

ANALYZING COHORT STUDIES WITH LEFT_TRUNCATED AND INTERVAL-
CENSORED DATA: A NEW MODEL-BASED LINEAR RANK-TYPE TEST

by

Yi Wang, MS

A thesis submitted to the Graduate Council of
Texas State University in partial fulfillment
of the requirements for the degree of
Master of Science
with a Major in Mathematics
December 2022

Committee Members:

Qiang Zhao, Chair

Alexander White

Xiaoxi Shen

COPYRIGHT

by

Yi Wang

2022

FAIR USE AND AUTHOR'S PERMISSION STATEMENT

Fair Use

This work is protected by the Copyright Laws of the United States (Public Law 94-553, section 107). Consistent with fair use as defined in the Copyright Laws, brief quotations from this material are allowed with proper acknowledgement. Use of this material for financial gain without the author's express written permission is not allowed.

Duplication Permission

As the copyright holder of this work I, Yi Wang, authorize duplication of this work, in whole or in part, for educational or scholarly purposes only.

DEDICATION

I would like to dedicate this work to my family. Without their spiritual and financial support, I would not be where I am today.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest appreciation to Dr. Zhao for giving me all help and suggestions to complete my thesis project. Dr. Zhao teaches Survival Analysis, which is an elective course for graduate students from the statistics concentration in the mathematics department. This course is my favorite, and I learned lots of knowledge relevant to survival analysis. Therefore, I decided to choose my thesis topic in this field. During my research, Dr. Zhao was always willing to assist me when I needed help. I am also thankful to my committee members Dr. White and Dr. Shen. I appreciate their time for reviewing my thesis project and serving on my thesis committee. I would also like to acknowledge Dr. Sun for teaching me mathematical statistics, regression analysis, and design and analysis of experiments, which supported my research.

TABLE OF CONTENTS

	Page
ACKNOWLEDGEMENTS.....	v
LIST OF TABLES.....	viii
LIST OF FIGURES.....	ix
ABSTRACT.....	xi
 CHAPTER	
I. INTRODUCTION.....	1
Introduction to Survival Analysis.....	1
Truncation and Censoring.....	2
Estimation of Survival Function.....	2
Comparison of Survival Distributions between Treatment Groups.....	3
Statistical Computation.....	5
Statement of Purpose	5
II. LITERATURE REVIEW.....	7
Left-truncated and Interval-censored (LTIC) Data.....	7
Estimation of Survival Function for LTIC Data.....	7
Comparison of Survival Distributions for LTIC Data	8
III. METHODOLOGY.....	11
Model.....	11
Proposed Test.....	11
IV. SIMULATION STUDIES.....	14
Simulation Settings.....	14
Simulation Results.....	17
Normality Check of the Proposed Test Statistic.....	20
Size of the Proposed Test Based on Different Models.....	21
Effect of Sample Size on the Size and Power of the Proposed Test.....	22

Effect of Level of Left Truncation on the Power Using the Proposed Test.....	24
Effect of Type of Major Censoring on the Power Using the Proposed Test.....	27
Comparison to an Existing Test	30
V. APPLICATION TO A SEMI-REAL DATA.....	39
VI. CONCLUSION.....	42
VII. DISCUSSION AND FUTURE RESEARCH.....	43
APPENDIX SECTION	44
REFERENCES.....	60

LIST OF TABLES

Table	Page
1. Parameters' Selection for the PRH Model.....	15
2. Parameters' Selection for the PH Model.....	15
3. Selected β Values for the PRH Model.....	16
4. Selected β Values for the PH Model.....	16
5. Power and Size of the Proposed Test for the PRH Model with $n = 100$	17
6. Power and Size of the Proposed Test for the PRH Model with $n = 200$	18
7. Power and Size of the Proposed Test for the PH Model with $n = 100$	19
8. Power and Size of the Proposed Test for the PH Model with $n = 200$	19
9. The Comparisons of Size and Power under the PRH Model with $n = 100$	32
10. The Comparisons of Size and Power under the PRH Model with $n = 200$	33
11. The Comparisons of Size and Power under the PH Model with $n = 100$	34
12. The Comparisons of Size and Power under the PH Model with $n = 200$	35
13. Results for the Analysis of the AIDS Blood Transfusion Data.....	41

LIST OF FIGURES

Figure	Page
1. Normality of the Proposed Test Statistic under the PRH Model with $n = 100$	20
2. The Size of the Proposed Test under the PRH and PH Models with $n = 100$	21
3. The Size of the Proposed Test under the PRH and PH Models with $n = 200$	22
4. The Size of the Proposed Test under on the PRH Model with $n = 100$ and $n = 200$	22
5. The Size of the Proposed Test under the PH Model with $n = 100$ and $n = 200$	22
6. The Power of the Proposed Test under the PRH Model with $n = 100$ and $n = 200$	23
7. The Power of the Proposed Test under the PH Model with $n = 100$ and $n = 200$	23
8. The Power for Levels of Left-Truncation under the PRH Model with $n = 100$	25
9. The Power for Levels of Left-Truncation under the PH Model with $n = 100$	25
10. The Power for Levels of Left-Truncation under the PRH Model with $n = 200$	26
11. The Power for Levels of Left-Truncation under the PH Model with $n = 200$	26
12. The Power for Types of Major Censoring under the PRH Model with $n = 100$	28
13. The Power for Types of Major Censoring under the PH Model with $n = 100$	28
14. The Power for Types of Major Censoring under the PRH Model with $n = 200$	29
15. The Power for Types of Major Censoring under the PH Model with $n = 200$	29
16. The Size Comparison under the PRH Model.....	36
17. The Size Comparison under the PH Model.....	36
18. The Comparison of Power under the PRH Model with a Small Deviation of β	37
19. The Comparison of Power under the PRH Model with a Large Deviation of β	37

20. The Comparison of Power under the PH Model with a Small Deviation of β	38
21. The Comparison of Power under the PH Model with a Large Deviation of β	38
22. The Comparison of Survival Distributions between Kids and Adults.....	40

ABSTRACT

Survival time is defined as the time duration until a certain event occurs. It is of interest for researchers to estimate the distribution of survival time, and to determine what factors and how these factors may be associated with survival time. Unlike other statistical data, survival data has its own features, which leads to many statistical methods not applicable. In addition to non-normality, censoring and truncation are two typical features. For left-truncated and interval-censored (LTIC) data, Shen (2015) derived a class of two-sample rank-based tests from weighted log-rank tests with weight parameters (ρ, γ) . However, it may be challenging to choose the weight parameters. In this thesis, a new model-based linear rank-type test is proposed to compare survival distributions for LTIC data, which is derived from a proportional reversed hazard (PRH) model. Extensive simulation studies are conducted to examine the performance of the proposed test including the test size and power, and normality of the test statistic. Two scenarios are considered to investigate the robustness of the proposed test: when assumed model is true or false. The proposed test is also compared with an existing method. For illustration, the proposed test is applied to the AIDS Blood Transfusion Data collected by the Centers for Disease Control (CDC) (Kalbfleisch and Lawless, 1989).

I. INTRODUCTION

Introduction to Survival Analysis

Survival analysis is a branch of statistics, which is applied to analyze survival data in many fields, such as medicine, biology, finance, insurance, engineering, and environmental science. Survival time T is the duration of time until an event occurs, for example, the incubation time from HIV infection to the development of AIDS. The following are important concepts in survival analysis.

The probability density function of T is defined as

$$f(t) = \lim_{\Delta t \rightarrow 0^+} \frac{P(t \leq T \leq t + \Delta t)}{\Delta t}, t \geq 0.$$

The cumulative distribution function and survival function are

$$F(t) = P(T \leq t) = \int_0^t f(s)ds, t \geq 0, \text{ and}$$

$$S(t) = P(T > t) = 1 - F(t), t \geq 0,$$

where $S(t)$ provides the probability that an individual survives beyond time t .

The hazard function, which is also called hazard rate function, is another important function used in survival analysis. It is defined as

$$\lambda(t) = \lim_{\Delta t \rightarrow 0^+} \frac{P(t \leq T \leq t + \Delta t | T \geq t)}{\Delta t} = \frac{f(t)}{S(t)},$$

which is instantaneous failure rate or the probability that the event will occur in the next infinitesimal time interval $(t, t + \Delta t)$, given an individual has survived up to time t .

It is of interest for researchers to estimate survival function, hazard function, and to investigate what factors and how these factors may be associated with the survival time.

Truncation and Censoring

Unlike other statistical data, survival data has its own features which makes many statistical methods inapplicable. Firstly, survival time rarely follows a normal distribution. Its distribution is usually skewed. Censoring is one unique feature of survival data. Survival time may not be observed exactly for every individual since some individuals may not have experienced the event in the study. This phenomenon is called censoring, and there are three different censoring types: left, right, and interval censoring. Left censoring occurs if we only know the event of interest occurred before a certain time. If the survival time is only known to be greater than time at the end of a study, right censoring occurs. It's also possible that we only know that the survival time is between two monitoring time points. This type of censoring is called interval censoring.

Truncation is another unique feature of many survival datasets, which may be confused with censoring. Truncation is resulted from sampling bias that only those subjects whose failure times are within a certain time interval can be observed or included in the analysis. Typically, there are left truncation and right truncation. For example, if the survival time of an individual is observable only if it is greater than a certain value, left truncation occurs. If the survival time is measurable only if it is less than a certain value, right truncation occurs. Censoring and/or truncation of survival data require special methods to analyze survival data.

Estimation of Survival Probability

Investigators are interested in estimating survival probabilities or survival function. A survival function can be estimated using parametric methods and nonparametric methods. In a parametric estimation, a parametric model is assumed for the distribution of

survival time, such as Exponential, Weibull, or Gamma distribution. If we already know survival time follows a certain distribution, it's more convenient to use parametric method. However, we usually don't know the actual underlying model of the survival time. If the parametric assumption is false, the parametric estimate may be biased and deviate from the truth. In order to resolve the above issue, nonparametric estimation is developed and widely used in survival analysis. For instance, a well-known maximum likelihood estimator of the survival function for right-censored data is Kaplan–Meier (KM) estimator by Kaplan and Meier (1958). To obtain a KM estimate, the observation time (failure time or censoring time) and a failure time indicator are required, and there is no need to assume a distribution for T .

Comparison of Survival Distributions between Treatment Groups

In survival analysis, a cohort study is a particular form of a longitudinal study. It usually investigates two or more treatment groups of people that have different factor levels. Through a cohort study, researchers try to understand the association between a shared factor of groups and an outcome and how the specific factor influences the outcome. For example, in a clinical trial, patients are randomly divided into two groups. One is called treatment group, in which patients are exposed to some type of intervention, like a drug. The other is called control group, in which patients are not exposed of intervention but receive a placebo instead. A cohort study could provide researchers an insight about the effects of the intervention over time and to the difference in survival time between the drug and placebo groups.

To compare survival distributions between two or more treatment groups in a cohort study, there are some nonparametric tests. The log-rank test (Peto and Peto, 1972)

is one of the most commonly used tests when comparing treatment groups for right censored data. The log-rank statistic compares observed number of events to the expected one over event occurring times for each group. Therefore, log-rank test is also viewed as a time-stratified Cochran–Mantel–Haenszel test (Cochran, 1954; Mantel and Haenszel, 1959). Weights for the difference between observed and expected counts could be added to the log-rank test, resulting in Fleming-Harrington $G^{\rho,\gamma}$ class (Fleming and Harrington, 1991). This class can be extended to the $G^{\rho,\gamma}$ class (Oller and Gómez, 2012) for interval-censored data. Choice of the (ρ, γ) pair can help detect early, middle, or late term differences between survival distributions. Shen (2015) also proposed a class of two-sample rank-based tests from weighted log-rank tests with weight parameters (ρ, γ) .

Cox proportional hazards (PH) model by Cox (1972) could also be used to compare survival distributions between two or more treatment groups, since it is a regression model allowing us to examine the association between the hazard rate and covariates. Unlike the log-rank test, as an alternative method, Cox PH model works for multiple covariates. Cox PH model examines how specified covariates affect hazard rate. The Cox PH model is written by the conditional hazard function denoted by $\lambda(t|\mathbf{Z})$ and it can be estimated as follow:

$$\lambda(t|\mathbf{Z}) = \lambda_0(t) \exp\{\beta_1 Z_1 + \beta_2 Z_2 + \cdots + \beta_p Z_p\},$$

where $\lambda_0(t)$ is an unspecified baseline hazard function, which can be viewed as infinite dimensional parameter. That's why Cox PH model is a semi-parametric regression model. \mathbf{Z} is a vector of p covariates, which many affect survival time, and β is a vector of p regression parameters. If $\mathbf{Z} = \mathbf{0}$, we have $\lambda(t|\mathbf{Z} = \mathbf{0}) = \lambda_0(t)$.

Statistical Computation

In this research, R, a programming language for Statistical Computing, was used to implement the proposed method and an existing method. In R, I simulated data, estimated survival functions, and conducted the proposed test to evaluate the performance of the proposed test. In order to reduce runtime, the high-performance computing cluster LEAP was also used.

Statement of Purpose

Left truncation and interval-censoring (LTIC) introduce complexity in analyzing data from cohort studies. However, existing tests have limitations when applying to data in cohort studies. For example, it is known that log-rank test shows great performance but only for right-censored data. For interval-censored data, when applying the $G^{\rho,\gamma}$ class by Oller and Gómez (2012), it may be difficult to choose proper ρ , γ values within the $G^{\rho,\gamma}$ class. Shen (2015) also has a same problem of the choices of ρ , γ values. Cox PH model is not compatible for situations where survival distributions have a decreasing hazard ratio, because hazard ration is constant under the Cox PH model. Therefore, in this thesis project, a new model-based rank-type test is proposed to compare survival distributions under the proportional reversed hazard (PRH) model for LTIC survival data. Using simulated data, the size and power of proposed test is evaluated in two scenarios. Proportional reversed hazard (PRH) model is applied in first scenario to investigate proposed test when assumed model is true. In second scenario, proportional hazard (PH) model is implemented to evaluate the robustness of proposed test when assumed model is false. Proposed test is also compared with Shen's rank-based test (Shen, 2015) using simulated data. AIDS Blood Transfusion Data collected by the Centers for Disease Control (CDC) (Kalbfleisch and

Lawless 1989) is also used to evaluate proposed test in real world.

II. LITERATURE REVIEW

Left-truncated and Interval-censored (LTIC) Data

Follow-up cohort studies often focus on time to certain event of interest. However, due to the limitations of these studies, investigators usually only know a time range rather than exact event time. Sometimes we notice that event occurs between two time points but don't know exact time, resulting in interval censoring. There are different types of interval censoring. For instance, there are case-I, case-II, case- k , and mixed-case depending on number of observation times for every individual and if they vary among individuals (Sun, 2006; Chen, Sun, and Peace, 2013; Bogaerts, Komarek, and Lesaffre, 2020).

Truncation also occurs in follow-up cohort studies, and it always introduces sampling bias. For example, in studies that allow late entry, subjects in a cohort must survive up to their entry times in order to be included, resulting in left truncated data. The survival probability will be overestimated comparing to that for the general population. The analysis of survival data becomes more complex when the data are subject to left truncation and interval censoring at the same time (Wang, 1991).

Estimation of Survival Function for LTIC Data

To estimate survival distribution, Turnbull (1976) proposed a self-consistency algorithm to obtain a nonparametric maximum likelihood estimator (NPMLE) of the survival function for interval-censored and truncated data. However, Turnbull's characterization of \hat{F} , the maximum likelihood estimator of the cumulative distribution function F , is not applicable in general. Frydman (1994) fixed the problem by modifying Turnbull's characterization of \hat{F} . Assume there are N observations in the data. For the i th observation, let $I_i = (L_i, R_i]$ be the censoring interval and $B_i = (V_i, U_i)$ be the truncation

interval for the survival time T_i . Left truncation corresponds to $U_i = \infty$ for all observations. Two sets are formed based on the end points of censoring intervals: $L = \{L_i, 1 \leq i \leq N\}$ and $R = \{R_i, 1 \leq i \leq N\}$. Combining B_i with sets L and R , two new sets \mathcal{L} and \mathcal{R} are generated by $\mathcal{L}' = L \cup \{U_i, 1 \leq i \leq N\}$, and $\mathcal{R}' = R \cup \{V_i, 1 \leq i \leq N\} \cup \{\infty\}$. Then, Turnbull intervals can be constructed based on the modified sets \mathcal{L}' and \mathcal{R}' .

However, Shen (2020) pointed out that the NPMLE based on the Turnbull intervals above can severely underestimate the survival probabilities for LTIC data, which is caused by inappropriate use of left-truncation time as the left end point of censoring interval for left-censored observations. Instead, he proposed to use the larger one between the left-truncated time and the smallest values of R_i with $\delta_{i1} = 1$, where $\delta_{i1} = 1$ indicates a left-censored observation. Based on this adjustment, Turnbull intervals $(q_j, p_j]$ will be constructed based on the following two set: $\hat{\mathcal{L}} = \{L_i: \delta_{i1} \neq 1; i = 1, \dots, N\} \cup \{\hat{L}_i: \delta_{i1} = 1; i = 1, \dots, N\}$ and \mathcal{R}' , where $\hat{L}_i = \max(V_i, \hat{a}_Q)$ and $\hat{a}_Q = \min_{\delta_{i1}=1} R_i$. Then, the NPMLE \hat{F} can be obtained using Turnbull's self-consistency algorithm (Turnbull, 1976).

Comparison of Survival distributions for LTIC Data

Based on different data types, researchers have proposed different tests to compare survival distributions in follow-up cohort studies. When survival data are complete, Mann–Whitney test (1947) and the Kruskal–Wallis test (1956) test are two commonly used nonparametric methods. The Gehan test (Gehan, 1965) and log-rank test (Peto and Peto, 1972) are applied when data is subject to right censoring. For right-censored data, Pepe and Fleming (1989) also derived a class of tests based on the integrated weighted difference in Kaplan–Meier estimates of survival probabilities. Fleming and Harrington (1991) also introduced the class $G^{\rho, \gamma}$ of weighted log-rank tests, considering for testing equality of

hazard functions against varied hazard ratio. Lagakos et al. (1988) introduced a weighted log-rank test to study randomly truncated data. When truncation distribution is parameterized, Bilker and Wang (1996) generated a semiparametric test. For left-truncated and right censored survival data, Shen (2007 and 2009) derived a weighted Kaplan–Meier test and a class of rank-based tests.

