

A Macro of Evaluating the Performance of the Log-rank test Using Different Weights for Enrichment Studies

Chuanwu Zhang M.S., Byron J. Gajewski Ph.D, and Jianghua (Wendy) He Ph.D. Department of Biostatistics & Data Science, University of Kansas Medical Center, Kansas City, KS

ABSTRACT

Log-rank test is a traditional nonparametric hypothesis test often used to compare the time-to-event data of two arms. Different weight functions applied to the log-rank test lead to different tests, e.g., the log-rank, Gehan, Peto-Peto, Fleming-Harrington, etc. Little is known regarding the performance of log-rank test with different weight functions when one arm contains subgroups with different hazard rates, which can be seen in enrichment studies. For example, in a two arms and two subgroups study, compared to the control arm, one treatment subgroup may benefit gradually, while the other treatment subgroup may benefit immediately then lose it. How can we know the most efficient (i.e. best performance) weight functions for log-rank test should be selected for analyzing such data?

In this paper, we will develop a macro to examine the performances of the log-rank test using different weight functions in terms of empirical power based on simulated data. Our macro contains three parts. Firstly, data is simulated with a basket of factors. They include the sample sizes and number of simulations, the type of distribution for data, the proportion of subgroups, the proportion of censoring within each arm and the pattern of hazard function for a subgroup. Secondly, the macro executes the log-rank test using different weight functions. Finally, the testing results are presented in a table and a graph to compare the performances. Users can invoke our macro to determine which weight function in log-rank test should be used to meet their needs.

INTRODUCTION

Log-rank test is a traditional nonparametric hypothesis test often used to compare the time-to-event data of K populations ($K \geq 2$). (Klein and Moeschberger, 2003). The hypotheses for multiple arms are:

$$H_0: h_1(t) \dots = h_k(t), \text{ for all } t \leq \tau, \text{ where } \tau \text{ is the largest time value at least one subjst at risk}$$

$$H_1: h_j(t) \neq h_g(t). \text{ For some } t \leq \tau, \forall j, g. j \leq k, g \leq k.$$

The test statistic is $(Z_1(\tau), Z_2(\tau) \dots Z_K(\tau)) \Sigma^{-1} (Z_1(\tau), Z_2(\tau) \dots Z_K(\tau))^T$. It is distributed as χ_{K-1}^2 under H_0 for large sample. $Z_j(\tau) = \sum_{i=1}^D W(t_i) \left(d_{ij} - Y_{ij} \frac{d_i}{Y_i} \right)$. $W(t_i)$ is a weight function defined by the pool data. d_{ij} is the number of events at t_i for the j th arm, and $i = 1, 2, \dots, D$ and $t_1 < t_2 \dots < t_D$, and $j = 1, 2, \dots, K$. Y_{ij} is the number of observations at risk right before t_i from the j th arm. $d_i = \sum_{j=1}^K d_{ij}$ and $Y_i = \sum_{j=1}^K Y_{ij}$. Σ is the variance-covariance matrix of $(Z_1(\tau), Z_2(\tau) \dots Z_K(\tau))^T$ with the dimension of $K \times K$. The variance of $Z_j(\tau)$: $\hat{\sigma}_{jj}(\tau) = \sum_{i=1}^D W^2(t_i) d_{ij} \frac{Y_{ij}}{Y_i} \left(1 - \frac{Y_{ij}}{Y_i} \right) \left(\frac{Y_i - d_i}{Y_i - 1} \right)$, and the covariance of $Z_j(\tau), Z_g(\tau)$: $\hat{\sigma}_{jg}(\tau) = - \sum_{i=1}^D W^2(t_i) d_i \frac{Y_{ij}}{Y_i} \frac{Y_{ig}}{Y_i} \left(\frac{Y_i - d_i}{Y_i - 1} \right)$, $j \neq g$.

Different weight functions applied to the log-rank test lead to different tests due to the different definitions of the weight function. Table 1 blow presents the common weight function expressions. Little is known regarding the performance of log-rank test with different weight functions when one arm has multiple subgroups with different hazard rates, which can be seen in enrichment studies. For example, compared to control arm having a hazard rate as the comparison, the subgroup in treatment arm may benefit gradually, i.e. the hazard rate of the subgroup in treatment arm starts with a similar value as that in the control arm and decrease gradually as the treatment proceeds. We can name such changing type of hazard rate of the subgroup in the treatment arm as "decrease". There is also a possibility that the related subgroup in the treatment arm may benefit immediately then lose it, which means the hazard rate for such a subgroup starts with a smaller value and then increase gradually as the study time elapses. We name such hazard rate changing type as "Increase". The subgroup in the treatment arm may also follow a

“concave” hazard rate changing type, which means that the hazard rate increases in the first half of the treatment duration and then decrease in the second half. Conversely, the “convex” hazard rate changing type means that the hazard rate firstly decreases then increases. In a specific study with the consideration of the subgroups, there should be at least two subgroups within each arm. Since one subgroup factor at least has two levels. For example, if we consider gender as the subgroup factor, female and male are the specific subgroups. How can we know the most efficient (i.e. best performance) weight functions for log-rank test should be selected for analyzing such data from enrichment studies?

