{\alpha}{2}}\hat{\tau}(t)\right)}$$

- logit transformation: The estimated variance of $\log\left(\frac{\hat{S}(t)}{1-\hat{S}(t)}\right)$ is

$$\hat{\tau}^2(t) = \frac{\hat{\sigma}^2(\hat{S}(t))}{\hat{S}^2(t)[1-\hat{S}(t)]^2}$$

The 100(1 – α)% confidence limits for $S(t)$ are given by

$$\frac{\hat{S}(t)}{\hat{S}(t) + \left[1-\hat{S}(t)\right]\exp\left(z_{\frac{\alpha}{2}}\hat{\tau}(t)\right)} \le S(t) \le \frac{\hat{S}(t)}{\hat{S}(t) + \left[1-\hat{S}(t)\right]\exp\left(-z_{\frac{\alpha}{2}}\hat{\tau}(t)\right)}$$

Quartile Estimation

The first quartile (25th percentile) of the survival time is the time beyond which 75% of the subjects in the population under study are expected to survive. For interval-censored data, it is problematic to define point estimators of the quartiles based on the survival estimate $\hat{S}(t)$ because of its undefined regions of Turnbull intervals. To overcome this problem, you need to impute survival probabilities within the Turnbull intervals. The previously defined estimator $\hat{S}_m(t)$ achieves this by placing all the estimated probabilities at the right boundary of the interval. The first quartile is estimated by

$$q_{.25} = \min\{t_j | \hat{S}_m(t_j) < 0.75\}$$

If $\hat{S}_m(t)$ is exactly equal to 0.75 from t_j to t_{j+1}, the first quartile is taken to be $(t_j + t_{j+1})/2$. If $\hat{S}_m(t)$ is greater than 0.75 for all values of t, the first quartile cannot be estimated and is represented by a missing value in the printed output.

The general formula for estimating the 100p percentile point is

$$q_p = \min\{t_j | \hat{S}_m(t_j) < 1 - p\}$$

The second quartile (the median) and the third quartile of survival times correspond to $p = 0.5$ and $p = 0.75$, respectively.

Brookmeyer and Crowley (1982) constructed the confidence interval for the median survival time based on the confidence interval for the survival function $S(t)$. The methodology is generalized to construct the confidence interval for the 100p percentile based on a g-transformed confidence interval for $S(t)$ (Klein and Moeschberger 1997). You can use the CONFTYPE= option to specify the g-transformation. The $100(1-\alpha)\%$ confidence interval for the first quantile survival time is the set of all points t that satisfy

$$\left| \frac{g(\hat{S}_m(t)) - g(1 - 0.25)}{g'(\hat{S}_m(t))\hat{\sigma}(\hat{S}(t))} \right| \le z_{1-\frac{\alpha}{2}}$$

where $g'(x)$ is the first derivative of $g(x)$ and $z_{1-\frac{\alpha}{2}}$ is the $100(1-\frac{\alpha}{2})$ percentile of the standard normal distribution.

Kernel-Smoothed Estimation

After you obtain the survival estimate $\hat{S}(t)$, you can construct a discrete estimator for the cumulative hazard function. First, you compute the jumps of the discrete function as

$$\hat{\lambda}_j = \frac{c_j \hat{\theta}_j}{\sum_{k=j}^{m} c_k \hat{\theta}_k}, \quad j = 1, \ldots, m$$

where the c_j's have been defined previously for calculating the Lagrange multiplier statistic.

Essentially, the numerator and denominator estimate the number of failures and the number at risks that are associated with the Turnbull intervals. Thus these quantities estimate the increments of the cumulative hazard function over the Turnbull intervals.

The estimator of the cumulative hazard function is

$$\hat{\Lambda}(t) = \sum_{k:p_k<t} \hat{\lambda}_k, \quad t \notin \text{any } I_j$$

Like $\hat{S}(t)$, $\hat{\lambda}(t)$ is undefined if t is located within some Turnbull interval I_j. To facilitate applying the kernel-smoothed methods, you need to reformulate the estimator so that it has only point masses. An ad hoc approach would be to place all the mass for a Turnbull interval at the right boundary. The kernel-based estimate of the hazard function is computed as

$$\tilde{h}(t,b) = -\frac{1}{b}\sum_{j=1}^{m} K\Big(\frac{t-p_j}{b}\Big)\hat{\lambda}_j$$

where $K(\cdot)$ is a kernel function and $b > 0$ is the bandwidth. You can estimate the cumulative hazard function by integrating $\tilde{h}(t,b)$ with respect to t.

Practically, an upper limit t_D is usually imposed so that the kernel-smoothed estimate is defined on $(0, t_D)$. The ICLIFETEST procedure sets the value depending on whether the right boundary of the last Turnbull interval is finite or not: $t_D = p_m$ if $p_m < \infty$ and $t_D = 1.2 * q_m$ otherwise.

Typical choices of kernel function are as follows:

- uniform kernel:

$$K_U(x) = \frac{1}{2}, \quad -1 \le x \le 1$$

- Epanechnikov kernel:

$$K_E(x) = \frac{3}{4}(1-x^2), \quad -1 \le x \le 1$$

- biweight kernel:

$$K_{BW}(x) = \frac{15}{16}(1-x^2)^2, \quad -1 \le x \le 1$$

For $t < b$, the symmetric kernels $K()$ are replaced by the corresponding asymmetric kernels of Gasser and Müller (1979). Let $q = \frac{t}{b}$. The modified kernels are as follows:

- uniform kernel:

$$K_{U,q}(x) = \frac{4(1+q^3)}{(1+q)^4} + \frac{6(1-q)}{(1+q)^3}x, \quad -1 \le x \le q$$

- Epanechnikov kernel:

$$K_{E,q}(x) = K_E(x)\frac{64(2-4q+6q^2-3q^3)+240(1-q)^2x}{(1+q)^4(19-18q+3q^2)}, \quad -1 \le x \le q$$

- biweight kernel:

$$K_{BW,q}(x) = K_{BW}(x)\frac{64(8-24q+48q^2-45q^3+15q^4)+1120(1-q)^3x}{(1+q)^5(81-168q+126q^2-40q^3+5q^4)}, \quad -1 \le x \le q$$

For $t_D - b \leq t \leq t_D$, let $q = \frac{t_D - t}{b}$. The asymmetric kernels for $t < b$ are used, with x replaced by $-x$.

The bandwidth parameter b controls how much "smoothness" you want to have in the kernel-smoothed estimate. For right-censored data, a commonly accepted method of choosing an optimal bandwidth is to use the mean integrated square error(MISE) as an objective criteria. This measure becomes difficult to adapt to interval-censored data because it no longer has a closed-form mathematical formula.

Pan (2000) proposes using a V-fold cross validation likelihood as a criterion for choosing the optimal bandwidth for the kernel-smoothed estimate of the survival function. The ICLIFETEST procedure implements this approach for smoothing the hazard function. Computing such a criterion entails a cross validation type procedure. First, the original data $\mathcal{D}$ are partitioned into V almost balanced subsets $\mathcal{D}^{(v)}$, $v = 1, \ldots, V$. Denote the kernel-smoothed estimate of the leave-one-subset-out data $\mathcal{D} - \mathcal{D}^{(v)}$ as $\hat{h}^{*(-v)}(t; b)$. The optimal bandwidth is defined as the one that maximizes the cross validation likelihood:

$$b_0 = \text{argmax}_b \sum_{v=1}^{V} L(\hat{h}^{*(-v)}(t; b) | \mathcal{D}^{(v)})$$

Comparison of Survival between Groups

If the TEST statement is specified, the ICLIFETEST procedure compares the K groups formed by the levels of the TEST variable using a generalized log-rank test. Let $S_k(t)$ be the underlying survival function of the kth group, $k = 1, \ldots, K$. The null and alternative hypotheses to be tested are

$$H_0 \colon S_1(t) = S_2(t) = \cdots = S_K(t) \text{ for all } t$$

versus

$$H_1 \colon \text{at least one of the } S_k(t)\text{'s is different for some } t$$

Let N_k denote the number of subjects in group k, and let n denote the total number of subjects ($n = N_1 + \cdots + N_K$).