There are also various nonparametric tests proposed relevant to interval-censored data, which are based on score tests statistics, weighted log-rank tests statistics, and linear rank like scores in general. For example, Sun (1996), Fay and Shih (1998), and Zhao and Sun (2004) proposed tests when data is subject to interval censoring. Oller and Gómez (2012) extended the class $G^{\rho,\gamma}$ of weighted log-rank tests that depends on parameters (ρ, γ) to interval-censored data. Recently Broët et al. (2022) proposed a linear rank-type test to investigate interval censored data facing decreasing hazard ratio based on a proportional reversed hazard model, which bypasses the choice of values for ρ, γ in the class $G^{\rho,\gamma}$.

When left-truncation coexists with interval censoring, Shen (2015) derived a class of two-sample rank-based tests from weighted log-rank tests with weight parameters (ρ, γ) . However, choices of proper values for (ρ, γ) pose challenges to investigators because they depend on knowledge about hazard ratio or more often on the observed test results. In addition, when left-truncated and interval-censored (LTIC) data exhibits decreasing hazard ratio, it may be more difficult to choose appropriate values for (ρ, γ) since fewer references are available. The above limitations inspired me to find an alternative method to compare survival distributions when facing complex survival data structure. As mentioned before, Broët et al. (2022) provided a way to bypass the choice of values for ρ, γ . To extend this method to LTIC data, in this thesis project, a new model-

based linear rank-type test is proposed to compare survival distributions. Based on the likelihood function under a proportional reversed hazard (PRH) model, a test statistic is derived to test the null hypothesis that there is no difference in survival among treatment groups against Lehmann-type alternatives.

III. METHODOLOGY

Model

For continuous survival time T , let $f(t)$ be its probability density function, $F(t)$ and $S(t)$ be its cumulative distribution function and survival function, respectively, with $S(t) = 1 - F(t)$. The hazard rate function of T is $\lambda(t) = f(t)/S(t)$. Compared with hazard rate function, the reverse hazard rate function is defined below by Barlow et al. (1963):

$$\eta(t) = \lim_{\Delta t \rightarrow 0^+} \frac{P(t - \Delta t \leq T \leq t | T \leq t)}{\Delta t} = \frac{f(t)}{F(t)}.$$

To compare k treatment groups, assume a proportional reserved hazard model

$$\eta(t|\mathbf{Z}) = \eta_0(t)e^{\boldsymbol{\beta}'\mathbf{Z}},$$

where $\eta_0(t)$ is an unspecified baseline reverse hazard function, \mathbf{Z} is a vector of $(k - 1)$ group indicators, and $\boldsymbol{\beta}$ is a vector of regression parameters. After an exponentiation transformation of the baseline cumulative distribution function, we have: $F(t|\mathbf{Z}) = F_0(t)e^{\boldsymbol{\beta}'\mathbf{Z}}$, where $F_0(t)$ is the baseline cumulative distribution function.

The ratio of hazard rates between group i and group j , $i \neq j$, is represented by:

$$\frac{\lambda(t|\mathbf{Z}_i)}{\lambda(t|\mathbf{Z}_j)} = \frac{e^{\boldsymbol{\beta}'\mathbf{Z}_i} \left(\frac{F_0(t)}{1 - F_0(t)e^{\boldsymbol{\beta}'\mathbf{Z}_i}} \right)}{\frac{F_0(t)}{1 - F_0(t)}} = e^{\boldsymbol{\beta}'(\mathbf{Z}_i - \mathbf{Z}_j)} \left(\frac{\frac{1}{F_0(t)} - 1}{\frac{1}{F_0(t)e^{\boldsymbol{\beta}'\mathbf{Z}_j}} - 1} \right).$$

Because $e^{\boldsymbol{\beta}'\mathbf{Z}}$ is always positive and $F_0(t)$ is converges from 0 to 1 when t goes to infinity.

Therefore, hazard ratio decreases as t increases and tends to unity.

Proposed Test

Assume observed data is $\{V_i, I_i = (L_i, R_i]; i = 1, \dots, n\}$, where V_i is the left truncation time and I_i is the censoring interval for observation i . To test the null hypothesis

$H_0: \boldsymbol{\beta} = \mathbf{0}$ (i.e., $F_0(t) = \dots = F_{k-1}(t) = F(t)$) against the alternative hypothesis $H_1: \boldsymbol{\beta} \neq \mathbf{0}$, the likelihood function based on LTIC data under the proportional reversed hazard model is derived with the adjusted Turnbull intervals $(q_j, p_j]$ by Shen, $j = 1, \dots, m$ and $0 \leq q_1 \leq p_1 < q_2 \leq p_2 \dots \leq p_m = \infty_+$:

$$Lik(\boldsymbol{\beta}, F_0(\cdot)) = \prod_{i=1}^n \sum_{j=1}^m \alpha_{ij} \left[\frac{F_0(p_j)^{e^{\boldsymbol{\beta}' \mathbf{z}_i}} - F_0(q_j)^{e^{\boldsymbol{\beta}' \mathbf{z}_i}}}{1 - F_0(V_i)^{e^{\boldsymbol{\beta}' \mathbf{z}_i}}} \right],$$

where $\alpha_{ij} = 1$ if $(q_j, p_j]$ belongs to I_i , and 0 otherwise.

Then the corresponding log-likelihood is:

$$LLik(\boldsymbol{\beta}, F_0(\cdot)) = \sum_{i=1}^n \log \left[\sum_{j=1}^m \alpha_{ij} \left[F_0(p_j)^{e^{\boldsymbol{\beta}' \mathbf{z}_i}} - F_0(q_j)^{e^{\boldsymbol{\beta}' \mathbf{z}_i}} \right] \right] - \sum_{i=1}^n \log [1 - F_0(V_i)^{e^{\boldsymbol{\beta}' \mathbf{z}_i}}].$$

The score statistic $\mathbf{U}_{(\boldsymbol{\beta}=\mathbf{0}, F_0)}$ can be obtained by taking the first partial derivative of the log-likelihood function with respect to $\boldsymbol{\beta}$ and then evaluated at $\boldsymbol{\beta} = \mathbf{0}$:

$$\begin{aligned} \mathbf{U}_{(\boldsymbol{\beta}=\mathbf{0}, F_0)} &= \sum_{i=1}^n \mathbf{z}_i \left\{ \frac{\sum_{j=1}^m \alpha_{ij} [F_0(p_j) \log(F_0(p_j)) - F_0(q_j) \log(F_0(q_j))]}{\sum_{j=1}^m \alpha_{ij} [F_0(p_j) - F_0(q_j)]} \right. \\ &\quad \left. + \frac{F_0(V_i) \log(F_0(V_i))}{1 - F_0(V_i)} \right\} \\ &= \sum_{i=1}^n \mathbf{z}_i c_i \end{aligned}$$

with $c_i = \frac{\sum_{j=1}^m \alpha_{ij} [F_0(p_j) \log(F_0(p_j)) - F_0(q_j) \log(F_0(q_j))]}{\sum_{j=1}^m \alpha_{ij} [F_0(p_j) - F_0(q_j)]} + \frac{F_0(V_i) \log(F_0(V_i))}{1 - F_0(V_i)}$. Here c_i can be

viewed as a pseudo-ranking function and $\mathbf{U}_{(\boldsymbol{\beta}=\mathbf{0}, F_0)}$ can be considered as a left truncated and interval censored linear rank statistic depending on F_0 (Hájek, Sidák, and Sen, 1999;

Kalbfleisch and Prentice, 2002). And the proposed test could be viewed as a linear rank-type test by $\sum_{i=1}^n \mathbf{z}_i c_i$.

The permutation distribution of \mathbf{U} is then obtained by permuting the labels and re-computing the test statistic for all possible rearranged labels (Oller and Gómez, 2012). The permutation can be computed exactly when the sample size is small. With large sample size, a version of the central Limit theorem for exchangeable random variables can be applied, which provides a normal approximation with expectation $E(\mathbf{U}) = n \bar{c} \bar{\mathbf{z}}$ and variance-covariance matrix $\mathbf{V}_{(\beta=0, F_0)} = \frac{1}{n-1} \{ \sum_{i=1}^n (c_i^2 - n \bar{c}^2) (\sum_{i=1}^n (\mathbf{z}_i \mathbf{z}_i' - \bar{\mathbf{z}} \bar{\mathbf{z}}') \}$, where \bar{c} and $\bar{\mathbf{z}}$ can be calculated as $\bar{c} = \frac{1}{n} \sum_{i=1}^n c_i$ and $\bar{\mathbf{z}} = \frac{1}{n} \sum_{i=1}^n \mathbf{z}_i$. Under H_0 , the test statistic based on the Mahalanobis distance is

$$\mathbf{U}_{(\beta=0, F_0)}' \mathbf{V}_{(\beta=0, \hat{F}_0)}^- \mathbf{U}_{(\beta=0, F_0)},$$

where A^- is a generalized inverse of matrix A. The test statistic asymptotically follows a χ_{k-1}^2 distribution under H_0 .

Since the distribution function F_0 is unknown, we could substitute $F_0(t)$ by $\hat{F}_0(t)$, Shen's modified maximum likelihood estimate (MMLE) (Shen, 2020). Finally, the proposed model-based linear rank-type test statistic is

$$\mathbf{U}_{(\beta=0, \hat{F}_0)}' \mathbf{V}_{(\beta=0, \hat{F}_0)}^- \mathbf{U}_{(\beta=0, \hat{F}_0)},$$

which also asymptotically follows a χ_{k-1}^2 distribution. The null hypothesis will be rejected for large values of the test statistic above.

For $k = 2$, the score statistic becomes a scalar, the proposed test statistic becomes

$$Z = \frac{U_{(\beta=0, \hat{F}_0)}}{\sqrt{V_{(\beta=0, \hat{F}_0)}}},$$

and $Z \sim N(0, 1)$ asymptotically.

IV. SIMULATION STUDIES

Simulation Settings

For the performance of the proposed test, extensive simulation studies were conducted to investigate test size, power, robustness, and comparison to an existing method. Two-sample comparisons ($k = 2$) are considered in this section. The two sample sizes are set to $n_1 = n_2 = n = 100$, or 200. The number of replications for each setting is 1000.

Scenario 1: Under the PRH model

Under the proportional reversed hazard model (PRH model), assume that survival time T_i^* of individual i follows an exponentiated Weibull distribution with CDF $F_z(t) = [1 - \exp\{(-at)^b\}]^{e^{\beta z}}$, where $z = 0$ or 1. That is, for the reference group, $F_0(t) = [1 - \exp\{(-at)^b\}]$ and, for the other group, $F_1(t) = [1 - \exp\{(-at)^b\}]^{e^\beta}$. Assume that the left-truncation time $V_i^* \sim U(0, \theta)$. To obtain untruncated survival times for each group, repeatedly generate random numbers (T_i^*, V_i^*) according to their distributions until n pairs with $T_i^* > V_i^*$ are obtained. To create interval-censoring for T_i^* , first generate number of monitoring times $X_i = 2 + B(4, 0.5)$ based on a binomial distribution. If $X_i = k$, generate k uniform random numbers $U_{ji} \sim i.i.d. U(0, \theta_d)$ ($j = 1, \dots, k$) and define monitoring times $Y_{i1}^* = V_i^* + U_{1i}$, $Y_{i2}^* = U_{2i} + Y_{i1}^*$, $Y_{i3}^* = U_{3i} + Y_{i2}^*$, \dots , $Y_{ik}^* = U_{ki} + Y_{i,k-1}^*$. Then, the censoring interval are formed as follows: $(L_i^*, R_i^*) = (V_i^*, Y_{i1}^*)$ if $V_i^* < T_i^* < Y_{i1}^*$; $(L_i^*, R_i^*) = (Y_{i,j-1}^*, Y_{ij}^*)$ if $V_{i,j-1}^* < T_i^* < Y_{ij}^*$ ($j = 1, \dots, k$); and $(L_i^*, R_i^*) = (Y_{ik}^*, 1000)$ if $T_i^* > Y_{ik}^*$.

To examine whether the proposed test performs well under different settings, we considered three proportions of left truncation and three types of dominating censoring for

the reference group, and five values of β , indicating the difference between the two groups. The obtain these settings, extensive searches were conducted to find corresponding combinations of values of a, b, θ and θ_d . The parameters a, b, θ and θ_d were chosen as follows.

Table 1. Parameter Selection for the PRH Model.

Left Truncation	Major Censoring	a	b	θ	θ_d
25%	Left	1	2	1	1.1
	Interval	1	2	1	0.5
	Right	1	2	1	0.25
55%	Left	2	1	1	1
	Interval	2	1	1	0.35
	Right	2	1	1	0.15
75%	Left	2	1	2	0.8
	Interval	2	1	2	0.35
	Right	2	1	2	0.2

Scenario 2: Under the PH model

When the PRH model is false, a proportional hazard (PH) model is applied. Specifically, let the survival time T_i^* follow an exponential distribution such as for the reference group $F_0(t) = 1 - \exp(-at)$ and for the other group $F_0(t) = 1 - \exp(-at)^{e^\beta}$. To achieve different levels of left-truncation and types of dominating censoring for the reference group, the parameters a , θ , and θ_d are chosen as follows.

Table 2. Parameter Selection for the PH Model.

Left Truncation	Major Censoring	a	θ	θ_d
25%	Left	1	0.5	1.5
	Interval	1	0.5	0.7
	Right	1	0.5	0.35
55%	Left	2	1	1
	Interval	2	1	0.35
	Right	2	1	0.2
	Left	2	2	0.8

75%	Interval	2	2	0.35
	Right	2	2	0.2

For test power under the alternative hypothesis, 0.5 (low power level) and 0.9 (high power level) are used as target values to determine four values of β (two negative and two positive), which determines how different the two groups are in survival distribution. Extensive experiments have been done for each setting. Different values of β has used to find the proper value at positive and negative 0.5 and 0.9 power levels. Based on the results of experiments, the final special value of β is chosen based on the range of observed β for each model. Tables 3 and 4 show details on how a single β value for each power level under each model is selected.

Table 3. Selected β Values for the PRH Model.

Left Truncation	Major Censoring	<i>Positive β for power 0.5</i>	<i>Negative β for power 0.5</i>	<i>Positive β for power 0.9</i>	<i>Negative β for power 0.9</i>
25%	Left	0.65	-1	1.1	-5.1
	Interval	0.55	-0.8	0.95	-3.7
	Right	0.55	-0.9	0.97	-2.9
55%	Left	0.65	-1.2	1.05	-5.1
	Interval	0.55	-0.8	0.85	-3.5
	Right	0.55	-0.8	0.95	-3.5
75%	Left	0.4	-0.5	0.65	-1.2
	Interval	0.34	-0.4	0.6	-0.8
	Right	0.36	-0.35	0.6	-0.85
β		0.5	-0.8	0.8	-3

Table 4. Selected β Values for the PH Model.

Left Truncation	Major Censoring	<i>Positive β for power 0.5</i>	<i>Negative β for power 0.5</i>	<i>Positive β for power 0.9</i>	<i>Negative β for power 0.9</i>
25%	Left	0.35	-0.35	0.63	-0.6
	Interval	0.37	-0.4	0.65	-0.65
	Right	0.4	-0.45	0.7	-0.73

55%	Left	0.35	-0.3	0.65	-0.55
	Interval	0.35	-0.35	0.65	-0.61
	Right	0.37	-0.4	0.65	-0.7
75%	Left	0.3	-0.28	0.5	-0.45
	Interval	0.27	-0.3	0.5	-0.47
	Right	0.3	-0.4	0.55	-0.52
β		0.35	-0.35	0.6	-0.6

Simulation Results

Scenario I: Under the PRH model

The finite sample properties of the proposed test were examined at two sample sizes. For $n = 100$, Table 5 shows that the observed sizes ($\beta = 0$) of the proposed test for different settings, $[0.040 - 0.057]$, are within the random sampling fluctuation of the nominal level (0.05), which indicates the proposed test has a proper size. For the power of the proposed test when $\beta \neq 0$, there is no surprise that the power increases as $|\beta|$ increases. One observable discovery is that at lower level of left truncation (especially at 25%) the proposed test is more powerful than higher level of left truncation, which makes sense because more information is available to investigators at a lower level of left truncation.

Table 5. Power and Size of the Proposed Test for the PRH Model with $n = 100$.

		β				
Left Truncation	Major Censoring	-3	-0.8	0	0.5	0.8
25%	Interval	0.995	0.690	0.051	0.522	0.927
	Left	0.954	0.520	0.057	0.399	0.818
	Right	0.993	0.691	0.046	0.511	0.905
55%	Interval	0.702	0.231	0.040	0.245	0.577
	Left	0.509	0.196	0.052	0.189	0.450
	Right	0.752	0.278	0.052	0.224	0.557

75%	Interval	0.661	0.236	0.048	0.243	0.518
	Left	0.542	0.170	0.040	0.157	0.410
	Right	0.735	0.262	0.043	0.198	0.524

For $n = 200$, Table 6 shows that the observed sizes of the proposed test, $[0.031 - 0.056]$, are still close to the nominal level (0.05). Note that the type I error rate is 0.031 for 55% left truncation and right major censoring, which is lower than expected. It may result from loss of more information when data is subject to heavy left truncation level and right censoring based on special selection of values for a, b, θ and θ_d . Comparing with Table 5, with the increase of the sample size, Table 6 shows that the power increases as expected. Also note that, when sample size increases, the difference of power resulted from different left truncation proportions is negligible at high power level of $|\beta|$.

Table 6. Power and Size of the Proposed Test for the PRH Model with $n = 200$.

Left Truncation	Major Censoring	β				
		-3	-0.8	0	0.5	0.8
25%	Interval	1.000	0.942	0.056	0.851	0.998
	Left	0.999	0.846	0.050	0.708	0.991
	Right	1.000	0.946	0.044	0.821	0.999
55%	Interval	0.954	0.522	0.041	0.425	0.866
	Left	0.827	0.390	0.040	0.329	0.730
	Right	0.959	0.534	0.031	0.412	0.869
75%	Interval	0.952	0.491	0.043	0.388	0.854
	Left	0.859	0.363	0.044	0.302	0.738
	Right	0.954	0.511	0.046	0.398	0.818

Scenario 2: Under the PH model

Table 7 shows that the observed sizes, $[0.045 - 0.059]$, are also within the random sampling fluctuation of the nominal significance level (0.05). It means that the proposed

test also has a proper size even if the model is incorrectly specified. The impact of left-truncation proportion is not as significant as under the PRH model.

Table 7. Power and Size of the Proposed Test for the PH Model with $n = 100$.