Weight Function	Expression
Log-rank	1
Gehan (Named Wilcoxon in SAS)	Y_i
Tarone-Ware	$\text{Sqrt}(Y_i)$
Peto-Peto	$\tilde{S}(t_i), \tilde{S}(t_i) = \begin{cases} 1; & \text{if } t < t_1 \\ \prod_{t_i \leq t} \left[1 - \frac{d_i}{Y_{i+1}} \right]; & \text{if } t \geq t_1 \end{cases}$
Modified Peto-Peto	$\tilde{S}(t_i) Y_i / (Y_i + 1)$
Fleming-Harrington	$[\hat{S}(t_{i-1})]^p [1 - \hat{S}(t_{i-1})]^q, p \geq 0, q \geq 0; \hat{S}(t_{i-1})$ is similar to $\tilde{S}(t_i)$ and replace $(Y_i + 1)$ to Y_i .

Table 1 The Expression Of Common Weight Functions

In this paper, we focus on the tests with different weight functions for two arms, and in each of arm there are two subgroups. We will develop a macro to examine the performances of the log-rank test using different weight functions in terms of empirical power based on simulated data. Our macro contains three parts. Firstly, data is simulated with a basket of factors that the sample sizes and number of simulations, the type of distribution for survival data, the proportion of subgroups, the proportion of censoring within each arm, and the pattern of hazard function for a subgroup. Secondly, the macro executes the log-rank test using different weight functions. Finally, the testing results are presented in a summary table and a graph to compare the performances. Users can invoke our macro to determine which weight function should be used for the log-rank test to meet their needs.

THE MACRO PART I: DATA SIMULATION

The part I of the macro is essential, which is used to simulate the survival data. The data simulation factor we consider include the sample sizes and number of simulations, the type of distribution for data, the proportion of subgroups, the proportion of censoring within each arm and the pattern of hazard function for a subgroup. Each of them is the key component for the survival data simulation, and we will specify them and interpret the related SAS code in the following parts.

THE SAMPLE SIZE AND NUMBER OF SIMULATIONS

we know that sample size is the key consideration for statisticians to design a study before the study execution. Statisticians incorporate the information of statistical knowledge, logistic considerations (such as budget, accrual rate, and so on) and simulation to finalize the sample size. You can assign a value for the sample size via the macro variable - `sample_size`. We assume the ratio of control and treatment arm is 1 vs.1. We also create the macro variable - `sim_number` for the user to provide how many simulation numbers they would like to conduct. The macro key parts involved in the input of sample size and simulation number display below.

```
do sim = 1 to &sim_number.;
  ID = 0;
  obs = 0;
do i = 1 to &sample_size.;
```

THE DISTRIBUTION OF THE SURVIVAL DATA

We provide two candidate distributions - Exponential and Weibull - to simulate the survival data, i.e., we assume the survival data is distributed as either Exponential or Weibull. **Error! Reference source not found.** below summarizes the hazard rates, survival functions, probability of density function (PDF) and

expected lifetime for the two candidate distributions. These quantities are with the related formats in accordance with SAS.

Distribution	Hazard Rate, $h(x)$	Survival Function, $S(x)$	PDF, $f(x)$	Expected lifetime, $E(x)$
Exponential	$\frac{1}{\sigma}$	$\exp\left(-\frac{x}{\sigma}\right)$	$f(x) = \frac{1}{\sigma} \exp\left(-\frac{x}{\sigma}\right)$	σ
Weibull	$\frac{a}{b^a} x^{a-1}$	$\exp\left(\left(-\frac{x}{b}\right)^a\right)$	$f(x) = \frac{a}{b^a} x^{a-1} \exp\left(\left(-\frac{x}{b}\right)^a\right)$	$b\Gamma\left(1 + \frac{1}{a}\right)$

Table 2. Hazard Rate, Survival Function, PDF, And Expected Lifetime For Exponential And Weibull Distributions

As the code from our macro below shows, you firstly assign Exponential or Weibull for the macro variable `-distribution-` to specify survival data follows which distribution. You then get the simulated data by the SAS function `-Rand("distribution", parameter-1, ..., parameter-k)`, providing the related distribution parameter values via macro variables.