Generalized Log-Rank Statistic

For the ith subject, let $\mathbf{z}_i = (z_{i1}, \ldots, z_{iK})'$ be a vector of K indicators that represent whether or not the subject belongs to the kth group. Denote $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K)'$, where β_k represents the treatment effect for the kth group. Suppose that a model is specified and the survival function for the ith subject can be written as

$$S(t|\mathbf{z}_i) = S(t|\mathbf{z}_i'\boldsymbol{\beta}, \boldsymbol{\gamma})$$

where $\boldsymbol{\gamma}$ denotes the nuisance parameters.

It follows that the likelihood function is

$$L = \prod_{i=1}^{n} [S(L_i|\mathbf{z}_i'\boldsymbol{\beta}, \boldsymbol{\gamma}) - S(R_i|\mathbf{z}_i'\boldsymbol{\beta}, \boldsymbol{\gamma})]$$

where (L_i, R_i) denotes the interval observation for the ith subject.

Testing whether or not the survival functions are equal across the K groups is equivalent to testing whether all the β_j's are zero. It is natural to consider a score test based on the specified model (Finkelstein 1986).

The score statistics for $\boldsymbol{\beta}$ are derived as the first-order derivatives of the log-likelihood function evaluated at $\boldsymbol{\beta} = \mathbf{0}$ and $\hat{\boldsymbol{\gamma}}$.

$$\mathbf{U} = (U_1, \ldots, U_K)' = \frac{\partial \log(L)}{\partial \boldsymbol{\beta}}\bigg|_{\boldsymbol{\beta}=\mathbf{0},\hat{\boldsymbol{\gamma}}}$$

where $\hat{\boldsymbol{\gamma}}$ denotes the maximum likelihood estimate for the $\boldsymbol{\gamma}$, given that $\boldsymbol{\beta} = \mathbf{0}$.

Under the null hypothesis that $\boldsymbol{\beta} = \mathbf{0}$, all K groups share the same survival function $S(t)$. It is typical to leave $S(t)$ unspecified and obtain a nonparametric maximum likelihood estimate $\hat{S}(t)$ using, for instance, Turnbull's method. In this case, $\boldsymbol{\gamma}$ represents all the parameters to be estimated in order to determine $\hat{S}(t)$.

Suppose the given data generates m Turnbull intervals as $\{I_j = (q_j, p_j], j = 1, \ldots, m\}$. Denote the probability estimate at the right end point of the jth interval by $\hat{\theta}_j$. The nonparametric survival estimate is $\hat{S}(t) = \sum_{k:p_k>t} \hat{\theta}_k$ for $t \notin$ any I_j.

Under the null hypothesis, Fay (1999) showed that the score statistics can be written in the form of a weighted log-rank test as

$$U_k = \sum_{j=1}^{m} U_{kj} = \sum_{j=1}^{m} v_j \left(d'_{kj} - \frac{n'_{kj}}{n'_j} d'_j \right)$$

where

$$v_j = \frac{[\hat{S}(p_j) - \hat{S}'(p_{j-1})][\hat{S}(p_{j-1}) - \hat{S}'(p_j)]}{\hat{S}(p_j)[\hat{S}(p_{j-1}) - \hat{S}(p_j)]}$$

and $S'(t)$ denotes the derivative of $S(t)$ with respect to $\boldsymbol{\beta}$.

d'_{kj} estimates the expected number of events within I_j for the kth group, and it is computed as

$$d'_{kj} = \sum_{i=1}^{n} z_{ik} \frac{\alpha_{ij} \hat{\theta}_j}{\sum_{l=1}^{m} \alpha_{il} \hat{\theta}_l}$$

d'_j is an estimate for the expected number of events within I_j for the whole sample, and it is computed as

$$d'_j = \sum_{k=1}^{K} d'_{kj}$$

Similarly, n'_{kj} estimates the expected number of subjects at risk before entering I_j for the kth group, and can be estimated by $n'_{kj} = \sum_{l=j}^{m} d'_{kl}$. n'_j is an estimate of the expected number of subjects at risk before entering I_j for all the groups: $n'_j = \sum_{k=1}^{K} n'_{kj}$.

Assuming different survival models gives rise to different weight functions v_j (Fay 1999). For example, Finkelstein's score test (1986) is derived assuming a proportional hazards model; Fay's test (1996) is based on a proportional odds model.

The choices of weight function are given in Table 62.3.

Table 62.3 Weight Functions for Various Tests

Test	v_j
Sun (1996)	1.0
Fay (1999)	$\hat{S}(p_{j-1})$
Finkelstein (1986)	$\frac{\hat{S}(p_{j-1})[\log\hat{S}(p_{j-1})-\log\hat{S}(p_j)]}{\hat{S}(p_{j-1})-\hat{S}(p_j)}$
Harrington-Fleming (*p*,*q*)	$[\hat{S}(p_{j-1})]^p[1-\hat{S}(p_{j-1})]^q, p \geq 0, q \geq 0$

Variance Estimation of the Generalized Log-Rank Statistic

Sun (1996) proposed the use of multiple imputation to estimate the variance-covariance matrix of the generalized log-rank statistic **U**. This approach is similar to the multiple imputation method as presented in "Variance Estimation of the Survival Estimator" on page 4684. Both methods impute right-censored data from interval-censored data and analyze the imputed data sets by using standard statistical techniques. Huang, Lee, and Yu (2008) suggested improving the performance of the generalized log-rank test by slightly modifying the variance calculation.

Suppose the given data generate m Turnbull intervals as $\{I_j = (q_j, p_j], j = 1, \ldots, m\}$. Denote the probability estimate for the *j*th interval as $\hat{\theta}_j$, and denote the nonparametric survival estimate as $\hat{S}(t) = \sum_{k:p_k>t}\hat{\theta}_k$ for $t \notin$ any I_j.

In order to generate an imputed data set, you need to randomly generate a survival time for every subject of the sample. For the *i*th subject, a random time T_i^* is generated randomly based on the following discrete survival function:

$$\hat{S}_i(T_i^* = p_j) = \frac{\hat{S}(q_j) - \hat{S}(R_i+)}{\hat{S}(L_i) - \hat{S}(R_i+)}, \quad p_j \in (L_i, R_i], \quad j = 1, \ldots, m$$

where $(L_i, R_i]$ denotes the interval observation for the subject.

For the *h*th imputed data set ($h = 1, \ldots, H$), let d_{kj}^h and n_{kj}^h denote the numbers of failures and subjects at risk by counting the imputed T_i^*'s for group *k*. Let d_j^h and n_j^h denote the corresponding pooled numbers.

You can perform the standard weighted log-rank test for right-censored data on each of the imputed data sets (Huang, Lee, and Yu 2008). The test statistic is

$$\mathbf{U}^h = (U_1^h, \ldots, U_K^h)'$$

where

$$U_k^h = \sum_{j=1}^{m} v_j \left(d_{kj}^h - \frac{n_{kj}^h}{n_j^h} d_j^h \right)$$

Its variance-covariance matrix is estimated by the Greenwood formula as

$$\mathbf{V}^h = \mathbf{V}_1^h + \cdots + \mathbf{V}_m^h$$

where

$$(\mathbf{V}_j^h)_{l_1 l_2} = \begin{cases} v_j^2 n_{l_1 j}^h (n_j^h - n_{l_1 j}^h d_j^h (n_j^h - d_j^h)(n_j^h)^{-2}(n_j^h - 1)^{-1}) & \text{when } l_1 = l_2 \\ -v_j^2 n_{l_1 j}^h n_{l_2 j}^h d_j^h (n_j^h - d_j^h)(n_j^h)^{-2}(n_j^h - 1)^{-1} & \text{when } l_1 \neq l_2 \end{cases}$$

After analyzing each imputed data set, you can estimate the variance-covariance matrix of **U** by pooling the results as

$$\hat{\mathbf{V}} = \frac{1}{H}\sum_{h=1}^{H}\mathbf{V}^h - \frac{1}{H-1}\sum_{h=1}^{H}[\mathbf{U}^h - \bar{\mathbf{U}}][\mathbf{U}^h - \bar{\mathbf{U}}]'$$

where

$$\bar{\mathbf{U}} = \frac{1}{H}\sum_{h=1}^{H}\mathbf{U}^h$$

The overall test statistic is formed as $\mathbf{U}'\mathbf{V}^{-}\mathbf{U}$, where $\mathbf{V}^{-}$ is the generalized inverse of $\mathbf{V}$. Under the null hypothesis, the statistic has a chi-squared distribution with degrees of freedom equal to the rank of $\mathbf{V}$. By default, the ICLIFETEST procedure perform 1000 imputations. You can change the number of imputations by the IMPUTE option in the PROC ICLIFETEST statement.

