

Survival Statistics with PROC LIFETEST and PROC PHREG: Pitfall-Avoiding Survival Lessons for Programmers

Yun (Julie) Zhuo, PRA Health Sciences

ABSTRACT

Survival statistics play a critical role in the analysis of efficacy in clinical trials. In SAS®, the LIFETEST procedure compares the survivor function between study arms, and the PHREG procedure estimates the effect of study treatments on hazard rates. This paper shares the lessons we have learned from programming survival analysis with SAS for multiple sponsor clients. Topics vary from P-values, handling warning messages, to some life-saving SAS options. The goal is to point programmers to potential pitfalls and show them ways around it. Geared towards programmers, this paper does not intend to explain any statistical underpinnings of the analysis.

INTRODUCTION

INTRODUCTION TO SURVIVAL STATISTICS

Survival statistics refer to a collection of statistics used to investigate the time it takes for an event of interest to occur. It is distinguished from and has advantages over conventional statistical methods because:

1. The analysis accounts for subjects who are lost to follow-up, withdraw from the study, or experience any competing event. These subjects will not be dropped out of the analysis. Instead, they will be censored, and their data will be analyzed up to the time that they are censored.
2. The analysis examines not just whether an event accrued but also how long it took to occur. Therefore, it is particularly useful for studies that evaluate a treatment designed to delay some undesired outcomes such as tumor progression or death.
3. Subjects do not need to be enrolled at the same time or be followed up for the same amount of time.

Due to its advantages, survival analysis plays a critical role in evaluating efficacy outcomes in clinical trials, especially in oncology trials whose primary efficacy endpoint is to prolong the progression-free time of the patients.

Survival statistics are centered around two related probabilities:

1. Survival probability is the probability that a subject survives from the time origin (e.g. randomization date) to a specified future time.

Most commonly reported survival probability statistics in clinical trials are Kaplan-Meier (KM) median survival time with 95% confidence interval and log-rank P-value. The purpose is to find out that, compared to the standard of care arm, whether the subjects in the treatment arm take statistically significantly longer time to reach a specific event such as death or disease progression.

2. Hazard probability is the probability that a subject who is under observation at a time has an event at that time.

Most commonly reported hazard probability statistics in clinical trials are stratified/unstratified hazard ratios with 95% confidence interval. It is an extension of the probability statistics assessing the risk of reaching a specific event compared to the standard of care arm. Usually, it will adjust for the effects of the stratification factors at randomization.

Statistical programmers are often requested to program the above survival statistics using SAS. Table 1 is an example of a typical shell table for survival analysis presented to programmers.

| Table #. Time to Progression-free Survival Per Central Assessment | | |
|---|---------------------------|----------------------------------|
| Full Analysis Set | | |
| | Treatment Arm (N = XX) | Standard of Care Arm (N = XX) |
| Number of subjects | xx | xx |
| Number with event | xx (xx.x) | xx (xx.x) |
| Number censored | xx (xx.x) | xx (xx.x) |
| Time to event (months) | | |
| Median | xx.x | xx.x |
| 95% confidence interval | (xx.x, xx.x) | (xx.x, xx.x) |
| 25 th -75 th percentiles | xx - xx | xx - xx |
| Min, Max | xx, xx | xx, xx |
| Stratified analysis | | |
| Log-rank P-value | x.xxxxx | |
| Hazard ratio (95% CI) | x.xx (x.xx, x.xx) | - |
| Unstratified analysis | | |
| Log-rank P-value | x.xxxxx | |
| Hazard ratio (95% CI) | x.xx (x.xx, x.xx) | - |

Table 1. Example Shell Table for Survival Statistics

INTRODUCTION TO SAS PROCEDURES FOR SURVIVAL STATISTICS

The LIFETEST Procedure

In SAS, The LIFETEST procedure generates the time-to-event summary statistics and performs statistical tests. Example code for an unstratified analysis:

```
ods output quartiles=quartiles homtests=homtests_unstrat;
proc lifetest data=adtte method=km;
  time aval*cnsr(1);
  strata trt01pn;
run;
```

and example code for a stratified analysis:

```
ods output homtests=homtests_strat;
proc lifetest data=adtte method=km;
  time aval*cnsr(1);
  strata stratr1n stratr2n / group=trt01pn;
run;
```

In the example code above:

1. Input data set ADTTE is a standard ADAM data set for time to event analysis. In ADTTE, the variable AVAL is calculated as the time between the time of origin (e.g., randomization date) to the analysis date (e.g., the date of the event, last known alive date, last evaluable disease assessment date, etc.). The CNSR variable indicates whether the subject is censored. TRT01PN indicates treatment assignment. STRATR1N and STRATR2N are stratification factors at randomization.