Left Truncation	Major Censoring	β				
		-0.6	-0.35	0	0.35	0.6
25%	Interval	0.775	0.365	0.053	0.398	0.824
	Left	0.857	0.403	0.046	0.411	0.837
	Right	0.615	0.248	0.048	0.325	0.732
55%	Interval	0.616	0.267	0.053	0.265	0.654
	Left	0.725	0.337	0.056	0.303	0.665
	Right	0.530	0.214	0.056	0.245	0.566
75%	Interval	0.529	0.261	0.046	0.278	0.632
	Left	0.635	0.278	0.045	0.290	0.618
	Right	0.456	0.202	0.059	0.252	0.582

For a large sample size, Table 8 shows that the observed sizes, ranging from 0.043 to 0.067, are also close the nominal level. Note that 0.063 and 0.067 are slightly higher than the nominal significant level 0.05. It may be due to the PH model applied in this scenario, which is not the true model. With the increase of sample size, there are noticeable power gains compared with small sample size in Table 7. The same trend for power gains is also observed in proportional reserved hazard model.

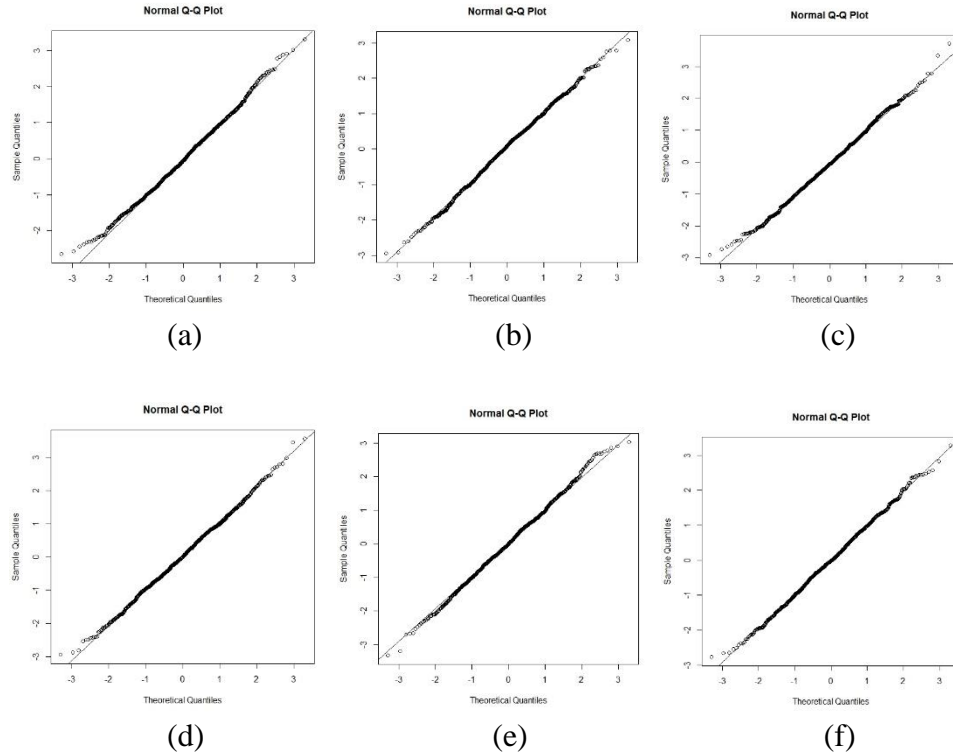
Table 8. Power and Size of the Proposed Test for the PH Model with $n = 200$.

Left Truncation	Major Censoring	β				
		-0.6	-0.35	0	0.35	0.6
25%	Interval	0.970	0.616	0.043	0.692	0.977
	Left	0.986	0.690	0.050	0.678	0.987
	Right	0.864	0.507	0.055	0.535	0.949
55%	Interval	0.891	0.488	0.051	0.532	0.917
	Left	0.955	0.588	0.067	0.532	0.928

	Right	0.814	0.386	0.051	0.481	0.866
75%	Interval	0.851	0.447	0.051	0.515	0.897
	Left	0.913	0.500	0.063	0.529	0.901
	Right	0.711	0.319	0.051	0.420	0.857

Normality Check of the Proposed Test Statistic

With $k = 2$, the test statistic of the proposed test asymptotically follows the standard normal distribution. To investigate the normality based on finite samples, normal QQ-plots were constructed. Figure 1 shows normal QQ-plot and QQ-line for the settings with $\beta = 0$ under the PRH model $n = 100$. It's validated from the QQ-plot that the test statistic follows a normal distribution. Normality check of the test statistic has been conducted for all designs. The results are similar.



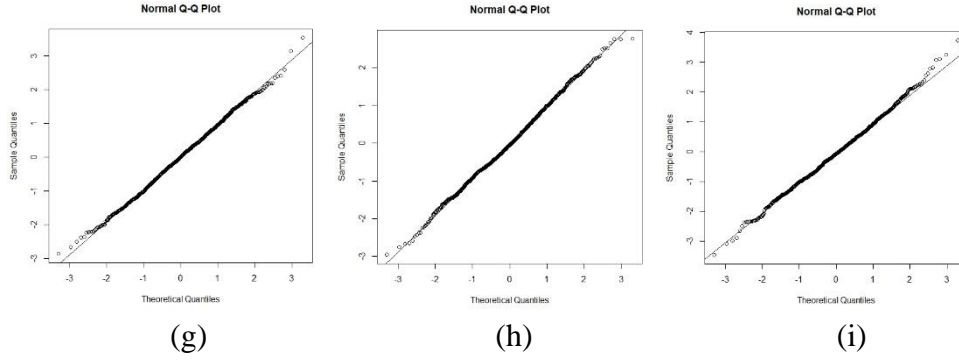


Figure 1. Normality of the Proposed Test Statistic under the PRH Model with $n = 100$:

(a) 25% left truncation with left-censoring; (b) 25% left truncation with interval-censoring; (c) 25% left truncation with right-censoring; (d) 55% left truncation with left-censoring; (e) 55% left truncation with interval-censoring; (f) 55% left truncation with right-censoring; (g) 75% left truncation with left-censoring; (h) 75% left truncation with interval-censoring; (i) 75% left truncation with right-censoring.

Size of the Proposed Test Based on Different Models

To check the robustness of the proposed test, two scenarios are considered. In the first scenario, a PRH model is used, which is the assumed model. In the second scenario, a PH model is used, i.e., the assumed model is false. From the Figures 2 and 3, it's obvious that the proposed test is robust and has a correct size in both scenarios. The observed sizes for all the settings are within the random sampling fluctuation of the nominal significance level for both sample sizes.

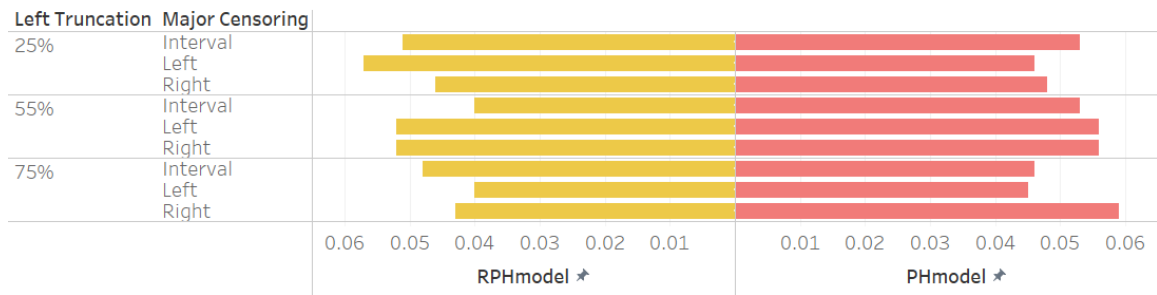


Figure 2. The Size of the Proposed Test under the PRH and PH Models with $n = 100$.

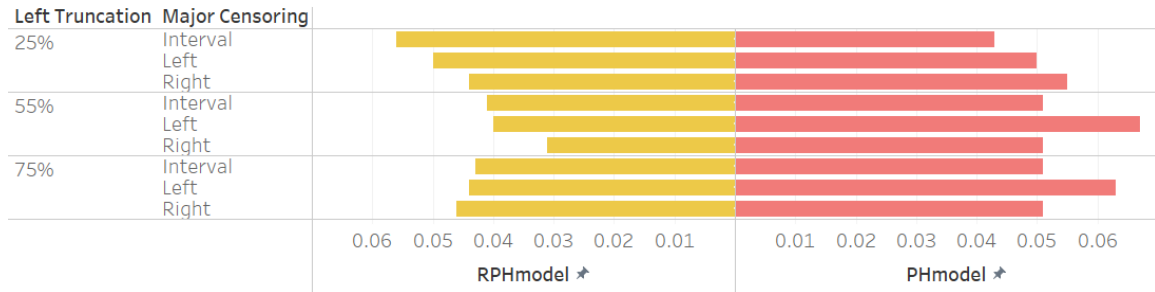


Figure 3. The Size of the Proposed Test under the PRH and PH Models with $n = 200$.

Effect of Sample Size on the Size and Power of the Proposed Test

Considering survival data in the real world, which might be from clinics and hospitals, it's not easy to conduct studies with a large sample size. Therefore, whether the proposed test works well with small sample size is of significance. It's remarkable that the observed type I error rates of all the settings are close to the nominal level for both sample sizes considered. When sample size increases, it's noticeable that the proposed test gains more power for both scenarios. Because the value of test statistic increases due to the decrease in variance, as the sample size gets larger.

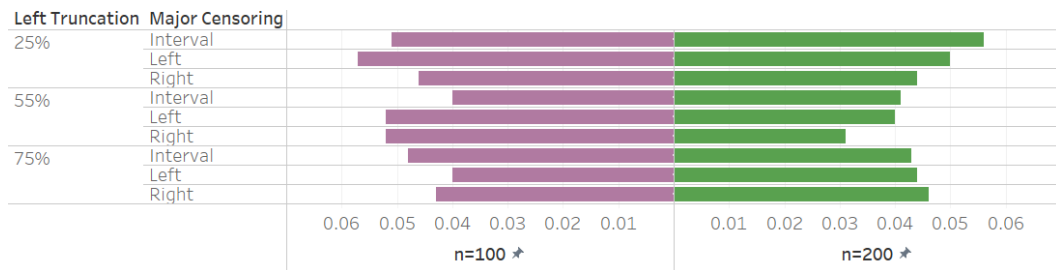


Figure 4. The Size of the Proposed Test under the PRH Model with $n = 100$ and $n = 200$.

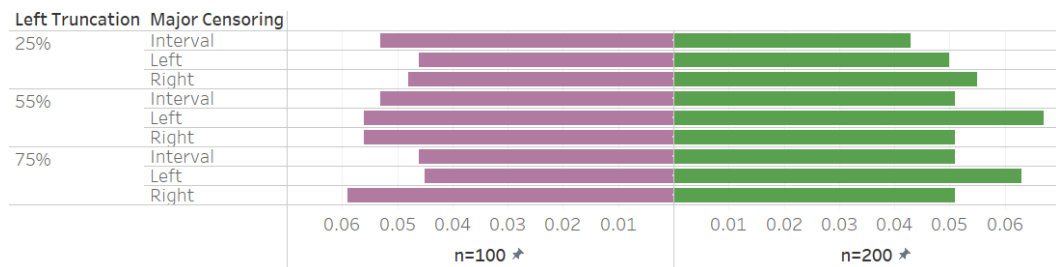


Figure 5. The Size of the Proposed Test under the PH Model with $n = 100$ and $n = 200$.

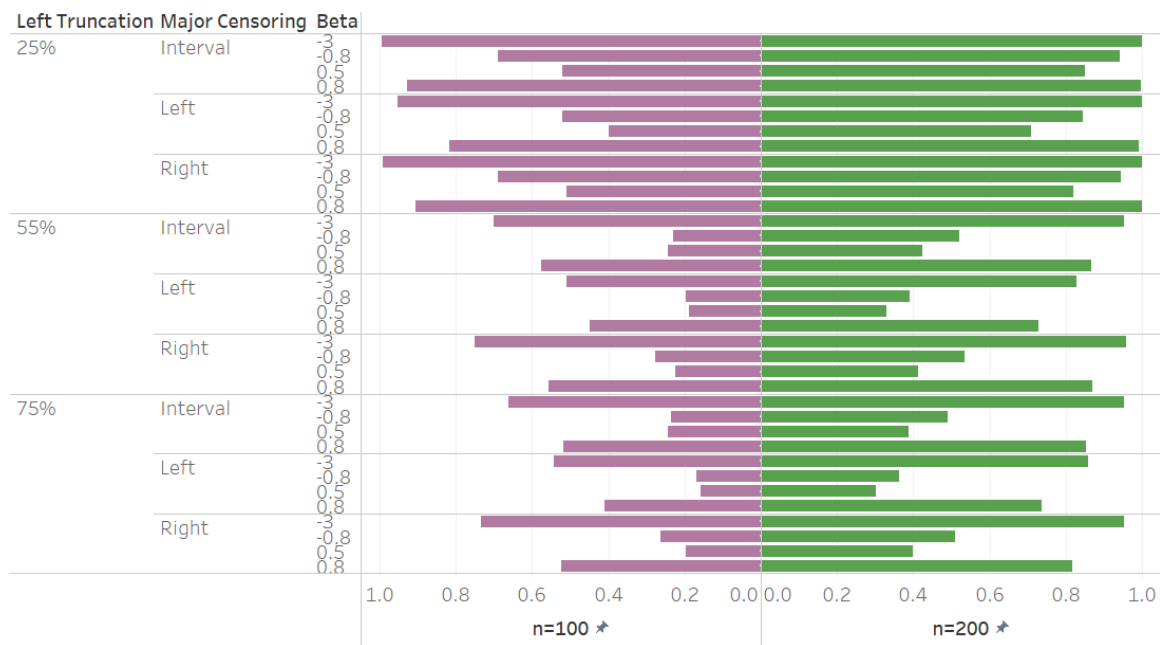


Figure 6. The Power of the Proposed Test under the PRH Model with $n = 100$ and $n = 200$.

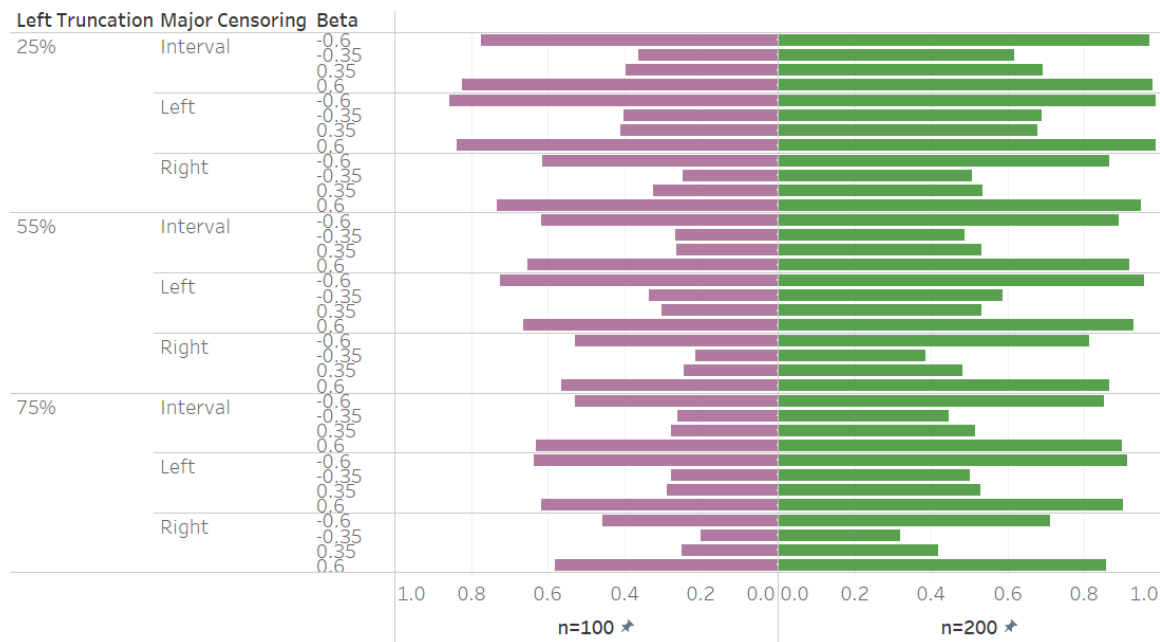


Figure 7. The Power of the Proposed Test under the PH Model with $n = 100$ and $n = 200$.

Effect of Level of Left Truncation on the Power Using the Proposed Test

To study the effect of left truncation, three levels of left truncation are considered in simulated data: 25%, 55%, and 75%. From Figures 8 and 9, it's obvious that for both models, the power for left truncation at 25% is larger than that at 55% and 75%. It may be due to that higher level of left truncation leads to more missing information when testing the difference between groups. For the PRH model, the difference is more obvious compared with that for the PH model. From Figures 10 and 11, it's observable that the difference at the lower power level still exists under both models even with an increased sample size. However, the difference at high power level decreases due to the power has reached its maximum.

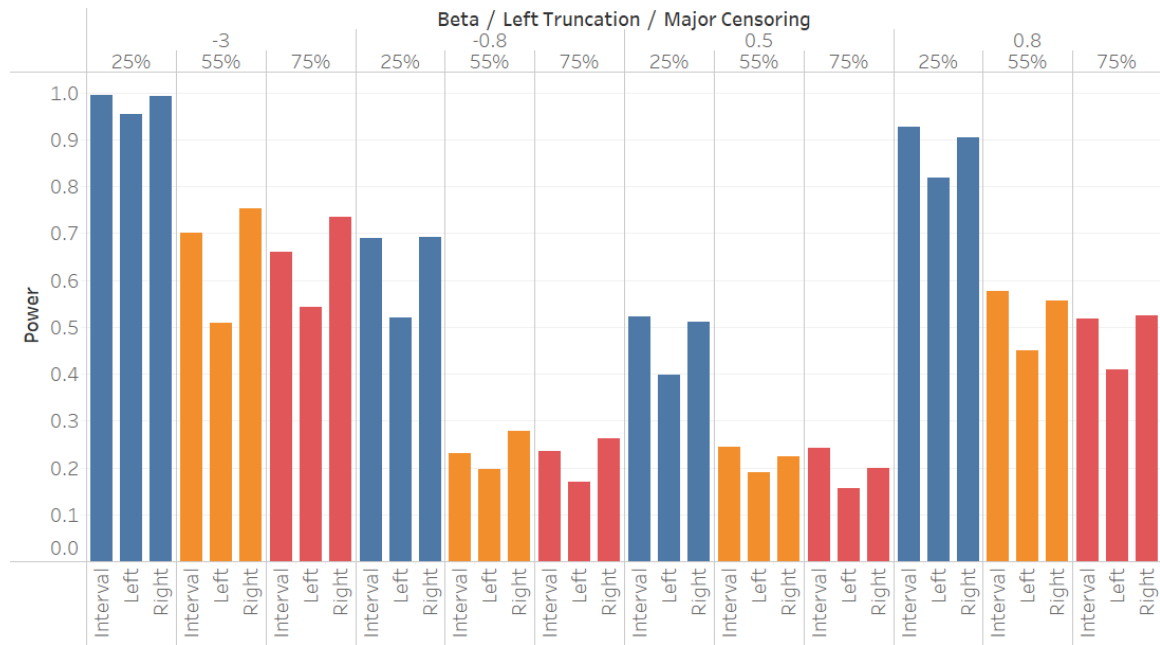


Figure 8. The Power for Levels of Left-Truncation under the PRH Model with $n = 100$.