Stratified Tests

Suppose the generalized log-rank test is to be stratified on the M levels that are formed from the variables that you specify in the STRATA statement. Based only on the data of the sth stratum ($s = 1, \ldots, M$), let $\mathbf{U}_{(s)}$ be the test statistic for the sth stratum and let $V_{(s)}$ be the corresponding covariance matrix as constructed in the section "Variance Estimation of the Generalized Log-Rank Statistic" on page 4691. First, sum over the stratum-specific estimates as follows:

$$\mathbf{U}. = \sum_{s=1}^{M}\mathbf{U}_{(s)}$$

$$\mathbf{V}. = \sum_{s=1}^{M}\mathbf{V}_{(s)}$$

Then construct the global test statistic as

$$\mathbf{U}.'\mathbf{V}.^{-}\mathbf{U}.$$

Under the null hypothesis, the test statistic has a chi-squared distribution with degrees of freedom equal to the rank of $\mathbf{V}.$. The ICLIFETEST procedure performs the stratified test only when the groups to be compared are balanced across all the strata.

Multiple-Comparison Adjustments

When you have more than two groups, a generalized log-rank test tells you whether the survival curves are significantly different from each other, but it does not identify which pairs of curves are different. Pairwise comparisons can be performed based on the generalized log-rank statistic and the corresponding variance-covariance matrix. However, reporting all pairwise comparisons is problematic because the overall Type I error rate would be inflated. A multiple-comparison adjustment of the p-values for the paired comparisons retains the same overall probability of a Type I error as the K-sample test.

The ICLIFETEST procedure supports two types of paired comparisons: comparisons between all pairs of curves and comparisons between a control curve and all other curves. You use the DIFF= option to specify the comparison type, and you use the ADJUST= option to select a method of multiple-comparison adjustments.

Let χ^2_r denote a chi-square random variable with r degrees of freedom. Denote ϕ and Φ as the density function and the cumulative distribution function of a standard normal distribution, respectively. Let m be the number of comparisons; that is,

$$m = \begin{cases} \frac{k(k-1)}{2} & \text{DIFF} = \text{ALL} \\ k-1 & \text{DIFF} = \text{CONTROL} \end{cases}$$

For a two-sided test that compares the survival of the *j*th group with that of *l*th group, $1 \leq j \neq l \leq r$, the test statistic is

$$z^2_{jl} = \frac{(U_j - U_l)^2}{V_{jj} + V_{ll} - 2V_{jl}}$$

and the raw p-value is

$$p = \Pr(\chi^2_1 > z^2_{jl})$$

For multiple comparisons of more than two groups ($r > 2$), adjusted p-values are computed as follows:

- Bonferroni adjustment:

  $$p = \min\{1, m\Pr(\chi^2_1 > z^2_{jl})\}$$

- Dunnett-Hsu adjustment: With the first group defined as the control, there are $r-1$ comparisons to be made. Let $\mathbf{C} = (c_{ij})$ be the $(r-1) \times r$ matrix of contrasts that represents the $r-1$ comparisons; that is,

  $$c_{ij} = \begin{cases} 1 & i = 1, \ldots, r-1, j = 2, \ldots, r \\ -1 & j = i+1, i = 2, \ldots, r \\ 0 & \text{otherwise} \end{cases}$$

  Let $\boldsymbol{\Sigma} \equiv (\sigma_{ij})$ and $\mathbf{R} \equiv (r_{ij})$ be covariance and correlation matrices of $\mathbf{Cv}$, respectively; that is,

  $$\boldsymbol{\Sigma} = \mathbf{CVC}'$$

  and

  $$r_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}}$$

  The factor-analytic covariance approximation of Hsu (1992) is to find $\lambda_1, \ldots, \lambda_{r-1}$ such that

  $$\mathbf{R} = \mathbf{D} + \boldsymbol{\lambda}\boldsymbol{\lambda}'$$

  where $\mathbf{D}$ is a diagonal matrix whose *j*th diagonal element is $1 - \lambda_j$ and $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_{r-1})'$. The adjusted p-value is

  $$p = 1 - \int_{-\infty}^{\infty} \phi(y) \prod_{i=1}^{r-1} \left[\Phi\left(\frac{\lambda_i y + z_{jl}}{\sqrt{1-\lambda_i^2}} \right) - \Phi\left(\frac{\lambda_i y - z_{jl}}{\sqrt{1-\lambda_i^2}} \right) \right] dy$$

  This value can be obtained in a DATA step as

  $$p = \text{PROBMC("DUNNETT2"}, z_{ij}, ., ., r-1, \lambda_1, \ldots, \lambda_{r-1}).$$

- Scheffé adjustment:

$$p = \Pr(\chi^2_{r-1} > z^2_{jl})$$

- Šidák adjustment:

$$p = 1 - \{1 - \Pr(\chi^2_1 > z^2_{jl})\}^m$$

- SMM adjustment:

$$p = 1 - [2\Phi(z_{jl}) - 1]^m$$

  This can also be evaluated in a DATA step as

$$p = 1 - \mathrm{PROBMC}(\text{"MAXMOD"}, z_{jl}, ., ., m).$$

- Tukey adjustment:

$$p = 1 - \int_{-\infty}^{\infty} r\phi(y)[\Phi(y) - \Phi(y - \sqrt{2}z_{jl})]^{r-1}\, dy$$

  This can be evaluated in a DATA step as

$$p = 1 - \mathrm{PROBMC}(\text{"RANGE"}, \sqrt{2}z_{jl}, ., ., r).$$

Trend Tests

Trend tests for right-censored data (Klein and Moeschberger 1997, Section 7.4) can be extended to interval-censored data in a straightforward way. Such tests are specifically designed to detect ordered alternatives as

$$H_1\colon S_1(t) \geq S_2(t) \geq \cdots \geq S_K(t), t \leq \tau, \text{ with at least one inequality}$$

or

$$H_2\colon S_1(t) \leq S_2(t) \leq \cdots \leq S_K(t), t \leq \tau, \text{ with at least one inequality}$$

Let $a_1 < a_2 < \cdots < a_K$ be a sequence of scores associated with the K samples. Let $\mathbf{U} = (U_1, \ldots, U_K)$ be the generalized log-rank statistic and $\mathbf{V} = (V_{jl})$ be the corresponding covariance matrix of size $K \times K$ as constructed in the section "Variance Estimation of the Generalized Log-Rank Statistic" on page 4691. The trend test statistic and its variance are given by $\sum_{j=1}^K a_j U_j$ and $\sum_{j=1}^K \sum_{l=1}^K a_j a_l V_{jl}$, respectively. Under the null hypothesis that there is no trend, the following z-score has, asymptotically, a standard normal distribution:

$$Z = \frac{\sum_{j=1}^K a_j U_j}{\sqrt{\{\sum_{j=1}^K \sum_{l=1}^K a_j a_l V_{jl}\}}}$$

The ICLIFETEST procedure provides both one-tail and two-tail p-values for the test.

Scores for Permutation Tests

The weighted log-rank statistic can also be expressed as

$$\mathbf{U} = \sum_{i=1}^{n} \mathbf{z}_i c_i$$

where c_i is the score from the ith subject and follows the form

$$c_i = \frac{\hat{S}'(L_i) - \hat{S}'(R_i)}{\hat{S}(L_i) - \hat{S}(R_i)}$$

where $S'(\cdot)$ denotes the derivative of $S(\cdot)$ with respect to $\boldsymbol{\beta}$, which is evaluated at $\boldsymbol{\beta} = \mathbf{0}$.

As presented in Table 62.4, Fay (1999) derives the forms of scores for three weight functions. Under the assumption that censoring is independent of the grouping of subjects, these derived scores can be used by permutation tests.

Table 62.4 Scores for Different Weight Functions

Test Weight	v_j
Sun (1996)	$-\frac{\hat{S}(L_i)\hat{\lambda}(L_i) - \hat{S}(R_i)\hat{\lambda}(R_i)}{\hat{S}(L_i) - \hat{S}(R_i)}$
Fay (1999)	$\hat{S}(L_i) + \hat{S}(R_i) - 1$
Finkelstein (1986)	$\frac{\hat{S}(L_i)\log[\hat{S}(L_i)] - \hat{S}(R_i)\log[\hat{S}(R_i)]}{\hat{S}(L_i) - \hat{S}(R_i)}$

You can output scores to a designated SAS data set by specifying the OUTSCORE= option in the TEST statement.