Display 1 is a partial snapshot of a dummy ADTTE data set with only selected variables used in the analysis. The dummy data set is modified from the *sashelp.bmt* data set, a publicly available SAS data set in the SAS library.

| USUBJID | PARAMCD | PARAM | TRT01PN | CNSR | AVAL | STRATR1N | STRATR2N |
|---------|---------|------------------------------------|---------|------|--------------|----------|----------|
| 23 | PFS | Progression-free Survival (Months) | 1 | 0 | 28 | 2 | 2 |
| 53 | PFS | Progression-free Survival (Months) | 1 | 1 | 7.2396694215 | 2 | 2 |
| 54 | PFS | Progression-free Survival (Months) | 1 | 1 | 20.033057851 | 1 | 1 |
| 55 | PFS | Progression-free Survival (Months) | 2 | 0 | 87.272727273 | 2 | 2 |
| 70 | PFS | Progression-free Survival (Months) | 2 | 1 | 0.0661157025 | 2 | 1 |

Display 1. Example ADTTE Data Set for Time-to-Event Data

- We use SAS ODS to generate output data sets containing the statistical results. Output data set QUARTILES contains summary statistics including median, 25th-75th percentiles, and their confidence intervals. Output data set HOMTESTS contains P-values for stratified or unstratified statistical tests such as the Log-Rank test and the Wilcoxon test.

Display 2 and Display 3 below show examples of the above output data sets generated out of the dummy ADTTE data set using the LIFETEST procedure.

| STRATUM | trt01pn | Percent | Estimate | Transform | LowerLimit | UpperLimit |
|---------|---------|---------|----------|-----------|------------|------------|
| 1 | 1 | 75 | 61.3884 | LOGLOG | 56.4959 | 79.6364 |
| 1 | 1 | 50 | 51.8347 | LOGLOG | 45.7521 | 60.9256 |
| 1 | 1 | 25 | 43.7686 | LOGLOG | 28.4298 | 50.4793 |
| 2 | 2 | 75 | 74.4463 | LOGLOG | 53.9174 | 87.2727 |
| 2 | 2 | 50 | 66.9091 | LOGLOG | 37.5537 | 74.4463 |
| 2 | 2 | 25 | 40.9256 | LOGLOG | 27.9339 | 66.9091 |

Display 2. The QUARTILES Output Data Set

| Test | ChiSq | DF | ProbChiSq |
|-----------|--------|----|-----------|
| Log-Rank | 1.0898 | 1 | 0.2965 |
| Wilcoxon | 0.5301 | 1 | 0.4666 |
| -2Log(LR) | 0.2995 | 1 | 0.5842 |

Display 3. The HomTests Output Data Set

The PHREG Procedure

In SAS, the PHREG procedure performs the regression analysis of survival data to generate hazard ratios and their corresponding 95% confidence intervals.

Example code for an unstratified analysis:

```
ods output ParameterEstimates = hazard_unstrat;
proc phreg data=adtte;
class trt01pn (ref="2");
model AVAL*CNSR(1)=TRT01PN/ties=breslow risklimits alpha=0.05 ;
run;
```

And example code for a stratified analysis:

```
ods output ParameterEstimates=hazard_strat;
proc phreg data=adtte;
class trt01pn (ref="2");
model AVAL*CNSR(1)=TRT01PN/ties=breslow risklimits alpha=0.05 ;
strata stratr1n stratr2n;
run;
```

Like the LIFETEST procedure, the PHREG procedure in the example codes also uses the standard ADAM data set ADTTE as the input data set.

In the procedure, the CLASS statement indicates that the treatment variable is the explanatory variable for the regression model. The MODEL statement constructs the statistical model. And the STRATA statement lists the stratification variables for a stratified analysis.