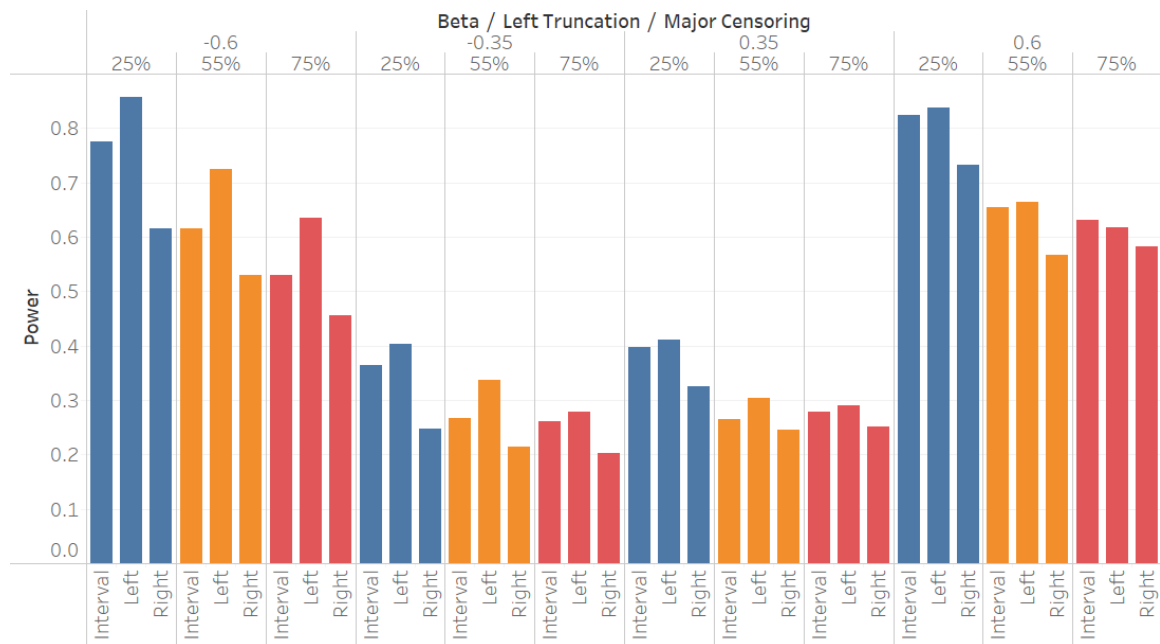


Figure 9. The Power for Levels of Left-Truncation under the PH Model with $n = 100$.

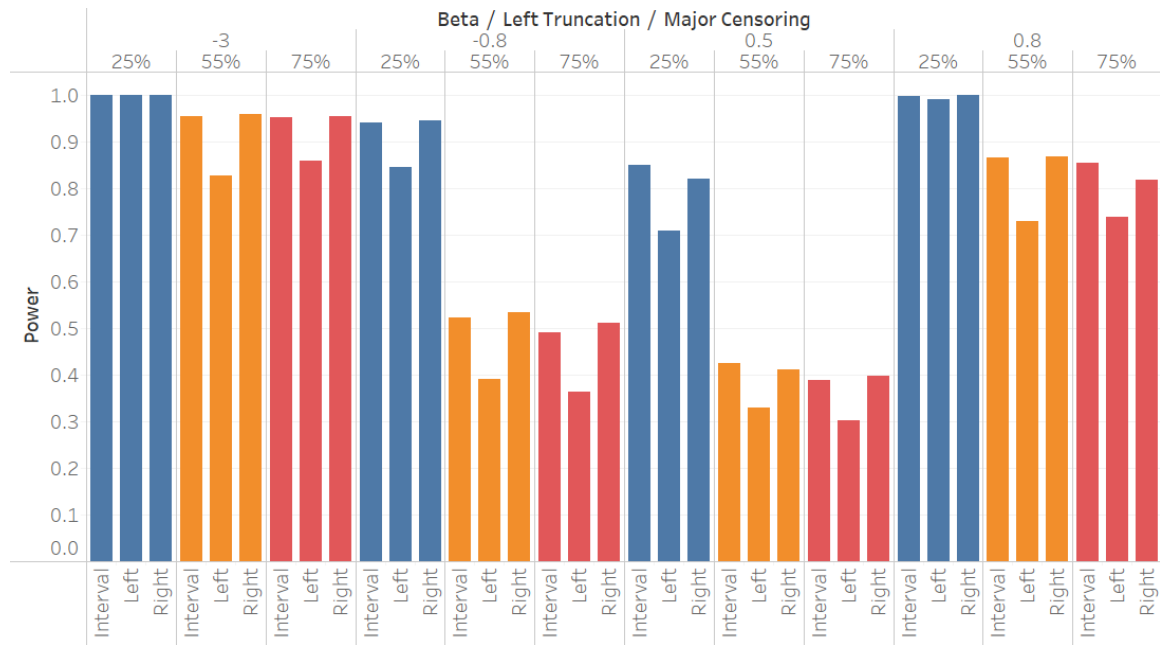


Figure 10. The Power for Levels of Left-Truncation under the PRH Model with $n = 200$.

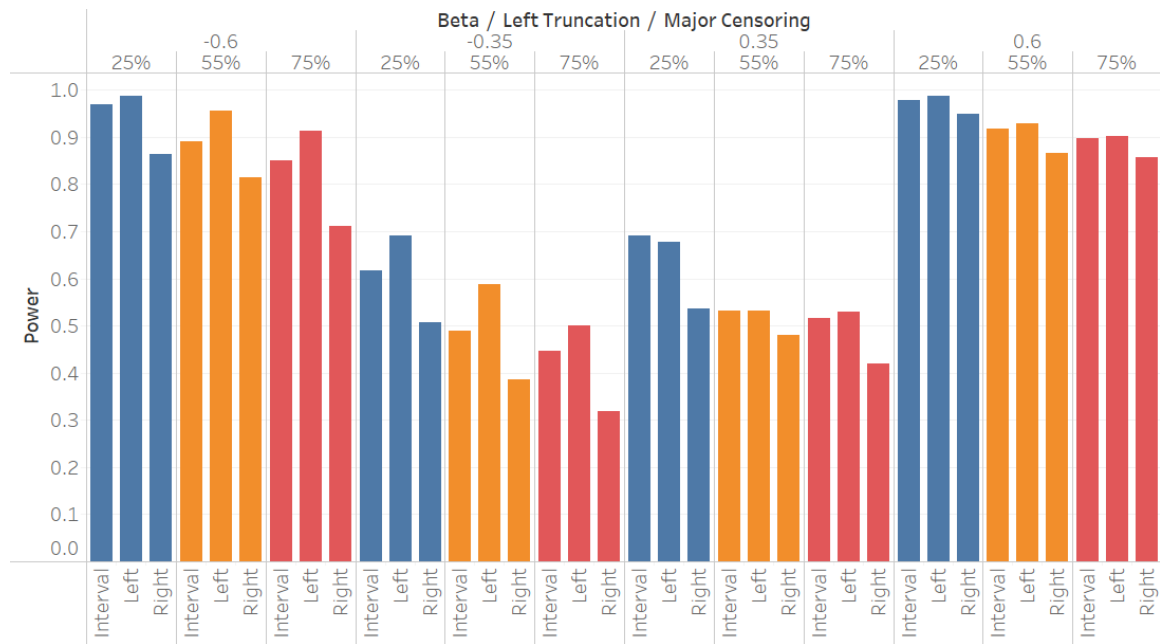


Figure 11. The Power for Levels of Left-Truncation under the PH Model with $n = 200$.

Effect of Type of Major Censoring on the Power Using the Proposed Test

To study the influence of censoring, three types of major or dominating censoring are considered in data generation: left, interval and right. If there are more than half observations are subject to left censoring, the major censoring is denoted as left. Interval censoring and right censoring are also defined in the same way. From Figures 12 and 13, there is a significant difference between the PRH and the PH models. Majorly left censored data show the least power for the PRH model, while majorly right censored data show the least power for the PH model. Based on Figures 14 and 15, the phenomenon is still observable when sample size is increased.

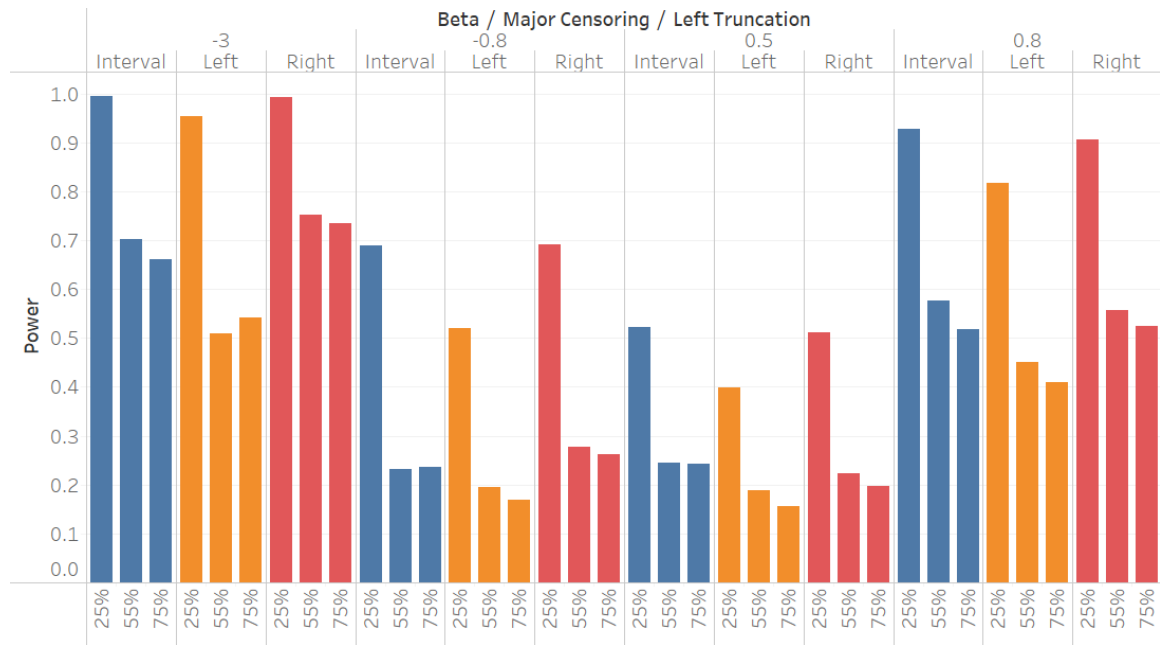


Figure 12. The Power for Types of Major Censoring under the PRH Model with $n = 100$.

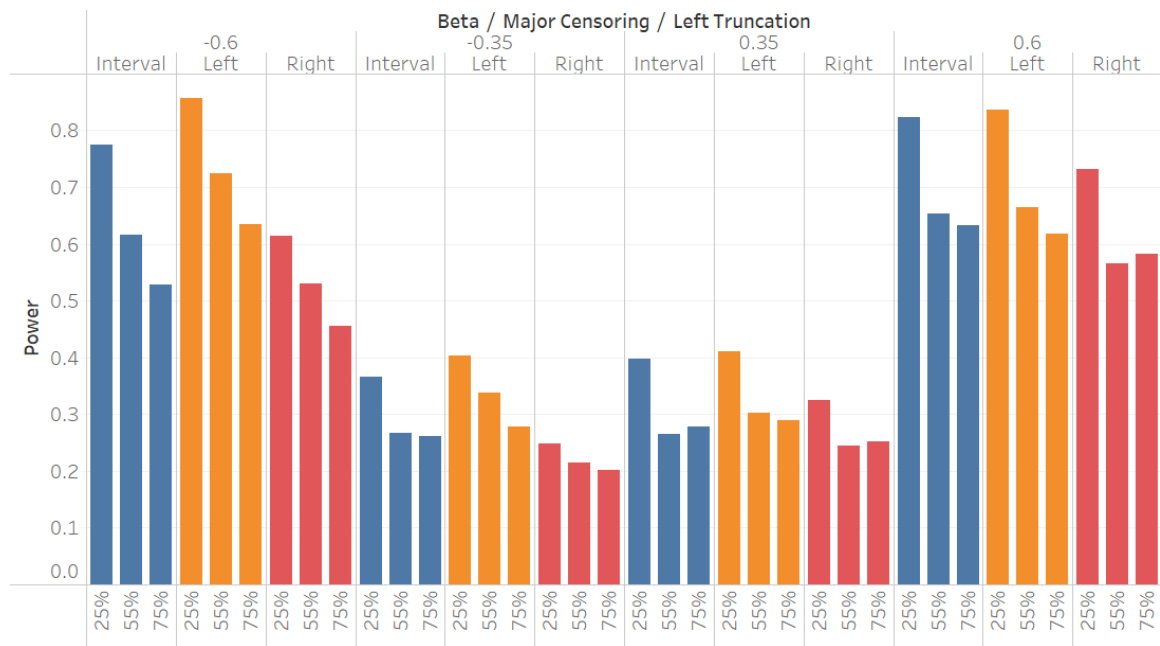


Figure 13. The Power for Types of Major Censoring under the PH Model with $n = 100$.

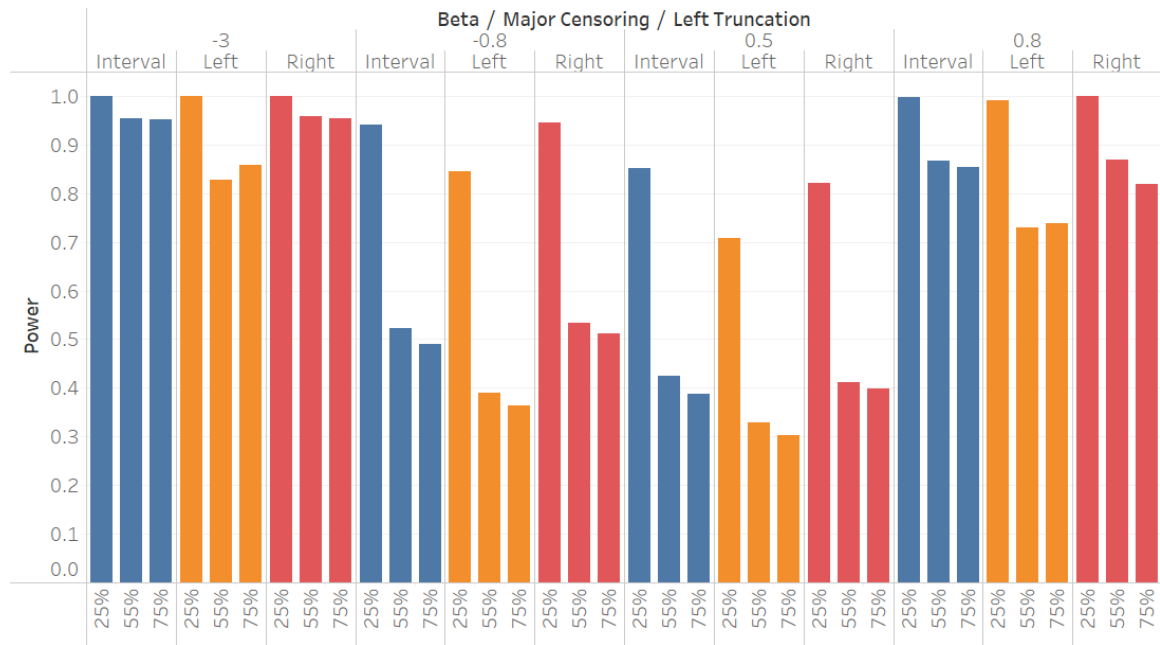


Figure 14. The Power for Types of Major Censoring under the PRH Model with $n = 200$.

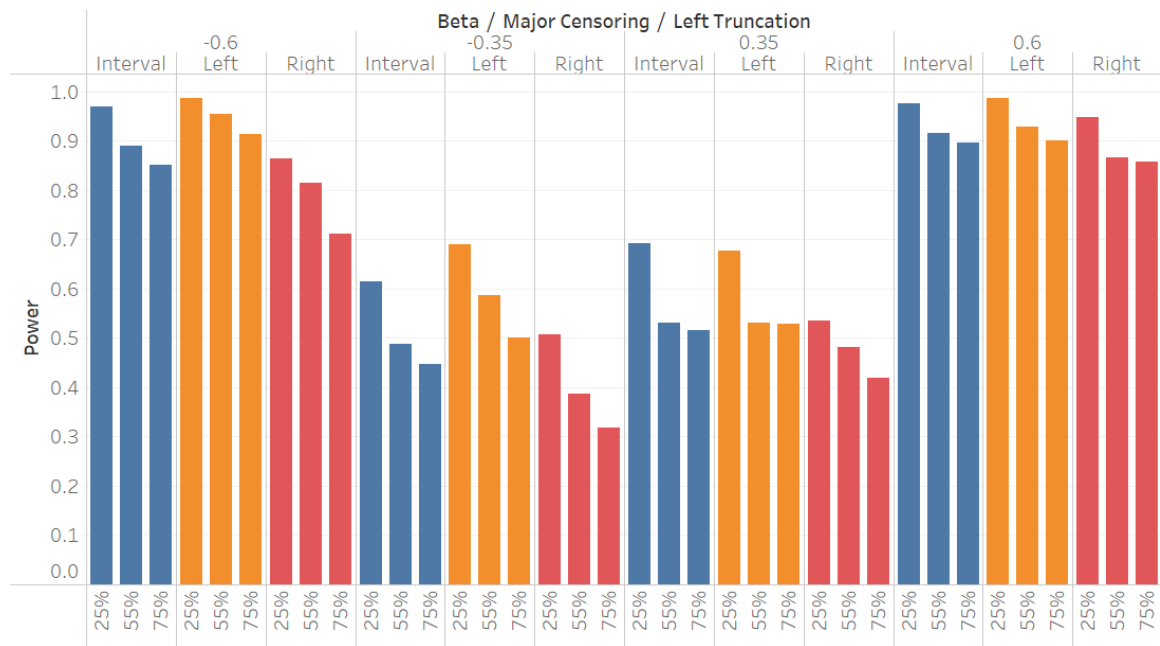


Figure 15. The Power for Types of Major Censoring under the PH Model with $n = 200$.