Missing Values

Observations that have a missing value for both the left boundary and the right boundary are not used in the analysis. If a STRATA or FREQ variable value is missing, the observation is not used. However, you can treat missing values as an independent stratum if you specify the MISSING option in the PROC ICLIFETEST statement. If any variable that you specify in the TEST statement has a missing value, that observation is not used in the calculation of test statistics unless you specify the MISSING option in the PROC ICLIFETEST statement.

Output Data Sets

OUTSURV= Data Set

You can specify the OUTSURV= option in the PROC ICLIFETEST statement to create an output data set that contains the following columns:

- any variables that are specified using the BY statement

- any variables that are specified using the STRATA statement
- any variable that is specified using the TEST statement
- LeftBoundary and RightBoundary, the two boundary variables that represent a Turnbull interval
- SurvProb, a variable that contains the survival function estimates at RightBoundary
- FailProb, a variable that contains the failure probability estimates at RightBoundary
- IMStderr, a variable that contains the standard errors that are estimated by using multiple imputations
- BTStderr, a variable that contains the bootstrap standard errors
- ConfType, a variable that contains the name of the transformation applied to the survival time in the computation of confidence intervals
- LagrangeMult, a variable that contains the Lagrange multiplier statistics for checking the Kuhn-Tucker conditions
- SurvProb_LCL, a variable that contains the lower limits of the pointwise confidence intervals for the survival function, using imputation standard errors
- SurvProb_UCL, a variable that contains the upper limits of the pointwise confidence intervals for the survival function, using imputation standard errors
- SurvProb_LCL_BT, a variable that contains the lower limits of the pointwise confidence intervals for the survival function, using bootstrap standard errors
- SurvProb_UCL_BT, a variable that contains the upper limits of the pointwise confidence intervals for the survival function, using bootstrap standard errors

OUTSCORE= Data Set

You can specify the OUTSCORE= option in the PROC ICLIFETEST statement to create an output data set that contains the following columns:

- any variables that are specified using the BY statement
- any variables that are specified using the STRATA statement
- any variable that is specified using the TEST statement
- Score, a variable that contains the scores for each subject
- ScoreType, a variable that contains the name of the weight function for which the scores are calculated

Displayed Output

If you use the NOPRINT option in the PROC ICLIFETEST statement, the procedure does not display any output.

Data and Method Information

The “Data and Method Information” table is displayed each time PROC ICLIFETEST is called. The table displays the following:

- input SAS data set name
- input left and right boundaries
- FREQ variable name
- method used to compute survival estimates
- number of observations read
- number of observations used
- number of bootstrap samples
- number of multiple imputation for calculating standard errors of the survival estimates
- number of multiple imputation for estimating covariance matrix of the generalized log-rank statistics (if the TEST statement is specified)
- seed for multiple imputation
- seed for bootstrap (if the BOOTSTRAP option is specified in the PROC ICLIFETEST statement)

The ODS name of this table is DataInfo.

Nonparametric Survival Estimates

The “Nonparametric Survival Estimates” table is displayed by default. The table displays the following:

- left boundary of time intervals
- right boundary of time intervals
- estimate of survival probability
- estimate of failure probability
- standard error of the survival estimate by the multiple imputation method
- bootstrapped standard error of the survival estimate (if the BOOTSTRAP option is specified in the PROC ICLIFETEST statement)
- Lagrange multiplier (if METHOD=TURNBULL|EM is specified in the PROC ICLIFETEST statement)

The ODS name of this table is ProbabilityEstimates.

Estimates at Right Boundaries of Turnbull Intervals

The “Estimates at Right Boundaries of Turnbull Intervals” table is displayed when you specify the SHOWTI option. The table displays the following:

- left boundary of Turnbull intervals
- right boundary of Turnbull intervals
- estimate of survival probability at the right boundary
- estimate of failure probability at the right boundary
- standard error of the survival estimate by the multiple imputation method
- bootstrapped standard error of the survival estimate (if the BOOTSTRAP option is specified in the PROC ICLIFETEST statement)
- Lagrange multiplier (if METHOD=TURNBULL is specified in the PROC ICLIFETEST statement)

The ODS name of this table is TurnbullIntervals.

Iteration History of Probability Estimates Associated with Turnbull Intervals

The “Iteration History of Probability Estimates Associated with Turnbull Intervals” table is displayed when you specify the ITHISTORY option. The table displays the following:

- iteration index
- type of updating method (EM or ICM) for the current iteration
- nonparametric log likelihood at each iteration
- probability estimate that is associated with each Turnbull interval

The ODS name of this table is IterHistory.

Summary of Censored and Uncensored Values

The “Summary of Censored and Uncensored Values” table is displayed by default. The table displays the following:

- STRATA variables, if there are any, and their values
- index of strata that are formed by the specified STRATA variables
- TEST variable, if there is one, and its values
- index of groups that are formed by the specified TEST variable
- frequency of exact observations

- frequency of left censorings
- frequency of interval censorings
- frequency of right censorings
- percentage of each type of censoring within a stratum

The ODS name of this table is CensoredSummary.

Quartile Estimates

The “Quartiles Estimates” table is displayed if nonparametric estimation finds an MLE successfully. The table displays the following:

- point estimates of the quartiles of the survival times
- lower and upper confidence limits for the quartiles

The ODS name of this table is Quartiles.

Generalized Log-Rank Statistics

The “Rank Statistics” table contains the test statistics of the nonparametric K-sample tests. The ODS name of this table is HomStats.

Covariance Matrix for the Generalized Log-Rank Statistics

The “Covariance Matrix for the Log-Rank Statistics” table is displayed together with the generalized log-rank statistics. The ODS name of this table is HomCov.

Test of Equality over Group

The “Test of Equality over Group” table is displayed if an unstratified K-sample test is performed. The table contains the chi-square statistics, degrees of freedom, and p-value of the generalized log-rank test. The ODS name of this table is HomTests.

Stratified Test of Equality over Group

The “Stratified Test of Equality over Group” table is displayed if a stratified test is carried out. The table contains the chi-square statistics, degrees of freedom, and p-values of the stratified tests. The ODS name of this table is HomTests2.

Scores for Trend Test

The “Scores for Trend Test” table is displayed if the TREND option is specified in the TEST statement. The table contains the set of scores that are used to construct the trend test. The ODS name of this table is TrendScores.

Trend Test

The “Trend Tests” table is displayed if the TREND option is specified in the TEST statement. The table contains the results of the trend test. The ODS name of this table is TrendTest.

Pairwise Group Comparisons

The “Pairwise Group Comparisons” table is displayed if a multiple-comparison adjustment method is specified. The table contains the chi-square statistics and the raw and adjusted p-values of the paired comparisons. The ODS name of this table is SurvDiff.

ODS Table Names

PROC ICLIFETEST assigns a name to each table that it creates. You can use these names to refer to the table when you use the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in Table 62.5. For more information about ODS, see Chapter 20, “Using the Output Delivery System.”

Table 62.5 ODS Tables Produced by PROC ICLIFETEST

ODS Table Name	Description	Statement / Option
CensoredSummary	Number of censored and uncensored observations	Default
ConvergenceStatus	Convergence status	Default
DataInfo	Data and methods information	Default
HomCov	Covariance matrix for the generalized log-rank statistics	TEST
HomStats	Generalized log-rank statistics	TEST
HomTests	Results of K-sample tests	TEST | TEST / WEIGHT=
IterHistory	Iteration history of nonparametric estimation	ITHISTORY
ProbabilityEstimates	Nonparametric survival estimates	Default
Quartiles	Quartile estimates	Default
SurvivalDiff	Adjustments for multiple comparisons	TEST / ADJUST= | TEST / DIFF=
TrendScores	Scores for trend test	TEST / TREND
TrendTest	Trend test results	TEST / TREND
TurnbullIntervals	Turnbull intervals	SHOWTI

ODS Graphics

Statistical procedures use ODS Graphics to create graphs as part of their output. ODS Graphics is described in detail in Chapter 21, "Statistical Graphics Using ODS."