Display 4 shows a sample output data set from the procedure with the hazard ratio statistics using our dummy ADTTE data set.

| Parameter | ClassVal0 | DF | Estimate | StdErr | ChiSq | ProbChiSq | HazardRatio | HRLowerCL | HRUpperCL |
|-----------|-----------|----|----------|---------|--------|-----------|-------------|-----------|-----------|
| trt01pn | 1 | 1 | 0.38601 | 0.37202 | 1.0766 | 0.2995 | 1.471 | 0.710 | 3.050 |

Display 4. The ParameterEstimates Output Data Set

PITFALLS AND CAUTIONS

The examples in the preceding introductory section may look straightforward. However, in real-life experience, we may inadvertently step on programming pitfalls and generate erroneous results. In this section, we will point out some of these potential pitfalls, and discuss the recommended actions we can take to steer away from them.

ONE-SIDED OR TWO-SIDED P-VALUE

The Pitfall

When the LIFETEST procedure performs the log-rank statistical test, it generates a P-value and saves it under the ProbChiSq column in the HomTests output data set. For example, in the HomTests output in Display 3, the log-rank P-value under the ProbChiSq column is 0.2965. If you report the P-value=0.2965 to your statistical report, you may have stepped on a pitfall trap.

The problem here is that neither the SAS output nor the SAS code explicitly indicates whether the P-value is one-sided or two-sided. Programmers do not typically understand the differences between a one-sided P-value and a two-sided P-value.

The Caution

The P-value SAS reports in the HomTests output is two-sided or two-tailed P-value. Two-sided P-value is very commonly used in statistical tests, because to test a hypothesis of 'is the effect consistent with chance', we usually must consider both tails of the distribution.

However, by contrast, in the survival analysis for clinical trials, we often test a one-sided hypothesis, which is whether the survival function in the treatment group is superior to the survival in the comparison group. Therefore, statisticians often request that we use one-sided P-value for the report.

The Recommendation

We need to carefully check the statistical specification to find out if one-sided or two-sided P-value should be reported. If the specification does not have the information, be sure to check with your statistician and ask them to add this information to the specification file.

If one-sided P-value should be reported, we can convert two-sided P-value to one-sided P-value by simply dividing the value by 2. For example, in the HomTests output data set in Display 3, the one-sided P-value for the Log-Rank test should be $0.2965/2 = 0.14825$.

INVERSE HAZARD RATIO AND CONFIDENCE LIMITS

The Pitfall

The hazard ratio statistics compares the treatment effect of our study drug to the effect of a reference group. In the PHREG procedure, the REF= option specifies the level of the treatment assignment variable (TRT01PN) to use as the reference group. If, by accident, the reference group in the model is the study

drug treatment instead of the comparison group, we will get inverse, and wrong, hazard ratio and confidence limits.

If we execute the following two PHREG procedures on our example ADTTE data, we will obtain identical results:

```
*PHREG with explicitly specified reference group*;
proc phreg data=adtte;
  class trt01pn (ref="2");
  model AVAL*CNSR(1)=TRT01PN/ties=breslow risklimits alpha=0.05 ;
run;
*PHREG without explicitly specified reference group*;
proc phreg data=adtte;
  class trt01pn;
  model AVAL*CNSR(1)=TRT01PN/ties=breslow risklimits alpha=0.05 ;
run;
```

The first Proc PHREG explicitly specifies that the reference group in the model is the group of subjects with TRT01PN=2 (the Standard of Care arm). The second Proc PHREG does not specify, and by default, when a reference group is not specified, SAS uses the last ordered level in the explanatory variable (TRT01PN) as reference. The two procedures above yield the same results on our example data because in our data, as shown in the FREQ output below in Display 5, the TRT01PN variable only has two levels, and the last ordered level is the Standard of Care group (TRT01PN=2).

| TRT01PN | TRT01P | Frequency | Percent | Cumulative Frequency | Cumulative Percent |
|---------|------------------|-----------|---------|----------------------|--------------------|
| 1 | Study Drug | 54 | 54.55 | 54 | 54.55 |
| 2 | Standard of Care | 45 | 45.45 | 99 | 100.00 |

Display 5. The Proc Freq Output for the Treatment Assignment Variable in the Example Data

In clinical trials, the last ordered level in the treatment assignment variable is usually the placebo or the Standard of Care arm. However, if you think it is safe to use the default setting and do not explicitly specify a reference level, you may run into trouble.