The value assignments for the distribution parameters are involved in the treatment effect. Specifically, for the exponential distribution, the macro variable, `exp_hr_h` which is correspondent to $\frac{1}{\sigma}$ for exponential distribution in Table 2, refers the hazard rate when treatment starts. It maintains identically during the whole treatment process for control arm, and you complete the event time simulation for control arm. The hazard rate generally changes as time elapses for treatment arm. We will specify the details regarding the event data update of treatment arm, however, the assigned value for macro variable `-exp_hr_h` currently is also used for simulation of initial event time for treatment arm. Similar logic for Weibull distribution, you assign the values for `weibull_a` and `Weibull_b`, which refer a and b in the Table 1; you then complete control arm event time data simulation, and event time in treatment arm will be updated in the following parts. We name the control and treatment arms as Arm 1 and 2, and we will use these terms interchangeably.

```

%**generate the random values for event time and censor time from
exponential or weibull distribution.;
%if %sysfunc(upcase(&distribution.)) = "EXPONENTIAL" %then %do;
  event_time = Rand("EXPONENTIAL", 1/(&exp_hr_h.));
%end;

%else %if %sysfunc(upcase(&distribution.)) = "WEIBULL" %then %do;
  event_time = Rand("WEIBULL", &weibull_a., &weibull_b.);
%end;

```

THE PROPORTION OF SUBGROUPS

As we have specified in introduction part, we assume there are two arms and two subgroups. Moreover, we assume different proportions of the subgroups may also affect the power of the log-rank test. For ease of the user exploring, we create a macro variable `-subgrp1_prop-` for the users to provide a specific value for subgroup 1, then the macro will automatically calculate the proportion for Subgroup 2 via $1 - \text{subgrp1_prop}$. Table 3 below presents several example cases of simulation studies with different proportions.

Scenario	Arm	Subgroup 1	Subgroup 2
Case 1	Control (1)	0.5	0.5
	Treatment (2)	0.5	0.5
Case 2	Control (1)	0.4	0.6
	Treatment (2)	0.4	0.6
Case 3	Control (1)	0.3	0.7
	Treatment (2)	0.3	0.7
Case N	Control (1)	:	:
	Treatment (2)	:	:

Table 3. Example Cases Of Simulation Two Arm Studies With Different Proportions Of Two Subgroups

THE PROPORTION OF CENSORING

The censor data is very important in survival analysis. To simplify the simulation and data analysis, we assume that the censoring is independent of event data. On the other hand, the censoring proportion plays a non-negligible effect in the survival data analysis, and we assume the censoring proportion is identical within the two arms.

We create a macro variable - `censor_prop` - for the user to provide a concrete proportion value ranging from 0 to 1. After the user specify the censor portion, we then derive the related distribution parameter values based on the relationship of censor proportion and hazard rate, i.e., $\frac{h_c(x)}{h_e(x)} = \frac{P_c}{P_e} = \frac{P_c}{1-P_c}$, where $h_c(x)$ and $h_e(x)$ are the hazard rate for censor and event data; P_c and P_e are the proportion of censor and event data. Specifically, as the SAS code presents below, since the hazard rate is constant under exponential distribution, the hazard rate of censor data is straightforward and is equal to $\frac{P_c}{1-P_c}h_e(x)$. You specify the values for P_c and $h_e(x)$ via the macro variables - `censor_prop` and `exp_hr_h`. The macro then creates the local macro variable - `exp_hazard_rate_cen` to calculate the hazard rate for censoring simulation under exponential distribution. For Weibull distribution, the hazard rate is $\frac{a}{b^a}x^{a-1}$, and we assume a is fixed, and for a specific event and censored data, we have $b_c = b_e \left(\frac{1-P_c}{P_c}\right)^{\frac{1}{a}}$, where b_e and b_c are the parameters of Weibull distribution for event and censored data. Then you use the calculated b_c and the fixed one - a - to simulate the censor data. The key component codes display below.

```

%*create two local macro variables for censor time simulation based on
related censor proportion and distribution parameter;
%let exp_hazard_rate_cen = (&censor_prop./ (1 - &censor_prop.)) *
&exp_hr_h.;
%let wei_cetime_b = &weibull_b.* ((1-&censor_prop.)/&censor_prop.)
** (1/(&weibull_a.));

%**generate the random values for event time and censor time from
exponential or weibull distribution.;
%if %sysfunc(upcase(&distribution.)) = "EXPONENTIAL" %then %do;
  censor_time = Rand("EXPONENTIAL", 1/(&exp_hazard_rate_cen.));
%end;

%else %if %sysfunc(upcase(&distribution.)) = "WEIBULL" %then %do;
  censor_time = Rand("WEIBULL", &weibull_a., &wei_cetime_b.); ;
%end;

```

Please be advised that the actual censored proportion from the simulated data may not be exactly identical to the value you specify. It depends on the sample size. From Table 4, you can identify that as the sample size increases, the censoring proportion is closer to the specified one.

Then based on the simulated event and censor time, the macro creates the survival time and censor status. As the code below indicates, if the censor time is less than the event time, then that observation is censored and the survival time is equal to censor time; otherwise, the survival time is equal to event time and no censoring. Please note that the censor status and survival time derive so far is for the control arm and original treatment arm. The treatment arm survival time and censor status will be updated based on the hazard rate changing type, which we introduce in the following part.

```

if censor_time lt event_time then do;
  censored = 1;
  survival_time = censor_time;
end;
else do;
  censored = 0;

```

```

survival_time = event_time;
end ;

```

Distribution	Sample Size	Specified Censor Proportion	Observed Censor Proportion
Exponential	100	0.2	0.1976
Exponential	200	0.2	0.1982
Exponential	500	0.2	0.2014
Exponential	800	0.2	0.1992
Weibull	100	0.2	0.2060
Weibull	200	0.2	0.2028
Weibull	500	0.2	0.2016
Weibull	800	0.2	0.2016

Table 4 The Summary Of The Specified And Observed Censor Proportion Of Survival Data Under Different Sample Size Via 1000 Simulation

THE PATTERN OF HAZARD FUNCTION FOR A SUBGROUP

This is the most important part of the macro to achieve the enrichment assumptions of the subgroups. We provide four changing types of the hazard rate, and they are decrease, increase, concave and convex.