Before you create graphs, ODS Graphics must be enabled (for example, by specifying the ODS GRAPHICS ON statement). For more information about enabling and disabling ODS Graphics, see the section "Enabling and Disabling ODS Graphics" on page 609 in Chapter 21, "Statistical Graphics Using ODS."

The overall appearance of graphs is controlled by ODS styles. Styles and other aspects of using ODS Graphics are discussed in the section "A Primer on ODS Statistical Graphics" on page 608 in Chapter 21, "Statistical Graphics Using ODS."

The survival plot is produced by default; other graphs are produced by specifying the PLOTS= option in the PROC ICLIFETEST statement. You can refer by name to every graph produced through ODS Graphics. The names of the graphs that PROC ICLIFETEST generates are listed in Table 62.6, along with the required keywords for the PLOTS= option.

Table 62.6 Graphs Produced by PROC ICLIFETEST

ODS Graph Name	Plot Description	PLOTS=Option
FailurePlot	Estimated failure function	SURVIVAL(FAILURE)
LogNegLogSurvivalPlot	Log of the negative log of the estimated survival function	LOGLOGS
NegLogSurvivalPlot	Negative log of the estimated survival function	LOGSURV

BandwidthGrid	Cross validation pseudo log likelihood by bandwidth	HAZARD(BW=*numeric-list*)
		HAZARD(BW=RANGE(*lower,upper*))

SmoothedHazard	Kernel-smoothed hazard estimate	HAZARD
SurvivalPlot	Estimated survival function	Default or SURVIVAL
SurvivalPlot	Estimated survival function with pointwise confidence limits	SURVIVAL(CL)
SurvivalPlot	Estimated survival function with K-sample test p-value	SURVIVAL(TEST)

Examples: ICLIFETEST Procedure

The following examples illustrate some of the capabilities of the ICLIFETEST procedure. They are not intended to represent definitive analyses of the data sets that are presented here. See the references for guidance on complete analysis of interval-censored data by appropriate statistical methods.

Example 62.1: Analyzing Data with Observations below a Limit of Detection

Data that have certain values below a limit of detection (LOD) are frequently encountered by toxicologists and environmental scientists. Such data are usually analyzed by imputing the unobserved values by LOD/2 or LOD/$\sqrt{2}$. This type of practice often raises the question of whether the population distributions can be estimated without bias. Gillespie et al. (2010) propose using a reverse Kaplan-Meier estimator, or equivalently, Turnbull's method (1976) by treating the unobserved data as left-censored. When the assumption of independent censoring holds, these estimators can unbiasedly estimate the population distribution functions.

The following hypothetical data have two values, 3 and 10, that are below the limit of detection:

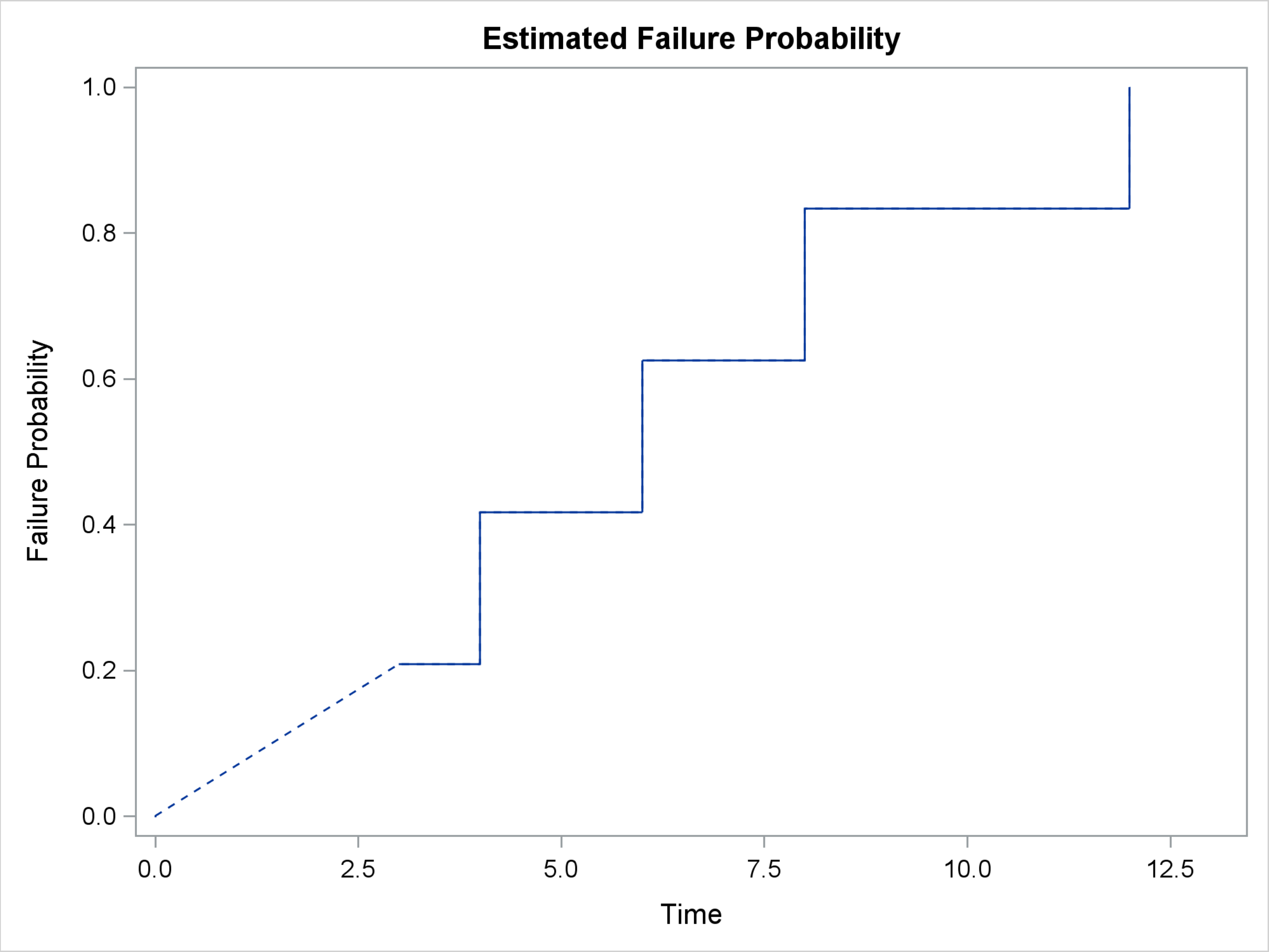
```
data temp;
   input C1 C2;
   datalines;
  .     3
  4     4
  6     6
  8     8
  .     10
  12    12
;
```

The following statements invoke PROC ICLIFETEST to estimate the population distribution function by using Turnbull's method:

```
proc iclifetest data=temp method=turnbull plots=survival(failure)
                impute(seed=1234);
   time (c1,c2);
run;
```

Specifying the PLOTS=SURVIVAL(FAILURE) option requests a failure probability plot. Results are shown in Output 62.1.1. Note that because the first Turnbull interval is $(0, 3)$, the failure probability function is undefined within that interval.

Output 62.1.1 Failure Probability Plot for Fictitious Nondetection Data



Output 62.1.2 presents the estimated failure probability, with standard errors that are estimated by the method of multiple imputations.