Comparison to an Existing Test

In order to further examine the performance of the proposed test, the proposed test is compared with a nonparametric rank-based test by Shen (2015). Based on the same simulated left-truncated and interval-censored datasets, the size and power of the proposed test are obtained in the two scenarios considered. For the weight $\widehat{W}(t) = [\widehat{S}(t-)]^r [1 - \widehat{S}(t-)]^s \widehat{W}_L(t)$ of Shen's test, I used various weight parameters: $(r, s) = (0, 0), (0.5, 0.5), (0, 1), (1, 0)$ as Shen used in his study, where $(r, s) = (0, 0)$ corresponds to the weight function of the extension of log-rank test. I used $B = 500$ for the number of bootstrap samples used in Shen's test.

Firstly, as shown in Tables 9 to 12, the observed type I error rates of all the settings are within the random sampling fluctuation of the nominal significance level for proposed tests. More details can be found in Figures 16 and 17: it's clear from the butterfly figures that the observed type I error rates of Shen's test is much lower than the nominal level 0.05, which may cause underestimation of the difference between treatment groups. It means that Shen's test rejects the null hypothesis less often than it should do.

The power is also investigated. For Shen's test, only the results from two selections of weight parameters, $(r, s) = (1, 0)$ and $(0.5, 0.5)$ are presented (denoted Shen(1,0) and Shen(0.5, 0.5)), which provide top 2 performance in power. From Figures 18, 19, 20 and 21, it is obvious that the proposed test performs better than Shen's tests under the PRH model for both small and large deviation of β . While Shen(1, 0) gains slightly more power than the proposed test with the PH model for both small and large deviation of β . Both tests show the same power trend when sample size increases. There is a remarkable power gain, which has also been discussed in the previous section. Results also show that power

decreases for Shen's test as the percentage of left truncation increases. In general, both tests product consistent results in two scenarios.

Table 9. The Comparisons of Size and Power under the PRH Model with $n = 100$.

		β													
		-3			-0.8			0		0.5			0.8		
Left Truncation	Major Censoring	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Shen (0.5,0.5)
25%	Interval	0.995	0.936	0.304	0.690	0.562	0.049	0.051	0.011	0.011	0.522	0.335	0.042	0.927	0.649
25%	Left	0.954	0.842	0.157	0.520	0.360	0.038	0.057	0.014	0.014	0.399	0.256	0.025	0.818	0.582
25%	Right	0.993	0.697	0.071	0.691	0.209	0.031	0.046	0.021	0.021	0.511	0.105	0.025	0.905	0.206
55%	Interval	0.702	0.638	0.090	0.231	0.223	0.017	0.040	0.020	0.020	0.245	0.162	0.024	0.577	0.391
55%	Left	0.509	0.489	0.139	0.196	0.203	0.072	0.052	0.020	0.020	0.189	0.202	0.046	0.450	0.463
55%	Right	0.752	0.321	0.044	0.278	0.093	0.022	0.052	0.016	0.016	0.224	0.059	0.018	0.557	0.127
75%	Interval	0.952	0.931	0.289	0.491	0.510	0.053	0.043	0.028	0.388	0.397	0.021	0.854	0.781	0.044
75%	Left	0.859	0.850	0.576	0.363	0.418	0.171	0.044	0.038	0.302	0.368	0.143	0.738	0.774	0.259
75%	Right	0.954	0.878	0.162	0.511	0.396	0.088	0.046	0.036	0.398	0.216	0.051	0.818	0.465	0.055

Table 10. The Comparisons of Size and Power under the PRH Model with $n = 200$.

		β													
		-3			-0.8			0		0.5			0.8		
Left Truncation	Major Censoring	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Shen (0.5,0.5)
25%	Interval	1.000	0.983	0.677	0.942	0.874	0.112	0.056	0.016	0.851	0.630	0.067	0.998	0.949	0.177
25%	Left	0.999	0.976	0.679	0.846	0.745	0.184	0.050	0.019	0.708	0.597	0.110	0.991	0.934	0.288
25%	Right	1.000	0.905	0.129	0.946	0.440	0.027	0.044	0.017	0.821	0.200	0.037	0.999	0.370	0.051
55%	Interval	0.954	0.902	0.199	0.522	0.469	0.050	0.041	0.009	0.425	0.321	0.032	0.866	0.712	0.065
55%	Left	0.827	0.799	0.545	0.390	0.384	0.168	0.040	0.022	0.329	0.355	0.128	0.730	0.741	0.313
55%	Right	0.959	0.630	0.103	0.534	0.225	0.043	0.031	0.020	0.412	0.126	0.039	0.869	0.254	0.059
75%	Interval	0.952	0.931	0.289	0.491	0.510	0.053	0.043	0.028	0.388	0.397	0.021	0.854	0.781	0.044
75%	Left	0.859	0.850	0.576	0.363	0.418	0.171	0.044	0.038	0.302	0.368	0.143	0.738	0.774	0.259
75%	Right	0.954	0.878	0.162	0.511	0.396	0.088	0.046	0.036	0.398	0.216	0.051	0.818	0.465	0.055

Table 11. The Comparisons of Size and Power under the PH Model with $n = 100$.

		β													
		-0.6			-0.35			0		0.35			0.6		
Left Truncation	Major Censoring	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Shen (0.5,0.5)
25%	Interval	0.775	0.593	0.225	0.365	0.307	0.110	0.053	0.023	0.023	0.398	0.436	0.179	0.824	0.859
25%	Left	0.857	0.904	0.679	0.403	0.495	0.362	0.046	0.028	0.028	0.411	0.399	0.339	0.837	0.730
25%	Right	0.615	0.274	0.135	0.248	0.123	0.072	0.048	0.017	0.017	0.325	0.151	0.068	0.732	0.448
55%	Interval	0.616	0.513	0.115	0.267	0.238	0.063	0.053	0.015	0.015	0.265	0.381	0.077	0.654	0.763
55%	Left	0.725	0.769	0.590	0.337	0.400	0.248	0.056	0.029	0.029	0.303	0.270	0.129	0.665	0.459
55%	Right	0.530	0.275	0.110	0.214	0.117	0.054	0.056	0.013	0.013	0.245	0.221	0.029	0.566	0.557
75%	Interval	0.529	0.601	0.123	0.261	0.380	0.120	0.046	0.036	0.036	0.278	0.430	0.261	0.632	0.786
75%	Left	0.635	0.764	0.417	0.278	0.405	0.199	0.045	0.038	0.038	0.290	0.350	0.284	0.618	0.619
75%	Right	0.456	0.364	0.169	0.202	0.221	0.147	0.059	0.048	0.048	0.252	0.351	0.226	0.582	0.718

Table 12. The Comparisons of Size and Power under the PH Model with $n = 200$.

		β													
		-0.6			-0.35			0		0.35			0.6		
		Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Our Test	Shen (1,0)	Shen (0.5,0.5)	Our Test	Shen (1,0)	Shen (0.5,0.5)
Left Truncation	Major Censoring														
25%	Interval	0.970	0.829	0.365	0.616	0.534	0.198	0.043	0.027	0.027	0.692	0.746	0.347	0.977	0.991
25%	Left	0.986	0.998	0.926	0.690	0.828	0.696	0.050	0.024	0.024	0.678	0.729	0.773	0.987	0.977
25%	Right	0.864	0.340	0.175	0.507	0.188	0.081	0.055	0.021	0.021	0.535	0.290	0.117	0.949	0.651
55%	Interval	0.891	0.801	0.235	0.488	0.541	0.104	0.051	0.013	0.013	0.532	0.728	0.223	0.917	0.979
55%	Left	0.955	0.977	0.942	0.588	0.713	0.707	0.067	0.012	0.012	0.532	0.588	0.593	0.928	0.873
55%	Right	0.814	0.482	0.190	0.386	0.273	0.105	0.051	0.019	0.019	0.481	0.516	0.094	0.866	0.924
75%	Interval	0.851	0.932	0.185	0.447	0.682	0.106	0.051	0.024	0.024	0.515	0.729	0.356	0.897	0.978
75%	Left	0.913	0.974	0.867	0.500	0.679	0.597	0.063	0.019	0.019	0.529	0.617	0.704	0.901	0.933
75%	Right	0.711	0.646	0.111	0.319	0.423	0.081	0.051	0.044	0.044	0.420	0.684	0.263	0.857	0.970

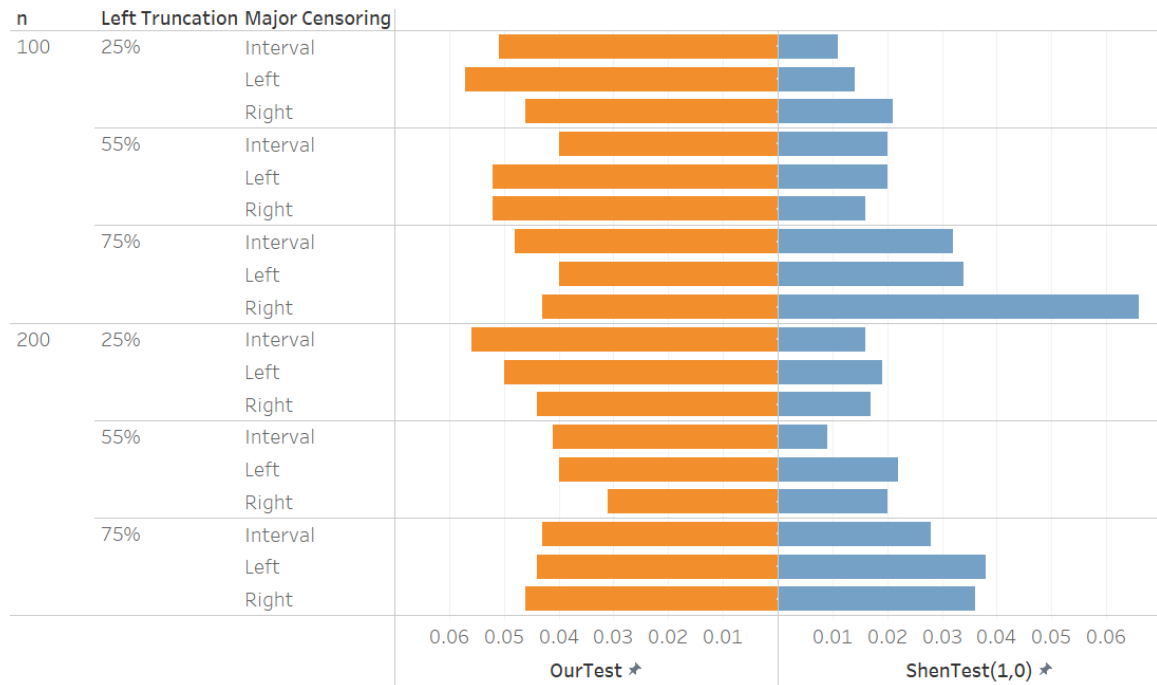


Figure 16. The Size Comparison under the PRH Model.

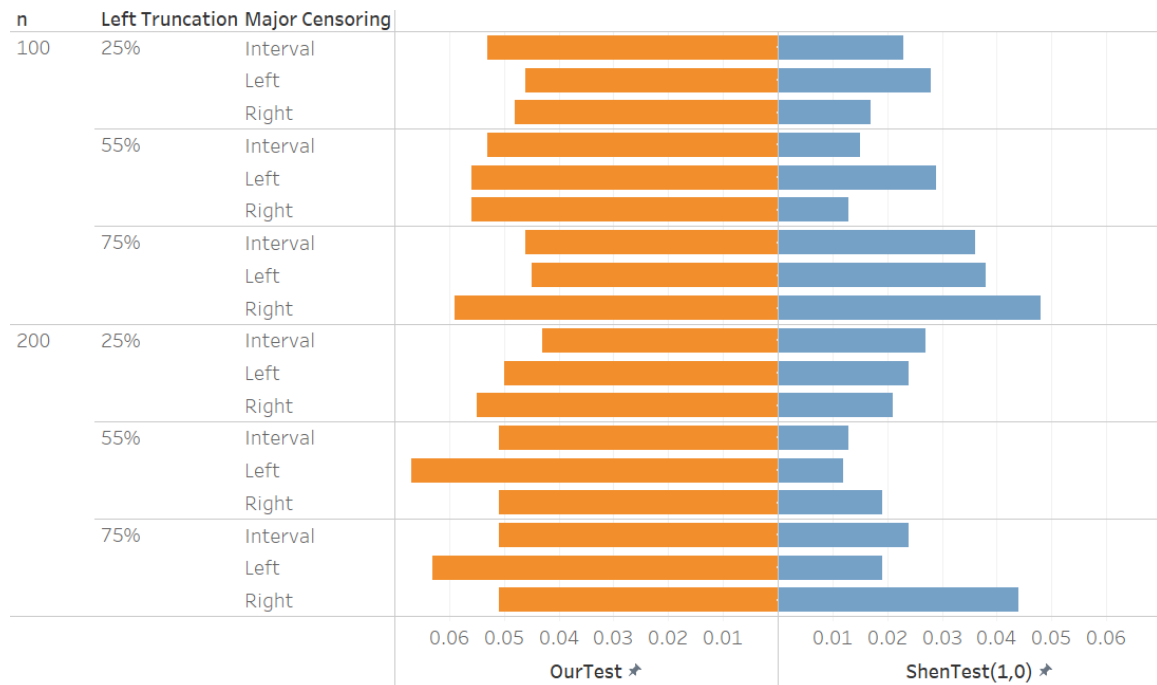


Figure 17. The Size Comparison under the PH Model.

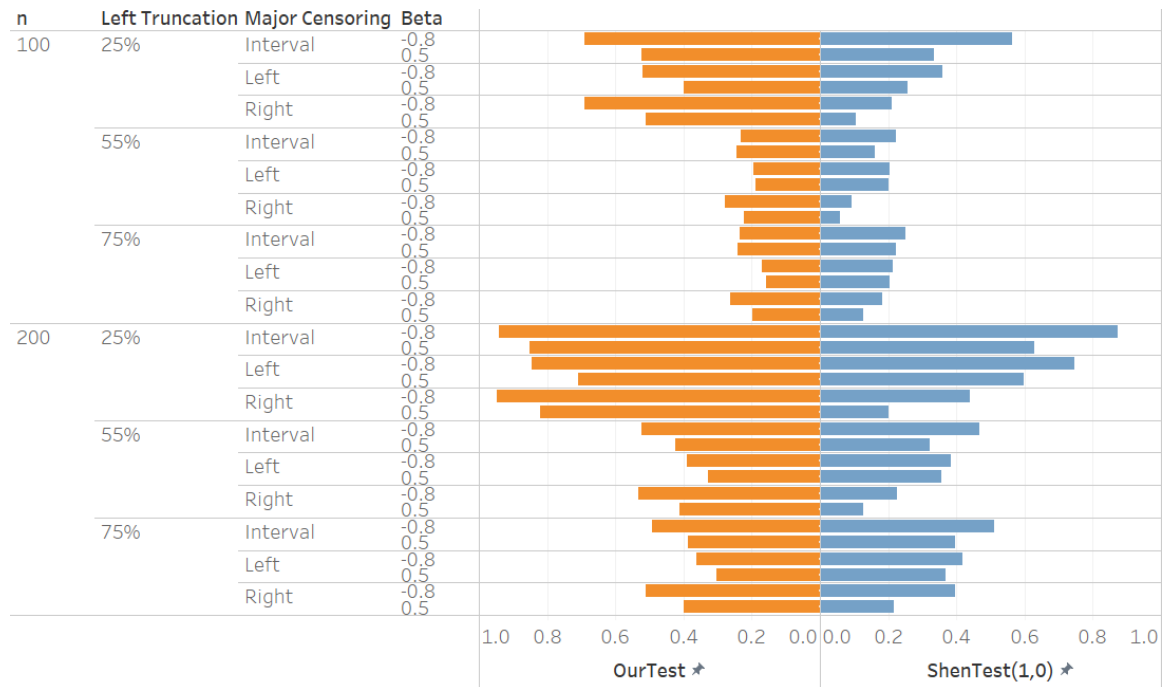


Figure 18. The Comparison of Power under the PRH Model with a Small Deviation of β .

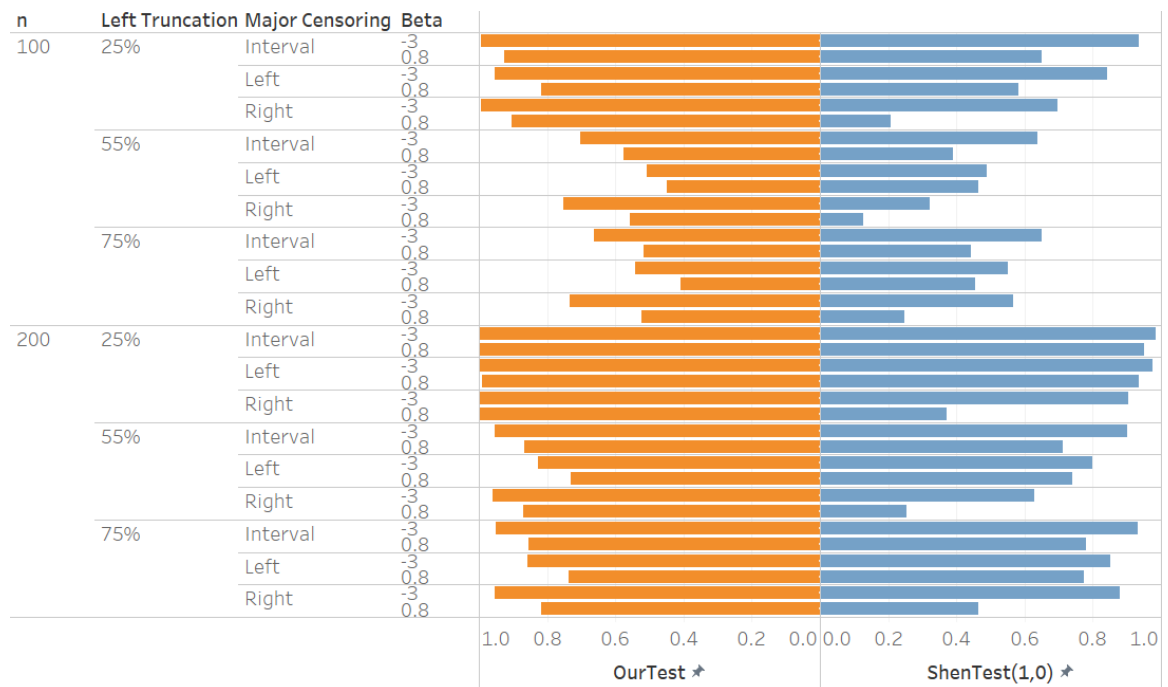


Figure 19. The Comparison of Power under the PRH Model with a Large Deviation of β .

