STATISTICAL ISSUES IN THE DESIGN AND ANALYSIS OF SEQUENTIALLY RANDOMIZED TRIALS

by

Jin Hui Ko

B.S., Myongji University, South Korea, 2000

M.E., Texas A&M University, 2003

Submitted to the Graduate Faculty of

the Department of Biostatistics in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

University of Pittsburgh

2010

UNIVERSITY OF PITTSBURGH

GRADUATE SCHOOL OF PUBLIC HEALTH

This dissertation was presented

by

Jin Hui Ko

It was defended on

July 26, 2010

and approved by

Dissertation Advisor:
Abdus S. Wahed, PhD
Associate Professor
Department of Biostatistics
Graduate School of Public Health
University of Pittsburgh

Committee Member:
Stewart Anderson, PhD
Professor
Department of Biostatistics
Graduate School of Public Health
University of Pittsburgh

Committee Member:
Jong-Hyeon Jeong, PhD
Associate Professor
Department of Biostatistics
Graduate School of Public Health
University of Pittsburgh

Committee Member:
Yu Cheng, PhD
Assistant Professor
Department of Statistics
School of Arts and Sciences
University of Pittsburgh

STATISTICAL ISSUES IN THE DESIGN AND ANALYSIS OF SEQUENTIALLY RANDOMIZED TRIALS

Jin Hui Ko, PhD

University of Pittsburgh, 2010

Adaptive treatment strategies are comprehensive methods for treating chronic diseases according to patients' needs and responses. They are useful in the treatment of diseases such as cancer or AIDS, where the treatment is frequently modified to adapt to the patients' health status. In the first part of this dissertation, we consider two commonly used randomization designs in clinical trial, namely, up-front randomized trial and sequentially randomized trial used to compare treatment strategies. Up-front randomization is the classical method of randomization where patients are randomized at the beginning of the study to pre-specified strategies. On the other hand, in sequentially randomization trials, patients are randomized sequentially to available treatment options over the duration of the therapy as they become eligible to receive them. We compare the efficiency of the traditional up-front randomized trials to that of sequentially randomized trials for comparing adaptive treatment strategies both analytically and numerically based on a continuous outcome.

In the second part of the dissertation, we consider analyzing right-censored survival data from two-stage sequentially randomized trials. In such analysis, it is often of interest to use median residual lifetime as the summary parameter to assess the treatment effectiveness. However, estimation of the median residual lifetime from sequentially randomized trials is not as straightforward because of its sequential randomization structure. We propose methods for estimating strategy-specific median residual life function from a two-stage sequentially randomized trial. Two types of estimators are proposed by inverting the inverse-probability-weighted estimated survival function and by using inverse-probability-weighted estimating

equation function. We provide methods for estimating variances of these estimators and compare them through a simulation study. Our simulation study shows that both methods produce approximately unbiased estimators in large samples. We demonstrate our methods by applying them to a sequentially randomized leukemia clinical trial data set.

Diseases such as cancer, leukemia, depression, and AIDS are major causes of morbidity and mortality in the United States. Medical research in the recent times has focused on finding optimal treatment strategies to manage or eradicate these diseases to reduce individual and community burdens. Statistical methodologies proposed in this dissertation will help appropriately design and analyze trials to develop effective treatment strategies and thus will be of significant use for improving public health in the United States and around the world.

Keywords: Adaptive Treatment Strategies, Dynamic Treatment Regime, Inverse Probability Weighting, Sequential Randomization, Two-Stage Randomization Design, Median Residual Life Function.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

PREFACE

I would like to express my sincere gratitude to my dissertation advisor, Dr. Abdus Wahed, for help and support through the whole process of producing this dissertation. I truly believe that I'm fortunate to have him as an advisor. Without his help, I couldn't finish this dissertation. Also, without his understanding, I couldn't stay with my husband in Virginia during this dissertation study. I would also like to thank my committees and all faculty members in this department for their time and support. Finally, I would like to thank my family and friends, above all, my husband Jeongheon Lee for all his love and support.

1.0 INTRODUCTION

1.1 ADAPTIVE TREATMENT STRATEGY (ATS)

Adaptive treatment strategies are comprehensive methods for treating chronic diseases according to patients' needs and responses. They are useful in the treatment of diseases such as cancer or AIDS, where the treatment is frequently modified to adapt to the patients' health status. For example, in the treatment of acute myeloid leukemia, it is of interest to know whether growth factor infusion should be used following induction chemotherapy, or whether a mono-therapy or a combination therapy should be used as maintenance following remission. Usually, these questions are addressed locally by comparing the treatments at specific stages of the disease. However, a locally optimum treatment (e.g. the best induction for archiving remission) may not be the best globally (e.g. extended overall survival). Therefore, instead of considering each treatment separately, ATSs are formed to consider all treatment options at once for the purpose of the analysis. For example, in the treatment of leukemia, one ATS might be "treat with induction chemotherapy and add a growth factor; if remission is achieved, treat with a combination maintenance therapy; if remission is not achieved, declare that patient to be a treatment failure". Another such strategy can just avoid the growth factor. Treatment options at each evaluation time determine the possible ATSs.

1.2 RANDOMIZATIONS IN CLINICAL TRIALS

Up-front randomized trial (URT) and sequentially randomized trial (SRT) are two methods to randomize patients to different strategies in order to compare treatment strategies. Up-front randomization is the classical method of randomization which randomizes patients at the beginning of the study to pre-specified strategies. Sequential randomization randomizes patients sequentially to possible treatments as they become eligible to receive subsequent treatment. An illustration of URT and SRT with two stages of treatment is presented in Figure 1.

Suppose A_i denotes the i^{th} induction treatment, B_k is the k^{th} maintenance treatment for the responders and C_l is the l^{th} alternative treatment for the non-responders. For simplicity, we assume that there are two varieties of A, B and C, and thus j, k and l can take values 1 and 2. In practice, the alternative treatment C can be one of the induction treatments that were not received by the patients at stage 1 (see Thall et al. [2007]). In some terminal illnesses, such as leukemia in elder patients, if a response (remission) is not achieved with an induction, no further treatment is given to reduce the treatment-related mortality and therefore C can be regarded as no treatment. In this case, patients without response to the induction treatment do not receive any further treatment.

As can be seen from Figure 1(a), in URT, patients are randomized upon entry to all possible treatment strategies (a total of 8, in this case), realizing that some of the patients will not receive the intended maintenance or alternative treatment. For example, patient randomized to $A_1B_1C_1$ will receive B_1 only if they respond to A_1 and C_1 only if they do not respond to A_1. Thus, even if randomized with equal probability, for example, to $A_1B_1C_1$ and $A_1B_1C_2$, the actual number of patients receiving C_1 and C_2 can be very different depending on the number of patients not responding to A_1 in the two arms. URT is simple to conduct and the traditional method of analysis can be easily applied to analyze the data from such trials. Statistical analysis for comparing strategies follows standard method of multiple treatment trials.

In SRT (Figure 1(b)), patients are randomized to one of the two first-stage treatments A_1 or A_2 at entry. Once a patient achieves response, he/she is randomized to the second-

stage treatments B_1 or B_2 and to C_1 or C_2 if not. As can be seen, patients responding to the induction treatment can be randomized to two maintenance treatments B_1 or B_2 by maintaining balance if desired. Similarly, if desired a balance in the number of patients between two alternative treatments C_1 and C_2 can be attained. SRTs are often referred to as SMART (Sequential Multiple Assignment Randomized Trials, Murphy [2005]). Even though more complex statistical methods are required to compare treatment strategies, SRT follows the traditional principle of randomizing patients at the time of ascertaining eligibility.

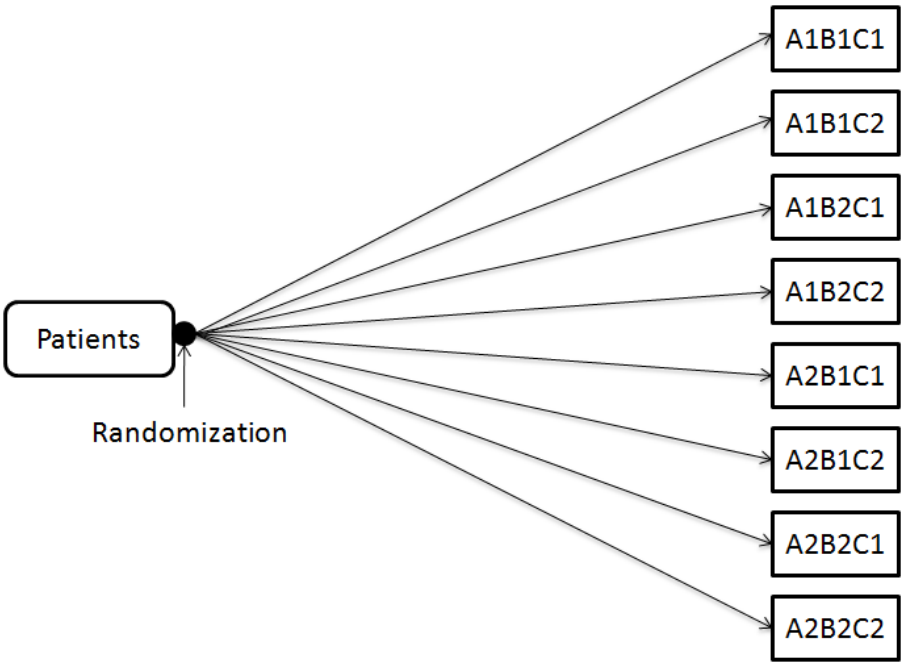
1.3 MEDIAN RESIDUAL LIFE FUNCTION

A MERL function is the median of the remaining life time at a specific time point. For overall survival T, median residual lifetime at time t_0 is defined as:

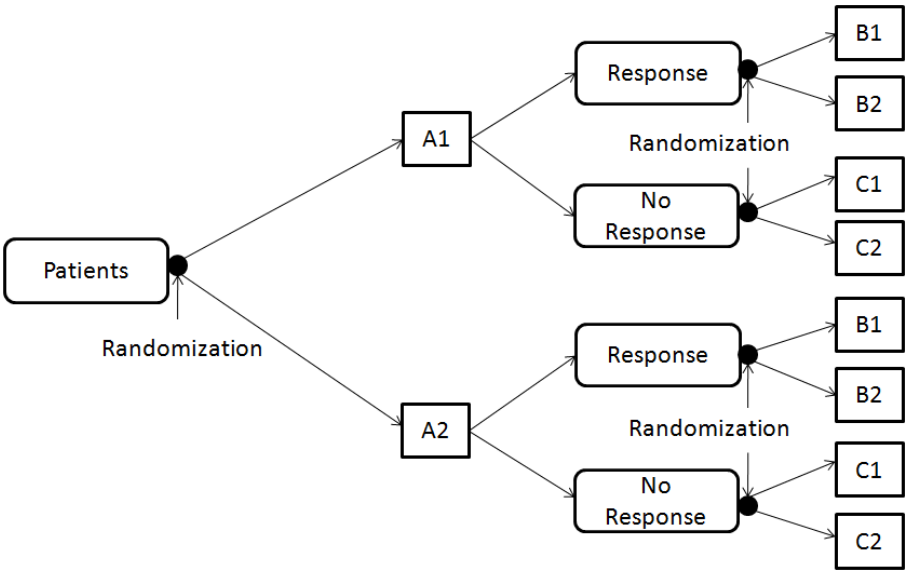
$$\theta(t_0) = median(T - t_0 | T \geq t_0),$$

where $median(X|Y)$ stands for the median of the conditional distribution of X given condition Y. In other words, $\theta(t_0)$ is the median time for those who survived beyond time t_0 in the population. The median residual lifetime is stable since it is not affected by outliers such as very long durations of survival time. In the case with the censored data, the use of a median for the residual lifetime not only exhibits a better location estimate than its mean but also shows less sensitivity to skewed distributions. The minimum of observed survival probability of the residual life distribution should be smaller than the 0.5, so that the median residual lifetime can be defined under censoring (Schmittlein et al. [1981]).

Various methods have been proposed in the literature to estimate the median residual lifetimes for one and two sample cases. Haines et al. [1974] introduced a general concept of the α-percentile residual life function ($0 < \alpha < 1$). Schmittlein et al. [1981] established a general concept of MERL function exploring non-uniqueness of the corresponding lifetime distribution. Csörgö et al. [1987] proposed a $100(1-p)$th percentile residual lifetime estimator for complete data, whereas Chung [1989] extended this idea to censoring cases. The nonparametric estimator of $100(1-p)$th percentile residual lifetime was proposed by Feng



(a) Up-front randomized trial



(b) Sequentially randomized trial

Figure 1: Illustrations of (a) URT and (b) SRT with two stages of treatment

et al. [1991], where inverse function of the Kaplan-Meier curve was used. Jeong et al. [2008] introduced a test statistic to compare two median residual lifetimes at a certain time point.

1.4 OBJECTIVE

The primary objective of this dissertation is two-fold : (a) compare the efficiency of the traditional up-front randomized trials to that of sequentially randomized trials for comparing adaptive treatment strategies and (b) propose methods for estimating strategy-specific median residual life function. We use the probability-adjusted survival estimator to estimate median residual life function.

To achieve the first goal, we construct unbiased estimators under both URT and SRT, and compared their analytical and estimated variances. We present the results in Chapter 3. We then propose two forms of estimators for the median residual lifetime from sequentially randomized trials. These are obtained by inverting the estimated survival function and using the estimating functions. The results are presented in Chapter 4.

2.0 METHODOLOGY REVIEWS

In the following, we review some of techniques used in this dissertation.

2.1 KAPLAN-MEIER ESTIMATOR

The Kaplan-Meier Estimator is a nonparametric method used to estimate a survival curve from lifetime data. Life time data is usually right censored and hence the empirical distribution function does not directly apply to the estimation of the survival distribution. Let $S(t)$ be the survival probability exceeding a time t. The Kaplan-Meier estimator is formulated as follows from the sample data containing N observed times (event/censoring times).

$$\hat{S}(t) = \Pi_{t_i<t} \frac{n_i - d_i}{n_i}, i = 1 \dots N,$$

where t_i's are event times, n_i is the number of survivors excluding censored cases at time t_i and d_i is the number of events occurred at time t_i. The variance of this estimator is generally approximated using the Greenwood formula:

$$var(\hat{S}(t)) = \hat{S}^2(t) \sum_{t_i<t} \frac{d_i}{n_i(n_i - d_i)}.$$

The Kaplan-Meier estimator is known to be unbiased in large samples and is asymptotically normally distributed with mean $S(t)$ and variance that can be estimated by the above formula.

2.2 INVERSE-PROBABILITY-WEIGHTED ESTIMATOR

The idea of inverse-probability-weighting originates from the survey sampling where the inverse of the probability of sampling an individual unit is used as weight for that unit to construct unbiased estimators (Horvitz et al. [1952]). The idea can be extended to the estimation of survival curve in the presence of censoring. When censoring is considered independent of the observed data, the complete observation can be viewed as independent sample of the whole data. Thus, only the complete observations can be used to empirically estimate the survival functions with individual probability of not being censored as the reciprocal of weights.

Let $U_i = min(T_i, C_i)$ be the observed survival time, $\Delta_i = I(T_i < C_i)$ be the event indicator where T_i is the event time and C_i is the potential censoring time for the i^{th} patient that for a fixed time t. The inverse probability of censoring weighted estimator of $S(t)$ is given by

$$\hat{S}_{IPW}(t) = n^{-1} \sum_{i=1}^{n} \frac{\Delta_i}{\hat{K}(U_i)} I(U_i > t)$$

where $\hat{K}(U_i)$ is the Kaplan-Meier estimator for censoring.

3.0 UP-FRONT VS. SEQUENTIAL RANDOMIZATION IN CLINICAL TRIALS

3.1 INTRODUCTION

Recently, SRTs have drawn considerable attention as an effective way of comparing ATSs. Analytical methods for comparing treatment strategies from SRTs are available in the literature (Lunceford et al. [2002], Murphy [2005], Robins et al. [1994], Wahed and Tsiatis [2004]). However, there is no literature on how SRT compares to URT in terms of statistical inference - particularly with respect to the efficiency of the estimators and the power of statistical hypothesis testing. In Wolbers et al. [2008], by using an estimator from Lunceford et al. [2002], several URTs were compared with a single SRT to show how new induction and maintenance treatments are beneficial over the existing treatment combinations. In their work, authors show that SRT is better than URT with respect to efficient use of information about patients. However, comparison does not include the efficiency of estimators.

In this chapter, we derive estimators of the mean of a continuous outcome for URT and SRT with two-stage designs by using the inverse-probability-weighting (IPW) (Robins et al. [1994]). Then we aim to evaluate the performance of the estimators for URT and SRT in terms of the efficiency of the estimators and the power of statistical hypothesis testing. Assumptions and notation used in this chapter are described in section 3.2. Inferential procedures in URT and SRT are discussed in section 3.3. We present the results of a simulation study in section 3.4.

3.2 NOTATION AND ASSUMPTIONS

Suppose there are two induction treatments A_j, $j = 1, 2$, two maintenance treatments for responders B_k, $k = 1, 2$, and two alternative treatments for non responders C_l, $l = 1, 2$.

Let us define Y to be the outcome variable of interest. We will treat Y as being continuous; however, the methodology can be applied analogously to binary responses. We will define the estimands in terms of patient-specific counterfactual variables (Holland [1986]). Let $Y_i(A_jB_k)$ denote the outcome of the individual i if he/she actually received the treatment sequence A_jB_k (Receive A_j, respond, and then receive B_k). Similarly, let $Y_i(A_jC_l)$ be the outcome of the same patient if he/she had received the sequence A_jC_l. Note that under the possible options for induction and maintenance, there are 8 such outcomes possible. In practice, only one of the sequences will be followed by one individual. Therefore, only one of these 8 outcomes will be observed If R_i denotes the response status ($R_i = 1$ for response, 0 for no response) for the patient i in the population, then the outcome Y for the patients i under strategy $A_jB_kC_l$ can be written as

$$Y_i(A_jB_kC_l) = R_iY_i(A_jB_k) + (1 - R_i)Y_i(A_jC_l), j, k, l = 1, 2. \tag{3.1}$$

To represent a generic individual, for simplicity, we will drop i from the variable notation, wherever possible.

Our goal is to estimate

$$\mu_{jkl} = \mathrm{E}\{Y(A_jB_kC_l)\}.$$

Note that conditioning on R, μ_{jkl} can be expressed as

$$\mu_{jkl} = \pi_{R_j}\mu_{A_jB_k} + (1 - \pi_{R_j})\mu_{A_jC_l},$$

where $\mu_{A_jB_k} = \mathrm{E}\{Y(A_jB_k)\}$, $\mu_{A_jC_l} = \mathrm{E}\{Y(A_jC_l)\}$ and π_{R_j} is the probability of response to induction treatment A_j. All the variables $Y(A_jB_k), Y(A_jC_l)$ and $Y(A_jB_kC_l), j, k, l = 1, 2$ are counterfactuals, as they all cannot be observed for the same individual. Regardless of the design of the study, that is, whether URT or SRT, the observed data can be written as $\{I_i(A_j), R_i, R_iI_i(B_k), (1 - R_i)I_i(C_l), Y_i, j, k, l = 1, 2\}, i = 1, 2, ..., n$, where Y_i is the observed

outcome for the i^{th} patient, $I_i(X)$ is the indicator function for treatment X. $I_i(X) = 1$ if the i^{th} patient was assigned to receive treatment X, and 0, otherwise.

3.3 ESTIMATION IN UP-FRONT AND SEQUENTIALLY RANDOMIZED TRIALS

In this section, we construct unbiased estimators for μ_{jkl} under the two randomized designs, URT and SRT, and derive their variances in order to compare their efficiency.

