

Implementing the Rank-Preserving Structural Failure Time Model in SAS[®] and R

Bradford J. Danner, GCE Solutions, Inc., Tennessee, USA

Indrani Sarkar, GCE Solutions, Inc., Indiana, USA

ABSTRACT

In randomized two-arm trials where survival is an endpoint of interest, patients may be required to stay on the assigned treatment until loss of clinical benefit or death. However, crossover from a reference to an active treatment is often allowed in oncology trials, after disease progression. Traditional intent-to-treat analysis can lead to underestimation of the treatment effect on overall survival when crossover is not taken into consideration. One possible method to minimize this estimation bias is the application of the rank-preserving structural failure time (RPSFT) model. A key feature of RPSFT model is to use a multiplicative factor to estimate the gain/loss in survival time from switching treatments contributed by the active treatment. While SAS PROC PHREG may be commonly used to perform a conventional Cox regression to estimate the hazard ratio between two treatment arms, an option for the RPSFT is not built-in. Therefore implementation in SAS requires a slightly more manual approach. Using R, the publicly available *RPSFTM* package may be used to perform an analysis utilizing the RPSFT model. The main purpose of this paper is to describe and illustrate with a straight-forward example the implementation of the RPSFT model using both SAS and R, and briefly compare and discuss the two approaches.

INTRODUCTION

In randomized trials two common efficacy endpoints are progression-free survival and overall survival. Utilizing a relatively simplistic oncology trial example, survival of patients randomized to an active test compound arm are compared to patients randomized to a reference, or conventional therapy arm, up to the point of progressive disease, death, or loss to follow-up in the case of progression-free survival, and death or loss to follow-up for overall survival. A standard convention employed will restrict the possibility for intent-to-treat patients from crossing over to the active therapy, or similar class of compounds, prior to the point of disease progression for patients in the placebo arm of the study. Following progression, study protocols often allow for patients in the placebo arm to crossover to an active treatment, in order to treat the disease under study, during the overall survival follow-up period. Analysis of both progression-free and overall survival are typically assessed with a conventional Cox regression analysis, taking into account only the treatment arm to which they were randomized. This conventional analysis does not account for any crossover to an alternative therapy, and the assessment of overall survival can become biased and underestimated, the magnitude of which will depend on the relative number of patients that switch to an active treatment.

One possible method to minimize this estimation bias when comparing the survival outcome is the application of the rank-preserving structural failure time (RPSFT) model. The RPSFT estimates the survival time gained or lost by receiving active treatment (Allison, 2010). Essentially, the model assumes the active therapy is acting on longevity by multiplying survival by a given factor once a patient starts receiving the treatment. This factor may be interpreted as the increase or decrease in survival by taking the active treatment compared to the control treatment. Once established, the survival duration of patients is reconstructed, and re-censored, as if they had never received the active compound. Other modeling methods are available to account for the crossover process (Korhonen et al., 1999, 2012), however, the RPSFT approach is simpler in that it only requires the arm patients are assigned to at baseline, and therefore results in a strictly randomization-based method for estimating treatment effect in the presence of crossover (Robins and Tsiatis, 1991).

While SAS PROC PHREG may be commonly used to perform a conventional Cox regression to estimate the hazard ratio between two treatment arms (Allison, 2010), no built-in option for implementing the

RPSFT is currently available. Implementation in SAS requires a slightly more manual approach. Using R, the publicly available *RPSFTM* package may be installed to perform an analysis utilizing the RPSFT model (Bond and Allison, 2017). The main purpose of this paper is to describe and illustrate with a straight-forward example the implementation of the RPSFT model using both SAS and R, and briefly compare and discuss the two approaches.

METHODS AND RESULTS

Detailed description and discussion of the RPSFT model may be found in Robins and Tsiatis (1991) and or White et al. (1999). A brief introduction is included here in order to act as a basis for the methods and results we subsequently present.

Consider the scenario we present in the introduction – a trial comparing survival of patients in an active treatment arm against survival of patients in a control or placebo arm, where crossover to an active compound is allowed following disease progression. We need to relate the individual patient’s observed event time T to the event time U that would have been observed had no active treatment been given. The time to event for individual patients (i) can be represented as: time off treatment + time on treatment:

$$T_i = T_{offi} + T_{oni}$$

The basic model relating T to U employed is the following causal accelerated life model:

$$U_i = \int_0^{T_i} \exp[\Psi X_i(t)] dt \dots\dots\dots (1)$$

where,

$$X_i(t) = \begin{cases} 1 & \text{in case of cross - over} \\ 0, & \text{Otherwise} \end{cases}$$

This model can be further explained under two possible scenarios:

Scenario 1: In the absence of crossover i.e. subject stays on the assigned treatment till end of study, $X_i(t)$ becomes 0. Then the equation (1) becomes:

$$U_i = \int_0^{T_i} \exp[\Psi * 0] dt \dots\dots\dots (2)$$

→ $U_i = T_i$

Scenario 2: In case of crossover $X_i(t)$ becomes 1.