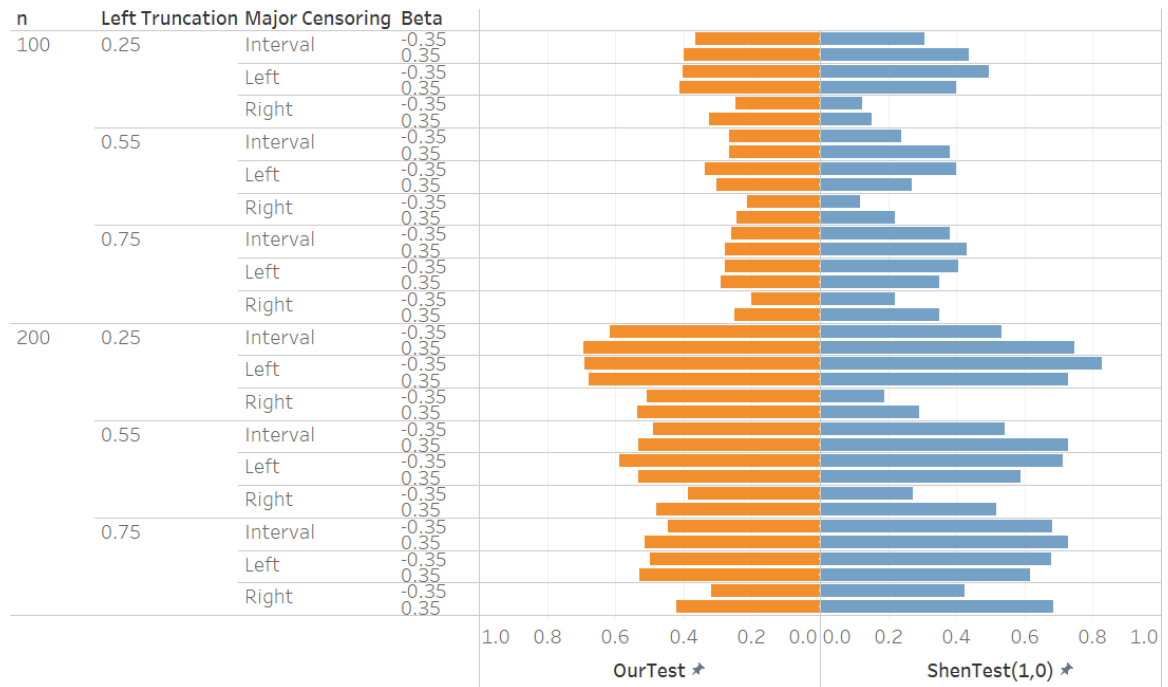


Figure 20. The Comparison of Power under the PH Model with a Small Deviation of β .

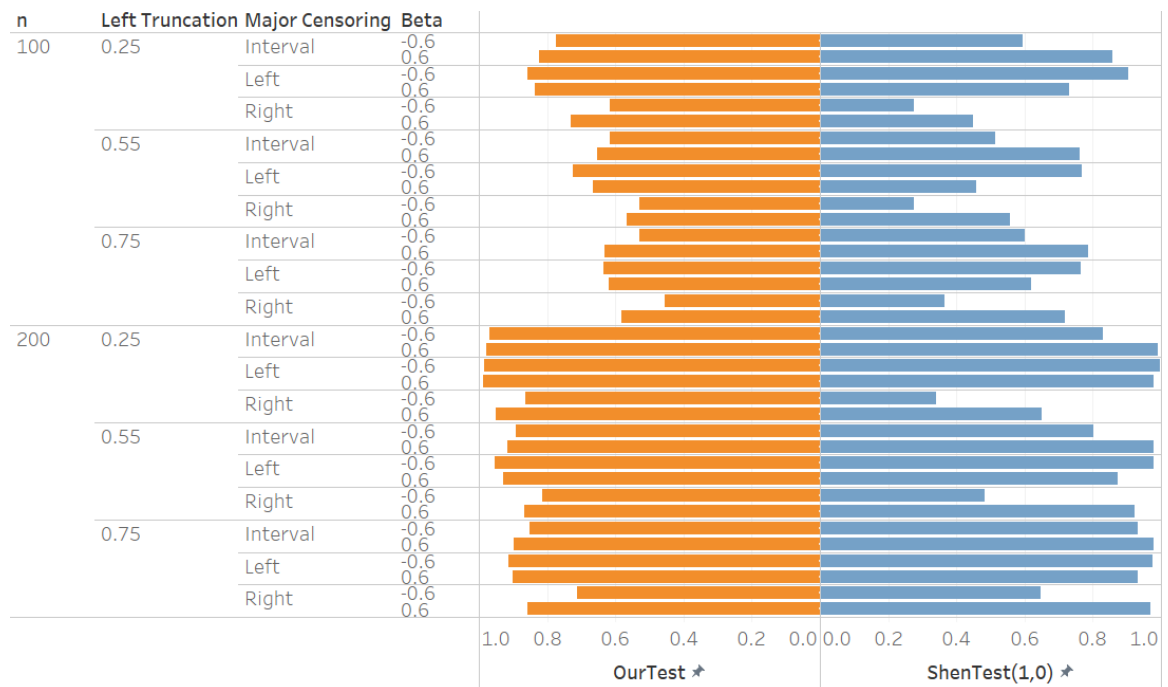


Figure 21. The Comparison of Power under the PH Model with a Large Deviation of β .

V. APPLICATION TO A SEMI-REAL DATA

In this section, the proposed test is applied to the AIDS Blood Transfusion Data (Kalbfleisch and Lawless, 1989) collected by the Centers for Disease Control (CDC) for illustration. The data contains all transfusion-associated AIDS cases in which the development of AIDS occurred prior to the end of the study, which was 06/30/1991. For the data, subjects who had developed AIDS prior to 07/01/1982 were excluded, which indicates that the data is subjected to left truncation. In order to investigate if age is associated to the HIV-AIDS incubation time, two subgroups are compared based on age: subjects whose age is between 0 and 5 belong to kids' group; and subjects whose age is greater than 5 but less than and equal to 50 belong to adults' group. The null hypothesis is no difference in survival distributions between two treatment groups. Because interval censoring was not present in the original data, censoring intervals are produced in the same way as discussed in the simulation section. A random number $X_i = 2 + B(m_c, 0.5)$ is generated. To make sure that censoring intervals are more compatible with the scale of the original data, we used $m_c = 16, 22,$ and 34 with $\theta_d = \frac{2}{3}, \frac{1}{2},$ and $\frac{1}{3},$ respectively. Figure 22 shows that it looks like there is a difference of survival distributions between kids and adults. In order to examine if this difference is statistically different, the proposed test is conducted. The proposed test is also compared with Shen's method using this semi-real data. The results show that both Shen(1,0) and the proposed test rejected the null hypothesis with all three settings for this semi-real LTIC data with p -values less than the nominal significant level 0.05. On the other hand, Shen(0,0), Shen(0,1) and Shen(0.5,05) sometimes failed to reject null hypothesis. This result is consistent with results from the simulation section, which indicates that Shen(1,0) is more powerful with the simulation data and this

semi-real data.

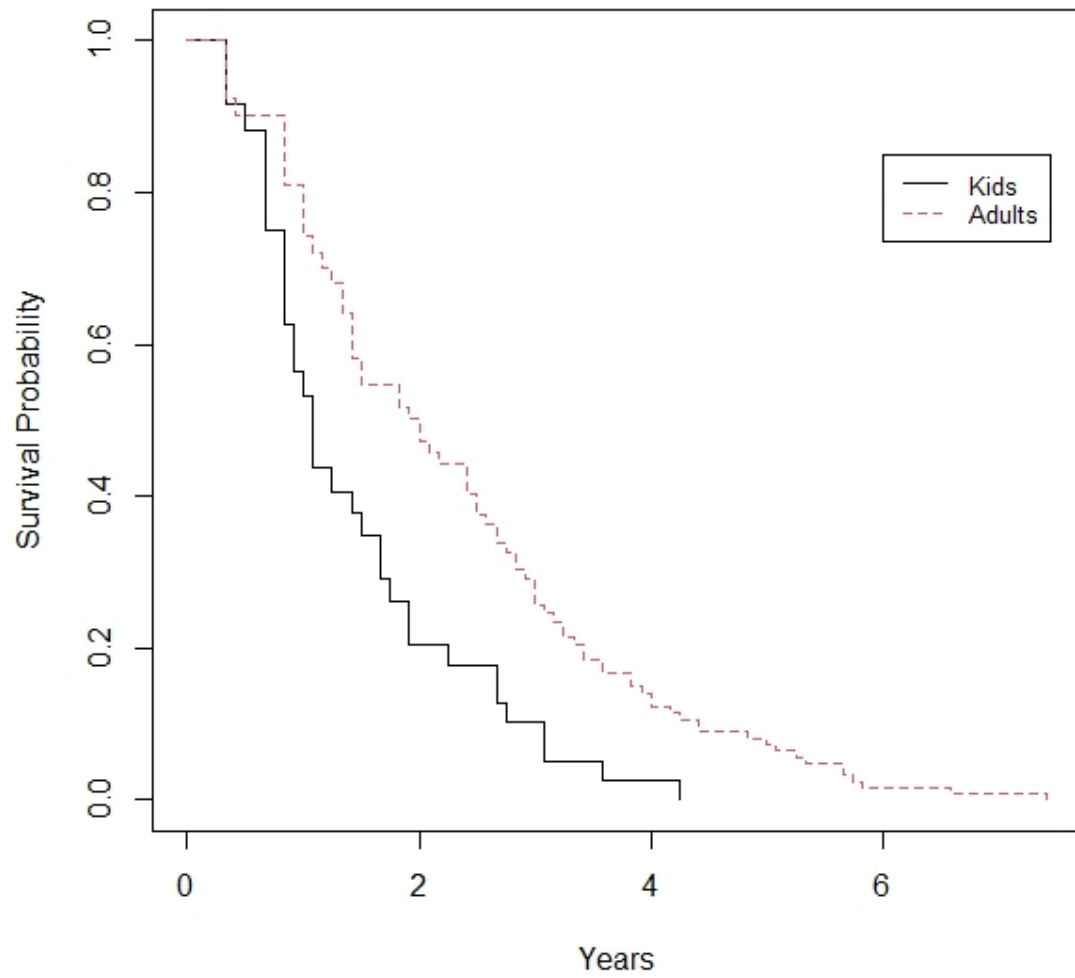


Figure 22. The Comparison of Survival Distributions between Kids and Adults.

Table 13. Results for the Analysis of the AIDS Blood Transfusion Data.

Mc	θ_d	Our Test Statistic	Our p -value	Shen(0,0) Test Statistic	Shen(0,0) p -value	Shen(0,1) Test Statistic	Shen(0,1) p -value	Shen(1,0) Test Statistic	Shen(1,0) p -value	Shen(0.5,0.5) Test Statistic	Shen(0.5,0.5) p -value
18	2/3	1.9549	0.0253	1.3342	0.0911	1.5908	0.0558	2.8081	0.0025	1.7508	0.0400
24	1/2	1.7589	0.0393	4.0138	0.0000	2.2536	0.0121	3.0930	0.0010	2.3252	0.0100
36	1/3	1.8357	0.0332	0.8937	0.1857	0.1050	0.4582	2.5527	0.0053	1.0810	0.1399

VI. CONCLUSION

To compare two or more survival distributions with left-truncated and interval-censored data, a new model-based linear rank-type test is proposed, which is derived from a proportional reversed hazard model. Under the null hypothesis, the proposed test statistic is asymptotically distributed as χ^2_{k-1} , where k is the number of groups. Two scenarios are considered in the simulation studies to evaluate performance and robustness of the proposed test: the underlying distributions follow a proportional reversed hazards (PRH) model or a proportional hazards (PH) model. The size and power of the proposed test are investigated in both scenarios. The simulation results show that the observed type I error rates in all settings considered are within a random sampling fluctuation of the nominal significance level 0.05, which is the probability of rejecting null hypothesis when there is no difference between groups. It means the proposed test is robust with different distributions. Besides, there is an observable power gain when sample size increases because of the increase of the value of the proposed test due to the decrease in its variance. Also, proportion of left truncation and type of interval censoring also play important roles. The results indicate heavier left truncation and left, or right censoring reduce the power of the proposed test. Comparing to Shen's test, the proposed test shows similar performance and bypass the choice of weight parameters. Unlike Shen's test, the proposed test does not need bootstrapping to estimate the variance of the test statistic. Thus, it is easier to implement and computationally inexpensive. For illustration, the proposed test is applied to a set of semi-real data based on the AIDS Blood Transfusion Data collected by the Centers for Disease Control (CDC).

VII. DISCUSSION AND FUTURE RESEARCH

It is noticeable that the proportion of left truncation and interval censoring have an influence on the power of proposed test. Also, when implementing the proposed test with semi-real data, the compatibility between simulated intervals and real data's scale is important. If the simulated intervals are too large or small, they may introduce a big increase in right or left censoring. Then we will lose some important information, leading to incorrect test results for hypothesis. Therefore, investigators should study the survival data first before choosing proper values for parameters in the proposed test. Survival data is complex with its own features, case-by-case analysis is necessary when applying the proposed test.

For future studies, it's possible to achieve the following:

- The proposed test assumes the PRH model. Model checking techniques are needed when analyzing real data.
- For the robust of the proposed test, only a PH model is considered in the simulation studies. Other models should be investigated when PRH model is not true.
- Due to the deviation of the size estimates of the proposed test for two settings, further simulation studies can be conducted, such as using 0.01 level of significance.
- Create an R package by implementing the proposed test for analyzing LTIC data under a PRH model.

APPENDIX SECTION

R code example for size test using proportional reversed hazard model with

sample size 100 in statistical tests section:

```
set.seed(1)
#Sys.time()
# Function to simply compute a Riemann-Stieltjes integral of the form integral[f(t)dg(t)]
#
# Expects:
# f = a function of t (with potential other parameters) that is the integrand
# g = a function of t (with potential other parameters) that is the integrator
# lower = minimum value to integrate from (inclusive)
# upper = maximum value to integrate to (inclusive)
# fparams = a list of optional parameters to pass to the function f
# gparams = a list of optional parameters to pass to the function g
# h = step-size to use in the integration. Is uniform over the interval [lower, upper].
# upper - lower should be divisible by h, otherwise upper will not be included in integration
#
# Returns:
# Estimate of Riemann-Stieltjes integral
rs_integrate = function(f, g, lower, upper, fparams = list(), gparams = list(), h = 0.001){

  t = seq(lower,upper,by = h)

  w = do.call(g,append(list(t[2:length(t)]),gparams)) - do.call(g,append(list(t[1:(length(t)-1)]),gparams))

  evals = (do.call(f,append(list(t[2:length(t)]),fparams)) + do.call(f,append(list(t[1:(length(t)-1)]),fparams)))
  / 2
  return(sum(w*evals))
}
# Function to evaluate the survival function from an estimate produced by Estimation_LTIC_Shen
#
# Expects:
# t = vector (or single value) of time points at which to evaluate the survival function
# est = output list produced by Estimation_LTIC_Shen
#
# Returns:
# A vector of S(t) values for each t given to the function
eval_St = function(t,est){
  # Read t as a vector and create return vector
  t = as.vector(t)
  ret_vec = rep(1,length(t))
  # Check for errors
  if(any(t < 0)){
    stop("Function only valid for t >= 0")
  } else {
```

```

# Get indices where  $S(t) < 1$ 
t_idx = which(t >= est$intmap[1,1])
# Takes care of cases where  $S(t)=1$  for all t
if(length(t_idx)>0){
  ret_vec[t_idx] = apply(cbind(t[t_idx]),1,function(w){
    # Get maximum index where  $t > q_j$ 
    qp_idx = max(which(est$intmap[1,] <= w))
    qj = as.numeric(est$intmap[1,qp_idx])
    pj = as.numeric(est$intmap[2,qp_idx])
    #  $S(t)$  constant for  $[p_j, q_j+1]$  (also captures  $q_j = p_j$ )
    if(w >= pj){
      est$sigma[qp_idx]
      #  $S(t)$  linear on  $[q_j, p_j]$  (also captures constant  $S(t)$  after  $p(J-1)$  for  $pJ = \text{Inf}$ )
    } else {
      # min resolves issues for t in  $[q_1, p_1]$ 
      min(est$sigma[qp_idx-1],1) - est$pf[qp_idx]*(w-qj)/(pj-qj)
    }
  })
}
return(ret_vec)
}

# Function to evaluate  $Y(t)$  from the weight function  $W(t)$  used in calculating the  $L_w$  statistic
#
# Expects:
# t = vector (or single value) of time points at which to evaluate  $Y(t)$ 
# vals = vector of values at which  $Y(t)$  can change
# Y = vector of  $Y(t)$  values corresponding with vals
#
# Returns:
# A vector of  $Y(t)$  values for each t given to the function
eval_Yt = function(t, vals, Y){
  # Read t as a vector
  t = as.vector(t)
  # Check for errors
  if(any(t < 0)){
    stop("Function only valid for  $t \geq 0$ ")
  } else {
    # Get indices for where t falls
    idx = apply(cbind(t, 1, function(w) max(which(vals <= w))),1,function(w) max(which(vals <= w)))
    return(Y[idx])
  }
}

# Function taking a data set and estimating its survival function using
# the methods of Shen (2020)
#
# Expecting data matrix data.shen with columns:
# 1) Left Truncation time
# 2) Left censoring time