3.3.1 Estimation in Up-front Randomized Trials

In URT, patients are randomized at the beginning into 8 eligible strategies. Therefore, the estimator for the mean outcome μ_{jkl} in URT is given by the sample mean of the outcome for patients randomized to the strategy $A_jB_kC_l$. Specifically,

$$\hat{\mu}_{jkl}^{URT} = \frac{\sum_{i=1}^{n} Y_i I_i(A_jB_kC_l)}{\sum_{i=1}^{n} I_i(A_jB_kC_l)} = \frac{\sum_{i=1}^{n} Y_i I_i(A_jB_kC_l)}{n_{jkl}},$$

where $I_i(A_jB_kC_l) = I_i(A_j)\{R_iI_i(B_k) + (1-R_i)I_i(C_l)\}$ and $n_{jkl} = \sum_{i=1}^{n} I_i(A_jB_kC_l)$. The variance of this estimator is given by

$$\mathrm{Var}(\hat{\mu}_{jkl}^{URT}) = \frac{\mathrm{Var}\{Y(A_jB_kC_l)\}}{n_{jkl}}.$$

Now, by the law of conditioned variance,

$$\mathrm{Var}\{Y(A_jB_kC_l)\} = \mathrm{Var}\big[\mathrm{E}\{Y(A_jB_kC_l)|R\}\big] + \mathrm{E}\big[\mathrm{Var}\{Y(A_jB_kC_l)|R\}\big]. \tag{3.2}$$

First, consider the first part of the right-hand side in Eq. (3.2). Using Eq. (3.1),

$$\begin{aligned}\mathrm{E}\{Y(A_jB_kC_l)|R\} =& R\mathrm{E}\{Y(A_jB_k)|R\} + (1-R)\mathrm{E}\{Y(A_jC_l)|R\} \\ =& R\mu_{A_jB_k} + (1-R)\mu_{A_jC_l}.\end{aligned}$$

Note that $\mathrm{E}[\mathrm{E}\{Y(A_jB_kC_l)|R\}] = \pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}$. Then, its variance would be

$$\begin{aligned}
\mathrm{Var}\big[\mathrm{E}\{Y(A_jB_kC_l)|R\}\big] =& \mathrm{E}\big[\mathrm{E}\{Y(A_jB_kC_l)|R\}^2\big] - \{\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\}^2 \\
=& \mathrm{E}\big[\{R\mathrm{E}(Y(A_jB_k)|R) + (1-R)\mathrm{E}(Y(A_jC_l)|R)\}^2\big] \\
& - \{\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\}^2 \\
=& \mathrm{E}\{R\mu^2_{A_jB_k} + (1-R)\mu^2_{A_jC_l}\} - \{\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\}^2 \\
=& \pi\mu^2_{A_jB_k} + (1-\pi)\mu^2_{A_jC_l} - \{\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\}^2. \qquad (3.3)
\end{aligned}$$

Similarly for the second term,

$$\begin{aligned}
\mathrm{Var}\{Y(A_jB_kC_l)|R\} =& \mathrm{Var}\{RY(A_jB_k) + (1-R)Y(A_jC_l)|R\} \\
=& \mathrm{E}\{(RY(A_jB_k) + (1-R)Y(A_jC_l))^2|R\} \\
& - [R\mathrm{E}\{Y(A_jB_k)|R\} + (1-R)\mathrm{E}\{Y(A_jC_l)|R\}]^2 \\
=& R\mathrm{E}\{Y(A_jB_k)^2|R\} + (1-R)\mathrm{E}\{Y(A_jC_l)^2|R\} \\
& - R[\mathrm{E}\{Y(A_jB_k)\}]^2 - (1-R)[\mathrm{E}\{Y(A_jC_l)\}]^2. \\
=& R(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) + (1-R)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l}) \\
& - R\mu^2_{A_jB_k} - (1-R)\mu^2_{A_jC_l},
\end{aligned}$$

whose expected value is given by

$$\begin{aligned}
\mathrm{E}\big[\mathrm{Var}\{Y(A_jB_kC_l)|R\}\big] =& \pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) + (1-\pi)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l}) \\
& - \pi\mu^2_{A_jB_k} - (1-\pi)\mu^2_{A_jC_l}. \qquad (3.4)
\end{aligned}$$

Then, by combining Eq. (3.3) and Eq. (3.4), the variance of $\hat{\mu}^{URT}_{jkl}$ is

$$\begin{aligned}
\mathrm{Var}(\hat{\mu}^{URT}_{jkl}) =& \frac{1}{n_{jkl}}\Big\{\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) + (1-\pi)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l}) \\
& - \big(\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\big)^2\Big\}. \qquad (3.5)
\end{aligned}$$

When randomization is done with equal probability to 8 different strategies, then $n_{jkl} \approx n/8$ and the variance in Eq. (3.5) can be written as

$$\frac{8}{n}\left\{\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) + (1-\pi)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l}) - \big(\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\big)^2\right\}. \qquad (3.6)$$

The variance of the estimator $\hat{\mu}^{URT}_{jkl}$ can be estimated by the sample variance for individual groups.

3.3.2 Estimation with Inverse-probability-weighting in Sequentially Randomized Trials

SRT randomizes patients first into one of the two A treatments (A_1, A_2), then depending on the response, further randomizes to B or C treatments. Responders are randomized to B_1 or B_2 while non-responders to C_1 or C_2. Consider only patients who received A_j (j=1 or 2). Using the idea of inverse-probability-weighting (IPW) (Robins et al. [1994]), an unbiased estimator for μ_{jkl} can be expressed as the average of the weighted outcome of responders and non-responders as follows

$$\hat{\mu}_{jkl}^{SRT} = \frac{1}{n_j} \sum_{i=1}^{n_j} \left\{ \frac{R_i I_i(B_k)}{P_k} + \frac{(1-R_i) I_i(C_l)}{Q_l} \right\} Y_i, \tag{3.7}$$

where P_k and Q_l are the probability of the responders receiving B_k and non-responders receiving C_l, respectively, and n_j is the number of patients assigned to A_j at the entry, i.e., $n_j = \sum_{i=1}^{n} I_i(A_j)$. The fact that $\hat{\mu}_{jkl}^{SRT}$ is unbiased can be shown by using the consistency assumption (Cole [2009]). In terms of the counterfactuals and the observed data, the consistency assumption can be written as

$$Y_i = \sum_{i=1}^{n} I_i(A_j) \left\{ R_i \sum_{k=1}^{n} I_i(B_k) Y_i(A_j B_k) + (1-R_i) \sum_{l=1}^{n} I_i(C_l) Y_i(A_j C_l) \right\}. \tag{3.8}$$

The consistency assumption states that a patient's counterfactual outcome under a given strategy equals the observed outcome under treatment assignment consistent to the same strategy. Now, under Eq. (3.8), Eq. (3.7) can be written as

$$\hat{\mu}_{jkl}^{SRT} = \frac{1}{n_j} \sum_{i=1}^{n_j} \left\{ \frac{R_i I_i(B_k)}{P_k} Y_i(A_j B_k) + \frac{(1-R_i) I_i(C_l)}{Q_l} Y_i(A_j C_l) \right\}.$$

Unbiasedness of this estimator can be shown by using the sequential randomization assumption as follows

$$\begin{aligned}
\mathrm{E}[\hat{\mu}_{jkl}^{SRT}] =& \frac{1}{n_j}\sum_{i=1}^{n_j} E\left[\frac{R_i I_i(B_k)}{P_k} Y_i(A_jB_k) + \frac{(1-R_i)I_i(C_l)}{Q_l} Y_i(A_jC_l)\right] \\
=& \frac{1}{n_j}\sum_{i=1}^{n_j}\left[\frac{1}{P_k} E\left\{R_i Y_i(A_jB_k)\mathrm{E}[I(B_k)|R_i, Y_i(A_jB_k)]\right\}\right. \\
& \left. + \frac{1}{Q_l} E\left\{(1-R_i)Y_i(A_jC_l)\mathrm{E}[I(C_l)|R_i, Y_i(A_jC_l)]\right\}\right] \\
=& \frac{1}{n_j}\sum_{i=1}^{n_j} \frac{1}{P_k} E\left[R_i Y_i(A_jB_k)P_k\right] + \frac{1}{Q_l} E\left[(1-R_i)Y_i(A_jC_l)Q_l\right] \\
=& \frac{1}{n_j}\sum_{i=1}^{n_j} \mathrm{E}[R_i Y_i(A_jB_k) + (1-R_i)Y_i(A_jC_l)] \\
=& \frac{1}{n_j}\sum_{i=1}^{n_j} \mathrm{E}[Y_i(A_jB_kC_l)] \\
=& \mu_{jkl},
\end{aligned}$$

where the sequential randomization assumption was used from going to the third line from the second, which states that the probability of receiving a treatment given the history of the observed data does not depend on counterfactuals. In the case of receiving treatment B_k, this assumption can be written as $I_i(B_k) \perp \{Y_i(A_jB_k), Y_i(A_jC_l)\}|R_i$.

The variance of $\hat{\mu}_{jkl}^{SRT}$ is obtained in a similar manner as $\mathrm{Var}(\hat{\mu}_{jkl}^{URT})$. Let $\hat{\mu}_{jkl}^{SRT} = (1/n_j)\sum_{i=1}^{n_j} H_i$ where $H_i = [\{Y_i(A_jB_k)R_iI_i(B_k)\}/P_k + \{Y_i(A_jC_l)(1-R_i)I_i(C_l)\}/Q_l]$. Then the variance of $\hat{\mu}_{jkl}^{SRT}$ can be expressed as

$$\mathrm{Var}(\hat{\mu}_{jkl}^{SRT}) = \frac{\mathrm{Var}(H)}{n_j},$$

where $H = \sum_{i=1}^{n} H_i$.
Now,

$$\mathrm{Var}(H) = \mathrm{E}\left[\mathrm{Var}\{H|R, Y(A_jB_k), Y(A_jC_l)\}\right] + \mathrm{Var}\left[\mathrm{E}\{H|R, Y(A_jB_k), Y(A_jC_l)\}\right]. \quad (3.9)$$

Starting with the conditional variance,

$$\mathrm{Var}(H|R, Y(A_jB_k), Y(A_jC_l)) = R(Y(A_jB_k))^2\frac{(1-P_k)}{P_k} + (1-R)(Y(A_jC_l))^2\frac{(1-Q_l)}{Q_l}.$$

The expected value of the conditional variance

$$
\begin{aligned}
&\mathrm{E}\{\mathrm{Var}(H|R, Y(A_jB_k), Y(A_jC_l)\} \\
=&\mathrm{E}\left\{R(Y(A_jB_k)^2\frac{(1-P_k)}{P_k} + (1-R)(Y(A_jC_l))^2\frac{(1-Q_l)}{Q_l}\right\} \\
=&\mathrm{E}\left[\mathrm{E}\left\{R\{Y(A_jB_k)\}^2\frac{(1-P_k)}{P_k} + (1-R)\{Y(A_jC_l)\}^2\frac{(1-Q_l)}{Q_l}|R\right\}\right] \\
=&\mathrm{E}\left\{R(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k})\frac{(1-P_k)}{P_k} + (1-R)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l})\frac{(1-Q_l)}{Q_l}\right\} \\
=&\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k})\frac{(1-P_k)}{P_k} + (1-\pi)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l})\frac{(1-Q_l)}{Q_l}. \qquad (3.10)
\end{aligned}
$$

For the second part of the right hand side in Eq. (3.9),

$$
\begin{aligned}
\mathrm{E}(H|R, Y(A_jB_k), Y(A_jC_l)) =&\left[\frac{R}{P_k}Y(A_jB_k)\mathrm{E}\{I(B_k)|R, Y(A_jB_k)\}\right. \\
&\left. + \frac{(1-R)}{Q_l}Y(A_jC_l)\mathrm{E}\{I(C_l)|R, Y(A_jC_l)\}\right] \\
=&RY(A_jB_k) + (1-R)Y(A_jC_l) \\
=&Y(A_jB_kC_l),
\end{aligned}
$$

the variance of which is

$$
\begin{aligned}
\mathrm{Var}[\mathrm{E}\{H|R, Y(A_jB_k), Y(A_jC_l)\}] = &\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) + (1-\pi)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l}) \\
&- \left(\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\right)^2. \qquad (3.11)
\end{aligned}
$$

By combining Eq. (3.10) and Eq. (3.11) together, the overall variance of $\hat{\mu}^{SRT}_{jkl}$ is

$$
\begin{aligned}
\mathrm{Var}(\hat{\mu}^{SRT}_{jkl}) =&\frac{1}{n_j}\left[\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k})\frac{(1-P_k)}{P_k}\right. \\
&+ (1-\pi)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l})\frac{(1-Q_l)}{Q_l} \\
&+ \pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) + (1-\pi)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l}) \\
&\left. - \left\{\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\right\}^2\right] \\
=&\frac{1}{n_j}\left[\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k})\frac{1}{P_k} + (1-\pi)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l})\frac{1}{Q_l}\right. \\
&\left. - \left\{\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\right\}^2\right]. \qquad (3.12)
\end{aligned}
$$

For the equal probability of the maintenance treatment (i.e. $P_k = Q_l = 1/2$), the variance of the estimator is

$$\begin{aligned}\mathrm{Var}(\hat{\mu}_{jkl}^{SRT}) =& \frac{1}{n_j}\Big[2\left\{\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) + (1-\pi)(\sigma^2_{A_jC_l} + \mu^2_{A_jC_l})\right\} \\ & - \left\{\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\right\}^2\Big]\end{aligned}$$

3.3.3 Estimation with Probability-adjusted-weights in Sequentially Randomized Trials

Another estimator commonly used in SRT is the probability-adjusted inverse-probability-weighted estimator (Lunceford et al. [2002]). We will call this estimator as the probability-adjusted estimator, thereafter. The probability-adjusted estimator is expressed as

$$\begin{aligned}\hat{\mu}_{jkl}^{SRTPA} =& \left[\sum_{i=1}^{n_j}\left\{\frac{R_iI_i(B_k)}{P_k} + \frac{(1-R_i)I_i(C_l)}{Q_l}\right\}\right]^{-1}\sum_{i=1}^{n_j}\left\{\frac{R_iI_i(B_k)}{P_k} + \frac{(1-R_i)I_i(C_l)}{Q_l}\right\}Y_i \\ =& \sum_{i=1}^{n_j} W_{ki}Y_i + \sum_{i=1}^{n_j} W_{li}Y_i \\ =& G_{nk} + G_{nl},\end{aligned}$$

where

$$W_{ki} = \frac{Q_lR_iI_i(B_k)}{Q_l\sum_{i=1}^{n_j}R_iI_i(B_k) + P_k\sum_{i=1}^{n_j}(1-R_i)I(C_l)},$$

and

$$W_{li} = \frac{P_k(1-R_i)I_i(C_l)}{P_k\sum_{i=1}^{n_j}R_iI_i(C_l) + Q_l\sum_{i=1}^{n_j}(1-R_i)I(B_k)}.$$

Note that

$$\mathrm{E}\{W_{ki}|R_i, Y(A_jB_k)\} \approx \frac{Q_lR_iP_k}{Q_lP_k\sum_{i=1}^{n_j}R_i + Q_lP_k\sum_{i=1}^{n_j}(1-R_i)} \approx \frac{R_i}{n_j}.$$

Therefore, $\mathrm{E}(W_{ki}) \approx \pi/n_j$ and similarly, $\mathrm{E}(W_{li}) \approx (1-\pi)/n_j$.

Unbiasedness of this estimator is shown as follows. We can write

$$\mathrm{E}(\hat{\mu}_{jkl}^{SRTPA}) = \mathrm{E}(G_{nk}) + \mathrm{E}(G_{nl}). \tag{3.13}$$

For the first part of Eq. (3.13),

$$\begin{aligned}
\mathrm{E}(G_{nk}) =& \mathrm{E}\left(\sum_{i=1}^{n_j} W_{ki} Y_i\right) \text{ (By consistency assumption)} \\
=& \mathrm{E}\left\{\sum_{i=1}^{n_j} W_{ki} Y_i(A_j B_k)\right\} \\
=& \mathrm{E}\left[\sum_{i=1}^{n_j} Y_i(A_j B_k) \mathrm{E}\{W_{ki} | R_i, Y_i(A_j B_k)\}\right] \\
\approx& \mathrm{E}\left\{\sum_{i=1}^{n_j} \frac{R_i}{n_j} Y_i(A_j B_k)\right\} \\
=& \frac{1}{n_j} \sum_{i=1}^{n_j} \mathrm{E}\left[R_i \mathrm{E}\{Y_i(A_j B_k) | R_i\}\right] \\
=& \frac{1}{n_j} \sum_{i=1}^{n_j} \pi \mu_{A_j B_k} \\
=& \pi \mu_{A_j B_k}.
\end{aligned}$$

Similarly, $\mathrm{E}(G_{nl}) = (1-\pi)\mu_{A_j C_l}$. Then we have

$$\mathrm{E}(\hat{\mu}_{jkl}^{SRTPA}) = \pi \mu_{A_j B_k} + (1-\pi)\mu_{A_j C_l} = \mu_{A_j B_k C_l}.$$

To derive the variance of $\hat{\mu}_{jkl}^{SRTPA}$, we note that

$$\mathrm{Var}(\hat{\mu}_{jkl}^{SRTPA}) = \mathrm{Var}(G_{nk}) + \mathrm{Var}(G_{nl}) + 2\mathrm{cov}(G_{nk}, G_{nl}). \tag{3.14}$$

Consider the first part of the Eq. (3.14) first,

$$\begin{aligned}
\mathrm{Var}(G_{nk}) =& \mathrm{Var}\left(\sum_{i=1}^{n_j} W_{ki} Y_i\right) \\
=& \mathrm{Var}\left\{\sum_{i=1}^{n_j} W_{ki} Y_i(A_j B_k)\right\} \\
=& \sum_{i=1}^{n_j} \mathrm{Var}\left\{W_{ki} Y_i(A_j B_k)\right\}.
\end{aligned}$$

Then,

$$\begin{aligned}\mathrm{Var}\{W_{ki}Y_i(A_jB_k)\} =& \mathrm{Var}[\mathrm{E}\{W_{ki}Y(A_jB_k)|R_i, Y_i(A_jB_k)\}] \\ &+ \mathrm{E}[\mathrm{Var}\{W_{ki}Y_i(A_jB_k)|R_i, Y_i(A_jB_k)\}] \\ \approx& \mathrm{E}\left\{\left(\frac{R_i}{n_j}Y_i(A_jB_k)\right)^2\right\} - \mathrm{E}\left\{\frac{R_i}{n_j}Y_i(A_jB_k)\right\}^2 \\ &+ \mathrm{E}[Y_i^2(A_jB_k)\mathrm{Var}\{W_{ki}|R_i, Y_i(A_jB_k)\}].\end{aligned}$$

The conditional variance of W_{ki} can be approximated as

$$\begin{aligned}\mathrm{Var}\{W_{ki}|R_i, Y_i(A_jB_k)\} &\approx \mathrm{E}\{W_{ki}^2|R_i, Y_i(A_jB_k)\} - \frac{R_i}{n_j^2} \\ &\approx \mathrm{E}\left[\frac{R_iI_i(B_k)Q_l^2/n_j^2}{\left\{Q_l\frac{\sum R_iI_i(B_k)}{n_j} + P_k\frac{\sum(1-R_i)I_i(C_l)}{n_j}\right\}^2}\right] - \frac{R_i}{n_j^2} \\ &\approx \mathrm{E}\left[\frac{R_iI_i(B_k)Q_l^2/n_j^2}{\{Q_l\pi P_k + P_k(1-\pi)Q_l\}^2}\right] - \frac{R_i}{n_j^2} \\ &\approx \frac{R_i}{n_j^2}\frac{1}{P_k} - \frac{R_i}{n_j^2} \\ &\approx \frac{R_i}{n_j^2}\left(\frac{1-P_k}{P_k}\right).\end{aligned}$$

Therefore,

$$\begin{aligned}\mathrm{Var}\{W_{ki}Y(A_jB_k)\} \approx& \frac{1}{n_j^2}\left\{\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) - \pi^2\mu^2_{A_jB_k}\right\} \\ &+ \mathrm{E}\left\{Y_i^2(A_jB_k)\frac{R_i}{n_j^2}\frac{1-P_k}{P_k}\right\} \\ \approx& \frac{1}{n_j^2}\left\{\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) - \pi^2\mu^2_{A_jB_k}\right\} \\ &+ \frac{1}{n_j^2}\left\{\left(\frac{1-P_k}{P_k}\right)\pi(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k})\right\} \\ =& \frac{\pi}{n_j^2}\left\{\left(1 + \frac{1-P_k}{P_k}\right)(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) - \pi\mu^2_{A_jB_k}\right\} \\ =& \frac{\pi}{n_j^2}\left\{\frac{1}{P_k}(\sigma^2_{A_jB_k} + \mu^2_{A_jB_k}) - \pi\mu^2_{A_jB_k}\right\}.\end{aligned}$$

Similarly,

$$\mathrm{Var}\{W_{li}Y(A_jC_l)\} \approx \frac{1-\pi}{n_j^2}\left\{\frac{1}{Q_l}(\sigma^2_{A_jC_l}+\mu^2_{A_jC_l}) - (1-\pi)\mu^2_{A_jC_l}\right\}.$$

For the covariance term,

$$\begin{aligned}
\mathrm{Cov}(G_{nk}, G_{nl}) =& \mathrm{E}(G_{nk}, G_{nl}) - \mathrm{E}(G_{nk})\mathrm{E}(G_{nl}) \\
=& \mathrm{E}\sum_{i=1}^{n_j}\sum_{j=i}^{n_j}(W_{ki}W_{lj}Y_iY_j) - \pi\mu_{A_jB_k}(1-\pi)\mu_{A_jC_l} \\
=& \frac{n_j(n_j-1)}{n_j^2}\{\pi\mu_{A_jB_k}(1-\pi)\mu_{A_jC_l}\} - \pi\mu_{A_jB_k}(1-\pi)\mu_{A_jC_l} \\
=& -\frac{1}{n_j}\pi\mu_{A_jB_k}(1-\pi)\mu_{A_jC_l}.
\end{aligned}$$