Output 62.1.2 Cumulative Probability Estimates

The ICLIFETEST Procedure

Nonparametric Survival Estimates

Time Interval		Probability Estimate		Imputation Standard Error	Lagrange Multiplier
		Failure	Survival		
3	4	0.2083	0.7917	0.1811	0.0000
4	6	0.4167	0.5833	0.2179	0.0000
6	8	0.6250	0.3750	0.2099	0.0000
8	12	0.8333	0.1667	0.1521	0.0000
12	Inf	1.0000	0.0000	0.0000	0.0000

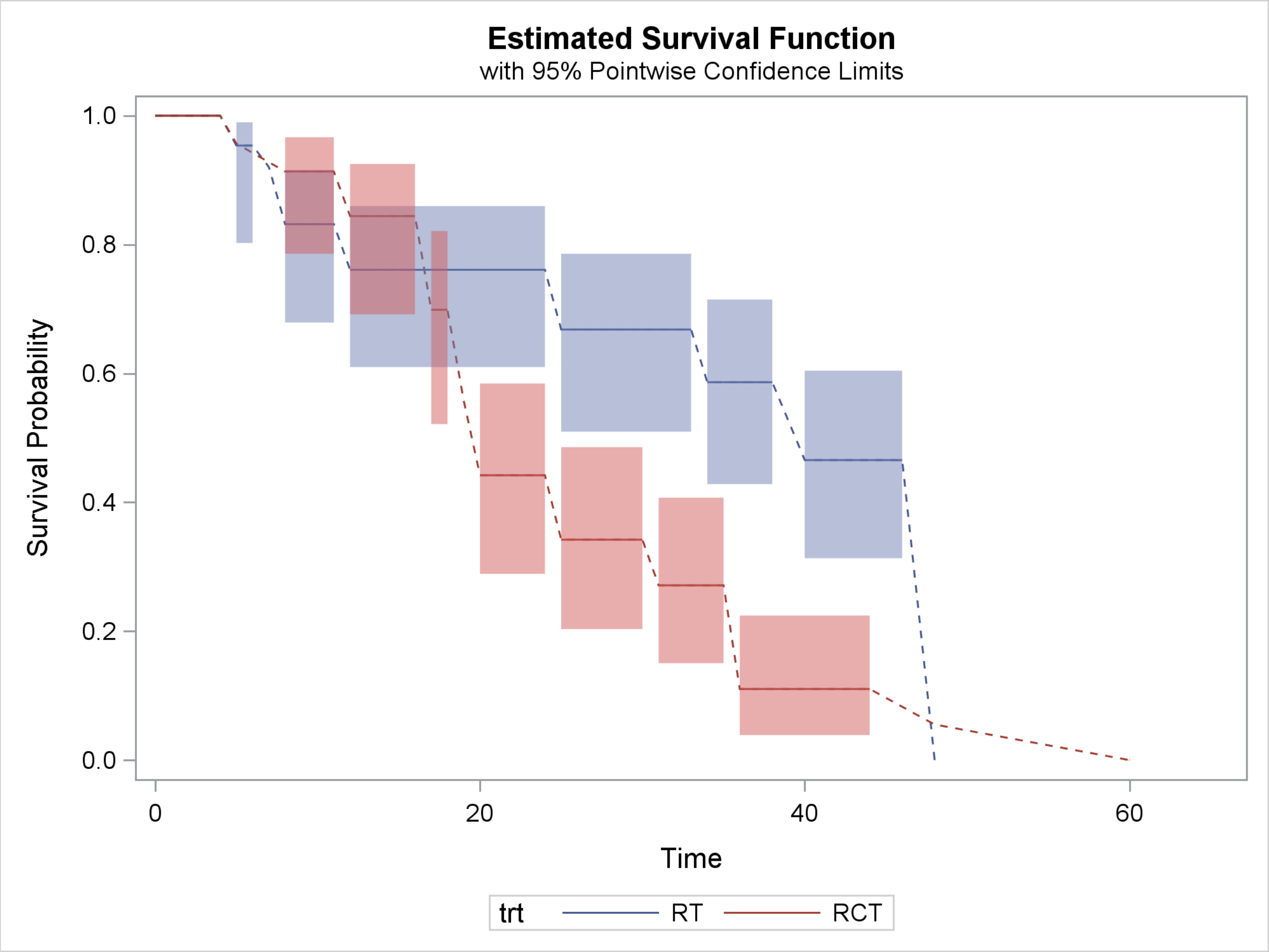
Example 62.2: Controlling the Plotting of Survival Estimates

Recall the breast cancer data in the example in the section "Getting Started: ICLIFETEST Procedure" on page 4663. The following statements compute complementary log-log transformed pointwise confidence limits for the two treatment groups and plot them:

```
proc iclifetest data=BCS plots=survival(cl) impute(seed=1234);
   strata trt;
   time (lTime, rTime);
run;
```

The two areas overlap a lot before the 20th month. After that, the radiation-only group tends to have higher survival rate than the radiation-plus-chemotherapy group. That is, patients who receive only radiation tend to survive longer to experience cosmetic deterioration than those who receive both treatments.

Output 62.2.1 Nonparametric Survival Estimates for Breast Cancer Data

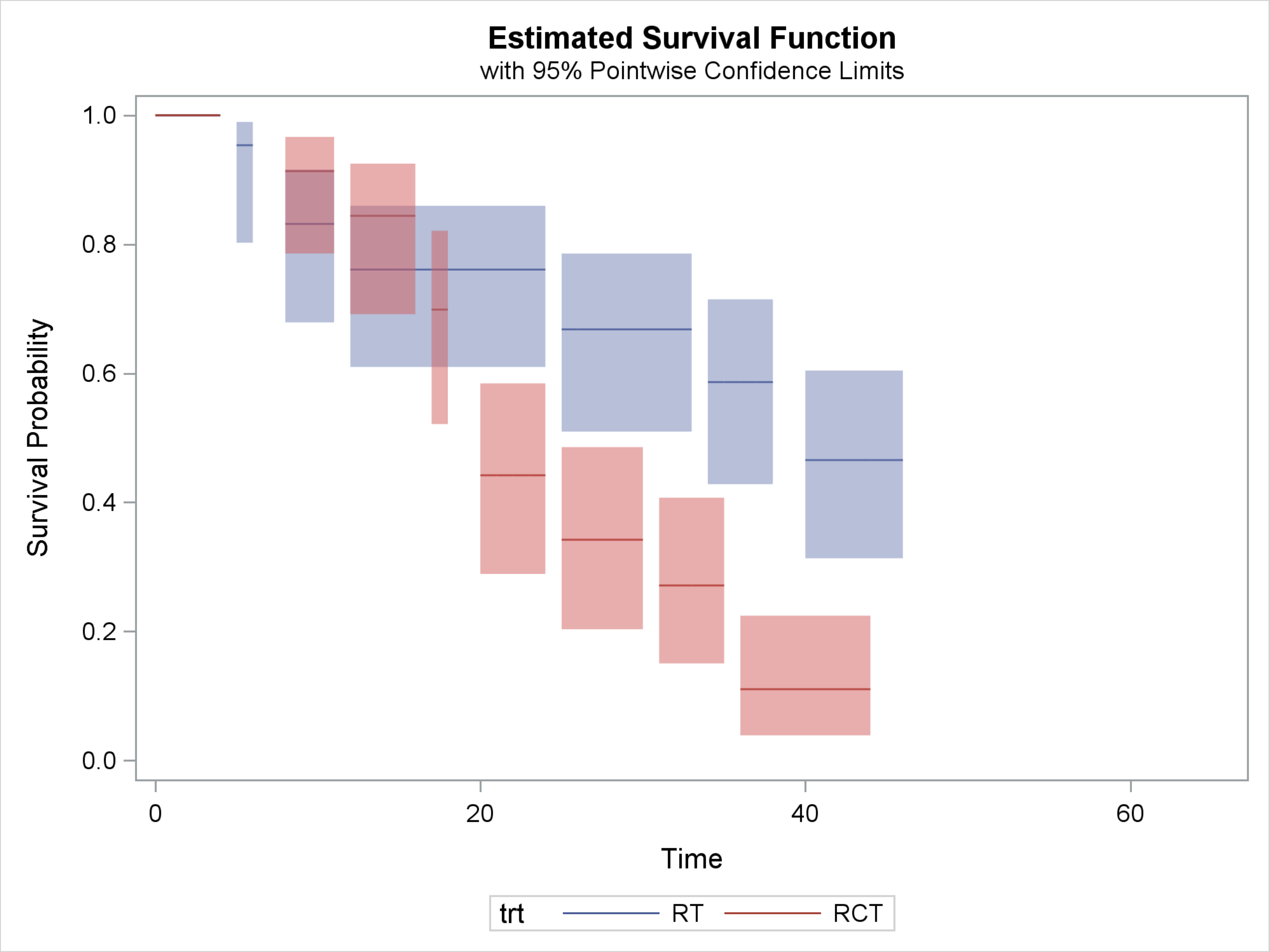


The dashed lines visually connect the survival estimates across Turnbull intervals for which the estimates are not defined. You can choose not to display the dashed lines by specifying the NODASH option as follows:

```
proc iclifetest data=BCS plots=survival(cl nodash) impute(seed=1234);
   strata trt;
   time (lTime, rTime);
run;
```

In Output 62.2.2, the survival estimates are plotted only for the time intervals for which estimates can be obtained.