The decrease type refers the hazard rate of treatment arm is similar but a little lower than that of control arm, and it decreases as time elapses. Correspondingly, the increase means that the hazard rate of treatment arm is much lower than that of control arm when the treatment starts, and it increases as treatment carries through but still lower than the control arm by the end of the study. For the concave type, it means that the hazard rate of the Arm 2 is similar to that in Arm 1 in the middle of the treatment duration, and it is lower at the beginning and end of the study. Conversely, the convex type means that the Arm 2 hazard rate is lower in the middle of the study, and it is similar to the one from Arm 1 when the treatment initiates and completes.

The core of this paper is that the hazard rate changing type of two subgroups within the Arm 2 are different. You can specify the specific types for the two subgroups via the two macro variables - `subgrp1_type` and `subgrp2_type`. The sample code below briefly demonstrates how the macro identify the four changing types. Totally, there are six different combinations of the hazard rate changing type for two subgroups as Table 5 presents.

```

%if %sysfunc(lowcase(&subgrp1_type.)) = "decrease" %then %do;
.....
%end;

%else %if %sysfunc(lowcase(&subgrp1_type.)) = "increase" %then %do;
.....
%end;

%else %if %sysfunc(lowcase(&subgrp1_type.)) = "concave" %then %do;
.....
%end;

%else %if %sysfunc(lowcase(&subgrp1_type.)) = "convex" %then %do;
.....
%end;

```

Combination	Subgroup 1	Subgroup 2
1	Increase	Decrease
2	Increase	Concave
3	Increase	Convex
4	Decrease	Concave
5	Decrease	Convex

6	Concave	Convex
---	---------	--------

Table 5 Six Different Combinations Of The Hazard Rate Changing Type For Two Subgroups in Treatment Arm

The algorithm below containing several steps achieves the goal of different changing type of hazard rate for each subgroup.

Step 1. Get the Percentile 20, 40, 60 and 80 of original event time for each subgroup within Arm 2 under each simulation. Key component of the code lists below.

```
proc means data = sa_enrich_raw p20 p40 p60 p80;
  where arm = 2;
  class sim sub_grp;
  var event_time;
run;
```

Step 2. As the Figure 1 below indicates, each subgroup within Arm 2 has been divided into five parts based on the percentiles, minimum and maximum. Each simulated observation with survival and censored time will be grouped in one of the five parts.

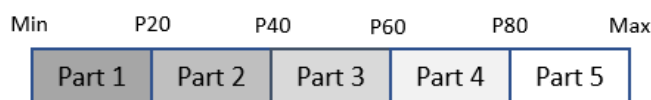


Figure.1 The Five Parts Of The Subgroup In Arm 2 Divided By Percentiles

Step 3. For hazard rate with a decrease type, it decreases gradually from Part 1 to 5.

Step 3.1. This sub-step is the specification of hazard rate with a decrease type for survival data following the Exponential distribution.

As table 2 presents, the hazard rate is a constant, which is the reciprocal of expected lifetime. In our macro, we provide two macro variables - `exp_hr_h` and `exp_hr_l` - for the user to input highest and lowest in treatment period.

For each part of subgroup with a decrease hazard rate changing type, the relationship of updated and original event time, updated hazard rate and original hazard rate is: $\frac{E_u}{E_s} = \frac{h_s}{h_u} = \frac{h_s}{h_s - (h_s - h_e) \times \frac{i}{5}} (1E_Drcs)$,

where E_u and E_s are the updated and original event time; h_s , h_u and h_e are the hazard rate at upmost, updated for each part and at lowest; $i = 1$ to 5 by 1 for each part of subgroup. Based on the equation (1E_Drcs), we can have the updated event time for each subgroup: $E_u = E_s \frac{h_s}{h_s - (h_s - h_e) \times \frac{i}{5}} (2E_Drcs)$. We

update event time for each part at subject level, i.e., we use the formula (2E_Drcs) to update the event time for each observation.

Same logic will be applied for updated censored time based on related hazard rates for censoring data. Then, final survival time and censor status for such a subgroup in treatment arm will be finalized based on the updated event and censor time, and the rationale is identical to that applied for control arm observation which is described in section of the proportion of censoring.

Step 3.2. This sub-step is the specification of hazard rate with a decrease type for survival data following the Weibull distribution.