```



```

# 3) Right censoring time
# 4) # of visits for right censoring time
# 5) Right truncation time (=Inf for LTIC data)
# n = sample size
# mv = maximum # of inspection times
# tol: convergence criterion
# adj.shen = adjustment
#
# Returns the matrix of censoring indices alpha, the matrix of
# truncation indices beta, the matrix of intervals for the survival
# function intmap, the estimated survival function sigma, and the
# vector of probabilities of the event occurring in each interval pf
Estimation_LTIC_Shen = function(data.shen, n, mv=12, p1=0.8, p2=0.5, p3=0.2, tol=1e-3, maxiter=1000,
adj.shen=1)
{
  K1=K2=K3=ind1=ind2=ind3=like=like1=like2=like3=like=loglike=tmpe = rep(NA,n)
  ER = array(NA,c(n,2))
  U = data.shen[,1] # LT time
  tmpe = data.shen[,2] # L
  tmpr = data.shen[,3] # R
  tmpind = data.shen[,4]
  V = data.shen[,5]
  if(sum(tmpind < 2) >= 1){
    data2 = rbind(data.shen[which(tmpind < 2),]) # subset of data w/
    minL=min(data2[,3]) #estimate left support for left interval censored data; QZ: \hat{a}_Q = smallest Ri
    with delta_{i1} = 1
    #print("minL:")
    #print(minL)
  }
  for(i in 1:n)
  {
    if(tmpind[i] < 2)
    {
      if(adj.shen == 1) # estimate S(t) instead of S(t | T > t0)
      {
        tmpe[i] = max(U[i],minL) #to obtain adjusted likelihood function
      }
      ind1[i]=0
      ind2[i]=0
      ind3[i]=1 #indicator for left censoring
    }
    else if(tmpind[i] > mv)
    {
      ind1[i]=0
      ind2[i]=1 #indicator for right censoring
      ind3[i]=0
    }
    else if((tmpind[i] < mv + 1) & (tmpind[i] > 1))
    {

```

```

ind1[i]=1 #indicator for interval censoring
ind2[i]=0
ind3[i]=0
}
ER[i,] = c(tmpe[i],tmpr[i])
}
data = cbind(U,ER)
q1=1-p1
q2=1-p2
q3=1-p3
t1=qgamma(q1,2,1)
t2=qgamma(q2,2,1)
t3=qgamma(q3,2,1)          #0.1,0.3,0.5,0.7,0.9
U = data[,1]
L = data[,2]
R = data[,3]
#V = data[,4]
minu=min(U)
maxv=max(V)
adj=exp(-minu)-exp(-maxv)
#print("adj")
#print(adj)
#=====
#data = data[which(ind3==0),]
#n = dim(data)[1]
ruev = cbind(c(data),c(rep(0,n),rep(1,n),rep(0,n)))
tmp = order(ruev[,1])
ruev = ruev[tmp,]
n2 = dim(ruev)[1]
P = Q = c()
for(i in 1:(n2-1))
{
  if((ruev[i,2]==1)&(ruev[i+1,2]==0))
  {
    Q = c(Q,ruev[i,1]);
    P = c(P,ruev[i+1,1])
  }
}
nJ = length(Q)
sj = rep(1/nJ,nJ)
alpha = (matrix(rep(Q,rep(n,nJ)),n,nJ)>=data[,2])&(matrix(rep(P,rep(n,nJ)),n,nJ)<=data[,3])
#beta = (matrix(rep(Q,rep(n,nJ)),n,nJ)>=data[,1])&(matrix(rep(P,rep(n,nJ)),n,nJ)<=data[,4])
beta = (matrix(rep(Q,rep(n,nJ)),n,nJ)>=data[,1])
fflike=0
times=0
repeat{
  times = times+1
  Msj = sum(1/apply(t(t(beta)*sj),1,sum))
  djsj = apply((alpha/apply(t(t(alpha)*sj),1,sum)) - (beta/apply(t(t(beta)*sj),1,sum)),2,sum)

```

```

sjn = (1 + djsj/Msj)*sj
sjn=ifelse(sjn < 0, 0.00001, sjn)
sjn=sjn/sum(sjn)
diff = abs(sjn-sj)
sj = sjn
slr = cbind(sj,Q,P)
# Likelihood calculation:
# for(i in 1:n)
# {
#   tK1 <- ifelse((P>R[i]),1,0)
#   K1[i]=sum(tK1*slr[,1])
#   tK2 <- ifelse((P>L[i]),1,0)
#   K2[i]=sum(tK2*slr[,1])
#   tK3 <- ifelse((P>U[i]),1,0)
#   K3[i]=sum(tK3*slr[,1])
#   like1[i]=(K2[i]-K1[i])**ind1[i]
#   like2[i]=(K3[i]-K1[i])**ind3[i]
#   like3[i]=(K2[i])**ind2[i]
#   like[i]=like1[i]*like2[i]*like3[i]/K3[i]
#   loglike[i]=log(like1[i])+log(like2[i])+log(like3[i])-log(K3[i])
# }
# #flike=log(prod(like1))+log(prod(like2))+log(prod(like3))
# flike=sum(loglike)
# diff=abs(fflike-flike)
if(sum(diff) < tol) break
if(times == maxiter) break
#cat(times," - ",fflike,flike,diff,"\n")
#fflike=flike
}
sigma = 1-cumsum(sj)
intmap = rbind(Q,P)
ret = list(alpha = 1*alpha, beta = 1*beta, intmap = intmap, sigma = sigma, pf = sj)
return(ret)
#print("Estimated PDF")
#print(slr)
# if(adj.shen == 1)
# {
#   tslr1 = slr[which(P>t1),]
#   #Fa.hat = sum(atslr[,1])
#   F1.hat = sum(tslr1[,1])
#   #F1.hat.seq[r]=F1.hat
#   tslr2 = slr[which(P>t2),]
#   F2.hat = sum(tslr2[,1])
#   #F2.hat.seq[r]=F2.hat
#   tslr3 = slr[which(P>t3),]
#   F3.hat = sum(tslr3[,1])
#   #F3.hat.seq[r]=F3.hat
#   print("F1.hat, F2.hat, F3.hat")
#   print(c(F1.hat, F2.hat, F3.hat))

```

```

# }
# else
# {
#   atslr = slr[which(P>minL),]
#   Fa.hat = sum(atslr[,1])
#   tslr1 = slr[which(P>max(t1,minL)),]
#   F1.hat = sum(tslr1[,1])
#   cF1.hat = F1.hat/Fa.hat
#   tslr2 = slr[which(P>max(t2,minL)),]
#   F2.hat = sum(tslr2[,1])
#   cF2.hat = F2.hat/Fa.hat
#   tslr3 = slr[which(P>max(t3,minL)),]
#   F3.hat = sum(tslr3[,1])
#   cF3.hat = F3.hat/Fa.hat
#   print("Fa.hat, F1.hat, F2.hat, F3.hat, cF1.hat, cF2.hat, cF3.hat")
#   print(c(Fa.hat, F1.hat, F2.hat, F3.hat, cF1.hat, cF2.hat, cF3.hat))
# }
}
# Function to compute the Lw statistic as in Shen (2014)
#
# Expects:
# A = data matrix with columns:
#   1) Left Truncation Time
#   2) Left Censoring Time
#   3) Right Censoring Time
#   4) # of Visits for Right Censoring Time
# trt = the vector of treatment groups for the data (should take the value of 0 or 1)
# mv = maximum number of visits for interval-censored data
# r = value of r to be used as the power for the weight of S(t) in the test statistic
# s = value of s to be used as the power for the weight of [1-S(t)] in the test statistic
# h = the step size to use when numerically estimating the Riemann-Stieltjes integral for the Lw statistic
# tol = tolerance for the convergence of the estimated survival function
# maxitter = maximum number of iterations for estimating the survival function
# inf = value representing infinity in the data for right censoring
# adj.shen = 0 or 1, whether or not to apply the adjustment introduced by Shen (2020)
#
# Returns:
# The computed Lw statistic
calc_Lw = function(A, trt, mv=13, r=0, s=0, h=0.005, tol=1e-06, maxiter=5000, inf1=Inf, adj.shen=1){
  # Set right censored observations to Inf
  A[A[,3]==inf1,3] = Inf
  # Check data format
  if(ncol(A) == 4 && all(A[,2] >= A[,1]) && all(A[,3] >= A[,2]) && all(A[,4]>0) && all(A[,4]<=mv+1)
  && length(unique(trt)) == 2 && all(trt>=0) && all(trt<2)){
    #Set parameters for easier calculation
    n = dim(A)[1]
    n1 = sum(trt==0)
    n2 = sum(trt==1)
    # Calculate the estimates for the 2 treatment groups

```

```

    est1 = Estimation_LTIC_Shen(cbind(A[trt==0,],rep(Inf,n1)), n1, mv=mv, tol=tol, maxiter=maxiter,
adj.shen=adj.shen)
    est2 = Estimation_LTIC_Shen(cbind(A[trt==1,],rep(Inf,n2)), n2, mv=mv, tol=tol, maxiter=maxiter,
adj.shen=adj.shen)
    # Create Delta(t) from S(t)
    Deltat = function(t,est){
        return(-log(eval_St(t,est)))
    }
    # Calculate Y1 and Y2
    vals = sort(unique(c(0,A[,1],A[,2],A[,3],Inf)))
    Y1 = rep(0,length(vals))
    Y2 = rep(0,length(vals))
    for(i in 1:length(vals)){
        Y1[i] = sum((1*(A[trt==0,1]<=vals[i]) + 1*(A[trt==0,2] > vals[i]))==2, (1*(A[trt==0,2] <= vals[i]) +
1*(A[trt==0,3] >= vals[i]) + 1*(is.finite(A[trt==0,3]))==3)
        Y2[i] = sum((1*(A[trt==1,1]<=vals[i]) + 1*(A[trt==1,2] > vals[i]))==2, (1*(A[trt==1,2] <= vals[i]) +
1*(A[trt==1,3] >= vals[i]) + 1*(is.finite(A[trt==1,3]))==3)
    }
    # Check if we need to calculate S(t)
    if(r==0 && s==0){
        # Create function for W(t) without S(t)
        Wt = function(t, vals, Y1, Y2){
            WL = (eval_Yt(t,vals,Y1)*eval_Yt(t,vals,Y2))/(eval_Yt(t,vals,Y1)+eval_Yt(t,vals,Y2))
            WL[is.nan(WL)] = 0
            return(WL)
        }
        # Find b
        b =
min(max(vals[eval_St(vals,est1)>0]),max(vals[eval_Yt(vals,vals,Y1)>0]),max(vals[eval_St(vals,est2)>0]),
max(vals[eval_Yt(vals,vals,Y2)>0]))
        # Calculate the test statistic
        Lw = rs_integrate(Wt, Deltat, 0, b, fparams=list(vals=vals,Y1=Y1,Y2=Y2), gparams=list(est=est1),
h=h)
        Lw = Lw - rs_integrate(Wt, Deltat, 0, b, fparams=list(vals=vals,Y1=Y1,Y2=Y2),
gparams=list(est=est2), h=h)
        Lw = sqrt(n/(n1*n2))*Lw
    } else {
        # Calculate S(t)
        est = Estimation_LTIC_Shen(cbind(A,rep(Inf,n)), n, mv=mv, tol=tol, maxiter=maxiter,
adj.shen=adj.shen)
        # Create function for W(t)
        Wt = function(t, est, vals, Y1, Y2, r, s){
            WL = (eval_Yt(t,vals,Y1)*eval_Yt(t,vals,Y2))/(eval_Yt(t,vals,Y1)+eval_Yt(t,vals,Y2))
            WL[is.nan(WL)] = 0
            return(WL*(eval_St(t,est)^r)*(1-eval_St(t,est))^s)
        }

        # Find b
        b =

```

```

min(max(vals[eval_St(vals,est1)>0]),max(vals[eval_Yt(vals,vals,Y1)>0]),max(vals[eval_St(vals,est2)>0]),
max(vals[eval_Yt(vals,vals,Y2)>0]))

```

```

# Calculate the test statistic
Lw = rs_integrate(Wt, Deltat, 0, b, fparams=list(est=est,vals=vals,Y1=Y1,Y2=Y2,r=r,s=s),
gparams=list(est=est1), h=h)
Lw = Lw - rs_integrate(Wt, Deltat, 0, b, fparams=list(est=est,vals=vals,Y1=Y1,Y2=Y2,r=r,s=s),
gparams=list(est=est2), h=h)
Lw = sqrt(n/(n1*n2))*Lw
}
return(Lw)
} else {
stop("Please Verify data format, # of samples, and treatment indicator!")
}
}

```

```

# Function to calculate the variance of a test statistic using the simple bootstrap method

```

```

#

```

```

# Expects:

```

```

# A = matrix of data

```

```

# trt = treatment group vector. If bootstrapping without treatment groups, set trt = rep(0,n)

```

```

# fun = function to calculate the test statistic of interest

```

```

# B = number of times to perform the bootstrap

```

```

# params = list of parameters to pass to the function calculating the test statistic.

```

```

# Requires all non-default parameters after the matrix of data to be included.

```

```

#

```

```

# Returns:

```

```

# The unbiased estimate of the variance of the test statistic calculated via the bootstrap

```

```

bootstrap_var = function(A, trt, fun, B = 100, params=list()){

```

```

# Create output vectors

```

```

test_stat = rep(0,B)

```

```

# Perform B bootstraps

```

```

for(i in 1:B){

```

```

AA = A

```

```

for(k in unique(trt)){

```

```

AA[trt==k,] = A[sample(which(trt==k),sum(trt==k),replace = TRUE),]

```

```

}

```

```

test_stat[i] = do.call(fun,append(list(AA),params))

```

```

}

```

```

# Return the variance of the test statistic

```

```

return(sum((test_stat-mean(test_stat))^2)/(B-1))

```

```

}

```

```

myfunction = function(n=400, a=4,b, beta=0, mv=12, theta=0.01, thetad=0.2, inf=10000000000,p1=0.8,

```

```

p2=0.5, p3=0.2,r1,s1, h=0.005, B=100, tol=1e-06, maxiter=1000, adj.shen=1)

```

```

{

```

```

#define and Store survival time (T),left truncation time(U), value representing infinity for right
censoring(V)

```

```

Z=T=U=V=X=rep(NA,n)

```

```

####for observations from i=1 to 1=n:

```

```

i=1
tot.samp = 0
repeat{
  #####Based on Exponentiated Weibull, creating tmpt#####
  tmpt=(1/a)*(-log(1-runif(1,0,1)^exp(-beta*0)))^(1/b) # distribution of survival time:  $T_i \sim$  Exponentiated
Weibull with a, b, beta and Z
  #####
  tmpt = round(tmpt, digits = 3) # Round generated value
  tmpu = theta*runif(1,0,1) #distribution of left truncation times
  tmpu = round(tmpu, digits = 3) # Round generated value
  tmpv=tmpu+inf
  ###When survival time > left truncated time, we can obtain data, and get
  if(tmpt>=tmpu){
    U[i] = tmpu # left truncated time
    V[i] = tmpv # value representing infinity for right censoring
    T[i] = tmpt
    Z[i] = 0# Survival time
    i=i+1
  }
  tot.samp = tot.samp + 1
  if(i==n/2+1) break
}
i=n/2+1
tot.samp = n/2
repeat{
  #####Based on Exponentiated Weibull, creating tmpt#####
  tmpt=(1/a)*(-log(1-runif(1,0,1)^exp(-beta*1)))^(1/b) # distribution of survival time:  $T_i \sim$  Exponentiated
Weibull with a, b, beta and Z
  #####
  tmpt = round(tmpt, digits = 3) # Round generated value
  tmpu = theta*runif(1,0,1) #distribution of left truncation times
  tmpu = round(tmpu, digits = 3) # Round generated value
  tmpv=tmpu+inf
  ###When survival time > left truncated time, we can obtain data, and get
  if(tmpt>=tmpu){
    U[i] = tmpu # left truncated time
    V[i] = tmpv # value representing infinity for right censoring
    T[i] = tmpt
    Z[i] = 1# Survival time
    i=i+1
  }
  tot.samp = tot.samp + 1
  if(i==n+1) break
}
#####after we set up survival time, left truncated time and theoretical infinity,
#####based on fixed visit interval (5 days), we create # of inspection times for each
observation,
#####then we could obtain Set ( $L_i, R_i$ ).
tmpx=rep(NA,n) # define and store # of inspection times for each observation i

```

```

tmpU=matrix(0,n,mv) # Store details of each inspection time (total mv inspection times ) for each
observation i (total n)
tmpZ=matrix(NA,n,mv+2) # Store LT time, inspection times, and 10000 (but repeats the largest
inspection time until 10000) of the subjects
tmpind=rep(NA,n) # define and store # of inspection times for each observation i
tmpe=rep(NA,n) # define and store Li
tmpr=rep(NA,n) # define and store Ri
for(i in 1:n){
  tmpx[i] = 2 + rbinom(1,mv-2,0.5) # generate #. of inspection times
  tmpU[i,] = c(thetad*runif(tmpx[i]),rep(0,mv-tmpx[i])) #based on #. of inspection times, we know visit
interval is 5 days and how many visits for each observation i
  tmpU[i,] = round(tmpU[i,], digits = 3) # Round generated values
  tmpZ[i,] = c(U[i],cumsum(tmpU[i,])+U[i],inf) #for each observation i, store, left truncated time U, last
visit time under left truncated, value representing infinity for right censoring
  tmpind[i] = max(which(T[i]>=tmpZ[i,])) #store # of inspection times for each observation i
  tmpe[i] = tmpZ[i,tmpind[i]] #store Li
  tmpr[i] = tmpZ[i,tmpind[i]+1] #store Ri
  X[i] = tmpx[i] # # of inspection times
}
data.EW= cbind(U,tmpe,tmpr,tmpind,V,Z,T)# contains LT time, L, R, # of visits for Right censoring time
#####
#####
K1=K2=K3=ind1=ind2=ind3=like=like1=like2=like3=like=loglike=tmpe = rep(NA,n)
ER = array(NA,c(n,2))
U = data.EW[,1] # LT time
tmpe = data.EW[,2] # L
tmpr = data.EW[,3] # R
tmpind = data.EW[,4]
V = data.EW[,5]
Z = data.EW[,6]
if(sum(tmpind < 2) >= 1){
  data2 = rbind(data.EW[which(tmpind < 2),]) # subset of data w/
  minL=min(data2[,3]) #estimate left support for left interval censored data; QZ: \hat{a}_Q = smallest Ri
with delta_{i1} = 1
  #print("minL:")
  #print(minL)
}
for(i in 1:n)
{
  if(tmpind[i] < 2)
  {
    if(adj.shen == 1) # estimate S(t) instead of S(t | T > t0)
    {
      tmpe[i] = max(U[i],minL) #to obtain adjusted likelihood function
    }
    ind1[i]=0
    ind2[i]=0
    ind3[i]=1 #indicator for left censoring
  }
}