We can further split the time to event into time from randomization to crossover (T_{off}) and time from crossover to event (T_{on}). Then equation (1) can be written into two parts:

$$U_i = \int_0^{T_{offi}} \exp[\Psi X_i(t)] dt + \int_{T_{offi}}^{T_i} \exp[\Psi X_i(t)] dt \dots\dots\dots (3)$$

$$U_i = T_{offi} + T_i \exp(\Psi) \dots\dots\dots (4)$$

The factor $\exp(\Psi)$ is called an acceleration factor, representing the ratio of lifetime if $X_i(t)=0$ to $X_i(t)=1$.

Estimation of Ψ is a key step in formulating RPSFT models. There are different ways this estimation can be performed. A commonly used method is the log-rank test where $Z(\Psi)$ is used as a test statistic for $U(\Psi)$. $Z(\Psi)$ should be equal to 0 at the optimal value of Ψ . The best estimate should have the same

observed time to event in the absence of any treatment. The time with no treatment is also known as counter-factual time. Same counter-factual time will generate similar Kaplan-Meier plot which will show optimal value of the parameter ψ has been reached.

An add-on package (*RPSFTM*) publically available for fitting rank preserving structural failure time models is available for R (Bond and Allison, 2017), and can be installed from the CRAN web portal. Included with the *RPSFTM* package for illustrative purposes is a simulated 1000 observation dataset (*immdef.csv*) based on a randomized controlled trial comparing immediate or deferred treatment of zidovudine in symptom free HIV+ individuals. We will present the use of *RPSFTM* in R with the example provided by Bond and Allison, and also briefly demonstrate SAS code to duplicate the basic determination of Ψ .

The *immdef.csv* dataset has 9 variables simulated for 1000 participants:

1. **id** – subject identifier
2. **def** - indicator for deferred treatment arm. Subjects who are on placebo and are likely to switch to active treatment at the onset of disease progression will be assigned 1. Subjects who are randomized to active treatment from the beginning will have a value of 0.
3. **imm** – indicator that assigns 1 to subjects who are randomized to active treatment from the beginning. The rest have value 0.
4. **censyrs** – time from randomization to end of trial
5. **xo** – indicator to assign 1 if patient switched treatment, otherwise 0
6. **xoyrs** - time from randomization to treatment crossover. Subjects on immediate treatment arm have value 0
7. **prog** – indicator to assign 1 for disease progression, 0 otherwise
8. **progyrs** – time from randomization to disease progression
9. **entry** – time from beginning of the study to subject's participation

To run the package, the proportion of time spent on treatment (rx) must also be created for all subjects. This proportion is 1 for all subjects in the immediate arm, since none could switch to the deferred treatment, 0 for subjects in the deferred group that never crossed over to receive treatment, and **1 – (xoyrs/progyrs)** in deferred subjects that did receive treatment.

We began with Minimal Kaplan-Meier results and Cox proportional hazards analysis with R:

```
KM_noadj <- survfit(Surv(progyrs, prog) ~ imm, data=immdef)
survdiff(Surv(progyrs, prog) ~ imm, data=immdef)
survfit(Surv(progyrs, prog) ~ imm, data=immdef)
coxph(Surv(progyrs, prog) ~ imm, data=immdef)
cox_noadj <- coxph(Surv(progyrs, prog) ~ imm, data=immdef)

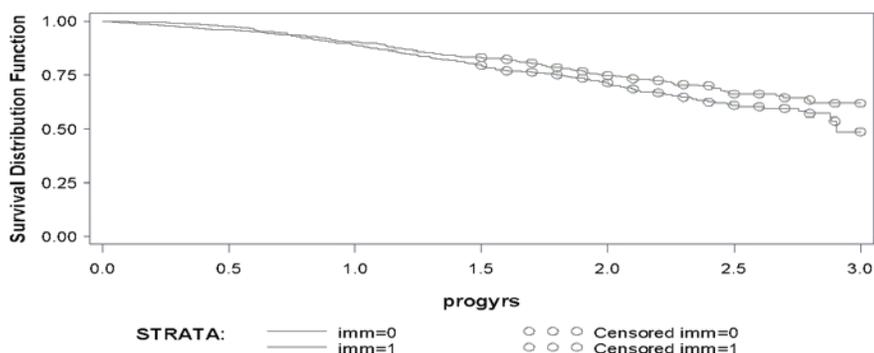
plot(KM_noadj, main=expression(paste("Kaplan-Meier-estimate without any
adjustment")), xlab="time", ylab="survival", lwd=2, col=1:3)
```

and also with SAS:

```
proc lifetest data = work.immdef plots=(s) method = km;
  time progyrs * prog (0); strata imm;

proc phreg data=work.immdef;
  class imm (ref='0');
  model progyrs*prog(0)=imm; ;
run;
```