Output 62.2.2 Nonparametric Survival Estimates

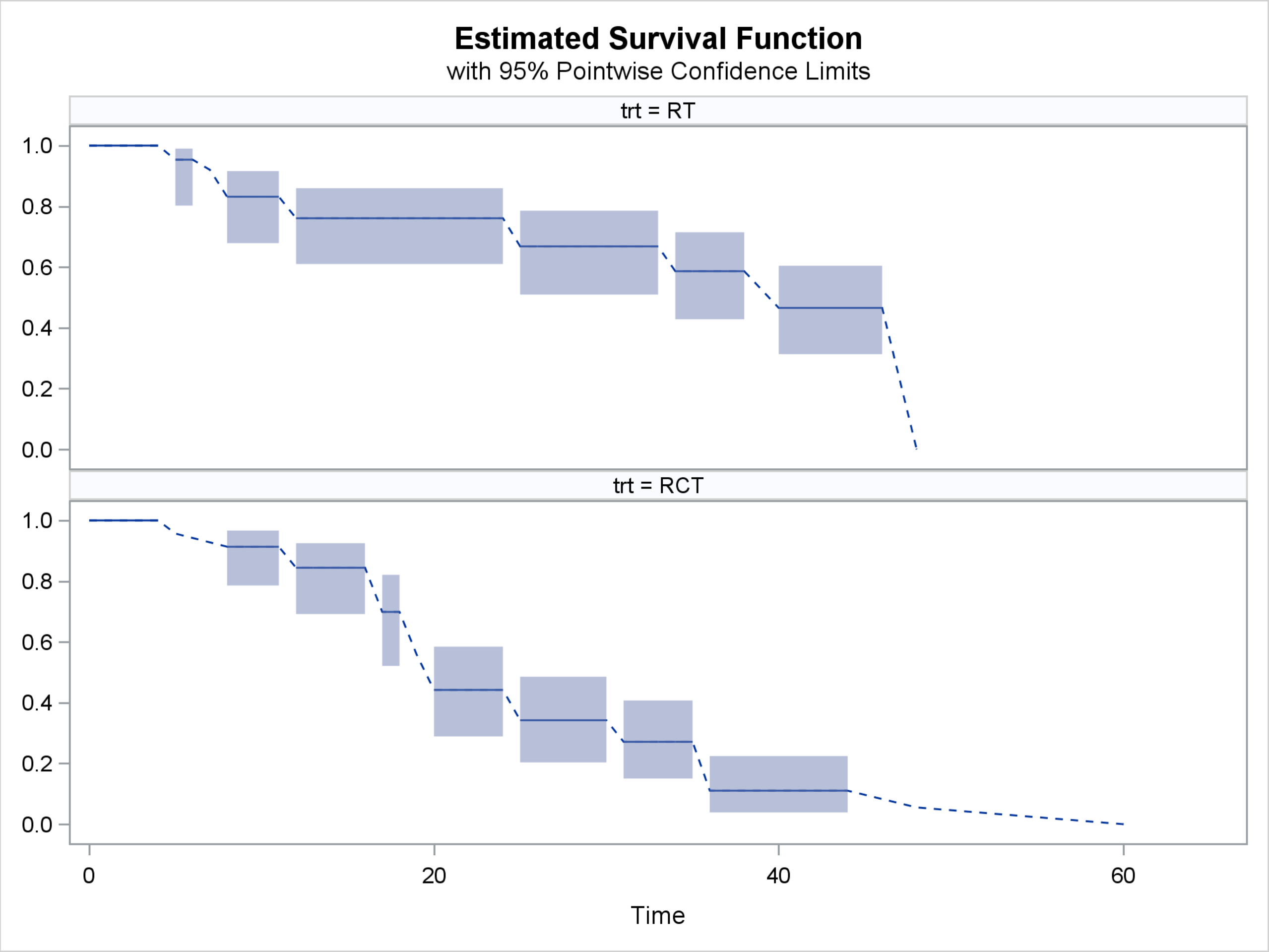


If the overlaid plot becomes too busy, then it might be necessary to plot the survival estimates separately. You can request a paneled display of individual survival estimate by specifying the STRATA=PANEL option as follows:

```
proc iclifetest data=BCS plots=survival(cl strata=panel) impute(seed=1234);
   strata trt;
   time (lTime, rTime);
run;
```

Now the output survival estimates for different treatments are plotted in different panels, as shown in Output 62.2.3.

Output 62.2.3 Paneled Survival Plot



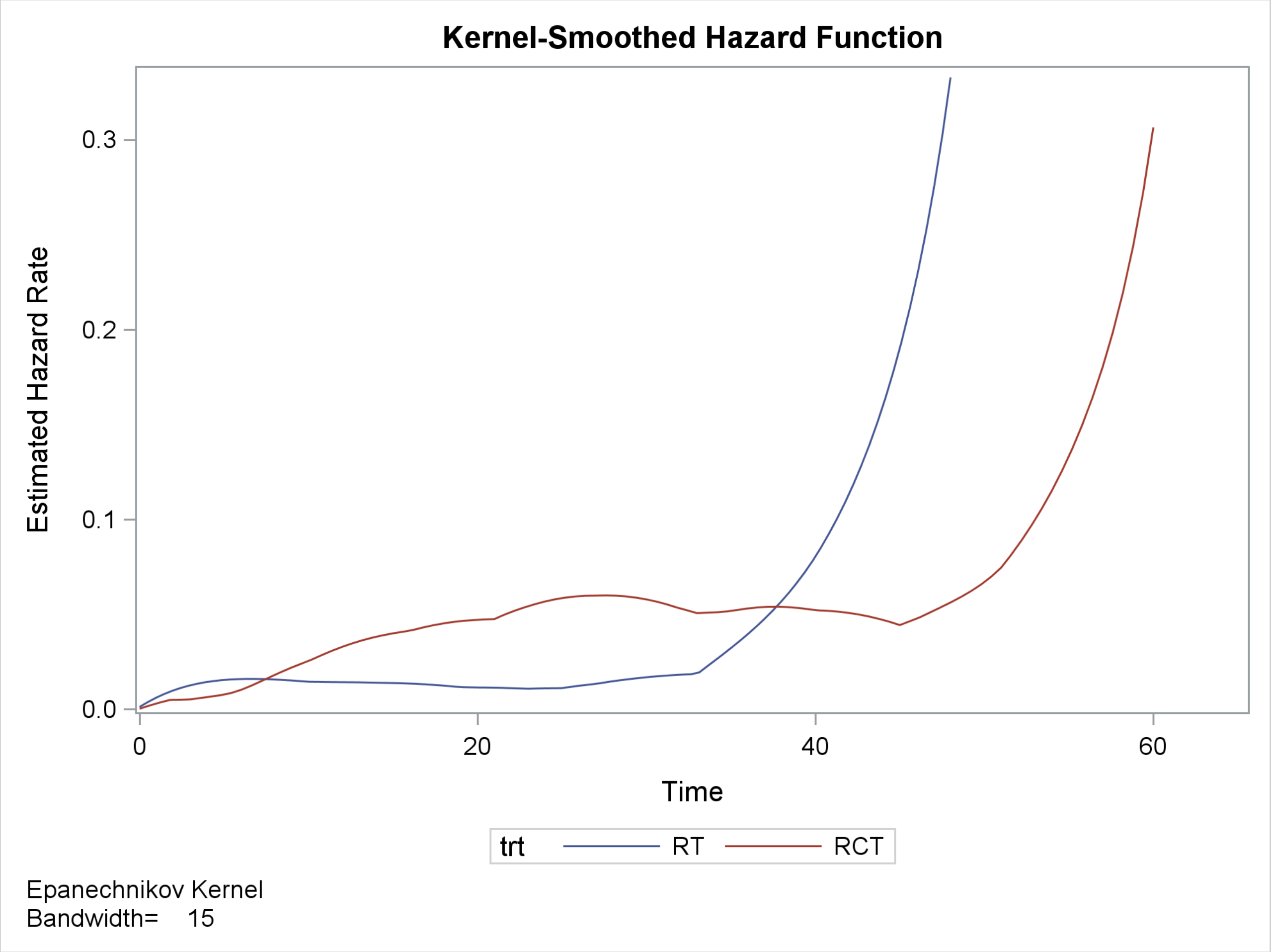
Example 62.3: Plotting Kernel-Smoothed Hazard Functions

The following statements compute the kernel-smoothed hazard functions of the two treatment groups for the breast cancer data:

```
proc iclifetest data=BCS plots=hazard(bandwidth=15) impute(seed=1234);
   strata trt;
   time (lTime, rTime);
run;
```

A bandwidth of 15 is specified. By default, the ICLIFETEST procedure uses the Epanechnikov kernel.