```



```

else if(tmpind[i] > mv)
{
  ind1[i]=0
  ind2[i]=1 #indicator for right censoring
  ind3[i]=0
}
else if((tmpind[i] < mv + 1) & (tmpind[i] > 1))
{
  ind1[i]=1 #indicator for interval censoring
  ind2[i]=0
  ind3[i]=0
}
ER[i,] = c(tmpe[i],tmpr[i])
}

data = cbind(U,ER)
q1=1-p1
q2=1-p2
q3=1-p3
t1=qgamma(q1,2,1)
t2=qgamma(q2,2,1)
t3=qgamma(q3,2,1)          #0.1,0.3,0.5,0.7,0.9
U = data[,1]
L = data[,2]
R = data[,3]
#V = data[,4]
minu=min(U)
maxv=max(V)
adj=exp(-minu)-exp(-maxv)
#print("adj")
#print(adj)
#=====
#data = data[which(ind3==0),]
#n = dim(data)[1]
ruev = cbind(c(data),c(rep(0,n),rep(1,n),rep(0,n)))
tmp = order(ruev[,1])
ruev = ruev[tmp,]
n2 = dim(ruev)[1]
P = Q = c()
for(i in 1:(n2-1))
{
  if((ruev[i,2]==1)&(ruev[i+1,2]==0))
  {
    Q = c(Q,ruev[i,1]);
    P = c(P,ruev[i+1,1])
  }
}
nJ = length(Q)
sj = rep(1/nJ,nJ)

```

```

alpha = (matrix(rep(Q,rep(n,nJ)),n,nJ)>=data[,2])&(matrix(rep(P,rep(n,nJ)),n,nJ)<=data[,3])
#beta = (matrix(rep(Q,rep(n,nJ)),n,nJ)>=data[,1])&(matrix(rep(P,rep(n,nJ)),n,nJ)<=data[,4])
beta = (matrix(rep(Q,rep(n,nJ)),n,nJ)>=data[,1])
fflike=0
times=0
repeat{
  times = times+1
  Msj = sum(1/apply(t(t(beta)*sj),1,sum))
  djsj = apply((alpha/apply(t(t(alpha)*sj),1,sum)) - (beta/apply(t(t(beta)*sj),1,sum)),2,sum)
  sjn = (1 + djsj/Msj)*sj
  sjn=ifelse(sjn < 0, 0.00001, sjn)
  sjn=sjn/sum(sjn)
  diff = abs(sjn-sj)
  sj = sjn
  slr = cbind(sj,Q,P)
  # Likelihood calculation:
  # for(i in 1:n)
  # {
  #   tK1 <- ifelse((P>R[i]),1,0)
  #   K1[i]=sum(tK1*slr[,1])
  #   tK2 <- ifelse((P>L[i]),1,0)
  #   K2[i]=sum(tK2*slr[,1])
  #   tK3 <- ifelse((P>U[i]),1,0)
  #   K3[i]=sum(tK3*slr[,1])
  #   like1[i]=(K2[i]-K1[i])**ind1[i])
  #   like2[i]=(K3[i]-K1[i])**ind3[i])
  #   like3[i]=(K2[i])**ind2[i])
  #   like[i]=like1[i]*like2[i]*like3[i]/K3[i]
  #   loglike[i]=log(like1[i])+log(like2[i])+log(like3[i])-log(K3[i])
  # }
  # #flike=log(prod(like1))+log(prod(like2))+log(prod(like3))
  # flike=sum(loglike)
  # diff=abs(fflike-flike)
  if(sum(diff) < tol) break
  if(times == maxiter) break
  #cat(times," - ",fflike,flike,diff,"\n")
  #fflike=flike
}
sigma = 1-cumsum(sj)
intmap = rbind(Q,P)
E.EW = list(alpha = 1*alpha, beta = 1*beta, intmap = intmap, sigma = sigma, pf = sj, Z = Z,Q=Q)
#####
#####
fpj=rep(NA,sum(E.EW[["pf"]]>=0))
alphaa=TOP=BOT=matrix(NA,n,sum(E.EW[["pf"]]>=0)) #calculation middleman
SUMTOP=SUMBOT=CC=ZZ=UU=rep(NA,n) #calculation middleman, sum step
Uscore=VVV=Test.Statistic=NA # score statistics
alphaa=E.EW[["alpha"]] # alpha censoring indicator for Turnbull intervals from above steps
fpj=cumsum(E.EW[["pf"]]) #probability at right-end p for Turnbull intervals from above steps

```

```

Sigma=E.EW[["sigma"]]
#####
fqj=c(0,head(fpj,sum(fpj>=0)-1)) #probability at left-end q for Turnbull intervals from above steps
ZZ=E.EW[["Z"]] #group indicator
for(i in 1:n) #n observations
{
  for(j in 1:sum(fqj>=0)) #j Turnbull intervals
  {
    if(fqj[j]==0 & fpj[j]==0)
    {
      TOP[i,j]=0
      BOT[i,j]=0
    }
    else if(fqj[j]==0 & fpj[j]!=0)
    {
      TOP[i,j]=alphaa[i,j]*fpj[j]*log(fpj[j])
      BOT[i,j] = alphaa[i,j]*fpj[j]
    }
    else if(fqj[j]!=0 & fpj[j]!=0)
    {
      TOP[i,j]=alphaa[i,j]*(fpj[j]*log(fpj[j])-fqj[j]*log(fqj[j]))
      BOT[i,j] = alphaa[i,j]*(fpj[j]-fqj[j])
    }
    #top[j]=fpj[j]*log(fpj[j])-fqj[j]*log(fqj[j])
    #TOP[i,j]=alpha[i,j]*(fpj[j]*log(fpj[j])-fqj[j]*log(fqj[j]))
    #if(alpha[i,j]*(fpj[j]*log(fpj[j])-fqj[j]*log(fqj[j])) < 10^(-10) & alpha[i,j]*(fpj[j]*log(fpj[j])-
    fqj[j]*log(fqj[j])) > -10^(-10)) # estimate S(t) instead of S(t | T > t0)
    #{
      #TOP[i,j]=0 #to obtain adjusted likelihood function
    #}
    #else
    #{
      #TOP[i,j]=alpha[i,j]*(fpj[j]*log(fpj[j])-fqj[j]*log(fqj[j]))
    #}
    #TOP[i,j]=alpha[i,j]*(fpj[j]*log(fpj[j])-fqj[j]*log(fqj[j])) ###fqj[1]=0 and Top[i,j] shows
    NAN!!!!!!!!!!!!!! ERROR!!!!!!!!!!!!!!
    #BOT[i,j] = alpha[i,j]*(fpj[j]-fqj[j])
  }
}
#####
FU=rep(NA,n)
CUMFU=1-Sigma
for(i in 1:n) #n observations
{
  if(data.EW[i,1] < E.EW[["intmap"]][2,1] & data.EW[i,1] >= E.EW[["intmap"]][1,1])
  {
    FU[i]=CUMFU[1]*(data.EW[i,1]-E.EW[["intmap"]][1,1])/(E.EW[["intmap"]][2,1]-
    E.EW[["intmap"]][1,1])
  }
}

```

```

else if(data.EW[i,1] < E.EW[["intmap"]][1,1])
{
  FU[i]=0
}
}

for(i in 1:n) #n observations
{
  for(j in 2:sum(fqj>=0)) #j Turnbull intervals
  {
    if(data.EW[i,1] < E.EW[["intmap"]][2,j] & data.EW[i,1] >= E.EW[["intmap"]][1,j])
    {
      FU[i]=CUMFU[j-1]+E.EW[["pf"]][j]*(data.EW[i,1]-E.EW[["intmap"]][1,j])/(E.EW[["intmap"]][2,j]-
E.EW[["intmap"]][1,j])
    }
    else if(data.EW[i,1] < E.EW[["intmap"]][1,j] & data.EW[i,1] >= E.EW[["intmap"]][2,j-1])
    {
      FU[i]=CUMFU[j-1]
    }
  }
}
TOPFU=BOTFU=rep(NA,n)
for(i in 1:n) #n observations
{
  if(FU[i]==0)
  {
    TOPFU[i]=0
  }
  else if(FU[i]!=0)
  {
    TOPFU[i]=FU[i]*log(FU[i])
  }
}
BOTFU=1-FU
CCFU=UUFU=rep(NA,n)
for(i in 1:n) #n observations
{
  CCFU[i]=TOPFU[i]/BOTFU[i]
  #UUFU[i]=ZZ[i]*CCFU[i]
}
#UFU=sum(UUFU)
#VFU=(1/(n-1))*sum((CCFU-mean(CCFU))^2)*sum((ZZ-mean(ZZ))^2)
#####
SUMTOP=rowSums(TOP)
SUMBOT=rowSums(BOT)
for(i in 1:n) #n observations
{
  CC[i]=SUMTOP[i]/SUMBOT[i]
  #UU[i]=ZZ[i]*CC[i]
}

```

```

}
CCC=CC+CCFU
UUU=ZZ*CCC
Uscore=sum(UUU)
VVV=(1/(n-1))*sum((CCC-mean(CCC))^2)*sum((ZZ-mean(ZZ))^2)
#Uscore=sum(UU) # find estimated score statistic
#VV=(1/(n-1))*sum((CC-mean(CC))^2)*sum((ZZ-mean(ZZ))^2) #permutation to gain Variance of
Uscore
#####
#UUU=Uscore+UFU
#VVV=VFU+VV
Test.Statistic=(Uscore^2)*VVV^-1 #final test statistic from  $U \cdot V^{(-1/2)}$  following chi-square with df=1
Indicator=ifelse(Test.Statistic>qchisq(0.95,1),1,0)
#####
AAA=data.EW[,-5]
A=AAA[,-6]
#####
#####
# Set right-truncated values to have Inf as their right censoring time
A[A[,3]==inf,3] = inf
# Check data format
if(ncol(A) == 5 && all(A[,2] >= A[,1]) && all(A[,3] >= A[,2]) && all(A[,4]>0) && all(A[,4]<=mv+1)
&& length(unique(A[,5])) == 2 && all(A[,5]>=0) && all(A[,5]< 2)){
  # split up input data to clean up notation
  AA = A[,-5]
  trt = A[,5]
  # Calculate test statistic
  Lw1 = calc_Lw(AA, trt, mv=mv, r=r1, s=s1, h=h, tol=tol, maxiter=maxiter, adj.shen=adj.shen)
  # Calculate test variance via bootstrap
  params1 = list(trt=trt, mv=mv, r=r1, s=s1, h=h, tol=tol, maxiter=maxiter, adj.shen=adj.shen)
  sig2_hat1 = bootstrap_var(AA, trt, fun=calc_Lw, B=B, params=params1)
  # Calculate Z-score and p-value
  Zshen1 = Lw1/sqrt(sig2_hat1)
  pshen1 = 1 - pnorm(abs(Zshen1),0,1)
  Indicator1=ifelse(pshen1<0.05,1,0)
  # Output the test results
} else {
  stop("Please Verify data format, # of samples, and treatment indicator!")
}
return(c(Indicator,Indicator1))
}
df=data.frame(n=rep(200,1000),
a=rep(2,1000),
b=rep(1,1000),
beta=rep(0,1000),
mv=rep(6,1000),
theta=rep(2,1000),
thetad=rep(0.35,1000),
inf=rep(1000,1000),

```

```

p1=rep(0.8,1000),
p2=rep(0.5,1000),
p3=rep(0.2,1000),
r1=rep(0,1000),
s1=rep(0,1000),
h=rep(0.005,1000),
B=rep(500,1000),
tol=rep(1e-06,1000),
maxiter=rep(1000,1000),
adj.shen=rep(1,1000))
dfm=as.matrix(df)
library(parallel)
SIZEmcl=mclapply(seq_len(nrow(dfm)),function(x) do.call(myfunction,as.list(dfm[x,])),mc.cores=28)
#testresult=myfunction(n=200, a=0.25, b=2, beta=0, mv=12, theta=0.5, thetad=1, inf=10000,tol=1e-6,
maxiter=1000, adj.shen=1)
#SIZE=mcmapply(myfunction,n, a, b, beta, mv, theta, thetad, inf,r, s, h, B, tol, maxiter,
adj.shen,mc.cores=parallel::detectCores()-1)
print(SIZEmcl)
#SIZE6=as.data.frame(t(SIZEmcm))
#write.csv(SIZEmcm,"0330-1.csv")
#Sys.time()
#nohup bash -c 'time Rscript --verbose mcl-real.R > mcl-real.Rout.txt'

```

REFERENCES

- Barlow, Richard E., Albert W. Marshall, Frank Proschan. 1963. "Properties of probability distributions with monotone hazard rate." *The Annals of Mathematical Statistics*. 34(2):375–389.
- Bilker, Warren B., Mei C. Wang. 1996. "A semiparametric extension of the Mann-Whitney test for randomly truncated data." *Biometrika*. 52(1):10–20.
- Bogaerts, Kris, Aronšt Komarek, Emmanuel Lesaffre. 2020. *Survival Analysis with Interval-Censored Data. A Practical Approach with Examples in R, SAS, and BUGS*. New York, NY: Chapman and Hall/CRC.
- Chen, Ding G., Jian G. Sun, Karl E. Peace. 2013. *Interval-censored time-to-event data: methods and applications*. No. 52 in Chapman & Hall/CRC biostatistics series, Boca Raton, Fla: CRC Press, Chapman & Hall.
- Cochran, William G.. 1954. "Some Methods for Strengthening the Common χ^2 Tests". *Biometrics*. 10:417–451.
- Cox, David R. 1972. "Regression Models and Life-Tables." *Journal of the Royal Statistical Society, Series B*. 34(2):187–220.
- Fay, Michael P., Joanna H. Shih. 1998. "Permutation tests using estimated distribution functions." *Journal of the American Statistical Association*. 93(441):387–396.
- Fleming, Thomas R., David P. Harrington. 1991. *Counting processes and survival analysis*. New York: John Wiley & Sons, Inc.
- Frydman, Halina. 1994. "A note on nonparametric estimation of the distribution function from interval-censored and truncated data." *Journal of the Royal Statistical Society, Series B*. 56(1):71–74.

Gehan, Edmund A.. 1965. "A generalized Wilcoxon test for comparing arbitrarily singly-censored samples." *Biometrika*. 52(1-2):203–223.

Hájek, Jaroslav, Zbynek Sidák, Pranab K. Sen. 1999. *Theory of rank tests. Probability and mathematical statistics*. San Diego, Calif.: Academic Press, 2nd ed.

Jantzen, Rodolphe, Pascale Tubert-Bitter, Philippe Broët. 2022. "Analyzing cohort studies with interval-censored data: A new model-based linear rank-type test." *Statistics in Medicine*. 41(18):3479-3491.

Kalbfleisch, John D., Jerald F. Lawless. 1989. "Inference Based on Retrospective Ascertainment: An Analysis of the Data on Transfusion-Related AIDS." *Journal of the American Statistical Association*. 84(406):360-372.

Kalbfleisch, John D., Ross L. Prentice. 2002. *The statistical analysis of failure time data. Wiley series in probability and statistics*. Hoboken, NJ: Wiley. 2nd. ed.

Kaplan, Edward L., Paul Meier. 1958. "Nonparametric Estimation from Incomplete Observations." *Journal of the American Statistical Association*. 53(282):457-481.

Kruskal, William H., W. A Wallis. 1952. "Use of ranks in one-criterion variance analysis." *Journal of the American Statistical Association*. 47(260):583–621.

Lagakos, Stephen W., Leila M. Barraj, V. De Gruttola. 1988. "Nonparametric analysis of truncated survival data with application to AIDS." *Biometrika*. 75(3):515–524.

Mann, Henry B., D. R. Whitney. 1947. "On a test of whether one of two random variables is stochastically larger than the other." *Annals of Mathematical Statistics*. 18(1):50–60.

Mantel, Nathan, William Haenszel. 1959. "Statistical aspects of the analysis of data from retrospective studies of disease." *Journal of the National Cancer Institute*. 22(4):719–748.

Marshall, Albert W., Ingram Olkin. 2007. *Life distributions*. New York, NY: Springer.

Oller, Ramon, Guadalupe Gómez. 2012. “A generalized Fleming and Harringtons class of tests for interval-censored data.” *The Canadian Journal of Statistics*. 40(3):501–516.

Pepe, Margaret S., Thomas R. Fleming. 1989. “Weighted Kaplan–Meier statistics: a class of distance tests for censored data.” *Biometrics*. 45(2):497–507.

Peto, Richard, Julian Peto. 1972. “Asymptotic efficient rank invariant test procedures.” *Journal of the Royal Statistical Society. Series A*. 135(2): 185–198.

Shen, Pao S. 2002. “Nonparametric estimators of survival function under the mixed case interval-censored model with left truncation.” *Lifetime Data Analysis*. 26(4):624–637.

Shen, Pao S. 2007. “A general class of test procedures for left-truncated and right-censored data.” *Communications in Statistics: Theory and Methods*. 36(16):2913–2925.

Shen, Pao S. 2009. “A class of rank-based test for left-truncated and right-censored data.” *Annals of the Institute of Statistical Mathematics*. 61(2):461–476.

Shen, Pao S. 2015. “Nonparametric tests for left-truncated and interval-censored data.” *Journal of Statistical Computation and Simulation*. 2015; 85(8):1544–1553.

Sun, Jian G. 1996. “A non-parametric test for interval-censored failure time data with application to AIDS studies.” *Statistics in Medicine*. 15(13):1387–1395.

Sun, Jian G. 2006. *The Statistical Analysis of Interval-censored Failure Time Data*. Statistics for Biology and Health, New York, NY: Springer New York.

Turnbull, Bruce W. 1976. "The empirical distribution function with arbitrarily grouped censored and truncated data." *Journal of the Royal Statistical Society, Series B*. 38(3):290–295.

Wang, Mei C. 1991. "Nonparametric estimation from cross-sectional survival data." *Journal of the American Statistical Association*. 86(413):130–143.

Zhao, Qiang, Jian G. Sun. 2004. "Generalized log-rank test for mixed interval-censored failure time data." *Statistics in Medicine*. 23(10):1621–1629.