Therefore, the variance for probability-adjusted SRT is

$$\begin{aligned}
\mathrm{Var}(\hat{\mu}^{SRTPA}_{jkl}) \approx& \frac{1}{n_j}\left\{\frac{\pi}{P_k}(\sigma^2_{A_jB_k}+\mu^2_{A_jB_k}) + \frac{(1-\pi)}{Q_l}(\sigma^2_{A_jC_l}+\mu^2_{A_jC_l})\right. \\
&\left. - \left(\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\right)^2\right\}. \qquad (3.15)
\end{aligned}$$

For the equal probability of the maintenance treatment, the variance of the estimator is

$$\begin{aligned}
\mathrm{Var}(\hat{\mu}^{SRTPA}_{jkl}) \approx& \frac{1}{n_j}\left\{2\left(\pi(\sigma^2_{A_jB_k}+\mu^2_{A_jB_k}) + (1-\pi)(\sigma^2_{A_jC_l}+\mu^2_{A_jC_l})\right)\right. \\
&\left. - \left(\pi\mu_{A_jB_k} + (1-\pi)\mu_{A_jC_l}\right)^2\right\}.
\end{aligned}$$

From Eq. (3.12) and Eq. (3.15), we see that, for sufficiently large n, the variance of the IPW and probability-adjusted estimators have the same variance. Therefore, when comparing SRT to URT, we will use the probability-adjusted estimator. The variance in Eq. (3.15) can be estimated by replacing the parameters with the corresponding sample estimates. For example,

$$\hat{\sigma}^2_{A_jB_k} = \frac{\sum_{i=1}^{n_j} I(B_k)\left\{Y - \frac{\sum_{i=1}^{n_j} I(B_k)Y}{\sum_{i=1}^{n_j} I(B_k)}\right\}}{\sum_{i=1}^{n_j} I(B_k) - 1}.$$

An analogous estimator can be used for $\hat{\sigma}^2_{A_jC_l}$. Instead of estimating sample variances individually, we use the fact that estimators ($\hat{\mu}^{SRT}_{jkl}$ and $\hat{\mu}^{SRTPA}_{jkl}$) are the solutions to the estimating equations

$$\sum_{i=1}^{n} \psi_i(\mu^{SRT}_{jkl}) = \sum_{i=1}^{n_j} \left[\left\{ \frac{R_i I_i(B_k)}{P_k} + \frac{(1-R_i) I_i(C_{kl})}{Q_l} \right\} Y_i - \mu^{SRT}_{jkl} \right] = 0,$$

and

$$\sum_{i=1}^{n} \psi_i(\mu^{SRTPA}_{jkl}) = \sum_{i=1}^{n_j} \left[\left\{ \frac{R_i I_i(B_k)}{P_k} + \frac{(1-R_i) I_i(C_{kl})}{Q_l} \right\} (Y_i - \mu^{SRTPA}_{jkl}) \right] = 0,$$

respectively. Then, the variances could be estimated by so-called sandwich variance estimator.

$$var(\hat{\mu}^{SRT}_{jkl}) = \left(\frac{1}{n_j} \right) \frac{B(\hat{\mu}^{SRT}_{jkl})}{A(\hat{\mu}^{SRT}_{jkl})^2}$$

,

where $A_n(x)$ is the empirical estimate of $A(x) = -\mathrm{E}\left[\frac{\delta}{\delta \mu} \{\psi_i(\mu)\} \right] |_{\mu=x} = 1$ and $B_n(x) = \frac{\sum \psi_i^2(x)}{n_j}$. Similarly, $\mathrm{Var}(\hat{\mu}^{SRTPA}_{jkl})$ also can be estimated.

The covariance of the two estimators can be estimated in a similar manner. For example, for $\hat{\mu}_{111}$ and $\hat{\mu}_{112}$, the covariance is

$$cov(\hat{\mu}^{SRT}_{111}, \hat{\mu}^{SRT}_{112}) = n_1^{-2} \sum \psi_i(\mu^{SRT}_{111}) \psi_i(\mu^{SRT}_{112}).$$

3.3.4 Hypothesis Testing

To test hypotheses regarding the strategy-specific means, we construct test statistics based on Wald-type tests. For example, to test the difference among μ_{111}, μ_{112}, μ_{121} and μ_{122}, constructed test statistic as follows :

$$\hat{\mu} = \begin{bmatrix} \hat{\mu}_{111} & \hat{\mu}_{112} & \hat{\mu}_{121} & \hat{\mu}_{122} \end{bmatrix}$$

$$\underset{\sim}{C} = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 1 & -1 & 0 & 0 \end{bmatrix}$$

$$\underset{\sim}{\Sigma} = \begin{bmatrix} var(\hat{\mu}_{111}) & cov(\hat{\mu}_{111}, \hat{\mu}_{112}) & cov(\hat{\mu}_{111}, \hat{\mu}_{121}) & cov(\hat{\mu}_{112}, \hat{\mu}_{122}) \\ cov(\hat{\mu}_{112}, \hat{\mu}_{111}) & var(\hat{\mu}_{112}) & cov(\hat{\mu}_{112}, \hat{\mu}_{121}) & cov(\hat{\mu}_{112}, \hat{\mu}_{122}) \\ cov(\hat{\mu}_{121}, \hat{\mu}_{111}) & cov(\hat{\mu}_{121}, \hat{\mu}_{112}) & var(\hat{\mu}_{121}) & cov(\hat{\mu}_{121}\hat{\mu}_{122}) \\ cov(\hat{\mu}_{122}, \hat{\mu}_{111}) & cov(\hat{\mu}_{122}, \hat{\mu}_{112}) & cov(\hat{\mu}_{122}, \hat{\mu}_{121}) & var(\hat{\mu}_{122}) \end{bmatrix}$$

$$Z^2 = \hat{\mu}^T \times \underset{\sim}{C}^T \times [\underset{\sim}{C} \times \underset{\sim}{\Sigma} \times \underset{\sim}{C}^T]^{-1} \times \underset{\sim}{C} \times \hat{\mu},$$

where T stands for transpose of a matrix. This statistic can be approximated by a χ^2 distribution with 3 degree of freedom under the null hypothesis of no difference. Similarly, test statistics for other hypotheses testing can be constructed.

3.4 SIMULATION STUDY

The true population in the simulation can be described in terms of the distribution of counterfactual variables. For each individual i in the population, assume that $Y_i(A_jB_k)$ and $Y_i(A_jC_l)$ are individually normally distributed with mean $\mu_{A_jB_k}$ and variance $\sigma^2_{A_jB_k}$, and mean $\mu_{A_jC_l}$ and variance $\sigma^2_{A_jC_l}$ respectively, $j, k, l = 1, 2$. The response indicator R_i was assumed to follow a Bernoulli distribution with probability of success π_{R_1} for the initial treatment A_1 and π_{R_2} for A_2.

We repeatedly (1000) sampled data from this population by using two forms of designs - URT and SRT. In the URT design, n individuals are randomized with equal probability to 8 different strategies $A_1B_1C_1, A_1B_1C_2, \ldots, A_2B_2C_2$. In the SRT, we randomized n patients first to two induction treatments A_1 and A_2 with probabilities π_{A_1} and π_{A_2} respectively (π_{A_1} + $\pi_{A_2} = 1$). For simplicity, we use $\pi_A = \pi_{A_1} = \pi_{A_2} = 1/2$, throughout the study. In both designs, the observed outcome was generated from the counterfactual distributions using the formula (3.8). The response status R_i for the i^{th} patient was generated according to a Bernoulli distribution with probability of success π. When R_i=1, we randomly assigned the patient to $B_1(B_2)$ with probability $P_1(P_2)$ ($P_1 + P_2 = 1$). Similarly, patients with $R_i = 0$ where randomized to $C_1(C_2)$ with probability $Q_1(Q_2)$ ($Q_1 + Q_2 = 1$).

We considered many different simulation scenarios by varying the values of population parameters $\mu_{A_jB_k}$, $\sigma^2_{A_jB_k}$, $\mu_{A_jC_l}$, $\sigma^2_{A_jC_l}$, π_{R_1}, π_{R_2}, P_k and Q_l, $j, k, l = 1, 2$. On the other hand, the first treatment assignment probability, π_A is set to 1/2 for all scenarios. Therefore, number of samples for both first treatments A_1 and A_2 are equal.

The first four scenarios represented here assumes that $\mu_{A_jB_k}$=$\mu_{A_jC_l} = 15$, $j, k, l = 1, 2$, and $\sigma^2_{A_jB_k} = 6$, $j, k = 1, 2$, and $\sigma^2_{A_jC_l} = 8$, $j, l = 1, 2$. In these situations, the eight strategy means are all equal to 15, i.e., $\mu_{A_1B_1C_1} = \mu_{A_1B_1C_2} = \mu_{A_1B_2C_1} = \mu_{A_1B_2C_2} = \mu_{A_2B_1C_1} = \mu_{A_2B_1C_2}$ $= \mu_{A_2B_2C_1} = \mu_{A_2B_2C_2}$=15. In our first simulation scenario, we assumed $\pi_{R_1} = \pi_{R_2} = P_1 = Q_1 = 1/2$. Then π_{R_1} and π_{R_2} are adjusted to see the change for other scenarios.

Table 1 summarizes the results of the simulation under designs from scenario 1-4, where all the strategy means are identical. The table lists the estimated strategy-specific means (Est.), Monte Carlo standard error (SE), and the relative bias (RB %). It also provides the coverage probability (CP) of 95% confidence interval based on normality assumptions. In addition, the table provides type I error rates for testing three hypotheses, namely H_1 : strategy means are identical for strategies sharing the same first stage treatment A_1 ($\mu_{A_1B_1C_1}$ $= \mu_{A_1B_1C_2} = \mu_{A_1B_2C_1} = \mu_{A_1B_2C_2}$), H_2 : strategy means are identical for strategies sharing the same first stage treatment A_2 ($\mu_{A_2B_1C_1} = \mu_{A_2B_1C_2} = \mu_{A_2B_2C_1} = \mu_{A_2B_2C_2}$) and H_3 : strategy means are identical for all strategies (H_3 : $\mu_{A_1B_1C_1} = \mu_{A_1B_1C_2} = \mu_{A_1B_2C_1} = \mu_{A_1B_2C_2} = \mu_{A_2B_1C_1}$ $= \mu_{A_2B_1C_2} = \mu_{A_2B_2C_1} = \mu_{A_2B_2C_2}$).

From the results in Table 1, we see that both URT and SRT produce unbiased estima-

tors of strategy means under the same sample size. Absolute relative biases were less than 0.7% for URT estimator and less than 0.5% for probability-adjusted estimator from the SRT. Standard errors of estimators are generally lager for the URT estimators as compared to the probability-adjusted estimator. In most cases, URT estimator fails to meet nominal coverage level for the 95% Wald confidential interval. On the other hand, probability-adjusted estimator provides coverage probabilities close to the nominal level. In most cases, rejection rates for the true null hypothesis match the nominal type I error rate of 5% for the probability-adjusted estimator in the SRT design, whereas for URT, the observed type I error rate were always larger than the nominal level. This was true irrespectively of the hypotheses being tested. And adjusting response rate does not change the results.

In the next four scenarios, we assume that $\mu_{A_jB_1} = 15$, $\mu_{A_jC_1} = 20$, $\mu_{A_jB_2} = 22$, $\mu_{A_jC_2} = 15$, j=1, 2, and $\sigma^2_{A_jB_k}$=6, j, k=1, 2, and $\sigma^2_{A_jC_l}$=8, j, l=1, 2 to test alternative hypothesis that all strategies have different means. In Tables 2(a) and 2(b), we assume that $\pi_{R_1} = \pi_{R_2} = P_1 = Q_1 = 1/2$ with sample sizes 150 and 200, respectively. In the next two scenarios (Tables 2(c) and 2(d)), response rates, π_{R_1} and π_{R_2} are adjusted to 0.1, 0.5 and 0.8 with $n = 200$.

Table 2 provides the powers for testing hypotheses to detect mean difference in addition to estimated strategy-specific means (Est.), MC standard error (SE), the relative bias (RB %) and the coverage probability (CP) of 95% confidence interval. Here, we use the same three hypotheses defined in Table 1. As can be seen from Table 2, both estimators are approximately unbiased. Relative biases were less than 0.8% for URT estimator and less than 0.6% for probability-adjusted estimator from the SRT. Standard errors of probability-adjusted estimator in SRT are smaller than the estimator in URT. And the probability-adjusted estimator in SRT achieves the nominal level of coverage probabilities and almost over 95% of power of the hypothesis to detect mean difference while coverage probabilities of the estimator in URT are less than the nominal level and its power is about 70%. Similar to the Table 1, adjusting response rate does not change the results. We note that both show better performance, such as smaller relative bias, smaller standard errors, smaller power and higher coverage probabilities with larger sample size.

Table 3 shows how the proportions of treatment assignment of B_k or C_l affect the estimates. Small probability of receiving B_k (i.e., $P_k \leq 0.5$) makes the variance of the $\hat{\mu}_{j1k}$ larger

and large proportion of receiving B_k (i.e., $P_k \geq 0.5$) makes the variance of $\hat{\mu}_{j2k}$ larger. This is analogous to the Q_l. From this scenario, we note that the probability-adjusted estimators in SRT for B_k have smaller standard errors when $P_k > 0.5$ during the simulation study. This applied to the C_l with Q_l similarly.

Overall, the simulation studies showed that the both estimators are approximately unbiased. The performance of the probability-adjust estimator in SRT was better than the estimator in URT in terms of unbiasedness, efficiency and power.

3.5 DISCUSSION

In this chapter, we have compared the estimators in up-front and sequential randomization trials using a statistical inferential approach. The estimator in URT can be obtained directly. For SRT, we employed the IPW estimator (Robins et al. [1994]) and the probability-adjusted estimator (Lunceford et al. [2002]). Using the sequential randomization assumption, we showed unbiasedness of estimators. Our simulation results indicated that all estimators are approximately unbiased. However, the probability-adjusted estimator in SRT has smaller standard errors and higher power than the estimator in URT. During this study, we also noted that the efficiency of the estimators could be related to the maintenance treatment proportion. Because the assigned sample size for a specific treatment strategy is affected by this proportion. In conclusion, the probability-adjusted estimator in sequential randomized trial performed better than the estimator in up-front randomized trial in terms of efficiency of the estimators and the power of statistical hypothesis testing.

Table 1: Simulation results of estimation with the same means

(a) $n = 150$, $\pi_A = 0.5$, $\pi_{R1} = 0.5$,$\pi_{R2} = 0.5$, $P_k = 0.5$, $Q_l = 0.5$

		Estimator in URT					Probability-adjusted estimator in SRT				
Policy	True	Est.	SE	RB(%)	CP	Type I	Est.	SE	RB(%)	CP	Type I
$A_1B_1C_1$	15.0	15.0	1.42	−0.1	0.89		15.0	1.14	0.0	0.94	
$A_1B_1C_2$	15.0	14.9	1.42	0.4	0.89	0.074^1	15.1	1.14	−0.3	0.94	0.065^1
$A_1B_2C_1$	15.0	15.0	1.41	0.3	0.89		15.0	1.14	0.3	0.93	
$A_1B_2C_2$	15.0	15.0	1.41	0.0	0.90		15.0	1.14	−0.1	0.94	
$A_2B_1C_1$	15.0	15.0	1.76	0.5	0.89		15.0	1.13	0.2	0.93	
$A_2B_1C_2$	15.0	15.0	1.76	0.4	0.89	0.073^2	15.0	1.14	0.3	0.94	0.069^2
$A_2B_2C_1$	15.0	15.0	1.77	−0.1	0.89		15.0	1.13	−0.2	0.94	
$A_2B_2C_2$	15.0	15.0	1.76	0.2	0.88		15.0	1.14	−0.0	0.94	
						0.057^3					0.053^3

(b) $n = 150$, $\pi_A = 0.5$, $\pi_{R1} = 0.1$,$\pi_{R2} = 0.5$, $P_k = 0.5$, $Q_l = 0.5$

$A_1B_1C_1$	15.0	15.0	1.84	−0.2	0.92		15.0	1.27	−0.3	0.94	
$A_1B_1C_2$	15.0	15.0	1.82	−0.1	0.92	0.090^1	15.0	1.26	0.3	0.95	0.042^1
$A_1B_2C_1$	15.0	15.0	1.85	−0.0	0.93		15.0	1.27	−0.1	0.94	
$A_1B_2C_2$	15.0	14.9	1.85	0.7	0.94		14.9	1.26	0.5	0.94	
$A_2B_1C_1$	15.0	14.9	1.77	0.4	0.93		15.0	1.14	−0.1	0.93	
$A_2B_1C_2$	15.0	15.1	1.74	−0.4	0.92	0.071^2	15.0	1.13	−0.0	0.94	0.077^2
$A_2B_2C_1$	15.0	15.1	1.78	−0.5	0.93		15.0	1.14	0.1	0.93	
$A_2B_2C_2$	15.0	15.0	1.77	−0.2	0.95		15.0	1.13	0.0	0.92	
						0.073^3					0.037^3

(c) $n = 150$, $\pi_A = 0.5$, $\pi_{R1} = 0.5$,$\pi_{R2} = 0.9$, P_k=0.5, $Q_l = 0.5$

$A_1B_1C_1$	15.0	15.0	1.51	−0.1	0.92		15.0	1.14	−0.0	0.93	
$A_1B_1C_2$	15.0	15.1	1.51	−0.9	0.90	0.069^1	15.0	1.14	−0.2	0.94	0.057^1
$A_1B_2C_1$	15.0	15.0	1.51	0.2	0.91		15.0	1.14	0.2	0.92	
$A_1B_2C_2$	15.0	15.0	1.52	0.2	0.92		15.0	1.14	0.1	0.95	
$A_2B_1C_1$	15.0	15.0	1.90	−0.1	0.90		15.0	1.01	0.1	0.93	
$A_2B_1C_2$	15.0	15.0	1.89	−0.0	0.90	0.066^2	15.0	1.01	0.0	0.93	0.052^2
$A_2B_2C_1$	15.0	15.0	1.88	−0.0	0.93		15.0	1.00	−0.0	0.95	
$A_2B_2C_2$	15.0	15.0	1.89	0.1	0.90		15.0	1.00	−0.1	0.95	
						0.049^3					0.036^3

(d) $n = 150$, $\pi_A = 0.5$, $\pi_{R1} = 0.2$,$\pi_{R2} = 0.9$, $P_k = 0.5$, $Q_l = 0.5$

$A_1B_1C_1$	15.0	15.0	1.75	0.1	0.93		15.0	1.24	−0.3	0.95	
$A_1B_1C_2$	15.0	15.0	1.77	−0.1	0.93	0.073^1	15.0	1.24	−0.3	0.94	0.056^1
$A_1B_2C_1$	15.0	15.0	1.76	−0.0	0.92		15.0	1.24	−0.2	0.94	
$A_1B_2C_2$	15.0	15.0	1.76	0.2	0.91		15.0	1.24	−0.1	0.9	
$A_2B_1C_1$	15.0	15.0	1.33	0.3	0.94		15.0	1.00	−0.2	0.93	
$A_2B_1C_2$	15.0	15.0	1.32	−0.0	0.92	0.067^2	15.0	1.00	−0.1	0.94	0.060^2
$A_2B_2C_1$	15.0	15.0	1.33	−0.1	0.93		15.0	1.01	−0.2	0.92	
$A_2B_2C_2$	15.0	15.0	1.33	−0.2	0.93		15.0	1.00	−0.1	0.93	
						0.057^3					0.044^3

$^1H_1 : \mu_{A_1B_1C_1} = \mu_{A_1B_1C_2} = \mu_{A_1B_2C_1} = \mu_{A_1B_2C_2}$, $^2H_2 : \mu_{A_2B_1C_1} = \mu_{A_2B_1C_2} = \mu_{A_2B_2C_1} = \mu_{A_2B_2C_2}$,
$^3H_3 : \mu_{A_1B_1C_1} = \mu_{A_1B_1C_2} = \mu_{A_1B_2C_1} = \mu_{A_1B_2C_2} = \mu_{A_2B_1C_1} = \mu_{A_2B_1C_2} = \mu_{A_2B_2C_1} = \mu_{A_2B_2C_2}$

Table 2: Simulation results of estimation with the different means

(a) $n = 150$, $\pi_A = 0.5$, $\pi_{R1} = 0.5$,$\pi_{R2} = 0.5$, $P_k = 0.5$, $Q_l = 0.5$

		Estimator in URT					Probability-adjusted estimator in SRT				
Policy	True	Est.	SE	RB(%)	CP	Power	Est.	SE	RB(%)	CP	Power
$A_1B_1C_1$	17.5	17.5	1.94	0.1	0.88		17.6	1.47	−0.6	0.91	
$A_1B_1C_2$	15.0	15.1	1.84	−0.3	0.91	0.454^1	15.1	1.39	−0.6	0.93	0.783^1
$A_1B_2C_1$	21.0	21.2	1.87	−0.7	0.88		21.0	1.40	−0.1	0.92	
$A_1B_2C_2$	18.5	18.5	2.03	0.1	0.88		18.5	1.39	0.1	0.92	
$A_2B_1C_1$	17.5	17.5	1.70	−0.3	0.89		17.5	1.46	0.2	0.92	
$A_2B_1C_2$	15.0	15.0	1.61	−0.3	0.89	0.445^2	15.0	1.38	−0.1	0.93	0.812^2
$A_2B_2C_1$	21.0	21.1	1.62	−0.3	0.88		21.1	1.40	−0.3	0.92	
$A_2B_2C_2$	18.5	18.6	1.82	−0.4	0.91		18.6	1.54	−0.5	0.90	
						0.573^3					0.940^3