Output 62.3.1 Kernel-Smoothed Hazards for Breast Cancer Data



Example 62.4: Outputting Scores for Permutation Tests

Recall the breast cancer data in the example in the section "Getting Started: ICLIFETEST Procedure" on page 4663. The following statements perform the generalized log-rank test to compare the survival distributions between two treatments by using Finkelstein's weights and save the corresponding scores to a SAS data set named Out:

```
proc iclifetest data=bcs impute(seed=123);
   time (ltime,rtime);
   test trt/weight=finkelstein outscore=out;
run;
```

Fay (1999) describes how to perform permutation tests by using these generated scores. PROC NPAR1WAY can do such tests and compute *p*-values based on normal theory approximation or Monte Carlo simulation. You need to specify the input data and specify SCORES=DATA, as follows:

```
proc npar1way data=out scores=data;
   class trt;
   var score;
   exact scores=data / mc seed=1234;
run;
```

The variable Trt is specified in the CLASS statement so that permutations are done for the groups formed by different levels of the variable. The MC option performs a Monte Carlo version of the permutation test and computes the *p*-values by using Monte Carlo samples. Output 62.4.1 shows the results from the test based on normal theory approximation.

Output 62.4.1 Asymptotic Permutation Test

The NPAR1WAY Procedure

Data Scores Two-Sample Test	
Statistic (S)	-9.9442
Z	-2.6839
One-Sided Pr < Z	0.0036
Two-Sided Pr > |Z|	0.0073

Output 62.4.2 displays the *p*-values that are calculated from the Monte Carlo samples.

Output 62.4.2 Monte Carlo Permutation Test

Monte Carlo Estimates for the Exact Test	
One-Sided Pr <= S	
Estimate	0.0033
99% Lower Conf Limit	0.0018
99% Upper Conf Limit	0.0048
Two-Sided Pr >= |S - Mean|	
Estimate	0.0064
99% Lower Conf Limit	0.0043
99% Upper Conf Limit	0.0085
Number of Samples	10000
Initial Seed	1234

Output 62.4.3 displays the results of the generalized log-rank statistics from using Finkelstein's weights. As you can see, it matches the two-sample test statistic of the permutation test.

Output 62.4.3 Generalized Log-Rank Statistics Using Finkelstein's Weights

The ICLIFETEST Procedure

Generalized Log-Rank Statistics	
trt	**Log-Rank**
RT	-9.94418
RCT	9.944182

References

Borgan, Ø., and Liestøl, K. (1990). "A Note on Confidence Interval and Bands for the Survival Curves Based on Transformations." *Scandinavian Journal of Statistics* 18:35–41.

Brookmeyer, R., and Crowley, J. (1982). "A Confidence Interval for the Median Survival Time." *Biometrics* 38:29–41.

Collett, D. (1994). *Modeling Survival Data in Medical Research*. London: Chapman & Hall.

Edwards, D., and Berry, J. J. (1987). "The Efficiency of Simulation-Based Multiple Comparisons." *Biometrics* 43:913–928.

Fay, M. P. (1996). "Rank Invariant Tests for Interval Censored Data under the Grouped Continuous Model." *Biometrics* 52:811–822.

Fay, M. P. (1999). "Comparing Several Score Tests for Interval Censored Data." *Statistics in Medicine* 18:273–285.

Finkelstein, D. M. (1986). "A Proportional Hazards Model for Interval-Censored Failure Time Data." *Biometrics* 42:845–854.

Fleming, T. R., and Harrington, D. P. (1981). "A Class of Hypothesis Tests for One and Two Samples of Censored Survival Data." *Communications in Statistics—Theory and Methods* 10:763–794.

Gasser, T., and Müller, H. G. (1979). "Kernel Estimation of Regression Functions." In *Smoothing Techniques for Curve Estimation: Proceedings of a Workshop Held in Heidelberg, April 2–4, 1979*, edited by T. Gasser, and M. Rosenblatt, 23–68. Vol. 757 of Lecture Notes in Mathematics. Berlin: Springer-Verlag.

Gentleman, R., and Geyer, C. J. (1994). "Maximum Likelihood for Interval Censored Data: Consistency and Computation." *Biometrika* 81:618–623.

Gillespie, B. W., Chen, Q., Reichert, H., Franzblau, A., Hedgeman, E., Lepkowski, J., Adriaens, P., Demond, A., Luksemburg, W., and Garabrant, D. H. (2010). "Estimating Population Distributions When Some Data Are Below a Limit of Detection by Using a Reverse Kaplan-Meier Estimator." *Epidemiology* 21, Suppl. 4:S64–S70.

Goodall, R. L., Dunn, D. T., and Babiker, A. G. (2004). "Interval-Censored Survival Time Data: Confidence Intervals for the Non-parametric Survivor Function." *Statistics in Medicine* 23:1131–1145. doi:10.1002/sim.1682.

Groeneboom, P., and Wellner, J. A. (1992). *Information Bounds and Nonparametric Maximum Likelihood Estimation*. Basel: Birkhäuser.

Harrington, D. P., and Fleming, T. R. (1982). "A Class of Rank Test Procedures for Censored Survival Data." *Biometrika* 69:133–143.

Hsu, J. C. (1992). "The Factor Analytic Approach to Simultaneous Inference in the General Linear Model." *Journal of Computational and Graphical Statistics* 1:151–168.

Huang, J., Lee, C., and Yu, Q. (2008). "A Generalized Log-Rank Test for Interval-Censored Failure Time Data via Multiple Imputation." *Statistics in Medicine* 27:3217–3226. doi:10.1002/sim.3211.

Jongbloed, G. (1998). "The Iterative Convex Minorant Algorithm for Nonparametric Estimation." *Journal of Computational and Graphical Statistics* 7:310–321. http://www.jstor.org/stable/1390706.

Kalbfleisch, J. D., and Prentice, R. L. (1980). *The Statistical Analysis of Failure Time Data*. New York: John Wiley & Sons.

Klein, J. P., and Moeschberger, M. L. (1997). *Survival Analysis: Techniques for Censored and Truncated Data*. New York: Springer-Verlag.

Kramer, C. Y. (1956). "Extension of Multiple Range Tests to Group Means with Unequal Numbers of Replications." *Biometrics* 12:307–310.

Lachin, J. M. (2000). *Biostatistical Methods: The Assessment of Relative Risks*. New York: John Wiley & Sons.

Meeker, W. Q., and Escobar, L. A. (1998). *Statistical Methods for Reliability Data*. New York: John Wiley & Sons.

Nair, V. N. (1984). "Confidence Bands for Survival Functions with Censored Data: A Comparative Study." *Technometrics* 26:265–275.

Ng, M. P. (2002). "A Modification of Peto's Nonparametric Estimation of Survival Curves for Interval-Censored Data." *Biometrics* 58:439–442. doi:10.1111/j.0006-341X.2002.00439.x.

Pan, W. (2000). "Smooth Estimation of the Survival Function for Interval Censored Data." *Statistics in Medicine* 19:2611–2624.

Peto, R. (1973). "Experimental Survival Curves for Interval-Censored Data." *Journal of the Royal Statistical Society, Series C* 22:86–91.

Sun, J. (1996). "A Nonparametric Test for Interval-Censored Failure Time Data with Application to AIDS Studies." *Statistics in Medicine* 15:1387–1395.

Sun, J. (2001). "Variance Estimation of a Survival Function for Interval-Censored Survival Data." *Statistics in Medicine* 20:1249–1257.

Turnbull, B. W. (1976). "The Empirical Distribution Function with Arbitrarily Grouped, Censored, and Truncated Data." *Journal of the Royal Statistical Society, Series B* 38:290–295.

Wellner, J. A., and Zhan, Y. (1997). "A Hybrid Algorithm for Computation of the Nonparametric Maximum Likelihood Estimator from Censored Data." *Journal of the American Statistical Association* 92:945–959.

Westfall, P. H., Tobias, R. D., Rom, D., Wolfinger, R. D., and Hochberg, Y. (1999). *Multiple Comparisons and Multiple Tests Using the SAS System.* Cary, NC: SAS Institute Inc.

Subject Index

Syntax Index