Still, as the presentation in table 2, the hazard rate is $\frac{a}{b^a} x^{a-1}$, which is a function of survival time at subject level. Suppose the original hazard rate is h_s , the hazard rate at lowest is h_e . Let the ratio of the two guys equal to HR_R, i.e., $\frac{h_e}{h_s} = HR_R$. You can assign a specific ratio value for HR_R via the macro variable - `wb_hr_ratio`.

Denote the updated hazard rate at each part of subgroup as h_u . To meet the hazard rate decrease type, we then have $\frac{h_u}{h_s} = 1 - (1 - HR_R) \frac{i}{5} = \frac{\frac{a}{b^a} E_u^{a-1}}{\frac{a}{b^a} E_s^{a-1}} = \left(\frac{E_u}{E_s}\right)^{a-1} (1W_Drcs)$, where E_s and E_u means the

original and update survival time for each subject; i has the identical meaning as in formula (1E_Drcs).

Based on (1W_Drcs), we have $E_u = E_s \left(1 - (1 - HR_R) \frac{i}{5}\right)^{\frac{1}{a-1}}$ (2W_Drcs). We then use the formula (2W_Drcs) to update the event time for each observation.

Regarding the updated censored data, final survival data and censoring status, same logic and procedures will be applied as those from step 3.1.

Step 4. For hazard rate with an increase type, it starts a smaller value at Part 1 and then increases gradually from Part 2 to 5.

Step 4.1. This sub-step is the specification of hazard rate with an increase type for survival data following the Exponential distribution.

The idea is similar to that from step 3.1. The only thing needed is to update the formula (1E_Drcs) since we have to meet the specification of increasing hazard rate: $\frac{E_u}{E_s} = \frac{h_s}{h_u} = \frac{h_s}{h_s - (h_s - h_e) \times \frac{(6-i)}{5}}$ (1E_Incs). Based on (1E_Incs), we have: $E_u = E_s \frac{h_s}{h_s - (h_s - h_e) \times \frac{(6-i)}{5}}$ (2E_Incs).

Still, we execute the formula (2E_Incs) at subject level to update the event time, and use same logic to update the censored data based on the related censored hazard rate. Then, finalize the survival time and censoring status based on the logic from control arm.

Step 4.2. This sub-step is the specification of hazard rate with an increase type for survival data following the Weibull distribution.

The idea is similar to that from step 3.2. The only thing needed is to update the formula (1W_Drcs) since we have to meet the specification of increasing hazard rate: $\frac{h_u}{h_s} = 1 - (1 - HR_R) \frac{(6-i)}{5} = \left(\frac{E_u}{E_s}\right)^{a-1}$ (1W_Incs). Based on (1W_Incs), we have: $E_u = E_s \left(1 - (1 - HR_R) \frac{(6-i)}{5}\right)^{\frac{1}{a-1}}$ (2W_Incs).

Still, we execute the formula (2W_Incs) to update the event time for each subject, and use same logic to update the censored data, finalize the survival time and censoring status.

Step 5: For hazard rate with a concave type, it is lower than that from control arm at the beginning and end of the study; then it has a similar value as in control arm in the middle of the study duration.

Step 5.1. This sub-step is the specification of hazard rate with a concave type for survival data following the Exponential distribution.

Still, the idea is similar to that from step 3.1. We need to update the formula (1E_Drcs) since we have to meet the specification that hazard rate is with a concave changing type: $\frac{E_u}{E_s} = \frac{h_s}{h_u} = \frac{h_s}{h_s - (h_s - h_e) \times \frac{\text{abs}(3-i)}{2}}$ (1E_Cncv). Based on (1E_Cncv), we have: $E_u = E_s \frac{h_s}{h_s - (h_s - h_e) \times \frac{\text{abs}(3-i)}{2}}$ (2E_Cncv).

Still, we execute the formula (2E_Cncv) at subject level to update the event time, and use same logic to update the censored data based on the related censored hazard rate. Then, finalize the survival time and censoring status based on the logic from control arm.

Step 5.2. This sub-step is the specification of hazard rate with a concave type for survival data following the Weibull distribution.

Still, the idea is similar to that from step 3.2. The only thing needed is to update the formula (1W_Drcs) since we have to meet the specification that hazard rate is with a concave changing type: $\frac{h_u}{h_s} = 1 - (1 - HR_R) \frac{\text{abs}(3-i)}{2} = \left(\frac{E_u}{E_s}\right)^{a-1}$ (1W_Cncv). Based on (1W_Cncv), we have: $E_u = E_s \left(1 - (1 - HR_R) \frac{\text{abs}(3-i)}{2}\right)^{\frac{1}{a-1}}$ (2W_Cncv).

Still, we execute the formula (2W_Cncv) to update the event time for each subject, and use same logic to update the censored data, finalize the survival time and censoring status.

Step 6: For hazard rate with a convex type, it is lower than that from control arm in the middle of the study duration; then it has a similar value as in control arm in the middle of the study duration at the beginning and end of the study.

Step 6.1. This sub-step is the specification of hazard rate with a convex type for survival data following the Exponential distribution.