(b) $n = 200$, $\pi_A = 0.5$, $\pi_{R1} = 0.5$,$\pi_{R2} = 0.5$, $P_k = 0.5$, $Q_l = 0.5$

$A_1B_1C_1$	17.5	17.5	1.49	0.1	0.95		17.5	1.05	0.1	0.92	
$A_1B_1C_2$	15.0	15.0	1.41	0.3	0.94	0.720^1	15.0	0.99	0.3	0.94	0.991^1
$A_1B_2C_1$	21.0	21.0	1.42	−0.0	0.94		21.0	0.99	−0.2	0.93	
$A_1B_2C_2$	18.5	18.5	1.57	0.1	0.94		18.5	1.10	−0.1	0.93	
$A_2B_1C_1$	17.5	17.4	1.60	0.7	0.94		17.5	1.05	0.1	0.93	
$A_2B_1C_2$	15.0	15.0	1.51	0.0	0.92	0.706^2	15.0	0.99	0.3	0.94	0.982^2
$A_2B_2C_1$	21.0	20.9	1.52	0.4	0.92		21.0	1.01	0.0	0.95	
$A_2B_2C_2$	18.5	18.5	1.68	0.2	0.93		18.5	1.11	0.1	0.94	
						0.874^3					1.000^3

(c) $n = 200$, $\pi_A = 0.5$, $\pi_{R1} = 0.5$,$\pi_{R2} = 0.8$, $P_k = 0.5$, $Q_l = 0.5$

$A_1B_1C_1$	17.5	17.6	1.41	−0.3	0.91		17.5	1.05	−0.1	0.94	
$A_1B_1C_2$	15.0	15.1	1.35	−0.3	0.92	0.721^1	15.1	0.99	−0.4	0.94	0.977^1
$A_1B_2C_1$	21.0	21.0	1.34	−0.0	0.91		21.0	1.00	0.2	0.93	
$A_1B_2C_2$	18.5	18.4	1.49	0.1	0.92		18.5	1.13	−0.0	0.94	
$A_2B_1C_1$	16.0	16.0	1.33	0.2	0.92		16.0	0.95	−0.1	0.95	
$A_2B_1C_2$	15.0	15.0	1.27	−0.0	0.91	0.965^2	15.0	0.91	−0.0	0.94	0.999^2
$A_2B_2C_1$	21.6	21.6	1.27	−0.0	0.93		21.6	0.91	0.1	0.95	
$A_2B_2C_2$	20.6	20.6	1.38	0.1	0.92		20.6	0.99	0.1	0.95	
						0.976^3					0.999^3

(d) $n = 200$, $\pi_A = 0.5$, $\pi_{R1} = 0.1$,$\pi_{R2} = 0.5$, $P_k = 0.5$, $Q_l = 0.5$

$A_1B_1C_1$	19.5	19.5	1.58	−0.2	0.92		19.5	1.12	0.1	0.94	
$A_1B_1C_2$	15.0	14.9	1.53	0.5	0.93	0.702^1	15.0	1.10	0.3	0.94	0.840^1
$A_1B_2C_1$	20.2	20.2	1.54	−0.1	0.92		20.2	1.10	−0.1	0.94	
$A_1B_2C_2$	15.7	15.7	1.59	0.8	0.91		15.7	1.10	0.2	0.95	
$A_2B_1C_1$	17.5	17.6	1.27	−0.4	0.93		17.5	1.05	0.1	0.94	
$A_2B_1C_2$	15.0	15.1	1.19	−0.5	0.94	0.733^2	15.0	1.00	0.1	0.93	0.986^2
$A_2B_2C_1$	21.0	21.0	1.20	−0.2	0.94		21.0	1.00	−0.1	0.94	
$A_2B_2C_2$	18.5	18.5	1.40	0.3	0.94		18.5	1.11	−0.0	0.93	
						0.875^3					0.998^3

$^1H_1 : \mu_{A_1B_1C_1} = \mu_{A_1B_1C_2} = \mu_{A_1B_2C_1} = \mu_{A_1B_2C_2}$, $^2H_2 : \mu_{A_2B_1C_1} = \mu_{A_2B_1C_2} = \mu_{A_2B_2C_1} = \mu_{A_2B_2C_2}$,
$^3H_3 : \mu_{A_1B_1C_1} = \mu_{A_1B_1C_2} = \mu_{A_1B_2C_1} = \mu_{A_1B_2C_2} = \mu_{A_2B_1C_1} = \mu_{A_2B_1C_2} = \mu_{A_2B_2C_1} = \mu_{A_2B_2C_2}$

Table 3: Simulation results of estimation by adjusting probability of maintenance treatment

$n = 200$, $\pi_A = 0.5$, $\pi_{R1} = \pi_{R2} = 0.5$, $P_k = 0.9$, $Q_l = 0.5$

		Estimator in URT					Probability adjusted estimator in SRT				
Policy	True	Est.	SE	RB(%)	CP	Power	Est.	SE	RB(%)	CP	Power
$A_1B_1C_1$	17.5	17.5	1.51	0.2	0.95		17.5	0.97	−0.2	0.95	
$A_1B_1C_2$	15.0	15.0	1.43	0.1	0.95	0.721^1	15.0	0.91	0.2	0.94	0.828^1
$A_1B_2C_1$	21.0	21.0	1.42	−0.2	0.92		21.0	1.52	0.2	0.91	
$A_1B_2C_2$	18.5	18.5	1.58	−0.2	0.94		18.3	1.72	0.8	0.89	
$A_2B_1C_1$	17.5	17.5	1.60	−0.3	0.94		17.5	0.96	−0.2	0.93	
$A_2B_1C_2$	15.0	15.0	1.52	0.3	0.94	0.753^2	15.1	0.91	−0.4	0.94	0.855^2
$A_2B_2C_1$	21.0	21.0	1.52	−0.2	0.93		21.0	1.50	−0.1	0.90	
$A_2B_2C_2$	18.5	18.5	1.68	0.1	0.93		18.4	1.69	0.5	0.89	
						0.896^3					0.969^3

4.0 NONPARAMETRIC INFERENCE ON MEDIAN RESIDUAL LIFE FUNCTION IN SEQUENTIALLY RANDOMIZED TRIALS

4.1 INTRODUCTION

Sequentially randomized trials (SRT) are effective methods for comparing ATSs. In SRT, patients are randomized sequentially to treatment options as the trial progresses and patients move from one stage to another.

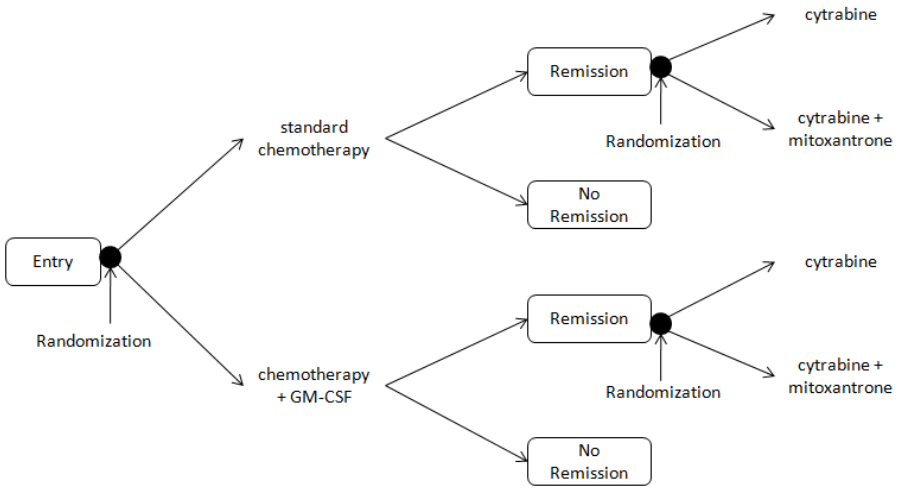


Figure 2: An example of Sequentially Randomization Trial from Cancer and Leukemia group B

Figure 2 shows an example of SRT with two stages, drawn from a leukemia study protocol from CALGB co-operative group. At entry, patients are randomized to one of the two first-stage treatments standard chemotherapy(A_1) or standard chemotherapy plus granulocyte-macrophage colony-stimulating factor(GM-CSF)(A_2). Once the patient achieves complete

remission, he/she is randomized to the second-stage treatments cytrabine(B_1) or cytrabine plus mitoxantrone(B_2). This SRT allows comparison of 4 treatment strategies A_jB_k, $j,k = 1,2$, where the strategy A_jB_k is defined as "Treat with A_j followed by B_k if a response to A_j is observed "(Lunceford et al. [2002]). Note that a patients belonging to any of the following scenarios is considered to be treated with strategy A_jB_k (consistent with strategy A_jB_k) :

1. Patient received A_j, and did not respond.
2. Patient received A_j, responded, and received B_k at the second stage.

The objective would be to assess which of these four strategies $A_jB_k, j,k = 1,2$ results in the best benefit for the patients. In survival analysis, the best benefit would be measured by longest survival time. The sample mean is the most common statistics in terms of summarizing the survival distributions. Survival probabilities at a fixed point of time or median survival are also used to summarize the data.

The estimation of mean survival time in SRT, especially in two-stage design, has been discussed extensively in the literature. Lunceford et al. [2002] proposed estimators for the survival distribution and mean restricted survival time under different treatment strategies using the concept of inverse-probability-weighting (IPW) (Robins et al. [1994]). Locally efficient estimators for survival distribution in two-stage design settings were introduced by Wahed and Tsiatis [2004], Wahed et al. [2006] for complete and right censored cases. The improvement in efficiency of these estimators was achieved through the use of additional information from auxiliary covariates. Guo [2005] proposed a weighted risk set estimator (WRSE) for the survival distribution. Even though the sample mean is the most commonly used index to summarize a distribution, this may not be a good summary representative of the overall distribution when the data are highly skewed. In such a case, percentiles can be a good index outlining the distribution instead of means.

In addition to percentile of the distribution (including median), the median residual life function (MERL function) is frequently used for the time-to-event data. A MERL function is the median of the remaining life time at a specific time point. For overall survival T, the median residual lifetime at time t_0 is defined as

$$\theta(t_0) = median(T - t_0 | T \geq t_0),$$

where $median(X|Y)$ stands for the median of the conditional distribution of X given Y. In other words, $\theta(t_0)$ is the median time for those who survived beyond time t_0 in the population. The median residual lifetime is stable since it is not affected by outliers such as very long durations of survival time. In the case with censored data, the use of a median for the residual lifetime not only exhibits a better location estimate than its mean but also shows less sensitivity to skewed distributions. The minimum of observed survival probability of the residual life distribution should be smaller than the 0.5, so that the median residual lifetime can be defined under censoring (Schmittlein et al. [1981]).

Various methods have been proposed in the literature to estimate the median residual lifetimes for one and two sample cases. Haines et al. [1974] introduced a general concept of the α-percentile residual life function ($0 < \alpha < 1$). Schmittlein et al. [1981] established a general concept of MERL function exploring non-uniqueness of the corresponding life distribution. Csörgö et al. [1987] proposed a $100(1-p)$th percentile residual lifetime estimator for complete data, whereas Chung [1989] extended this idea to censoring cases. The nonparametric estimator of $100(1-p)$th percentile residual lifetime was proposed by Feng et al. [1991], where inverse function of the Kaplan-Meier curve was used. Jeong et al. [2008] introduced a test statistic to compare two median residual lifetimes at a fixed time point. However, these methods cannot be applied to estimate median residual lifetime for treatment strategies in two-stage designs as they do not account for sequential randomization structure.

In this study, we aim to develop methods for nonparametric estimation of strategy-specific median residual lifetime from two-stage randomization designs. Our method uses the marginal mean models for survival data introduced by Lunceford et al. [2002] for sequentially randomized trials. We also use inverse-probability weighted estimating equations directly for this purpose.

Notation used throughout this chapter is described in Section 4.2. Estimation of MERL function using different approaches is discussed in Section 4.3. We present a simulation study to demonstrate the performance of proposed methods in Section 4.4. Then, we apply these methods to a leukemia data set in Section 4.5.

4.2 NOTATION AND ASSUMPTIONS

We consider the two-stage design described in the previous section. Our development of methods borrows the idea of counterfactual variables from the causal inference literature (Holland [1986]). Consider only the population that receives A_1 as the initial treatment (development for the population receiving A_2 is analogous). For the i^{th} individual in the population, let R_i be the response indicator: $R_i = 1$ if a response to the first-stage treatment is achieved, 0 otherwise. If the patient did not respond, his/her survival time is denoted by T_{NRi}. Let T^*_{ki} be the post-response survival time for patient i if the patient had responded and received B_k as the maintenance treatment. Let T_{ai} be the time to the starting of the second-stage treatment, if the ith patient proceeded to the second stage. Thus, under the strategy A_1B_k, the overall survival time for patients i can be defined as

$$T_{ki} = (1 - R_i)T_{NRi} + R_i(T_{ai} + T^*_{ki}), k = 1, 2.$$

Note that, T_1 and T_2 both cannot be observed for the same patient since a patient either does not receive any maintenance treatment, or receives only one of the two maintenance treatments, B_1 or B_2. For such, these variables are referred to as potential outcomes or counterfactuals (Holland [1986]). The interest lies in estimating the MERL function for the strategy A_1B_k or equivalently for the overall survival T_{ki}. Now, the survival time for the i^{th} patient, T_i, if observed for the i^{th} individual, can be expressed in terms of T_1 and T_2 by means of the consistency assumption (Cole [2009]) as follows.

$$T_i = Z_{1i}T_{1i} + Z_{2i}T_{2i}, \tag{4.1}$$

where Z_{ki} is the B_k treatment assignment indicator, i.e. $Z_{ki} = 1$ if the i^{th} patient was assigned to treatment B_k, 0 otherwise. Note that, Eq. (4.1) can alternatively be expressed as,

$$T_i = (1 - R_i)T_{NRi} + R_i\{T_{ai} + Z_{1i}T^*_{1i} + Z_{2i}T^*_{2i}\}.$$

Unfortunately, T_i also may not be observed for the i^{th} individual, since it may be censored. The observed data from a two-stage design described in Figure. 2 can be denoted

by $(R_i, R_i Z_{ki}, U_i, \Delta_i)$ $i = 1...n, k = 1, 2$, where R_i and Z_{ki} are as defined before, Δ_i is the complete case indicator, and U_i is the event (survival or censoring) time. If C_i is used to denote the censoring time for the i^{th} individual in the sample, then $U_i = min(T_i, C_i)$ and $\Delta_i = I(T_i < C_i)$. Note that, in practice, patients may be censored prior to a response being observed; such patients are historically treated as non responders in the analysis. We assume that censoring does not depend on T_i or any other observed or counterfactual data. Denote by $K(t)$ the survival distribution function of censoring time C_i, i.e., $K(t) = P(C_i \geq t)$. Let $\pi_1 = P(Z_{1i} = 1|R_i = 1)$ and $\pi_2 = 1 - \pi_1$ denote the probability of receiving treatment B_1 or B_2, respectively. Further let $S_k(t) = P(T_k > t)$ denote the survival probability at time t under strategy $A_1 B_k$ for $k = 1, 2$.

Again, our goal is to find an estimator of the MERL function under strategy $A_1 B_k$ for $k = 1, 2$. In other words, we would like to estimate $\theta_k(t_0)$, where

$$\theta_k(t_0) = median(T_k - t_0 | T_k > t_0). \tag{4.2}$$

4.3 ESTIMATION OF THE MEDIAN RESIDUAL LIFE FUNCTION

We first express $\theta_k(t_0)$ as a function of $S_k(t_0)$ so that the median residual lifetime can be estimated indirectly from the estimated survival curve. The MERL function defined in Eq. (4.2) can be expressed as $P(T_k - t_0 \geq \theta_k(t_0)|T_k \geq t_0) = \frac{1}{2}$, thereby suggesting that

$$\theta_k(t_0) = S_k^{-1}\left[\frac{1}{2} S_k(t_0)\right] - t_0, \tag{4.3}$$

where $S_k^{-1}(u)$ is the inverse of the survival function at u, i.e., $S_k^{-1}(u) = inf\{t : S_k(t) < u\}$. Thus, one way to estimate $\theta_k(t_0)$ is to estimate the survival distribution $S_k(t_0)$ and use Eq. (4.3) to obtain $\hat{\theta}_k(t_0)$ as

$$\hat{\theta}_k^{(1)}(t_0) = \hat{S}_k^{-1}\left[\frac{1}{2} \hat{S}_k(t_0)\right] - t_0, \tag{4.4}$$

where $\hat{S}_k(t)$ is a consistent estimator of $S_k(t)$ (Feng et al. [1991], Jeong et al. [2008]). As long as the largest observation in the sample is uncensored, the MERL function can be appropriately estimated.

One way to estimate the survival curve for a given strategy A_1B_k is the probability-adjusted inverse-probability-weighted estimator defined as:

$$\hat{S}_k(t_0) = \left\{ \sum_{i=1}^{n} \frac{\Delta_i Q_{ki}}{\hat{K}(U_i)} \right\}^{-1} \left\{ \sum_{i=1}^{n} \frac{\Delta_i Q_{ki}}{\hat{K}(U_i)} I(U_i > t_0) \right\}, \tag{4.5}$$

where, $\hat{K}(U_i)$ is the Kaplan-Meier estimator of $K(t)$ and $Q_{ki} = 1 - R_i + \frac{R_i Z_{ki}}{\pi_k}$, $k = 1, 2$ is the weight function. This function assigns appropriate weight to each observation that is consistent with the strategy A_1B_k. In the estimation of the survival function S_k, every uncensored U_i is weighted by $\frac{1}{K(U_i)}$ to reflect the fact that there were $\frac{P(C_i \leq U_i)}{P(C_i > U_i)} = \frac{1}{K(U_i)} - 1$ expected censored individuals in the population at time U_i. Therefore, the response for an uncensored individual counts for him/herself and additional censored individual $\frac{1}{K(U_i)} - 1$. The weights Q_{ki} are defined based on the fact that patients who did not respond are consistent with both strategies A_1B_1 and A_1B_2, and those who responded, only a portion of them are treated with A_1B_1 (A_1B_2) with probability π_1 (π_2). Thus, for the treatment strategy A_1B_1, $Q_{1i} = 0$ if individual i received B_2 treatment (i.e., $R_i = 1$ and $Z_i = 0$), while $Q_{1i} = 1$ if $R_i = 0$, and $Q_{1i} = \pi_1^{-1}$ if $R_i = 1$ and $Z_i = 1$ to reflect the weights due to randomization. Similarly, when an individual receives a treatment which is consistent with the treatment strategy A_1B_2, Q_{2i} acts as a weight.

Lunceford et al. [2002] showed that the estimator $\hat{S}_k(t_0)$ defined by Eq. (4.5) is a consistent and asymptotically normal estimator with mean $S_k(t_0)$ and variance $\sigma_k^2(t_0)$ that can be estimated by

$$\begin{aligned} \hat{\sigma}_k^2(t_0) = & \frac{1}{n} \left[\frac{1}{n} \sum_{i=1}^{n} \frac{\Delta_i Q_{ki}}{\hat{K}(U_i)} \Big\{ I(U_i > t_0) - 1 + \hat{S}_k(t_0) \Big\}^2 \right. \\ & \left. + \int_0^L \frac{dN^c(u)}{\hat{K}(u)Y(u)} \hat{E}\{L_{ki}(t,u)\}^2 \right], \end{aligned}$$

where

$$\int_0^L \frac{dN^c(u)}{\hat{K}(u)Y(u)}\hat{E}\{L_{ki}(t,u)\}^2$$
$$=\sum_{j=1}^{n}\left\{\frac{1-\Delta_j}{\hat{K}(U_j)Y(V_j)}\left\{\frac{1}{n}\sum_{i=1}^{n}\Delta_i\Big\{Q_{ki}\big(I(U_i>t_0)-1+\hat{S}_k(t_0)\big)\right.\right.$$
$$\left.\left.-\hat{G}_k(t,U_i)\Big\}^2\frac{I(U_i\geq u)}{\hat{K}(U_i)}\right\}\right\},$$
$$\hat{G}_{ki}(t,u)=\frac{1}{n\hat{S}(u)}\left\{\sum_{i=1}^{n}\Delta_i Q_{ki}\big(I(U_i>t_0)-1+\hat{S}_k(t_0)\big)\frac{I(U_i\geq u)}{\hat{K}(U_i)}\right\}.$$

Here, $N^c(u)=\sum N_i^c(u)=\sum I(U_i<u,\Delta_i=0)$ and $Y(u)=\sum Y_i(u)=\sum I(U_i\geq u)$.