Results were consistent regardless of package used. There were 143 progression events observed in the immediate treatment arm, with 71% of subjects censored. In the deferred arm, there were 169 progression events with 66% of subjects censored. Comparing between the two arms a log-rank $\chi^2=3.7$, $p=0.0556$. The maximum likelihood estimate of the hazard ratio was 0.805, and the following KM plot illustrates survival of subjects in the two treatment arms, obtained from SAS:



Next, the RPSFT model was run in R using the *RPSFTM* package with the following results (excerpted):

```
rpsftm_fit_lr <- rpsftm(formula=Surv(progyrs, prog) ~ rand(imm, rx),
                        data=immdef, censor_time=censyrs)
summary(rpsftm_fit_lr)

psi: -0.181
exp(psi): 0.834
Confidence Interval, psi -0.349 0.002
Confidence Interval, exp(psi) 0.705 1.002
```

In SAS, the steps necessary to run the RPSFT model (Appendix A) were implemented using a combination of DATA step programming (to establish the model) and iterative macro processing in order to obtain the estimate for Ψ . For purposes of this example, we used a search grid of -0.5 to 0.1, with an interval of 0.001, since we had an indication available already from R.

Using PROC LIFETEST, the immediate and deferred treatment arms were compared at every guess of Ψ , and logrank test statistics and variances were collected to create Z scores. The point estimate for Ψ is the value where $Z(\Psi)=0$ and confidence interval for Ψ is the range of values for which $|Z(\Psi)| < z_{1-\alpha/2}$. Our results from SAS using this search grid and interval, to 3 decimal places, were as follows:

```
psi: -0.181
exp(psi): 0.834
Confidence Interval, psi -0.349 0.002
Confidence Interval, exp(psi) 0.705 1.002
```

As mentioned earlier, there are other approaches to estimate Ψ apart from log rank test. Either a Cox regression model or Weibull model can be used to find the optimal value of Ψ . In R, *coxph* and *survreg*

can be specified in the *RPSFTM* test option for Cox regression model and Weibull model respectively. $Z(\Psi)$ should be equal to 0 for Weibull model and for Cox model $Z(\Psi)$ should cross 0. The Weibull model and Cox regression approaches to estimate Ψ may also be implemented in SAS.

As mentioned earlier, a key assumption of the RPSFT model is to include the treatment effect for crossover patients in the analysis. Time to event for these patients is re-calculated based on the optimal Ψ . This assumption also needs to be considered when determining event status for patients on the deferred arm. For patients in deferred arm, time to event will only include time spent off treatment and may show more events relative to the treatment arm. This will cause biased estimation of survival time. A correction to minimize this effect in censoring time is mentioned in Robins and Tsiatis (1991). The following equation (5) shows how censoring time is considered:

$$D_i = \text{Min}(C_i, C_i \exp(\Psi)) \quad (5)$$

where C_i is the predetermined censoring time, most often the end of study, or data cutoff.

The RPSFT model re-establishes censoring based on the optimal Ψ and equation 5. New values are only applicable for subjects on the deferred arm. For the patients in the deferred arm, the value of the censoring indicator will change if the updated U_i (obtained from equation 4) is greater than D_i . The rest of the patients in the group will remain censored. There will be no change in censoring status for patients in the immediate arm.

We incorporated the above assumptions in both R and SAS (see Appendix A, Step 4 dataset) to recalculate censoring status and event times for patients in the deferred treatment arm. Standard Kaplan-Meier methods to estimate median time to event and Cox regression may be performed to investigate the updated hazard ratio and assess possible bias in the intent-to-treat analysis, which does not take into account the possibility of crossing-over to an active treatment compound. Results were consistent regardless of package used. There were 143 progression events observed in the immediate treatment arm, with 71% of subjects censored. In the deferred arm, there were 143 progression events with 71% of subjects censored. Comparing between the two arms a log-rank $\chi^2=5.1$, $p=0.0237$. The maximum likelihood estimate of the hazard ratio was 0.761.

CONCLUSION

This paper describes implementation of the RPSFT model using both SAS and R. The RPSFT model provides a randomization based method by which investigators can review the impact, and potential bias, of treatment change over the course of a clinical study. Conventional Cox regression is a common analysis technique for time-to event endpoints, and for programmers and statisticians accustomed to working with SAS, is easily accomplished through the PHREG procedure. Unfortunately, a built-in functionality to conduct the RPSFT model in SAS is not widely available, as far as we were aware, though a well-documented and efficient add-on package (*RPSFTM*) in R was. While implementation of the technique is relatively straight-forward using SAS, consistent results were obtained more efficiently using the publicly available add-on in R.