The idea is similar to that from step 3.1. The only thing needed is to update the formula (1E_Drcs) since we have to meet the specification that hazard rate is with a convex changing type: $\frac{E_u}{E_s} = \frac{h_s}{h_u} = \frac{h_s}{h_s - (h_s - h_e) \times \frac{2 - \text{abs}(3-i)}{2}}$ (1E_Cnvx). From (1E_Cnvx), we have: $E_u = E_s \frac{h_s}{h_s - (h_s - h_e) \times \frac{2 - \text{abs}(3-i)}{2}}$ (2E_Cnvx).

Similarly, we execute the formula (2E_Cnvx) at subject level to update the event time, and use same logic to update the censored data based on the related censored hazard rate. Then, finalize the survival time and censoring status based on the logic from control arm.

Step 6.2. This sub-step is the specification of hazard rate with a convex type for survival data following the Weibull distribution.

The idea is similar to that from step 3.2. The only thing needed is to update the formula (1W_Drcs) since we have to meet the specification that hazard rate is with a concave changing type: $\frac{h_u}{h_s} = 1 - (1 - HR_R) \frac{(3 - \text{abs}(3-i))}{2} = \left(\frac{E_u}{E_s}\right)^{a-1}$ (1W_Cnvx). Based on (1W_Cnvx), we have: $E_u = E_s \left(1 - (1 - HR_R) \frac{(3 - \text{abs}(3-i))}{2}\right)^{\frac{1}{a-1}}$ (2W_Cnvx).

Similarly, we execute the formula (2W_Cnvx) to update the event time for each subject, and use same logic to update the censored data based on the related censored hazard rate. Then, finalize the survival time and censoring status based on the logic from control arm.

The code below is the key component of SAS macro to implement the algorithm to achieve the different hazard rate changing type.

```
%if %sysfunc(upcase(&distribution.)) = "EXPONENTIAL" %then %do;

%if %sysfunc(lowcase(&subgrp2_type.)) = "decrease" %then %do;
  event_time_2 = event_time * &exp_hr_h./(&exp_hr_h. - (&exp_hr_h.-
&exp_hr_l.)*time_percentile_grp/5);
  censor_time_2 = censor_time * &exp_hr_h./(&exp_hr_h. - (&exp_hr_h.-
&exp_hr_l.)*time_percentile_grp/5);
%end;

%else %if %sysfunc(lowcase(&subgrp2_type.)) = "increase" %then %do;
  event_time_2 = event_time * &exp_hr_h./(&exp_hr_h. - (&exp_hr_h.-
&exp_hr_l.)*(6-time_percentile_grp)/5);
  censor_time_2 = censor_time * &exp_hr_h./(&exp_hr_h. - (&exp_hr_h.-
&exp_hr_l.)*(6-time_percentile_grp)/5);
%end;

%else %if %sysfunc(lowcase(&subgrp2_type.)) = "concave" %then %do;
  event_time_2 = event_time * &exp_hr_h./(&exp_hr_h. - (&exp_hr_h.-
&exp_hr_l.)*abs(3-time_percentile_grp)/2);
  censor_time_2 = censor_time * &exp_hr_h./(&exp_hr_h. - (&exp_hr_h.-
&exp_hr_l.)*abs(3-time_percentile_grp)/2);
%end;

%else %if %sysfunc(lowcase(&subgrp2_type.)) = "convex" %then %do;
  event_time_2 = event_time * &exp_hr_h./(&exp_hr_h. - (&exp_hr_h.-
&exp_hr_l.)*abs(2-abs(3-time_percentile_grp))/2);
```



```

        censor_time_2 = censor_time * &exp_hr_h./(&exp_hr_h. - (&exp_hr_h.-
        &exp_hr_l.)*abs(2-abs(3-time_percentile_grp))/2);
    %end;

%end;

%else %if %sysfunc(upcase(&distribution.)) = "WEIBULL" %then %do;

%if %sysfunc(lowcase(&subgrp2_type.)) = "decrease" %then %do;
    event_time_2 = event_time * (1-(1-
    &wb_hr_ratio.)*time_percentile_grp/5)**(1/(&weibull_a.-1));
    censor_time_2 = censor_time * (1-(1-
    &wb_hr_ratio.)*time_percentile_grp/5)**(1/(&weibull_a.-1));
%end;

%else %if %sysfunc(lowcase(&subgrp2_type.)) = "increase" %then %do;
    event_time_2 = event_time * (1-(1-&wb_hr_ratio.)*(6-
    time_percentile_grp)/5)**(1/(&weibull_a.-1));
    censor_time_2 = censor_time * (1-(1-&wb_hr_ratio.)*(6-
    time_percentile_grp)/5)**(1/(&weibull_a.-1));
%end;

%else %if %sysfunc(lowcase(&subgrp2_type.)) = "concave" %then %do;
    event_time_2 = event_time * (1-(1-
    &wb_hr_ratio.)*abs(time_percentile_grp-3)/2)**(1/(&weibull_a.-1)) ;
    censor_time_2 = censor_time * (1-(1-
    &wb_hr_ratio.)*abs(time_percentile_grp-3)/2)**(1/(&weibull_a.-1));
%end;

%else %if %sysfunc(lowcase(&subgrp2_type.)) = "convex" %then %do;
    event_time_2 = event_time * (1-(1-&wb_hr_ratio.)*abs(2-
    abs(time_percentile_grp-3))/2)**(1/(&weibull_a.-1)) ;
    censor_time_2 = censor_time * (1-(1-&wb_hr_ratio.)*abs(2-
    abs(time_percentile_grp-3))/2)**(1/(&weibull_a.-1)) ;
%end;

%end;