4.3.1 Survival Function Based Estimator

Substituting Eq.(4.5) in Eq.(4.4), we obtain the first estimator for the MERL function. This estimator will be referred to as survival function based (SFB) estimator, $\hat{\theta}_k^{(1)}(t_0)$. To calculate the variance of this estimator, we first note that $\hat{\theta}_k^{(1)}(t_0)$ satisfies $\hat{S}_k(t_0+\hat{\theta}_k^{(1)}(t_0))-\frac{1}{2}\hat{S}_k(t_0)=0$. Let us denote the left hand side of the above equation by $\psi_k(\hat{\theta}_k^{(1)}(t_0))$. Expanding $\psi_k(\hat{\theta}_k^{(1)}(t_0))$ around $\theta_k(t_0)$, we obtain,

$$\psi_k(\hat{\theta}_k^{(1)}(t_0))\approx\psi_k(\theta_k(t_0))+(\hat{\theta}_k^{(1)}(t_0)-\theta_k(t_0))\frac{\delta\psi_k(\theta_k(t_0))}{\delta\theta_k(t_0)}.$$

Assuming sufficient regularity, it follows that variance of $\hat{\theta}_k(t_0)$ could be approximated by

$$\mathrm{Var}(\hat{\theta}_k^{(1)})\approx\frac{var[\psi_k(\hat{\theta}_k^{(1)}(t_0))]}{\left[\frac{\delta\psi_k(\theta_k(t_0))}{\delta\theta_k(t_0)}\right]^2}.$$

Note that,

$$\frac{\delta\psi_k(\theta)}{\delta\theta}=-f_k(t_0+\theta),$$

leading to

$$\mathrm{Var}(\hat{\theta}_k^{(1)}(t_0))\approx\frac{var(\psi_k(\hat{\theta}_k^{(1)}(t_0)))}{f_k^2(t_0+\hat{\theta}_k^{(1)}(t_0))},\tag{4.6}$$

where $f_k(\cdot)$ is the density function of T_k, which can be estimated by using nonparametric smoothing technique such as the kernel density estimator. To estimate the variance in Eq. (4.6), we first note that $\psi_k(\hat{\theta}_k^{(1)}(t_0))$ can be written as

$$\left\{\sum_{i=1}^{n}\frac{\Delta_i Q_{ki}}{\hat{K}(U_i)}\right\}^{-1}\left\{\sum_{i=1}^{n}\frac{\Delta_i Q_{ki}}{\hat{K}(U_i)}h\{U_i;\hat{\theta}_k^{(1)}(t_0)\}\right\},$$

where $h\{U_i;\theta\} = I(U_i > \theta + t_0) - \frac{1}{2}I(U_i > t_0)$. Following Lunceford et al. [2002], we obtain a consistent estimator of $\mathrm{Var}(\hat{\psi}_k(\hat{\theta}_k^{(1)}(t_0))$ as follows:

$$\begin{aligned} var(\hat{\psi}_k(\hat{\theta}_k^{(1)}(t_0))) =& \frac{1}{n}\left[\frac{1}{n}\sum_{i=1}^{n}\frac{\Delta_i Q_{ki}}{\hat{K}(U_i)}\Big\{h(U_i)\Big\}^2\right. \\ &\left. + \int_0^L \frac{dN^c(u)}{\hat{K}(u)Y(u)}\hat{E}\{L_{ki}(t,u)\}^2\right], \end{aligned} \tag{4.7}$$

where

$$\begin{aligned} &\int_0^L \frac{dN^c(u)}{\hat{K}(u)Y(u)}\hat{E}\{L_{ki}(t,u)\}^2 \\ &=\sum_{j=1}^{n}\left\{\frac{1-\Delta_j}{\hat{K}(U_j)Y(V_j)}\left\{\frac{1}{n}\sum_{i=1}^{n}\Delta_i\Big\{Q_{ki}\big(h\{U_i;\hat{\theta}_k^{(1)}(t_0)\}\big)\right.\right. \\ &\left.\left. - \hat{G}_k(t,U_i)\Big\}^2\frac{I(U_i\geq u)}{\hat{K}(U_i)}\right\}\right\}, \\ \hat{G}_{ki}(t,u) =& \frac{1}{n\hat{S}(u)}\left\{\sum_{i=1}^{n}\Delta_i Q_{ki}\big(h\{U_i;\hat{\theta}_k^{(1)}(t_0)\}\big)\frac{I(U_i\geq u)}{\hat{K}(U_i)}\right\}. \end{aligned}$$

Eq.(4.7), along with an inverse-probability-weighted kernel density estimator of f_k, can then be used in Eq.(4.6) to estimate the variance of $\hat{\theta}_k^{(1)}(t_0)$.

4.3.2 Estimating Equation Based Approach

An alternative method of estimating the MERL function at time t_0 would be to directly solve an appropriate estimating equation of the form

$$\sum_{i=1}^{n} \psi_i(\theta_k(t_0)) = 0, \tag{4.8}$$

where $\psi_i(\theta) = \frac{\Delta_i Q_{ki}}{K(U_i)}[I(U_i > t_0 + \theta) - \frac{1}{2}I(U_i > t_0]$. Note that $K(\cdot)$ is unknown and hence it must be estimate by some consistent method. Since this equation is not linear in θ, we can use numerical methods such as the Secant method or Newton-Raphson method to solve the equation for θ. We used the Secant method for this purpose to avoid computing numerical derivative in the Newton-Raphson method. The solution to Eq.(4.8) will be referred to as $\hat{\theta}_k^{(2)}(t_0)$.

To estimate the variance, we will use the so-called sandwich estimator. (4.8). Explicitly,

$$var(\hat{\theta}_k^{(2)}(t_0)) = \left(\frac{1}{n}\right) \frac{B(\hat{\theta}_k^{(2)}(t_0))}{A(\hat{\theta}_k^{(2)}(t_0))^2},$$

where $A(\theta) = \hat{S}_k'(t_0 + \theta)$ and $B(\theta) = \left\{\sum_{i=1}^{n} \frac{\Delta_i Q_{ki}}{\hat{K}(U_i)}\right\}^{-1} \sum_{i=1}^{n} \{\psi_i(\theta)\}^2$.

4.4 SIMULATION STUDY

We conducted a simulation study to evaluate the proposed methods. Since initial treatment A_1 and A_2 are assigned by randomization, we only designed simulation studies for the induction treatment A_1. We generate n independent sets of observations described as follows. Indicator of patient's response R was generated from a Bernoulli (π_R) distribution, where π_R was varied between 0.1 and 0.7. For responders ($R = 1$), the B_1 treatment assignment indicator Z_1 was generated from Bernoulli (π_1) distribution. We then defined $Z_2 = 1 - Z_1$. For non-responders ($R = 0$), we generated survival time T_{NR} from exponential distribution with mean $\lambda = 0.5$. For responders, a response time T_a was generated from exponential ($\alpha = 0.1$). Patients receiving maintenance treatment B_k, post remission survival time T_k^* was generated from three-parameter Weibull distribution as $T_k^* = (\frac{1}{\rho})(-exp(-\beta_k)log(1-u))^{(\frac{1}{\eta})}$

, $k = 1, 2$. Following Jeong et al. [2008], the parameter values for ρ and η are set to 0.2 and 2, respectively, and $\beta_1 = 0.6$ and $\beta_2 = 0.9$. Therefore, the survival time for the i^{th} patient, $T_i = (1 - R_i)T_{NRi} + R_i\{T_{ai} + Z_{1i}T^*_{1i} + Z_{2i}T^*_{2i}\}$. The censoring time C_i was taken from a uniform distribution between 0 and θ_C to reflect about 30% of censoring and assumed independent of all other variables. The observed survival U_i is $min(T_i, C_i)$ and the complete case indicator Δ_i is $I(T_i < C_i)$.

Under different simulation scenarios obtained under different combination of parameters, we generated 1000 samples of size n (200 and 500) and then estimated median residual lifetime at 183 and 365 days.

In Table 4 and Table 5, we summarize estimators of the median residual lifetimes (Estimates) and standard error (S.E.) for these estimators, along with 95% coverage probability of estimates (CP). Table 4 gives the results from scenarios where patients were assigned to the maintenance treatment with equal allocation. As can be seen, estimates and coverage probabilities were similar for the SFB and EEB. For the SFB and EEB estimators, the coverage probabilities in most cases achieved by the nominal level. These estimators were approximately unbiased as shown by the small observed biases. Standard errors for the EEB estimators were larger than the SFB estimators.

The cases with unequal probability of assignment to maintenance treatment are shown in Table 5. Here 30% of the responders were assigned to B_1 while 70% to B_2. Performance of the estimators, coverage probabilities and standard errors remain similar to the case where the randomization was done with equal probability in Table 5. In this scenario, estimators corresponding to the maintenance treatment with higher probability of randomization had smaller standard errors compared to the strategy which shared the other maintenance treatment. Specifically, standard errors for A_1B_1 was larger than that for A_1B_2 with 30% of responders being assigned to B_1. This is expected since this form of randomization increases the effective sample size of strategy A_1B_2 and reduces the number of patients followed in the strategy A_1B_1.

In both Tables 4 and 5, both estimators gained efficiency with increasing sample size. Standard errors of the estimators and Monte Carlo standard errors were close to each other, implying consistency of variance estimators for both methods.

Overall, the estimator from SFB method has smaller standard errors than the EEB estimator, and EEB estimator shows higher coverage rate than the SFB estimator.

4.5 ANALYSIS OF LEUKEMIA CALGB 8923 TRIAL

In this section, we apply the methods described in previous sections to estimate the median residual life time of patients treated with various treatment strategies. The data is collected from a two-stage randomized clinical trial conducted by the Cancer and Leukemia Group B (CALGB). This is a double-blinded placebo-controlled trial designed to examine the effects of infusions of granulocyte-macrophage colony-stimulating factor (GM-CSF) following induction chemotherapy in acute myelogenous leukemia (AML) patients. Patients are initially randomized to GM-SCF (n =193) or placebo (n=195) following treatment with standard chemotherapy. Patients who achieved remission and consented to second stage treatment (79 out of 193 in GM-CSF and 90 out of 195 in the placebo group) were re-randomized into one of the intensification therapies: cytarabine (intensification I) and cytarabine+mitoxantrone (Intensification II). In this process, 37 patients were assigned to intensification I in the GM-CSF group whereas 45 were assigned to the same intensification in the placebo group. The remaining patients in each group received intensification II. Therefore, there are 4 possible treatment strategies in this trial, namely, GM-CSF/I, GM-CSF/II, Placebo/I, and Placebo/II, where for example, "GM-CSF/I" stands for "add GM-CSF by infusion after chemotherapy followed by maintenance cytarabine, if respond to chemotherapy with GM-CSF infusion".

In Table 6, we presented the estimators of the median residual lifetime at specific time points for the treatment strategies, GM-CSF/I, GM-CSF/II, Placebo/I and Placebo/II, using the SFB [Eq. (4.4)] and EEB [Eq. (4.8)] methods. The median residual lifetimes at three different time points of 150, 250, and 350 days were estimated. From the results presented in Table 6, the estimates under both methods SFB and EEB were similar to each other, although the standard errors of the EEB estimates were larger compared to the SFB estimates. Estimated MERL was smallest for strategy GM-CSF/I whereas it was largest for

Table 4: Simulation results of estimation of MERL at two time points with $\pi_Z = 0.5$

n	π_R	t_0	Method	Strategy	$\theta(t_0)$	Estimates	S.E.	MC S.E.	C.P.
200	0.4	183	SFB	A_1B_1	267.5	267.9	48.81	51.63	0.92
				A_1B_2	235.2	235.2	44.59	47.31	0.92
			EEB	A_1B_1	267.5	277.9	58.59	53.50	0.96
				A_1B_2	235.2	242.3	49.96	48.17	0.96
		365	SFB	A_1B_1	232.6	229.5	62.83	64.15	0.91
				A_1B_2	193.8	190.0	53.42	57.73	0.91
			EEB	A_1B_1	232.6	253.0	80.37	66.98	0.96
				A_1B_2	193.8	209.7	63.10	62.85	0.96
	0.7	183	SFB	A_1B_1	341.9	332.3	51.25	51.68	0.93
				A_1B_2	285.6	283.6	47.58	44.42	0.94
			EEB	A_1B_1	341.9	340.7	61.29	52.47	0.95
				A_1B_2	285.6	289.8	52.33	44.52	0.98
		365	SFB	A_1B_1	262.9	253.7	56.70	55.26	0.93
				A_1B_2	209.5	205.1	48.63	48.33	0.93
			EEB	A_1B_1	262.9	270.0	69.62	57.97	0.97
				A_1B_2	209.5	214.9	54.11	49.47	0.96
500	0.4	183	SFB	A_1B_1	267.5	269.6	31.48	34.33	0.92
				A_1B_2	235.2	233.0	28.66	29.82	0.94
			EEB	A_1B_1	267.5	272.6	37.59	34.92	0.96
				A_1B_2	235.2	235.4	31.51	30.18	0.96
		365	SFB	A_1B_1	232.6	235.7	39.66	40.90	0.93
				A_1B_2	193.8	193.1	33.91	34.99	0.93
			EEB	A_1B_1	232.6	241.2	48.20	41.58	0.96
				A_1B_2	193.8	197.7	37.49	35.31	0.96
	0.7	183	SFB	A_1B_1	341.9	337.9	32.81	33.63	0.92
				A_1B_2	285.6	282.6	30.02	28.70	0.95
			EEB	A_1B_1	341.9	341.1	39.89	34.09	0.96
				A_1B_2	285.6	284.6	32.97	28.61	0.97
		365	SFB	A_1B_1	262.9	256.2	57.37	56.22	0.93
				A_1B_2	209.5	206.8	49.33	46.65	0.95
			EEB	A_1B_1	262.9	271.7	69.58	58.27	0.97
				A_1B_2	209.5	218.2	54.97	48.53	0.98

Table 5: Simulation results of estimation of MERL at two time points with $\pi_Z = 0.3$

N	π_R	t_0	Method	Strategy	$\theta(t_0)$	Estimates	S.E.	MC S.E.	C.P.
200	0.4	183	SFB	A_1B_1	273.9	265.3	61.47	66.04	0.88
				A_1B_2	239.9	235.5	37.56	39.23	0.93
			EEB	A_1B_1	273.9	280.4	71.66	87.88	0.93
				A_1B_2	239.9	240.7	42.80	38.93	0.96
		365	SFB	A_1B_1	234.3	227.1	82.50	80.26	0.90
				A_1B_2	193.4	189.6	44.68	46.31	0.91
			EEB	A_1B_1	234.3	262.0	105.79	85.32	0.95
				A_1B_2	193.4	201.5	51.76	49.13	0.96
	0.7	183	SFB	A_1B_1	343.5	331.1	67.15	67.59	0.91
				A_1B_2	285.1	284.6	38.79	36.89	0.96
			EEB	A_1B_1	343.5	344.8	77.48	65.53	0.94
				A_1B_2	285.1	289.3	43.81	37.28	0.97
		365	SFB	A_1B_1	262.1	252.3	76.80	74.32	0.90
				A_1B_2	210.4	206.5	40.55	38.73	0.95
			EEB	A_1B_1	262.1	278.9	93.83	77.28	0.95
				A_1B_2	210.4	214.0	46.30	39.53	0.97
500	0.4	183	SFB	A_1B_1	273.9	270.8	40.19	42.62	0.92
				A_1B_2	239.9	235.8	24.34	25.34	0.93
			EEB	A_1B_1	273.9	275.4	46.67	43.25	0.95
				A_1B_2	239.9	237.4	27.45	25.25	0.96
		365	SFB	A_1B_1	234.3	234.1	52.10	53.73	0.93
				A_1B_2	193.4	193.7	28.47	29.14	0.95
			EEB	A_1B_1	234.3	244.9	62.80	57.57	0.95
				A_1B_2	193.4	196.9	32.27	29.67	0.97
	0.7	183	SFB	A_1B_1	343.5	339.1	43.15	45.37	0.92
				A_1B_2	285.1	282.2	24.37	23.63	0.95
			EEB	A_1B_1	343.5	344.3	50.15	45.55	0.95
				A_1B_2	285.1	283.9	27.44	23.69	0.97
		365	SFB	A_1B_1	262.1	258.9	49.04	48.26	0.92
				A_1B_2	210.4	207.6	25.54	25.31	0.93
			EEB	A_1B_1	262.1	268.5	58.73	52.75	0.96
				A_1B_2	210.4	209.7	28.80	25.19	0.96

Table 6: Estimated median residual lifetime for CALGB 8923 data at different days

Strategy	t0 (days)	Methods	
		SFB	EEB
GM-CSF/I	150	290.0 (21.07)	309.2 (47.11)
	250	260.0 (23.93)	275.5 (52.30)
	350	233.0 (26.92)	246.4 (54.40)
GM-CSF/II	150	288.0 (21.66)	281.8 (45.08)
	250	233.0 (22.80)	239.8 (50.20)
	350	192.0 (27.08)	235.4 (66.27)
Placebo/I	150	313.0 (27.08)	318.0 (52.10)
	250	285.0 (27.59)	303.1 (59.51)
	350	271.0 (31.55)	271.2 (64.28)
Placebo/II	150	401.0 (45.47)	409.6 (106.80)
	250	395.0 (47.62)	459.2 (153.03)
	350	500.0 (68.60)	496.4 (230.50)

the strategy Placebo/II at all three time points. When the time of interest increased, median residual lifetimes decreased, except for the strategy "do not give GM-CSF after chemotherapy, and when a response is observed treat with second maintenance". The estimates of the median residual lifetimes over time are presented in Figure 3 for all four treatment strategies. It shows the evolution of the median residual lifetime over the duration of the trial.

For all strategies, the median residual lifetimes initially increased, most likely due to the effect of initial treatment. After about 50 days, the median residual lifetimes began to decrease for the strategies GM-CSF/I, GM-CSF/II and Placebo/I. Placebo/II shows the increasing median residual lifetimes throughout which may be due to the fact that some of the patients survived extremely long under this strategy. This result has been consistent with the findings from previous studies based on mean survival or survival curve (Lunceford et al. [2002]; Wahed and Tsiatis [2004]) where it has been showed that the infusion with GM-CSF was not beneficial to patients in terms of overall survival.

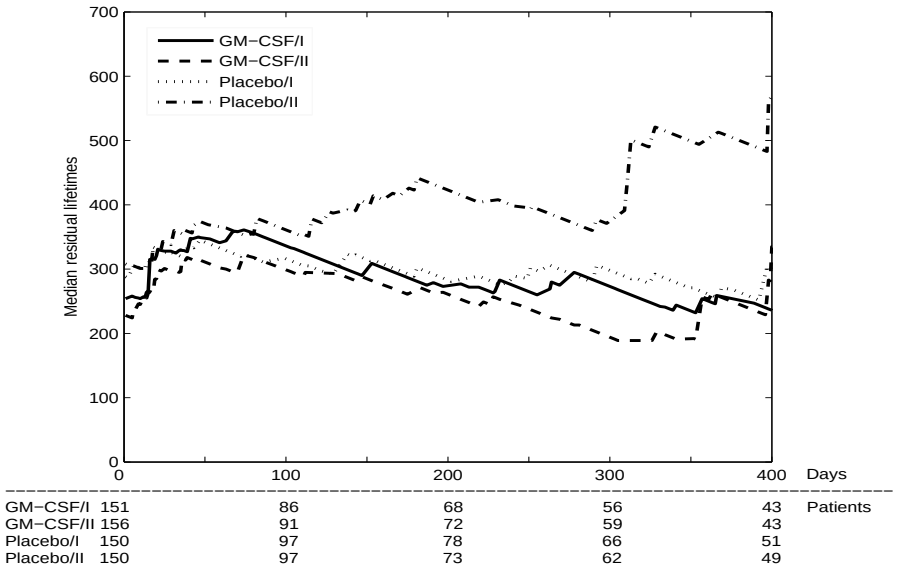


Figure 3: Survival function based median residual lifetimes for CALGB 8923 data with number of the patients at risk for each strategy

4.6 DISCUSSION

In this chapter, two methods were proposed to estimate the median residual life function for sequentially randomized trials. First, we estimated the median residual lifetime indirectly from the estimated survival curve. The survival curve estimation was done based on the inverse-probability-weighting. One important limitation of this estimator is that we need to estimate the density function to estimate the variance. In addition to SFB method, the median residual lifetime can also be estimated by directly solving the estimating equation of the median residual lifetime.

As can be seen from our simulation study, the SFB and EEB estimators are approximately unbiased. The estimator from SFB method has smaller standard errors than the EEB estimator which leads to higher coverage rates for the latter. Future research in this area might consider including covariates in the estimation process. Other important research might include developing test statistics to compare MERL across different strategies.

5.0 CONCLUSION

We first have compared up-front and sequential randomization designs as prospective designs for comparing dynamic treatment strategies. Most previous studies investigated these methods but few compared them with respect to efficiency. Our simulation results indicated that the sequential randomization provides better estimator for ATS's compared to up-front randomization designs. Specifically, sequential randomization is better than up-front randomization in terms of efficiency of the estimators and the power of statistical hypothesis testing. In the latter half of the study, we proposed nonparametric estimation methods to estimate the MERL function for SRT. One estimator for MERL function was obtained by using probability-adjusted estimator (Lunceford et al. [2002]). The other estimator was obtained by directly solving the estimating equation. We demonstrated our methods by applying to a Leukemia clinical trial data set. Our simulation study shows that SFB and EEB estimators for SRT are approximately unbiased in large samples.