REFERENCES

- Allison, Paul D. 2010. *Survival Analysis Using SAS®: A Practical Guide*, Second Edition, Cary, NC: SAS® Institute Inc.
- Bond, S. and A. Allison. 2017. *rpsftm: Rank Preserving Structural Failure Time Models*. R package version 1.2.1. <http://CRAN.R-project.org/package=rpsftm>
- Korhonen, P., N. Laird, and J. Palmgren. (1999). Correcting for Non-Compliance in Randomized Trials: An Application to the ATBC Study. *Statistics in Medicine* 18:2879–2897.

Korhonen, P., E. Zuber, M. Branson, N. Hollaender, N. Yateman, T. Katiskalahti, D. Lebwohl and T. Haas. (2012). Correcting Overall Survival for the Impact of Crossover Via a Rank-Preserving Structural Failure Time (RPSFT) Model in the RECORD-1 Trial of Everolimus in Metastatic Renal-Cell Carcinoma. *Journal of Biopharmaceutical Statistics* 22:1258-1271.

Robins, J. and A.Tsiatis. (1991). Correcting for Non-Compliance in Randomized Trials Using Rank Preserving Structural Failure Time Models. *Communication in Statistics – Theory and Methods* 20(8):2609-2631.

White, I. A. Babiker, S. Walker and J. Darbyshire. (1999). Randomization-Based Methods for Correcting for Treatment Changes: Examples from the Concorde Trial. *Statistics in Medicine*, 18: 2617-2634.

ACKNOWLEDGMENTS

We would like to thank our colleagues in the Biostatistics and Statistical Programming groups we work with on a daily basis, and that provided real-world opportunities which provided inspiration for the paper topic itself. Their insights and guidance are appreciated.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Please contact the authors at:

Bradford J Danner
E-mail: bradford.danner@gcesolutions.com

Indrani Sarkar
E-mail: indrani.sarkar@gcesolutions.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

APPENDIX

```

data immdef_dataset;
  infile test(immdef.csv) dsd delimiter=', ' firstobs=2;
  input obnum id def imm censyrs xo xoyrs prog progyrs entry;
  length immcat $20.;
  if imm=0 then immcat='Defer';
  else if imm=1 then immcat='Immediate';
  drop obnum;
run;

%macro find_U(psi = );
  if imm eq 1 then do;
    t_on = progyrs;
    t_off = 0;
  end;
  if imm eq 0 then do;
    if xo eq 1 then do;
      t_on = progyrs - xoyrs;
      t_off = xoyrs;
    end;
    else do;
      t_on = 0;
      t_off = progyrs;
    end;
  end;

  T = t_off + t_on;
  U = t_off + exp(&psi) * t_on;

  cut_U = min(exp(&psi)*censyrs,censyrs);

  if prog eq 1 then do;
    if U > cut_U then do;
      U = cut_U;
      event_U = 0;
    end;
    else do;
      event_U = 1;
    end;
  end;
  else do;
    event_U = 0;
    if U > cut_U then U = cut_U;
  end;

%mend find_U;

data step1;
  set immdef_dataset;
  do psi_search = -0.5 to 1.0 by 0.001;
    %find_U(psi = psi_search);
  output;
  end;
run;

proc sort data = step1; by psi_search id; run;

```

```

ods listing close;
ods select homstats logrankhomcov;
ods output homstats = hstats (keep = psi_search immcat logrank)
           LogRankHomCov = hcov (keep = psi_search immcat defer );
proc lifetest data = step1 ;
  by psi_search;
  time U*event_U(0);
  strata immcat / test = logrank ;
run;
ods listing;

data step2 (keep = psi_search logrank z_value);
  merge hstats hcov;
  by psi_search immcat;
  if immcat = "Defer";
  Z_value = logrank / sqrt(defer);
run;

data step3;
  set step2;
  by psi_search;
  retain last_Z_value;
  if Z_value eq 0 then output;
  if _n_ ne 1 then do;
    t = Z_value * last_Z_value;
    if Z_value * last_Z_value < 0 then do;
      psi_search = psi_search - 0.0001/2;
      output;
    end;
  end;
  last_Z_value = Z_value;
run;

proc sql noprint;
  select distinct psi_search into :psi0 separated by "," from step3;
quit;

title1 'Plot of Z_value vs. psi try';
proc gplot data = step2;
  plot Z_value * psi_search / vref = 0 href = &psi0;
run;
quit;

data step4;
  set immdef_dataset;
  %find_U(psi = &psi0);
  if imm = 1 then do;
    progyrs_U = progyrs;
    prog_U = prog;
  end;
  else if imm=0 then do;
    progyrs_U = U;
    prog_U = event_U;
  end;
run;

```