```

THE MACRO PART II: LOG-RANK TEST

The second part of the macro is to implement the log-rank test for each simulated sample. It is straightforward. The common and specific weight functions provided by SAS include Logrank, Gehan (Which is also called Wilcoxon or Breslow, SAS names it as Wilcoxon), Tarone-Ware, Peto-Peto, Modified Peto-Peto, Fleming- Harrington, etc. You can input the one or more key words for the weight functions via the macro variables, `weight`, to execute related log-rank tests. Please refer Table 6 below for the related SAS keywords for the functions.

Please be advised that the macro assigns value one for censoring data, and value zero for event data.

```

ods output HomTests = pre_power;
proc lifetest data = PamaSUG.&final_ds.;
    by sim;
    time survival_time * censored (1);
    strata /test =&weight. group = arm;
run;

data power;
set pre_power;

```

```

if ProbChiSq le 0.05 then identify = 1;
else if ProbChiSq gt 0.05 then identify = 0;
run;

```

Weight Function	SAS Keyword
All Weight Functions	ALL; and Fleming- Harrington weight will be specified as FLEMING (1, 0).
Log-rank	LOGRANK
Gehan	WILCOXON
Tarone-Ware	TARONE
Peto-Peto	PETO
Modified Peto-Peto	MODPETO
Fleming- Harrington	FLEMING (Num-1, Num-2)

Table 6 The SAS Keyword For Common Weight Functions

THE MACRO PART III: CREAT REPORT AND GRAPHS TO SUMMARIZE THE POWER

The Part III of the macro is straightforward as well. Firstly, the `proc freq` will get the summary of the empirical power. The procedure - `proc sgplot`- will get the basic bar chart to demonstrate the empirical power for each weight function you specify in Part II. The key component of the code that achieves such objectives displays below. You need to specify the name of report containing the summary and graphics of the empirical power via the macro variable - `report_name`.

```

ods output CrossTabFreqs = pre_final;
proc freq data = power;
  table Test * identify/nocol nopercnt;
run;

options orientation=portrait;
ods pdf file = "&path.\&report_name..pdf" startpage = never;
title "Summary of emperical power profile for log-rank test with different
weight functions";
Proc print data = PamaSUG.&power_smry_ds. noobs label;
run;

ods graphics/noborder;
proc sgplot data = PamaSUG.&power_smry_ds.;
  vbar Test/response = RowPercent barwidth = 0.3;
  yaxis max = 1;
  label Test = "Weight Function";
run;
ods pdf close;

```

CASE STUDIES RESULT PRESENTATION

In this section, we introduce two case studies, and then describe how to invoke the macro based on the study information, and then present the summary report.

Table 7 and 8 present the related information of the case studies and how to invoke the macro accordingly. Figure 2 and 3 present the summary report.

For the overall macro and how to invoke the macro based on the two examples, please refer the supplement document.

Study Information	Macro Invocation
<ul style="list-style-type: none"> • Simulation number: 1000; • Sample size: 100; • Two arms, and randomization ratio: 1 vs. 1; • Proportion of Subgroups: 0.5 vs. 0.5 • Survival data distribution assumption: Exponential • Hazard rate highest in treatment period: 4 • Hazard rate lowest in treatment period: 1 • Censor Proportion: 0.2 • Subgroup 1 in arm 2 hazard rate changing type: Decrease • Subgroup 2 in arm 2 hazard rate changing type: Increase • Weight functions for log-rank test: logrank fleming(0,1) wilcoxon 	<pre> %PharmaSUG_SA_enrich (random_seed = 2046, sim_number = 1000, sample_size = 100, subgrp1_prop = 0.5, distribution = "Exponential", exp_hr_h = 4, exp_hr_l = 1, weibull_a = , weibull_b = , wb_hr_ratio = , censor_prop = 0.2, subgrp1_type = "Decrease", subgrp2_type = "Increase", path = C:\Users\Chwzhang2015\Desktop\PharmaSUG 2019 June 15-20\!!!Code & dataset\PharmaSUG, final_ds = exp_1, weight = logrank fleming(0,1) Wilcoxon, power_smry_ds = exp_smry_1, report_name = exp_smry_1_report); </pre>