One advantage of the SFB methods is that it can be calculated directly from the survival curve estimator. One limitation of this approach is that it requires density estimation for the purpose of estimating its variance. On the other hand, EEB estimator although less efficient, does not require any such limitation.

APPENDIX

PROGRAMS WRITTEN IN MATLAB©

A.1 UP-FRONT AND SEQUENTIAL RANDOMIZATION IN CLINICAL TRIALS

```
clear all;

% ========================================================
% Function to find nearest value
% ========================================================
function K = fOne(U,tk,Ku);
n = length(U);
for i=1:n
    index = find(U(i)==tk);
    if length(index)==0
        [val1 index] = min(abs(U(i)-tk));
 end
    K(i) = Ku(index(1));
end

% ========================================================
% Function to find Kaplan Meier Estimator
```

```
% ========================================================
function [K S] = kmest(U,D)
    D1 = 1-D;
    n = length(D);
[K,x] = ecdf(U,'censoring',D,'function','survivor');
[S,x1] = ecdf(U,'censoring',D1,'function','survivor');
K = fOne(U,x,K);
S = fOne(U,x1,S);
   for j = 2:n
    if K(j) == 0 K(j) = K(j-1); end
    if S(j) == 0 S(j) = S(j-1); end
   end
K = K';
S = S';

% ========================================================
% Function to find Density function
% ========================================================
function [Den1_R0 Den1_t0_R0] = denS(U_R0, Q, D1_R0, t0)
n_R0 = length(U_R0);
[den1_R0,x1] = ksdensity(U_R0,'weights',Q,'censoring',D1_R0,
'npoints',n_R0);
Den1_R0 = fOne(U_R0,x1,den1_R0);
Den1_t0_R0 = fOne(t0,U_R0,Den1_R0);
function f = secfunc1(p1)
load FileforSecant
f = (sum((D./K.*Q1.*((U>t0+p1)-0.5.*(U>t0))))/sum(D./K.*Q1));

% ========================================================
% Function to call secant function
```

```
% ========================================================
function f = secfunc1(p1)
load FileforSecant
f = (sum((D./K.*Q1.*((U>t0+p1)-0.5.*(U>t0))))/sum(D./K.*Q1));

% ========================================================
% Function to find Inverse Survival
% ========================================================
function [S_half1_pmb Sinv1_pmb Sinv1_pmb_t0 S_half1_pmb_t0] =
S_find(S1pmb, U_pmb, t0)
    S_half1_pmb = 0.5*S1pmb;
    Sinv1_pmb = fOne(S_half1_pmb,S1pmb,U_pmb);
    Sinv1_pmb_t0 =  fOne(t0,U_pmb,Sinv1_pmb);
    S_half1_pmb_t0 = fOne(t0,U_pmb,S_half1_pmb);

% ========================================================
% Function to calculate Coverage Probabilities
% ========================================================
function [Var331 Var332 Se331 Se332 Cp331 Cp332]
= CP1(Den31_t0, Den32_t0, D, K, Q1, Q2, U, t0, Theta31, Theta32,
 mu31, mu32)
n = length(U);
 AA31 = Den31_t0^2;
    AA32 = Den32_t0^2;
    BB31 = sum(((D./K.*Q1.*((U>t0+Theta31)-0.5.*(U>t0)))).^2)/n;
    BB32 = sum(((D./K.*Q2.*((U>t0+Theta32)-0.5.*(U>t0)))).^2)/n;
    Var331 =  (BB31/AA31)/n;
    Var332 =  (BB32/AA32)/n;
    Se331 = sqrt(Var331);
    Se332 = sqrt(Var332);
```

```
    CL_L331 = Theta31 -1.96* Se331;
    CL_U331 = Theta31 + 1.96*Se331;
    CL_L332 = Theta32 -1.96* Se332;
    CL_U332 = Theta32 + 1.96*Se332;
    Cp331 = (mu31 >= CL_L331) * (CL_U331 >= mu31);
    Cp332 = (mu32 >= CL_L332) * (CL_U332 >= mu32);


% ========================================================
% Begin Main codes
% ========================================================
clear all;
nos = 100; % iterations
nT = 200; % sample size
%%make effect size small like 0.25? 0.1

piA = 0.5; %for Treatment A
piR1 = 0.5; %response for A1
piR2 = 0.5; %response for A2
piB = 0.5; %proportion for B
piC = 0.5; %proportion for C
%%%%%% same true mean
% m11d = 15;
% m1d1 = 15;
% m12d = 15;
% m1d2 = 15;
% m21d = 15;
% m2d1 = 15;
% m22d = 15;
% m2d2 = 15;
```

```
%%%%% different true mean
            m11d = 15;
            m1d1 = 20;
            m12d = 22;
            m1d2 = 15;
            m21d = 15;
            m2d1 = 20;
            m22d = 22;
            m2d2 = 15;
n1 = piA*nT; % # of people in A1
n2 = (1-piA)*nT; % # of people in A2
v1 = 6;
v2 = 8;
for k = 1:nos
    U01 = rand(n1,1);
    U02 = rand(n2,1);
    U1 = rand(n1,1);
    U2 = rand(n1,1);
    V1 = rand(n1,1);
    V2 = rand(n1,1);
    V21 = rand(n2,1);
    V22 = rand(n2,1);
    U21 = rand(n2,1);
    U22 = rand(n2,1);
% ====================================================================
%       For A1
% ====================================================================
    for i = 1:n1
        %indicator for receiving second treatment
        if V1(i) < piR1 R1(i) = 1;
```

```
else R1(i) = 0;
end
%indicator for second treatment method ( responders piB
%non-responders piC )
if U1(i) < piB Z1(i) = 1;
else Z1(i) = 0;
end
if U2(i) < piC Z3(i) = 1;
else Z3(i) = 0;
end
%indicator for 4 groups in URT
if U01(i) < 0.25 X1(i) = 1;
else X1(i) = 0;
end
if (0.25 <= U01(i)) && (U01(i)< 0.5) X2(i) = 1;
else X2(i) = 0;
end
if (0.5 <= U01(i)) && (U01(i) < 0.75) X3(i) = 1;
else X3(i) = 0;
end
if U01(i) >= 0.75 X4(i) = 1;
else X4(i) = 0;
end
%distribution of observed data
y11d = randn(n1,1)*v1 + m11d;
y1d1 = randn(n1,1)*v2 + m1d1;
y12d = randn(n1,1)*v1 + m12d;
y1d2 = randn(n1,1)*v2 + m1d2;
%split data into 4 different groups in URT
y111(i) = (R1(i)*y11d(i) + (1-R1(i))*y1d1(i))*X1(i);
```

```
        y112(i) = (R1(i)*y11d(i) + (1-R1(i))*y1d2(i))*X2(i);
        y121(i) = (R1(i)*y12d(i) + (1-R1(i))*y1d1(i))*X3(i);
        y122(i) = (R1(i)*y12d(i) + (1-R1(i))*y1d2(i))*X4(i);
        %total observed data in SRT with counterfactuals
        y_tot(i) = R1(i)*(Z1(i)*y11d(i) + (1-Z1(i))*y12d(i)) +
(1-R1(i))*(Z3(i)*y1d1(i) + (1-Z3(i))*y1d2(i));
        %divide into 4 groups based on second treatment agreement and
        %treatment method in SRT
        W111(i) = (((R1(i)*Z1(i))/piB) + ((((1-R1(i))*Z3(i)))/piC)); % IPW
        W112(i) = (((R1(i)*Z1(i))/piB) + ((((1-R1(i))*(1-Z3(i))))/(1-piC)));
        W121(i) = (((R1(i)*(1-Z1(i)))/(1-piB)) + ((((1-R1(i))*Z3(i)))/piC));
        W122(i) = (((R1(i)*(1-Z1(i)))/(1-piB)) +
((((1-R1(i))*(1-Z3(i))))/(1-piC)));
        y1_2(i) = (((R1(i)*Z1(i))/piB) + ((((1-R1(i))*Z3(i)))/piC))
*y_tot(i);
        y2_2(i) = (((R1(i)*Z1(i))/piB) + ((((1-R1(i))*(1-Z3(i))))/(1-piC)))
*y_tot(i);
        y3_2(i) = (((R1(i)*(1-Z1(i)))/(1-piB)) + ((((1-R1(i))*Z3(i)))/piC))
*y_tot(i);
        y4_2(i) = (((R1(i)*(1-Z1(i)))/(1-piB)) +
(((1-R1(i))*(1-Z3(i))))/(1-piC)))*y_tot(i);
    end
% ========================================================================
%       For A2
% ========================================================================
    for i = 1:n2
        %indicator for receiving second treatment
        if V22(i) < piR2 R2(i) = 1;
        else R2(i) = 0;
        end
```

```
%indicator for second treatment method
if U21(i) < piB Z2(i) = 1;
else Z2(i) = 0;
end
if U22(i) < piC Z4(i) = 1;
else Z4(i) = 0;
end
%indicator for 4 groups in design 1
if U02(i) < 0.25 X1(i) = 1;
else X1(i) = 0;
end
if (0.25 <= U02(i)) && (U02(i)< 0.5) X2(i) = 1;
else X2(i) = 0;
end
if (0.5 <= U02(i)) && (U02(i) < 0.75) X3(i) = 1;
else X3(i) = 0;
end
if U02(i) >= 0.75 X4(i) = 1;
else X4(i) = 0;
end
%distribution of observed data
y21d = randn(n2,1)*v1 + m21d;
y2d1 = randn(n2,1)*v2 + m2d1;
y22d = randn(n2,1)*v1 + m22d;
y2d2 = randn(n2,1)*v2 + m2d2;
%split data into 4 different groups in URT for rest of R
y211(i) = ((R2(i))*y21d(i) + (1-R2(i))*y2d1(i))*X1(i);
y212(i) = ((R2(i))*y21d(i) + (1-R2(i))*y2d2(i))*X2(i);
y221(i) = ((R2(i))*y22d(i) + (1-R2(i))*y2d1(i))*X3(i);
y222(i) = ((R2(i))*y22d(i) + (1-R2(i))*y2d2(i))*X4(i);
```

```
        %total observed data in SRT with counterfactuals
        y_tot_2a(i) = (R2(i))*(Z2(i)*y21d(i) + (1-Z2(i))*y22d(i)) +
 (1-R2(i))*(Z4(i)*y2d1(i) + (1-Z4(i))*y2d2(i));
        %divide into 4 groups based on second treatment agreement and
        %treatment method in SRT
        W211(i) = ((((R2(i))*Z2(i))/piB) + ((((1-R2(i))*Z4(i)))/piC));
% IPW
        W212(i) = ((((R2(i))*Z2(i))/piB) + ((((1-R2(i))*
(1-Z4(i))))/(1-piC)));
        W221(i) = ((((R2(i))*(1-Z2(i)))/(1-piB)) +
((((1-R2(i))*Z4(i)))/piC));
        W222(i) = ((((R2(i))*(1-Z2(i)))/(1-piB)) +
((((1-R2(i))*(1-Z4(i))))/(1-piC)));
        y1_2_2a(i) = ((((R2(i))*Z2(i))/piB) + ((((1-R2(i))*Z4(i)))/piC))
*y_tot_2a(i);
        y2_2_2a(i) = ((((R2(i))*Z2(i))/piB) +
((((1-R2(i))*(1-Z4(i))))/(1-piC)))*y_tot_2a(i);
        y3_2_2a(i) = ((((R2(i))*(1-Z2(i)))/(1-piB)) +
((((1-R2(i))*Z4(i)))/piC))*y_tot_2a(i);
        y4_2_2a(i) = ((((R2(i))*(1-Z2(i)))/(1-piB)) +
((((1-R2(i))*(1-Z4(i))))/(1-piC)))*y_tot_2a(i);
    end

    %%remove all zeros from each outcome vector in SRT
    index111 = find(y111);y1_n1 = y111(index111);
    index112 = find(y112);y2_n1 = y112(index112);
    index121 = find(y121);y3_n1 = y121(index121);
    index122 = find(y122);y4_n1 = y122(index122);
    index211 = find(y211);y1_n1_2a = y211(index211);
    index212 = find(y212);y2_n1_2a = y212(index212);
```

```
    index221 = find(y221);y3_n1_2a = y221(index221);
    index222 = find(y222);y4_n1_2a = y222(index222);
    %%counting elements after removing zeros
    size1 = length(y1_n1);
    size2 = length(y2_n1);
    size3 = length(y3_n1);
    size4 = length(y4_n1);
    size1_2a = length(y1_n1_2a);
    size2_2a = length(y2_n1_2a);
    size3_2a = length(y3_n1_2a);
    size4_2a = length(y4_n1_2a);
% ========================================================================
%  estimate means
% ========================================================================
    %% mean from URT
    m111(k) = mean(y1_n1);
    m112(k) = mean(y2_n1);
    m121(k) = mean(y3_n1);
    m122(k) = mean(y4_n1);
    m211(k) = mean(y1_n1_2a);
    m212(k) = mean(y2_n1_2a);
    m221(k) = mean(y3_n1_2a);
    m222(k) = mean(y4_n1_2a);
    %% mean in SRT
    nmu111(k) = mean(y1_2);
    nmu112(k) = mean(y2_2);
    nmu121(k) = mean(y3_2);
    nmu122(k) = mean(y4_2);
    nmu211(k) = mean(y1_2_2a);
    nmu212(k) = mean(y2_2_2a);
```

```
    nmu221(k) = mean(y3_2_2a);
    nmu222(k) = mean(y4_2_2a);
    %% mean in SRT probability adjusted
    mu111(k) = sum(y1_2)/sum(W111);
    mu112(k) = sum(y2_2)/sum(W112);
    mu121(k) = sum(y3_2)/sum(W121);
    mu122(k) = sum(y4_2)/sum(W122);
    mu211(k) = sum(y1_2_2a)/sum(W211);
    mu212(k) = sum(y2_2_2a)/sum(W212);
    mu221(k) = sum(y3_2_2a)/sum(W221);
    mu222(k) = sum(y4_2_2a)/sum(W222);
% ================================================================
%       variance for estimators
% ================================================================


    %% sample variance for URT
    s111_1(k) = (sum((y1_n1 - m111(k)).^2)/(size1-1));
    s112_1(k) = (sum((y2_n1 - m112(k)).^2)/(size2-1));
    s121_1(k) = (sum((y3_n1 - m121(k)).^2)/(size3-1));
    s122_1(k) = (sum((y4_n1 - m122(k)).^2)/(size4-1));
    s111_1_2a(k) = (sum((y1_n1_2a - m211(k)).^2)/(size1_2a-1));
    s112_1_2a(k) = (sum((y2_n1_2a - m212(k)).^2)/(size2_2a-1));
    s121_1_2a(k) = (sum((y3_n1_2a - m221(k)).^2)/(size3_2a-1));
    s122_1_2a(k) = (sum((y4_n1_2a - m222(k)).^2)/(size4_2a-1));

    %% M-est for var. in SRT
    B111(k) = (sum((W111.*y_tot-nmu111(k)).^2)/n1^2);
    B112(k) = (sum((W112.*y_tot-nmu112(k)).^2)/n1^2);
    B121(k) = (sum((W121.*y_tot-nmu121(k)).^2)/n1^2);
```

```
    B122(k) = (sum((W122.*y_tot-nmu122(k)).^2)/n1^2);
    B211(k) = (sum((W211.*y_tot_2a-nmu211(k)).^2)/n2^2);
    B212(k) = (sum((W212.*y_tot_2a-nmu212(k)).^2)/n2^2);
    B221(k) = (sum((W221.*y_tot_2a-nmu221(k)).^2)/n2^2);
    B222(k) = (sum((W222.*y_tot_2a-nmu222(k)).^2)/n2^2);
    %% covariance for SRT
    C12_2(k) = sum((W111.*y_tot-nmu111(k)).*(W112.*y_tot-nmu112(k)))/(n1^2);
    C13_2(k) = sum((W111.*y_tot-nmu111(k)).*(W121.*y_tot-nmu121(k)))/(n1^2);
    C14_2(k) = sum((W111.*y_tot-nmu111(k)).*(W122.*y_tot-nmu122(k)))/(n1^2);
    C23_2(k) = sum((W112.*y_tot-nmu112(k)).*(W121.*y_tot-nmu121(k)))/(n1^2);
    C24_2(k) = sum((W112.*y_tot-nmu112(k)).*(W122.*y_tot-nmu122(k)))/(n1^2);
    C34_2(k) = sum((W121.*y_tot-nmu121(k)).*(W122.*y_tot-nmu122(k)))/(n1^2);
    C12_2_2a(k) = sum((W211.*y_tot_2a-nmu211(k)).*
(W212.*y_tot_2a-nmu212(k)))/(n2^2);
    C13_2_2a(k) = sum((W211.*y_tot_2a-nmu211(k)).*
(W221.*y_tot_2a-nmu221(k)))/(n2^2);
    C14_2_2a(k) = sum((W211.*y_tot_2a-nmu211(k)).*
(W222.*y_tot_2a-nmu222(k)))/(n2^2);
    C23_2_2a(k) = sum((W212.*y_tot_2a-nmu212(k)).*
(W221.*y_tot_2a-nmu221(k)))/(n2^2);
    C24_2_2a(k) = sum((W212.*y_tot_2a-nmu212(k)).*
(W222.*y_tot_2a-nmu222(k)))/(n2^2);
    C34_2_2a(k) = sum((W221.*y_tot_2a-nmu221(k)).*
(W222.*y_tot_2a-nmu222(k)))/(n2^2);

    %% M-est for var. in SRTPA
    BM111(k) = (sum((W111.*(y_tot-mu111(k))).^2)/(n1*sum(W111)^2));
    BM112(k) = (sum((W112.*(y_tot-mu112(k))).^2)/(n1*sum(W112)^2));
    BM121(k) = (sum((W121.*(y_tot-mu121(k))).^2)/(n1*sum(W121)^2));
    BM122(k) = (sum((W122.*(y_tot-mu122(k))).^2)/(n1*sum(W122)));
```

```
    BM211(k) = (sum((W211.*(y_tot_2a-mu211(k))).^2)/(n2*sum(W211)^2));
    BM212(k) = (sum((W212.*(y_tot_2a-mu212(k))).^2)/(n2*sum(W212)^2));
    BM221(k) = (sum((W221.*(y_tot_2a-mu221(k))).^2)/(n2*sum(W221)^2));
    BM222(k) = (sum((W222.*(y_tot_2a-mu222(k))).^2)/(n2*sum(W222)^2));

    %% covariance for SRTPA
    C12_3(k) = sum((W111.*(y_tot-mu111(k))).*(W112.*(y_tot-mu112(k))))
/(n1^2);
    C13_3(k) = sum((W111.*(y_tot-mu111(k))).*(W121.*(y_tot-mu121(k))))
/(n1^2);
    C14_3(k) = sum((W111.*(y_tot-mu111(k))).*(W122.*(y_tot-mu122(k))))
/(n1^2);
    C23_3(k) = sum((W112.*(y_tot-mu112(k))).*(W121.*(y_tot-mu121(k))))
/(n1^2);
    C24_3(k) = sum((W112.*(y_tot-mu112(k))).*(W122.*(y_tot-mu122(k))))
/(n1^2);
    C34_3(k) = sum((W121.*(y_tot-mu121(k))).*(W122.*(y_tot-mu122(k))))
/(n1^2);
    C12_3_2a(k) = sum((W211.*(y_tot_2a-mu211(k))).*
(W212.*(y_tot_2a-mu212(k))))/(n2^2);
    C13_3_2a(k) = sum((W211.*(y_tot_2a-mu211(k))).*
(W221.*(y_tot_2a-mu221(k))))/(n2^2);
    C14_3_2a(k) = sum((W211.*(y_tot_2a-mu211(k))).*
(W222.*(y_tot_2a-mu222(k))))/(n2^2);
    C23_3_2a(k) = sum((W212.*(y_tot_2a-mu212(k))).*
(W221.*(y_tot_2a-mu221(k))))/(n2^2);
    C24_3_2a(k) = sum((W212.*(y_tot_2a-mu212(k))).*
(W222.*(y_tot_2a-mu222(k))))/(n2^2);
    C34_3_2a(k) = sum((W221.*(y_tot_2a-mu221(k))).*
(W222.*(y_tot_2a-mu222(k))))/(n2^2);
```

```
    %% to test 8 strategies
    new_contrast = [1 0 -1 0 0 0 0 0 ;0 1 0 -1 0 0 0 0 ; ...
        1 -1 0 0 0 0 0 0 ;0 0 0 0 1 0 -1 0;0 0 0 0 0 1 0 -1;
0 0 0 0 1 -1 0 0];
    %% testing in URT
    new_mean1 = [m111(k) m112(k) m121(k) m122(k) m211(k) m212(k)
 m221(k) m222(k)];
    %% var-cov matrix for URT
    new_cov1 = [s111_1(k)/size1 0 0 0 0 0 0 0;
0 s112_1(k)/size2 0 0 0 0 0 0;...
        0 0 s121_1(k)/size3 0 0 0 0 0; 0 0 0 s122_1(k)/size4 0 0 0 0;...
        0 0 0 0 s111_1_2a(k)/size1_2a 0 0 0;
0 0 0 0 0 s112_1_2a(k)/size2_2a 0 0;...
        0 0 0 0 0 0 s121_1_2a(k)/size3_2a 0;
0 0 0 0 0 0 0 s122_1_2a(k)/size4_2a];
    ts_tot_1(k) = new_mean1*new_contrast'*
inv(new_contrast*new_cov1*new_contrast')*new_contrast*new_mean1';
    %% testing in SRTPA
    new_mean = [mu111(k) mu112(k) mu121(k) mu122(k) mu211(k)
mu212(k) mu221(k) mu222(k)];
    new_cov = [BM111(k) C12_3(k) C13_3(k) C14_3(k) 0 0 0 0 ;
C12_3(k) BM112(k) C23_3(k) C24_3(k) 0 0 0 0;...
        C13_3(k) C23_3(k) BM121(k) C34_3(k) 0 0 0 0;
C14_3(k) C24_3(k) C34_3(k) BM122(k) 0 0 0 0 ;...
        0 0 0 0 BM211(k) C12_3_2a(k) C13_3_2a(k) C14_3_2a(k);
0 0 0 0 C12_3_2a(k) BM212(k) C23_3_2a(k) C24_3_2a(k);...
        0 0 0 0 C13_3_2a(k) C23_3_2a(k) BM221(k) C34_3_2a(k);
0 0 0 0 C14_3_2a(k) C24_3_2a(k) C34_3_2a(k) BM222(k)];
    ts_tot_3(k) = new_mean*new_contrast'*
```

```
inv(new_contrast*new_cov*new_contrast')*new_contrast*new_mean';

    %% A1, A2 testing
    %% mean matrices
    mean_1 = [m111(k) m112(k) m121(k) m122(k)]; % URT A1
    mean_2 = [nmu111(k) nmu112(k) nmu121(k) nmu122(k)]; % SRT A1
    mean_3 = [mu111(k) mu112(k) mu121(k) mu122(k)]; % SRTPA A1
    mean_1_2a = [m211(k) m212(k) m221(k) m222(k)]; %URT A2
    mean_2_2a = [nmu211(k) nmu212(k) nmu221(k) nmu222(k)]; %SRT A2
    mean_3_2a = [mu211(k) mu212(k) mu221(k) mu222(k)]; %SRTPA A2
    contrast = [1 0 -1 0;0 1 0 -1; 1 -1 0 0];
    %% Var-Cov matrix for URT
    v_cov_1 = [s111_1(k)/size1 0 0 0; 0 s112_1(k)/size2 0 0;...
        0 0 s121_1(k)/size3 0; 0 0 0 s122_1(k)/size4];
    v_cov_1_2a = [s111_1_2a(k)/size1_2a 0 0 0;
0 s112_1_2a(k)/size2_2a 0 0;...
        0 0 s121_1_2a(k)/size3_2a 0; 0 0 0 s122_1_2a(k)/size4_2a];
    %% Var-Cov matrix for SRT
    v_cov_2 = [B111(k) C12_2(k) C13_2(k) C14_2(k);
C12_2(k) B112(k) C23_2(k) C24_2(k);...
        C13_2(k) C23_2(k) B121(k) C34_2(k); C14_2(k) C24_2(k)
C34_2(k) B122(k)];
    v_cov_2_2a = [B211(k) C12_2_2a(k) C13_2_2a(k) C14_2_2a(k);
C12_2_2a(k) B212(k) C23_2_2a(k) C24_2_2a(k);...
        C13_2_2a(k) C23_2_2a(k) B221(k) C34_2_2a(k); C14_2_2a(k)
C24_2_2a(k) C34_2_2a(k) B222(k)];
    %% Var-Cov matrix for SRTPA
    v_cov_3 = [BM111(k) C12_3(k) C13_3(k) C14_3(k); C12_3(k)
BM112(k) C23_3(k) C24_3(k);...
        C13_3(k) C23_3(k) BM121(k) C34_3(k); C14_3(k) C24_3(k)
```

```
C34_3(k) BM122(k)];
    v_cov_3_2a = [BM211(k) C12_3_2a(k) C13_3_2a(k) C14_3_2a(k);
C12_3_2a(k) BM212(k) C23_3_2a(k) C24_3_2a(k);...
        C13_3_2a(k) C23_3_2a(k) BM221(k) C34_3_2a(k); C14_3_2a(k)
C24_3_2a(k) C34_3_2a(k) BM222(k)];
    %% Test Statistics
    %% URT A1
    ts_1(k) = mean_1*contrast'*inv(contrast*v_cov_1*contrast')*
contrast*mean_1';
    %% SRTPA A1
    ts_3(k) = mean_3*contrast'*inv(contrast*v_cov_3*contrast')*
contrast*mean_3';
    %% URT A2
    ts_1_2a(k) =
mean_1_2a*contrast'*inv(contrast*v_cov_1_2a*contrast')*contrast*mean_1_2a';
    %% SRTPA A2
    ts_3_2a(k) =
mean_3_2a*contrast'*inv(contrast*v_cov_3_2a*contrast')*contrast*mean_3_2a';
% ========================================================================
%       comparing test statistics with chi-sq critical values
% ========================================================================
    %% for URT
    if abs(ts_1(k)) >= 7.82
        count1(k) = 1;
    else count1(k) = 0;
    end
    if abs(ts_1_2a(k)) >= 7.82
        count1_2a(k) = 1;
    else count1_2a(k) = 0;
    end
```

```
    if abs(ts_tot_1(k)) >= 14.02
        count_tot_1(k) = 1;
    else count_tot_1(k) = 0;
    end
    %% for SRTPA
    if abs(ts_tot_3(k)) >= 14.02
        count_tot_3(k) = 1;
    else count_tot_3(k) = 0;
    end
    if abs(ts_3(k)) >= 7.82
        count3(k) = 1;
    else count3(k) = 0;
    end
    if abs(ts_3_2a(k)) >= 7.82
        count3_2a(k) = 1;
    else count3_2a(k) = 0;
    end

end


%%=========================================================================
%       Coverage probabilities
%%=========================================================================
%% true means for A1
t_mu111 = piR1*m11d + (1-piR1)*m1d1;
t_mu112 = piR1*m11d + (1-piR1)*m1d2;
t_mu121 = piR1*m12d + (1-piR1)*m1d1;
t_mu122 = piR1*m12d + (1-piR1)*m1d2;
%% true means for A2
```

```
t_mu211 = piR2*m21d + (1-piR2)*m2d1;
t_mu212 = piR2*m21d + (1-piR2)*m2d2;
t_mu221 = piR2*m22d + (1-piR2)*m2d1;
t_mu222 = piR2*m22d + (1-piR2)*m2d2;
%% Confidence interval in URT
U_CL111 = m111 - 1.96*sqrt(s111_1/size1);
U_CU111 = m111 + 1.96*sqrt(s111_1/size1);
U_CL112 = m112 - 1.96*sqrt(s112_1/size2);
U_CU112 = m112 + 1.96*sqrt(s112_1/size2);
U_CL121 = m121 - 1.96*sqrt(s121_1/size3);
U_CU121 = m121 + 1.96*sqrt(s121_1/size3);
U_CL122 = m122 - 1.96*sqrt(s122_1/size4);
U_CU122 = m122 + 1.96*sqrt(s122_1/size4);
U_CL211 = m211 - 1.96*sqrt(s111_1_2a/size1_2a);
U_CU211 = m211 + 1.96*sqrt(s111_1_2a/size1_2a);
U_CL212 = m212 - 1.96*sqrt(s112_1_2a/size2_2a);
U_CU212 = m212 + 1.96*sqrt(s112_1_2a/size2_2a);
U_CL221 = m221 - 1.96*sqrt(s121_1_2a/size3_2a);
U_CU221 = m221 + 1.96*sqrt(s121_1_2a/size3_2a);
U_CL222 = m222 - 1.96*sqrt(s122_1_2a/size4_2a);
U_CU222 = m222 + 1.96*sqrt(s122_1_2a/size4_2a);
%% coverage probability in URT
U_CP111 = (t_mu111 >= U_CL111) .* (t_mu111 <= U_CU111);
U_CP112 = (t_mu112 >= U_CL112) .* (t_mu112 <= U_CU112);
U_CP121 = (t_mu121 >= U_CL121) .* (t_mu121 <= U_CU121);
U_CP122 = (t_mu122 >= U_CL122) .* (t_mu122 <= U_CU122);
U_CP211 = (t_mu211 >= U_CL211) .* (t_mu211 <= U_CU211);
U_CP212 = (t_mu212 >= U_CL212) .* (t_mu212 <= U_CU212);
U_CP221 = (t_mu221 >= U_CL221) .* (t_mu221 <= U_CU221);
U_CP222 = (t_mu222 >= U_CL222) .* (t_mu222 <= U_CU222);
```

```
%% Confidence interval in SRTPA
CL111 = mu111 - 1.96*sqrt(BM111);
CU111 = mu111 + 1.96*sqrt(BM111);
CL112 = mu112 - 1.96*sqrt(BM112);
CU112 = mu112 + 1.96*sqrt(BM112);
CL121 = mu121 - 1.96*sqrt(BM121);
CU121 = mu121 + 1.96*sqrt(BM121);
CL122 = mu122 - 1.96*sqrt(BM122);
CU122 = mu122 + 1.96*sqrt(BM122);
CL211 = mu211 - 1.96*sqrt(BM211);
CU211 = mu211 + 1.96*sqrt(BM211);
CL212 = mu212 - 1.96*sqrt(BM212);
CU212 = mu212 + 1.96*sqrt(BM212);
CL221 = mu221 - 1.96*sqrt(BM221);
CU221 = mu221 + 1.96*sqrt(BM221);
CL222 = mu222 - 1.96*sqrt(BM222);
CU222 = mu222 + 1.96*sqrt(BM222);
%% coverage probability in SRTPA
CP111 = (t_mu111 >= CL111) .* (t_mu111 <= CU111);
CP112 = (t_mu112 >= CL112) .* (t_mu112 <= CU112);
CP121 = (t_mu121 >= CL121) .* (t_mu121 <= CU121);
CP122 = (t_mu122 >= CL122) .* (t_mu122 <= CU122);
CP211 = (t_mu211 >= CL211) .* (t_mu211 <= CU211);
CP212 = (t_mu212 >= CL212) .* (t_mu212 <= CU212);
CP221 = (t_mu221 >= CL221) .* (t_mu221 <= CU221);
CP222 = (t_mu222 >= CL222) .* (t_mu222 <= CU222);

% latex(P2,'%.3f')
% savefile1 = ['VC55' num2str(piR1*10) '51.mat'];
% save(savefile1, 'P*')
```

```
% save savefile Policy*
% msave('VC5551.mat','P*');
% save VC55581.mat Policy*

%         end
%     end
% end
%  diary vc_table_out11


PPi = [piA piR1 piR2 piB piC ];
PPower = [sum(count_tot_1)/nos sum(count_tot_3)/nos];
P1 = [ mean(m111') sqrt(mean(s111_1)/size1)
((t_mu111 - mean(m111'))/t_mu111)*100  mean(U_CP111') sum(count1)/nos...
    mean(mu111') sqrt(mean(BM111))  ((t_mu111 - mean(mu111'))/t_mu111)*
100 mean(CP111') sum(count3)/nos;...
    mean(m112') sqrt(mean(s112_1)/size2) ((t_mu112 - mean(m112'))/t_mu112)*
100 mean(U_CP112') sum(count1)/nos...
    mean(mu112') sqrt(mean(BM112))  ((t_mu112 - mean(mu112'))/t_mu112)*
100 mean(CP112') sum(count3)/nos;...
    mean(m121') sqrt(mean(s121_1)/size3) ((t_mu121 - mean(m121'))/t_mu121)*
100  mean(U_CP121') sum(count1)/nos...
    mean(mu121') sqrt(mean(BM121))  ((t_mu121 - mean(mu121'))/t_mu121)*
100 mean(CP121') sum(count3)/nos;...
    mean(m122') sqrt(mean(s122_1)/size4)  ((t_mu222 - mean(m122'))/t_mu122)*
100 mean(U_CP122') sum(count1)/nos...
    mean(mu122') sqrt(mean(BM122))  ((t_mu122 - mean(mu122'))/t_mu122)*
100 mean(CP122') sum(count3)/nos;...
    mean(m211') sqrt(mean(s111_1_2a)/size1_2a)
((t_mu211 - mean(m211'))/t_mu211)*100  mean(U_CP211') sum(count1_2a)/nos...
```

```
    mean(mu211') sqrt(mean(BM211))  ((t_mu211 - mean(mu211'))/t_mu211)*
100 mean(CP211') sum(count3_2a)/nos;...
    mean(m212') sqrt(mean(s112_1_2a)/size2_2a)
((t_mu212 - mean(m212'))/t_mu212)*100 mean(U_CP212') sum(count1_2a)/nos...
    mean(mu212') sqrt(mean(BM212))  ((t_mu212 - mean(mu212'))/t_mu212)*
100 mean(CP212') sum(count3_2a)/nos;...
    mean(m221') sqrt(mean(s121_1_2a)/size3_2a)
((t_mu221 - mean(m221'))/t_mu221)*100 mean(U_CP221') sum(count1_2a)/nos...
    mean(mu221') sqrt(mean(BM221))  ((t_mu221 - mean(mu221'))/t_mu221)*
100 mean(CP221') sum(count3_2a)/nos;...
    mean(m222') sqrt(mean(s122_1_2a)/size4_2a)
((t_mu222 - mean(m222'))/t_mu222)*100 mean(U_CP222') sum(count1_2a)/nos...
    mean(mu222') sqrt(mean(BM222))  ((t_mu222 - mean(mu222'))/t_mu222)*
100 mean(CP222') sum(count3_2a)/nos];...


P2 =[mean(nmu111') sqrt(mean(B111)) ((t_mu111 - mean(nmu111'))/t_mu111)*100;
    mean(nmu112') sqrt(mean(B112)) ((t_mu112 - mean(nmu112'))/t_mu112)*100;
    mean(nmu121') sqrt(mean(B121)) ((t_mu121 - mean(nmu121'))/t_mu121)*100;
    mean(nmu122') sqrt(mean(B122)) ((t_mu122 - mean(nmu122'))/t_mu122)*100;
    mean(nmu211') sqrt(mean(B211)) ((t_mu211 - mean(nmu211'))/t_mu211)*100;
    mean(nmu212') sqrt(mean(B212)) ((t_mu212 - mean(nmu212'))/t_mu212)*100;
    mean(nmu221') sqrt(mean(B221)) ((t_mu221 - mean(nmu221'))/t_mu221)*100;
    mean(nmu222') sqrt(mean(B222)) ((t_mu222 - mean(nmu222'))/t_mu222)*100];

%diary off

disp('pi0 & piR1 & piR2 & piB1 & piB2 & piC1 & piC2')
latex(PPi,'%.1f')
latex(PPower,'%.3f')
```

```
disp('True & Estimates &  SE &  RelativeBias &  CP  & Power')
latex(P1,'%.2f')
sum(count_tot_1)/nos
sum(count_tot_3)/nos
% latex(P2,'%.2f')
%% True values are same under null no matter what response rates are
%% applied
```