The Caution

In real world clinical trials, the study never remains static. Protocol modifications and amendments are commonplace. A change in the treatment arm assignment may cause problem if reference group is not explicitly specified. For example, one of the studies we worked on added a new treatment arm in the middle of the study:

| TRT01PN | TRT01P | Frequency | Percent | Cumulative Frequency | Cumulative Percent |
|---------|-----------------------------|-----------|---------|----------------------|--------------------|
| 1 | Study Drug (Dose Regimen 1) | 54 | 39.42 | 54 | 39.42 |
| 2 | Standard of Care | 45 | 32.85 | 99 | 72.26 |
| 3 | Study Drug (Dose Regimen 2) | 38 | 27.74 | 137 | 100.00 |

← New treatment arm

Display 6. The Proc Freq Output for the New Treatment Assignment Variable after Study Modification

Display 6 above shows that after adding a new treatment arm, the last ordered level of the explanatory variable TRT01PN is not the comparator group (Standard of Care arm). The PHREG procedure will produce inverse hazard ratio measuring instead the effect of Standard of Care versus the effect of study Drug Dose Regimen 2.

Another common mistake that may result in inverse hazard ratios is to omit the CLASS statement in the PHREG procedure altogether. For example, we execute the following SAS codes on the dummy ADTTE data shown in Display 1. Note that the CLASS statement is not included, and the code runs without any error or warning message:

```

*PHREG without the CLASS statement*;
proc phreg data=adtte;
  model AVAL*CNSR(1)=TRT01PN/ties=breslow risklimits alpha=0.05 ;
run;

```

However, as shown in Display 7, the result from the above codes could be an inverse hazard ratio, compared to the result from the following PHREG procedure that has the CLASS statement:

```

*PHREG with the CLASS statement*;
proc phreg data=adtte;
  class trt01pn;
  model AVAL*CNSR(1)=TRT01PN/ties=breslow risklimits alpha=0.05 ;
run;

```

| Output from the PHREG procedure with the CLASS statement | | | | | | | | | | |
|--|-----------|----|----------|---------|--------|-----------|-------------|-----------|-----------|--|
| Parameter | ClassVal0 | DF | Estimate | StdErr | ChiSq | ProbChiSq | HazardRatio | HRLowerCL | HRUpperCL | |
| TRT01PN | 1 | 1 | 0.38601 | 0.37202 | 1.0766 | 0.2995 | 1.471 | 0.710 | 3.050 | |

| Output from the PHREG procedure without the CLASS statement | | | | | | | | | | |
|---|----|----------|---------|--------|-----------|-------------|-----------|-----------|--|--|
| Parameter | DF | Estimate | StdErr | ChiSq | ProbChiSq | HazardRatio | HRLowerCL | HRUpperCL | | |
| TRT01PN | 1 | -0.38601 | 0.37202 | 1.0766 | 0.2995 | 0.680 | 0.328 | 1.409 | | |

Inverse Hazard Ratio

Display 7. A Comparison of the Outputs from PHREG Procedures with and without the CLASS Statement

The Recommendation

Therefore, when we use the PHREG procedure to generate the hazard ratio statistics, it is always a good practice to include the CLASS statement. It is also always a good practice to explicitly specify the reference level that we use as the comparison group. We want to be clear in the codes about our intentions, even though sometimes we may get the same results without the CLASS statement or the explicitly specified reference level.

CENSORING VARIABLE AND ITS VALUE CODING

The Pitfall

The concept of censoring is unique and most difficult to understand in survival analysis. Creating the censoring variable and incorporating it into the statistical models is most challenging for many programmers. Not surprisingly, many programmers turn to online articles as well as scholarly resources for sample codes and other assistance.

Below is a snapshot of a sample code from the SAS User's Guide for the LIFETEST procedure:

```

ods graphics on;
proc lifetest data=Exposed plots=(survival(atrisk) logsurv);
  time Days*Status(0);
  strata Treatment;
run;
ods graphics off;

```

And below is another snapshot of a sample code from an online scholarly article:

```
ods output ProductLimitEstimates = ple;
proc lifetest data=whas500(where=(fstat=1)) nelson outs=outwhas500;
time lenfol*fstat(0);
run;
```

In the above sample codes, variables 'status' and 'fstat' are censoring variables. As you may have noticed, both examples use the coding value '0' (that is, censoring variable = 0) in the parenthesis after the censoring variable in the TIME statement. Using the coding value of '0' in the model is very commonly seen in online resources.

If you do not dive deeper to understand the concept of censoring and the requirement of the TIME statement, you may simply copy the sample codes for your analysis. Then you may be walking into a pitfall.