Table 7 The Information And How To Invoke The Macro For Case One Study

Study Information	Macro Invocation
<ul style="list-style-type: none"> • Simulation number: 1000; • Sample size: 100; • Two arms, and randomization ratio: 1 vs. 1; • Proportion of Subgroups: 0.5 vs. 0.5 • Survival data distribution assumption: Weibull • Parameter A in Weibull distribution: 2 • Parameter B in Weibull distribution: 2 • Ratio of Hazard Rate lowest vs. highest in the treatment period: 0.5 • Censor Proportion: 0.2 • Subgroup 1 in arm 2 hazard rate changing type: Convex • Subgroup 2 in arm 2 hazard rate changing type: Concave • Weight functions for log-rank test: logrank fleming(0,1) wilcoxon 	<pre> %PharmaSUG_SA_enrich (random_seed = 2046, sim_number = 1000, sample_size = 100, subgrp1_prop = 0.5, distribution = "Weibull", exp_hr_h = , exp_hr_l = , weibull_a = 2, weibull_b = 2, wb_hr_ratio = 0.5, censor_prop = 0.2, subgrp1_type = "Convex", subgrp2_type = "Concave", path = C:\Users\Chwzhang2015\ Desktop\PharmaSUG 2019 June 15-20\!!!Code & dataset\PharmaSUG, final_ds = wei_2, weight = logrank fleming(0,1) wilcoxon, power_smry_ds = wei_smry_2, report_name = wei_smry_2_report); </pre>

Table 8 The Information And How To Invoke The Macro For Case Two Study

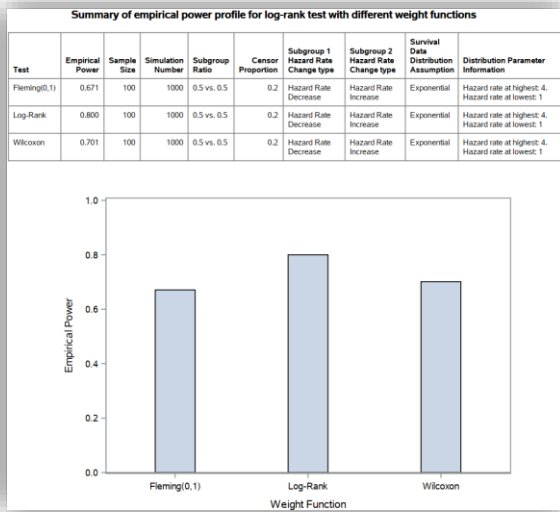


Fig.2 Summary Report Of Case One Study

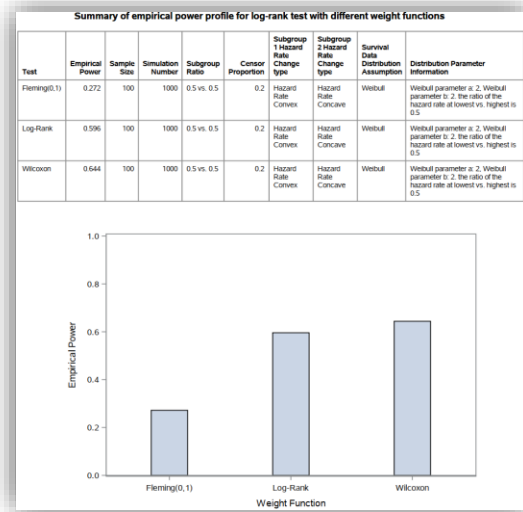


Fig.3 Summary Report Of Case Two Study

DISCUSSION

Firstly, we did not explore the empirical type I error rate because it is simple – just to simulate two arm data from the same distribution then execute the log-rank test with different weight functions.

Secondly, we arbitrarily divide the treatment subgroup into five parts, and then execute related SAS code to achieve related hazard rate changing type. The part number could be other numbers as well.

Thirdly, in this macro, we provide two survival data distributions, which are Exponential and Weibull, and the general idea and SAS logic proposed in this paper can also be applied to other distributions.

Lastly, we assume there are two arms and two subgroups in our exploring. In reality, a survival study may have more arms and more subgroups.

We try to cover as many as considerations, however, the macro will become more complicated in case it covers more different designs. We have to check and balance between the macro complication and the application scope. It's more important for the user to read the macro with ease, and they can update the macro accordingly to achieve their specific explorations.

CONCLUSION

The user can invoke our macro to explore the empirical power of the log-rank test with different weight functions for two arms with two subgroups, along with a basket of other considerations. The basket factors include the sample size, the number of simulations, the data distribution assumption (Exponential or Weibull), the proportions of the subgroups and censoring data, and the pattern (increase, decrease, concave or convex) of hazard function for a subgroup.

CONTACT INFORMATION

Any input, comment and question are valued and encouraged. Please feel free to contact the author at:

Chuanwu Zhang

Department of Biostatistics & Data Science, University of Kansas Medical Center

c Zhang4@kumc.edu

REFERENCES

Klein, J., Moeschberger, M. *Survival Analysis: Techniques For Censored And Truncated Data*. 2nd ed. New York, NY: Springer-Verlag.