A.2 NONPARAMETRIC INFERENCE ON MEDIAN RESIDUAL LIFE FUNCTION IN SEQUENTIALLY RANDOMIZED TRIALS

```
clear all;

%same functions needed from previous code
%parameters
% ======================================================================
% Leukemia data set analysis
% ======================================================================
Pi_z = 0.5; % equally assigned to both maintenance treatments
t0 = 250; % time of interest
trtA = 1; %depend on induntion treatment option
% trtA = 2;
[id trt1 resp resp_time consent sec_rand_time trt2 death U age wbc]
= textread('C:\calgb1_.txt','%d %d %d %d %d %d %d %d %d %d %f ');
R = resp.*consent;
if trt2 == 2
    trt2 = 0;
```

```
end
Z = trt2;
D = death; % 0 for censored
n = length(D);
for i=1:n
    if (Z(i)==2)
       Z(i) = 0;
     else Z(i) = Z(i);
     end
     if D(i)==0
        D1(i)=1;
     else D1(i)=0;
     end
end
D1=D1';

%  for first or second treatment option in induction treatment
index_vector = find(trt1==trtA);
R = R(index_vector);
Z = Z(index_vector);
D = D(index_vector);
D1 = D1(index_vector);
U = U(index_vector);
resp_time = resp_time(index_vector);

n = length(D);

% Kaplan-Meier Estimators
[K,x] = ecdf(U,'censoring',D,'function','survivor');
[S,x1] = ecdf(U,'censoring',D1,'function','survivor');
```

```
K = fOne(U,x,K);
S = fOne(U,x1,S);

for j = 1:n
    if K(j) == 0 K(j) = K(j-1); end
    if S(j) == 0 S(j) = S(j-1); end
end

% weight functions
Q2=1-R + ((R.*(1-Z))/(1- Pi_z));
Q1=1-R + ((R.*Z)/Pi_z);

Cens = D./K';
K = K';
save FileforSecant D K Q1 Q2 U t0 n;

%Secant method to find Theta
[Theta31,fncvalue,err,hist] = secant1('secfunc1',0,500,1.0e-16,
1.0e-16,10000);
[Theta32,fncvalue,err,hist] = secant1('secfunc2',0,500,1.0e-16,
1.0e-16,10000);
K = K';

% Survival for each time point;
for j = 1:n
    S01 = Cens.*Q1.*(U>U(j));
    S02 = Cens.*Q2.*(U>U(j));
    S1(j) = sum(S01)/sum(Cens.*Q1);
    S2(j) = sum(S02)/sum(Cens.*Q2);
end
```

```
        S1_t0 =  fOne(t0,U,S1);
        S2_t0 =  fOne(t0,U,S2);

        S_half1 = 0.5*S1;
        S_half2 = 0.5*S2;
        S_half1_t0 =  fOne(t0,U,S_half1);
        S_half2_t0 =  fOne(t0,U,S_half2);

        %Find S inverse for 0.5*S(t)
        Sinv1 = fOne(S_half1, S1, U);
        Sinv2 = fOne(S_half2, S2, U);
        Sinv1_t0 =  fOne(t0,U,Sinv1);
        Sinv2_t0 =  fOne(t0,U,Sinv2);

        %kernel density to find density of S
        [den1,xi1] = ksdensity(U,'weights',Q1,'censoring',D1,
'npoints',n);
        [den2,xi2] = ksdensity(U,'weights',Q2,'censoring',D1,
'npoints',n);
        Den21 = fOne(U,xi1,den1);
        Den22 = fOne(U,xi2,den2);
        Den21_t0 = fOne(t0,U,Den21);
        Den22_t0 = fOne(t0,U,Den22);

% MERL Theta 1
for j = 1:n
    Theta11(j) = Sinv1(j) - U(j);
    Theta12(j) = Sinv2(j) - U(j);
end
```

```
        Theta11_t0 = fOne(t0,U,Theta11);
        Theta12_t0 = fOne(t0,U,Theta12);

        % density for theta 1 +t0 and theta3+t0
        Den31_the = fOne(Theta31+t0,U,Den21);
        Den32_the = fOne(Theta32+t0,U,Den22);
        Den11_the_new = fOne(Theta11_t0+t0, U,Den21);
        Den12_the_new = fOne(Theta12_t0+t0, U,Den22);

        U_new1 = U' + Theta11;
        U_new2 = U' + Theta12;

        [den1_new,xi1_new] = ksdensity(U_new1,'npoints',n,
'weights',Q1);
        [den2_new,xi2_new] = ksdensity(U_new2,'npoints',n,
'weights',Q2);
        Den11_the = fOne(U_new1,xi1_new,den1_new);
        Den12_the = fOne(U_new2,xi2_new,den2_new);



% /* Y(u) and Y(u)^-1
for j = 1:n
    Y(j) = sum(U >= U(j));
    Y_inv(j) = (sum(U >= U(j))).^-1;
end

%Aalan-Nelson Est.
for j = 1:n
    if (U(j) < t0) & (D(j) ==1)
```

```
        ANe(j) = sum(Y_inv(j));
    end
end

%finding variances
for j = 1:n
    for k = 1:n
        G11(j,k) = sum(Cens.*Q1.*((U>U_new1(j))- 0.5.*(U>U(j)))
.*(U>=U(k))./(n.*S(k));
        G12(j,k) = sum(Cens.*Q2.*((U>U_new1(j))- 0.5.*(U>U(j)))
.*(U>=U(k))./(n.*S(k));
        E11(j,k) = sum(Cens.*(Q1.*((U>U_new1(j))- 0.5.*(U>U(j)))
-G11(j,k)).^2.*(U >= U(k)))/n;
        E12(j,k) = sum(Cens.*(Q2.*((U>U_new1(j))- 0.5.*(U>U(j)))
-G12(j,k)).^2.*(U >= U(k)))/n;
        G21(j,k) = sum(Cens.*Q1.*(U>U(j)).*(U>=U(k))./(n.*S(k));
        G22(j,k) = sum(Cens.*Q2.*(U>U(j)).*(U>=U(k))./(n.*S(k));
        E21(j,k) = sum(Cens.*(Q1.*(U>U(j))-G21(j,k)).^2.*
(U >= U(k)))/n;
        E22(j,k) = sum(Cens.*(Q2.*(U>U(j))-G22(j,k)).^2.*
(U >= U(k)))/n;
    end
end

for j = 1:n
    V_second11(j) = sum((1-D)'.*E11(j,:)./(K.*Y));
    V_second12(j) = sum((1-D)'.*E12(j,:)./(K.*Y));
    V_second21(j) = sum((1-D)'.*E21(j,:)./(K.*Y));
    V_second22(j) = sum((1-D)'.*E22(j,:)./(K.*Y));
    V_first11(j) = sum(Cens.*((Q1.*((U>U_new1(j))- 0.5
```

```
.*(U>U(j))))).^2))/n;
    V_first12(j) = sum(Cens.*((Q2.*((U>U_new1(j))- 0.5
.*(U>U(j))))).^2))/n;
    V_first21(j) = sum(Cens.*((Q1.*(U>U(j))-S1(j)).^2))/n;
    V_first22(j) = sum(Cens.*((Q2.*(U>U(j))-S2(j)).^2))/n;
    var11(j) = (V_first11(j) + V_second11(j))/n;
    var12(j) = (V_first12(j) + V_second12(j))/n;
    var21(j) = (V_first21(j) + V_second21(j))/n;
    var22(j) = (V_first22(j) + V_second22(j))/n;
    Var11(j) = var11(j)/(Den11_the(j)^2);
    Var12(j) = var12(j)/(Den12_the(j)^2);
    Var21(j) = var21(j)/(4*(Den21(j)^2));
    Var22(j) = var22(j)/(4*(Den22(j)^2));
    Se11(j) = sqrt(Var11(j));
    Se12(j) = sqrt(Var12(j));
    Se21(j) = sqrt(Var21(j));
    Se22(j) = sqrt(Var22(j));
    CL_L11(j) = Theta11(j) -1.96* Se11(j);
    CL_U11(j) = Theta11(j) + 1.96*Se11(j);
    CL_L12(j) = Theta12(j) -1.96* Se12(j);
    CL_U12(j) = Theta12(j) + 1.96*Se12(j);
    CL_L21(j) = Theta11(j) -1.96* Se21(j);
    CL_U21(j) = Theta11(j) + 1.96*Se21(j);
    CL_L22(j) = Theta12(j) -1.96* Se22(j);
    CL_U22(j) = Theta12(j) + 1.96*Se22(j);
end

Se11_t0 = fOne(t0,U,Se11);
Se12_t0 = fOne(t0,U,Se12);
Se21_t0 = fOne(t0,U,Se21);
```

```
Se22_t0 = fOne(t0,U,Se22);


CL_L11_t0 = fOne(t0,U,CL_L11);
CL_U11_t0 = fOne(t0,U,CL_U11);
CL_L12_t0 = fOne(t0,U,CL_L12);
CL_U12_t0 = fOne(t0,U,CL_U12);

% sandwich estimator for theta 1
AA1 = Den11_the_new.^2;
AA2 = Den12_the_new.^2;
BB1 = sum(((Cens.*Q1.*((U>t0+Theta11_t0)-0.5.*(U>t0)))).^2)/
(sum(Cens.*Q1));
BB2 = sum(((Cens.*Q2.*((U>t0+Theta12_t0)-0.5.*(U>t0)))).^2)/
(sum(Cens.*Q2));
Var31 =  (BB1/AA1)/n;
Var32 =  (BB2/AA2)/n;
Se31 = sqrt(Var31);
Se32 = sqrt(Var32);

% sandwich estimator for theta 3 from secant
AA31 = Den31_the.^2;
AA32 = Den32_the.^2;
BB31 = sum(((Cens.*Q1.*((U>t0+Theta31)-0.5.*(U>t0)))).^2)/
(sum(Cens.*Q1));
BB32 = sum(((Cens.*Q2.*((U>t0+Theta32)-0.5.*(U>t0)))).^2)/
(sum(Cens.*Q2));
Var331 =  (BB31/AA31)/n;
Var332 =  (BB32/AA32)/n;
Se331 = sqrt(Var331);
Se332 = sqrt(Var332);
```

```
 K = K';
 psi11 = (D./K).*Q1.*(U>t0).*R.*(resp_time<=t0).*Z;
    psi12 = (D./K).*Q1.*(U>t0).*(1-R).*(resp_time<=t0).*Z;

    psi31 = (D./K).*Q2.*(U>t0).*R.*(resp_time<=t0).*Z;
    psi32 = (D./K).*Q2.*(U>t0).*(1-R).*(resp_time<=t0).*Z;

    psi21 = (D./K).*R.*(resp_time<=t0);
    psi22 = (D./K).*(1-R).*(resp_time<=t0);

    %=========================================================================
    % Variance by M-estimation
    %=========================================================================
    A1 = [Den11_the_new 0; 0 -1];
    B1 = [sum(psi11.^2)/n sum(psi11.*psi21)/n; sum(psi11.*psi21)/n
 sum(psi21.^2)/n];
    invA1 = inv(A1);
    V1 = (invA1*B1*invA1')./n;

    A2 = [Den12_the_new 0; 0 -1];
    B2 = [mean(psi11.^2) mean(psi12.*psi22); mean(psi12.*psi22)
mean(psi22.^2)];
    invA2 = inv(A2);
    V2 = (invA2*B2*invA2')./n;

    new_var = V1+V2;
```

```
out = [Theta11_t0 Se11_t0 Se31 Theta31 Se331 Theta41_t0 Se41];
out1= [Theta12_t0 Se12_t0 Se32 Theta32 Se332 Theta42_t0 Se42];
latex(out,'%.2f')
latex(out1,'%.2f')

% ========================================================================
% Simulation study for Median Residual Life Function
% ========================================================================

A = 3.5;
Pi_r = 0.4;
Pi_z = 0.5;
t0 = 365;
nu = 2;
rho = 0.2;
%1000 iterations and sample size
nos = 1000;
n = 200;

%% values for different scenrios
% mu31 = 267; mu32 = 235; %183, 0.4, 0.5
mu31=234.3;  mu32=193.4; %365, 0.4, 0.5

% %  mu31 = 341.9; mu32 = 285.6; %183 0705
% mu31 = 262.9; mu32 = 209.5; %365 0705


% mu31 = 273.9; mu32 = 239.9; %183 0403
```

```
% mu31 = 234.3; mu32 = 193.4; %365 0403
% mu31 = 343.5; mu32 = 285.1; %183 0703
% mu31 = 262.1; mu32 = 210.4; %365 0703

for l = 1:nos
    %C for Censroing~Uni(0,A);
    c = rand(n,1);
    C = A*c;
    ind1 = 0;
    ind2 = 0;
    index2 = 0;
    index3 = 0;
    index4 = 0;
    r = rand(n,1);
    z = rand(n,1);
    % **Generating R from b(1,Pi_r) for Consent indicator;
    R = zeros(n,1);
    Z = zeros(n,1);
    Tnr = zeros(n,1);
    Ta = zeros(n,1);
    T1 = zeros(n,1);
    T2 = zeros(n,1);

    for i = 1:n
        % indicator for receiving second treatment
        if r(i) < Pi_r
            R(i,1) = 1;
        else
            R(i,1) = 0;
        end
```

```
    end
    for i = 1:n
        if R(i) == 1
            % Generating Z from bin(1, Pi_z) for trt B1 indicator;
            if z(i) < Pi_z
                Z(i,1) = 1;
            else
                Z(i,1) = 0;
            end
        else
            Z(i,1) = -1;
        end
    end

    ind1 = find(R==0);
    ind2 = find(R==1);
    Tnr(ind1) = expinv(rand(length(ind1),1),0.5);
% failure times are exponential(exp(beta1))under R=0
    Ta(ind2) = expinv(rand(length(ind2),1),0.1);
% failure times are exponential(exp(beta1))under R=1
    u11 = rand(length(ind2),1);
    u12 = rand(length(ind2),1);
    T1(ind2) = ((1/rho).*(-exp(-0.6).*log(1-u11))).^(1/nu);
    T2(ind2) = ((1/rho).*(-exp(-0.9).*log(1-u12))).^(1/nu);
    T = ((R.*((Z).*(Ta + T1) + (1-Z).*(Ta + T2)) + (1-R).*Tnr));
    Tr = R.*Ta + (1-R).*Tnr;
    U = min(T,C);
    D = (T <= C); % 1 for not censoring
    data = [U R Z D Tr Ta Tnr];
    data = sortrows(data);
```

```
U = data(:,1);
R = data(:,2);
Z = data(:,3);
D = data(:,4);
Tr = data(:,5);
Ta = data(:,6);
Tnr = data(:,7);
D(n,1) = 1;
D1 = 1-D;
U=U*365.25;
Tr = Tr*365.25;
Ta = Ta*365.35;
Tnr = Tnr*365.25;

% weight
Q1 = (1-R) + ((R.*Z)./Pi_z);
Q2 = (1-R) + ((R.*(1-Z))./(1-Pi_z));

%KMe
[K,sk] = kmest(U,D);
[S,ss] = kmest(U,D1);
% save files for EEB method
save FileforSecant D K Q1 Q2 U t0 n;

%=========================================================================
%       SSB method & Taylor
%=========================================================================
% SampleSize adjusted survival estimates for each time point;
S1 = 0; S2 = 0;
for j = 1:n
```

```
    S1(j) = 1-sum(D./K.*Q1.*(U<=U(j)))/sum(D./K.*Q1);
    S2(j) = 1-sum(D./K.*Q2.*(U<=U(j)))/sum(D./K.*Q2);
end
%     S1 = S_1(:,l);
%     S2 = S_2(:,l);
S1_t0 = fOne(t0,U,S1);
S2_t0 = fOne(t0,U,S2);

[S_half1 Sinv1 Sinv1_t0(l) S_half1_t0] = S_find(S1, U, t0);
[S_half2 Sinv2 Sinv2_t0(l) S_half2_t0] = S_find(S2, U, t0);
[Den21 Den21_t0] = denS(U, Q1, D1, t0);
[Den22 Den22_t0] = denS(U, Q2, D1, t0);

% **MERL Theta 1 and 2**;
Theta11_t0(l) = Sinv1_t0(l) - t0;
Theta12_t0(l) = Sinv2_t0(l) - t0;
Theta21_t0(l) = S_half1_t0/Den21_t0;
Theta22_t0(l) = S_half2_t0/Den22_t0;

% density for theta1+t0
Den11_the_new = fOne(Theta11_t0(l)+t0, U,Den21);
Den12_the_new = fOne(Theta12_t0(l)+t0, U,Den22);
%theta plus t0
U_new1_t0 = t0 + Theta11_t0(l);
U_new2_t0 = t0 + Theta12_t0(l);
% Y(u) and Y(u)^-1
Y = 0; Y_inv = 0;
for j = 1:n
    Y(j) = sum(U >= U(j));
    Y_inv(j) = (sum(U >= U(j))).^-1;
```

```
    end
    % Var and CP by Lunceford
    [Se11(l) Cp11(l)] = CP5(D, K, Q1, U, U_new1_t0, t0, S, Y,
Theta11_t0(l), mu31, Den11_the_new);
    [Se12(l) Cp12(l)] = CP5(D, K, Q2, U, U_new2_t0, t0, S, Y,
Theta12_t0(l), mu32, Den12_the_new);
    [Se21(l) Cp21(l)] = CP6(D, K, Q1, U, t0, S, Y, Theta11_t0(l),
mu31, Den21_t0, S1_t0);
    [Se22(l) Cp22(l)] = CP6(D, K, Q2, U, t0, S, Y, Theta12_t0(l),
mu32, Den22_t0, S2_t0);
    % Var and CP by Sandwich for theta 1
    [Var31 Var32 Se31(l) Se32(l) Cp31(l) Cp32(l)] =
       CP1(Den11_the_new, Den12_the_new, D, K, Q1, Q2, U, t0,
        Theta11_t0(l), Theta12_t0(l), mu31, mu32);


    %=========================================================================
    %       EEB method
    %=========================================================================
    % estimates for MERL times by EEB
    [Theta31(l),fncvalue31(l),err,hist] =
secant1('secfunc1',0,400,1.0e-16,1.0e-16,10000);
    [Theta32(l),fncvalue32(l),err,hist] =
secant1('secfunc2',0,400,1.0e-16,1.0e-16,10000);
    % Density for theta3 + t0
    Den31_t0 = fOne(Theta31(l)+t0,U,Den21);
    Den32_t0 = fOne(Theta32(l)+t0,U,Den22);
    % Var and CP for theta 3 from secant by sandwich
    [Var331 Var332 Se331(l) Se332(l) Cp331(l) Cp332(l)] = ...
        CP1(Den31_t0, Den32_t0, D, K, Q1, Q2, U, t0,
Theta31(l), Theta32(l), mu31, mu32);
```

```
end

Out = [mean(Theta11_t0) mean(Se11) sqrt(var(Theta11_t0)) mean(Cp11) ;
    mean(Theta12_t0) mean(Se12) sqrt(var(Theta12_t0)) mean(Cp12);
    mean(Theta11_t0) mean(Se31) sqrt(var(Theta11_t0)) mean(Cp31) ;
    mean(Theta12_t0) mean(Se32) sqrt(var(Theta12_t0)) mean(Cp32) ;
    mean(Theta21_t0) mean(Se21) sqrt(var(Theta21_t0)) mean(Cp21) ;
    mean(Theta22_t0) mean(Se22) sqrt(var(Theta22_t0)) mean(Cp22) ;
    mean(Theta31) mean(Se331) sqrt(var(Theta31)) mean(Cp331) ;
    mean(Theta32) mean(Se332) sqrt(var(Theta32)) mean(Cp332);
    mean(Theta41_t0) mean(Se41) sqrt(var(Theta41_t0)) mean(Cp41);
    mean(Theta42_t0) mean(Se42) sqrt(var(Theta42_t0)) mean(Cp42)];
out = latex(Out,'%.2f')

[mean(Se51) mean(Se52)]



% save MERL_0812_R0705_N200_T183.mat
%Relative bias
RB11 = ((mu31 - mean(Theta11_t0))/mu31)*100;
RB12 = ((mu32 - mean(Theta12_t0))/mu32)*100;
RB31 = ((mu31 - mean(Theta31))/mu31)*100;
RB32 = ((mu32 - mean(Theta32))/mu32)*100;
RB41 = ((mu31 - mean(Theta41_t0))/mu31)*100;
RB42 = ((mu32 - mean(Theta42_t0))/mu32)*100;

Out1 = [RB11 RB12 RB31 RB32 RB41 RB42];
out1 = latex(Out1,'%.1f')
```