The Caution

In the LIFETEST procedure, the TIME statement includes a failure time variable, a censoring variable, and a parenthetical list of numeric values that correspond to right censoring. For example, in the TIME statement below:

```
TIME t*c(1, 2)
```

Variable c is a censoring variable. If a subject has C=1 or C=2, it means that the subject is right censored.

But what is right censoring? In clinical trials, right censoring is when the event (e.g., death, disease progression) does NOT occur while the subject is being observed in the study or because they are lost to follow-up or otherwise leave the study for any reason.

In the two sample examples we borrowed from online resources, when status=0 or fstat=0, the subject is right censored and does not have any event. In other words, in their time-to-event input data set, they use the value code of 0 to indicate right censoring.

By contrast, in the guideline from the ADaM Basic Data Structure (BDS) for Time-to-Event (TTE) Analyses v1.0, the standard for the censoring variable is the following:

| Variable Name | Variable Label | Type | Codelist / Controlled Terms | Core | CDISC Notes |
|---------------|----------------|------|-----------------------------|------|---|
| CNSR | Censor | Num | * | Cond | CNSR is a required variable for a time-to-event analysis dataset, though it is a conditionally required variable with respect to the ADaM BDS. For example, CNSR = 0 for event and CNSR > 0 for censored records. |

Table 2. Standard Censoring Variable per CDISC

Although the guideline does not require the censoring variable to code CNSR>0 for censored records, typical time-to-event data sets in the industry follow the example in the CDISC notes and use CNSR=0 for event and CNSR>0 for censors.

Table 3 below is an example metadata for the CNSR variable in the dummy data set ADTTE (Display 1) that we use in this paper.

| Dataset Name | Variable Name | Variable Label | Variable Type | Display Format | Codelist / Controlled Terms | Source / Derivation |
|--------------|---------------|---------------------|---------------|----------------|-----------------------------|--|
| ADTTE | CNSR | Censoring Indicator | Integer | 1.0 | 0, 1 | Set to 0 if patient has died or has disease progression; else set to 1 if there is at least one post-baseline disease assessment record. |

Table 3. Example Metadata for the CNSR Censoring Variable in the Dummy Data Set

We can see from Table 2 and Table 3 that typically in ADTTE, we use value coding CNSR>0 to indicate censored subjects. Therefore, we need to plug in (1), instead of (0), to the TIME statement if we run the survival analysis on our example dummy data:

```
proc lifetest data=adtte method=km;
  time aval*cnsr(1);
  strata trt01pn;
run;
```

The Recommendation

The TIME statement requires that the parenthetical list of values should correspond to right censoring. Therefore, it is important that we review the metadata carefully in order to understand how the censoring variable is coded. We should only plug in coded numeric, non-missing values that indicate censored records.

THE HOMOGENEITY WARNING

The Pitfall

We often encounter the following warning message when we run the LIFETEST procedure, especially when the analysis is stratified or performed for subgroups with very few records.

```
WARNING: The likelihood ratio test for strata homogeneity is
questionable since some strata have no events.
```

The message indicates that due to no event data in at least one of the strata in the model, the statistical analysis method is inappropriate for the data. How do we deal with the warning message? The first impulse is to skip the entire LIFETEST procedure when at least one of the strata does not have any event, since the analysis is inappropriate for the data.

The Caution

Although skipping the entire procedure can get rid of the warning message, you also lose all the other valuable, appropriate results the procedure can produce. For example, the LIFETEST procedure produces, in addition to the statistical P-values, the Kaplan-Meier median survival time and its 95% confidence interval, which are often requested in most statistical specifications. In the case when there are multiple strata or groups, the LIFETEST procedure can also perform appropriate statistical tests on those strata where there are adequate number of events.

There are two alternative methods to handle the warning message.

If the specification you received request summary statistics for time-to-event data such as median survival time, confidence intervals, and 25th – 75th percentiles, you still need to use the LIFETEST procedure and you can use the NOTEST option to skip the inappropriate statistical test. For example, the following codes will not issue the warning message because it does not perform the inappropriate statistical test, but it will still give you the summary statistics.

```
ods output Quartiles=quartiles;
proc lifetest data=adtte method=KM alpha=0.05;
  by subgroup;
  time AVAL*CNSR(1);
  strata trt01pn/notest;
run;
```

If the data contains more than two strata, you can still perform the analysis on those strata with events. Firstly, perform a FREQ procedure to identify the strata that has no event:

```
proc freq data=adtte;
  tables trt01pn*CNSR / list norow nocol nopercnt;
run;
```

Or if the analysis is performed by a subgroup (a categorical variable):

```
proc freq data=adtte;
  tables subgroup*trt01pn*CNSR / list norow nocol nopercnt;
run;
```

Then we can exclude the problematic strata from our analysis with a WHERE statement. For example:

```
ods output Quartiles=quartiles;
proc lifetest data=adtte method=KM alpha=0.05;
  where trt01pn ^= 3;
  time AVAL*CNSR(1);
  strata trt01pn;
run;
```

After excluding the problematic strata, we can still get appropriate results for other strata without the warning message.