BIBLIOGRAPHY

Bandos H. (2007). Regression on Median Residual Life Function for Censored Survival Data. *University of Pittsburgh, Pittsburgh.*

Chung, C. F. (1989). Confidence bands for percentile residual lifetime under random censorship model. *Journal of Multivariate Analysis*, **29**, 94–126

Cole, S. R. and Frangakisb, C. E.(2009). The Consistency Statement in Causal Inference : A Definition or an Assumption?. *Epidemiology*, **20**, 1, 3–5

Cox, D. R. (1972). Regression models and life tables (with Discussion). *J. R. Statist. Soc. B*, **34**, 187–220.

Csörgö, M. and Csörgö, S. (1987). Estimation of percentile residual life. *Operations Research*, **35**, 598–606.

Feng, W. and Wahed, A. S.(2008). Supremum weighted log-rank test and sample size for comparing two-stage adaptive treatment strategies. *Biometrika*, **95**, 695–707.

Feng, Z. and Kulasekera, K. B. (1991). Nonparametric estimation of the percentile residual life function. *Communication in Statistics: Theory and Methods*,**20**, 87–105.

Guo, X. (2005). Statistical Analysis in Two Stage Randomization Designs in Clinical Trials. *North Carolina State University,Raleigh.*

Haines, A. L. and Singpurwalla, N. D. (1974). Some contributions to the stochastic characterization of wear. *Reliability and Biometry*, **47-80**, F. Proschan and R.J. Serfling (eds.). SIAM, Philadelphia

Horvitz, D. G. and Thompson, M. E. (1952). A generalization of sampling without replacement from a finite universe. Journal of American Statistical Association, **47**, 663–685.

Holland, P. W.(1986). Statistics and causal inference. *Journal of American Statistical Association*, **81**, 945–970.

Jeong, J., Jung, S. and Costantino, J. P. (2008). Nonparametric Inference on Median Residual Life Function. *Biometrics*, **64**, 157–163.

Kaplan, E. L. and Meier, P. (1958). Nonparametric Estimator from Incomplete Observations. *J.Am.Statist.Assoc*, **53**, 457–481.

Lunceford, J. K., Davidian, M. and Tsiatis, A. A. (2002). Estimation of Survival Distributions of Treatment Policies in Two-Stage Randomization Designs in Clinical Trials. *Biometrics*, **58**, 48–57.

The MathWorks, Inc. (2005). Version 7.1 (R14SP3) MATLAB Software *The MathWorks, Inc., Natick, MA*.

Murphy, S., van der Laan, M. J. and Robins,J. M. (2001). YEAR = 2001, Marginal mean models for dynamic regimes. *J.Amer.Statist.Assoc.*, **96**, 1410-1423.

Murphy, Susan A. (2005). An Experimental Design for the Development of Adaptive Treatment Strategies. *Statistics in Medicine* **24**, 1455–1481.

Robins, J. M. and Rotnitzky, A. (1992). Recovery of information and adjustment for dependent censoring using surrogate markers.. In: Jewell N, Dietz K, Farewell V, eds. *AIDS Epidemiology: Methodological Issues.* Boston, MA: Birkhauser; 24–33

Robins, J. M., Rotnitzky, A. and Zhao, L. P. (1994). Estimation of Regression Coefficients when Some Regressors are not always Observed. *Journal of American Statistical Association*, **89**, 846–866.

Schmittlein, D. C. and Morrison, D. G. (1981). The median residual lifetime: A characterization theorem and an application. *Operations Research*, **29**, 392–399.

Stefanski, L. A. and Boos, D. D. (2001). The Calculus of M-Estimation. *North Carolina State University, Institute of Statistics Mimeo Series No.2528*.

Stone, R. M., Berg, D. T.,George, S. L., Dodge, R. K.,Paciucci, P. A., Schulman, P.,Lee, E. J., Moore, J. O.,Powell, B. L. and Schiffer,C.A. (1995). Granulocytemacrophage Colony-stimulating Factor after Initial Chemotherapy for Elderly Patients with Primary Acute Myelogenous Leukemia. *The New England Journal of Medicine*, **332**, 1671–1677.

Stone, R. M., Berg, D. T.,George, S. L., Dodge, R. K.,Paciucci, P. A., Schulman, P.,Lee, E. J., Moore, J. O.,Powell, B. L. , Baer,M.R., Bloomfield,C.D. and Schiffer,C.A. (2001). Postremission therapy in older patients with de novo acute myeloid leukemia: a randomized trial comparing mitoxantrone and intermediate-dose cytarabine with standard-dose cytarabine. *Blood*, **98**, 548–553.

Thall, P. F., Wooten,L. H., Logothetis, C. J., Millikan, R. E. and Tannir, N. M. (2007). Bayesian and frequentist two-stage treatment strategies based on sequential failure times subject to interval censoring. *Statistics in Medicine*, **26**, 4687–4702.

Wahed, A. S. and Tsiatis, A. A. (2004). Optimal Estimator for the Survival Distribution and Related Quantities for Treatment Policies in Two-Stage Randomization Designs in Clinical Trials. *Biometrics*, **60**, 124–133.

Wahed, A. S. and Tsiatis, A. A. (2006). Semiparametric efficient estimation of survival distributions in two-stage randomisation designs in clinical trials with censored data. *Biometrika*, **93**, 163–177

Wahed, A. S. (2009a). Estimation of survival quantiles in two-stage randomization designs. *Journal of Statistical Planning and Inference*, **139**, 2064–2075

Wahed, A. S. (2009b). Inference for two-stage adaptive treatment strategies using mixture distributions. *Journal of the Royal Statistical Society, in press*

Wang, J. L. and Hettmansperger, T. P. (1990). Two-sample inference for median survival times based on one-sample procedures for censored survival data. *Journal of the American Statistical Association*, **85**, 529–536.

Wolbers, M. and Helterbrand, J.D. (2008). Two-stage Randomization Designs in Drug Development. *Statistics in Medicine*. Published onlie in Wiley Interscience, DOI: 10.1002/sim.3309.

Zhao, H. and Tsiatis, A. A. (1997). A Consistent Estimator for the Distribution of Quality Adjusted Survival Time. *Biometrika*, **84**, 339–348.