The Recommendation

The LIFETEST procedure performs multiple tasks at the same time. Instead of skipping the entire procedure, we may use alternative methods to handle the warning message. Depending on what is required in the statistical specification, we can use the NOTEST option to skip the statistical tests or use the WHERE statement to exclude problematic strata from the analysis.

THE CONVERGENCE WARNING

The Pitfall

Another mysterious warning message we may encounter is the following convergence warning from the PHREG procedure:

```
WARNING: Convergence was not attained in 25 iterations.
```

This type of warning is most typically seen in regression model fitting procedures including the PROC PHREG. It is because the algorithm behind regression models is iterative. There is no closed form solution and we run iterations until some convergence criteria is reached. The warning message indicates that the model converging criteria has not been reached within 25 iterations, and the results from the model cannot really be trusted.

How do we handle the warning message? When we search through the SAS options for the PHREG procedure, we will find the MAXITER= option. For example, the MAXITER= option can be added to the MODEL statement:

```
ods output ParameterEstimates=hazard_strat;
proc phreg data=adtte;
  by subgroup;
```

```
class trt01pn (ref="2");  
model AVAL*CNSR(1)=TRT01PN/ties=breslow risklimits alpha=0.05 maxiter=500;  
strata stratr1n stratr2n;  
run;
```

The MAXITER= option specifies the maximum number of iterations the procedure will try to run in order to reach convergence. The default value is 25. Increasing the value to a very large number might do the trick and get rid of the warning message.

The Caution

Increasing the number of iterations is a justifiable option and might do the trick in some instances. However, in real-life experiences, the convergence warning is usually caused by a lack of event in one of the analysis categories. This is especially true if we conduct stratified analysis or by-group analysis, and there is sparse data in one of the analysis categories. The question becomes whether we truly need to fit into the model all the groups and strata as defined in the specification.

The Recommendation

Therefore, a good practice is to, similar to the methods discussed in the previous section, use a FREQ procedure to find out if there is any specific strata or group that does not have any event. If so, then, we can work with statisticians to find out if the problematic group or strata can be removed, collapsed, or re-defined.

CONCLUSION

Today survival statistics are commonly used in clinical trials, especially oncology studies, to measure important efficacy endpoints. It is critical that we program the survival statistics properly without falling into potential pitfalls. In this paper, we introduced the survival statistics and the related SAS procedures from programmers' perspective. We discussed the potential programming pitfalls and shared our perspectives on the best practices to avoid those pitfalls.

REFERENCES

- Brucken, N. 2012. "Creating the Time to Event ADaM Dataset: The Nuts and Bolts." Proceeding of PharmaSUG 2012, SAS Institute. Available at <https://www.pharmasug.org/proceedings/2012/HW/PharmaSUG-2012-HW03.pdf>
- CDISC 2012. "The ADaM Basic Data Structure for Time-to-Event Analyses." Clinical Data Interchange Standards Consortium, Inc.
- Clark, T G. 2003. "Survival Analysis Part I: Basic concepts and first analyses." *British Journal of Cancer*, 89(2): 232-238.
- Columbia Public Health. "Time-To-Event Data Analysis." Accessed March 27, 2021. <https://www.publichealth.columbia.edu/research/population-health-methods/time-event-data-analysis>
- SAS Institute Inc. 2014. *SAS/STAT® 13.2 User's Guide*. Cary, NC: SAS Institute Inc.
- Williams, C. 2008. "Surviving Survival Analysis – An Applied Introduction." Proceeding of NESUG 2008, SAS Institute. Available at <https://www.lexjansen.com/nesug/nesug08/sa/sa20.pdf>

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Yun (Julie) Zhuo
PRA Health Sciences
ZhuoYun@PRAHS.com
jzhuo@KITEPHARMA.com