A SAS macro for sample size adjustment and randomization test for internal pilot study

Suzhen Wang, Jielai Xia, Lili Yu, Chanjuan Li, Li Xu

Department of Health Statistics, Faculty of Preventative Medicine, Fourth Military Medical University, No. 17 Changle West Road, Xi’an, Shaanxi 710032, China

EPS China Co., Ltd., No. 18 Caoxi North Road, Shanghai 200030, China

Article history:
Received 7 April 2007
Received in revised form 28 October 2007
Accepted 27 November 2007

Keywords:
Clinical trials
Internal pilot study
Sample size adjustment
Two-stage adaptive design
Randomization test
Monte Carlo simulation
SAS/IML

Abstract
An unnecessarily high or inadequately low sample size often occurs in clinical trials if the planned variance of the trials is overestimated or underestimated. Internal pilot study which utilizes the information from the patients recruited up to interim stage can solve this problem well by re-estimating the variance and re-calculating the sample size. The trial may get a satisfactory power but the type I error rate may be inflated while the t-test is adopted to make hypothesis test because condition of t-distribution is not sufficed any more with variation of the sample size resulted from internal pilot design. If blind variance estimators of the internal pilot are used for sample size recalculation and randomization test is used to accomplish the final hypothesis test, not only the blindness of the internal pilot is preserved but also the ability to control the type I error rate is guaranteed. A SAS macro is programmed to simulate the process of sample size adjustment and randomization test.

1. Introduction

When we design the clinical trial, calculating the required sample size is a key step. In order to determine the sample size, we need to know the type I and type II error rates, the relevant effect size to be detected and the variance of the primary outcome variable. But the exact variance of the current trial remains unknown until the trial is finished and unblinded. So the estimate of the variance from previous trials is often used to calculate the sample size for current trial. However, the conditions of the previous trials and the current one are seldom exactly same with respect to many aspects. Therefore, considerable uncertainties may exist about whether the assumed variance value is appropriate for the current trial when using only external information from previous trials for sample size calculation.

An approach named internal pilot study introduced by Wittes and Brittain [1] enables us be able to use the estimated variance obtained from an interim analysis to recalculate the sample size during an ongoing trial. The internal pilot study is also a kind of adaptive design which has two stages during the course of the trial’s performance. It can use the data of the patients recruited in the first stage which is called internal pilot study to re-estimate the design parameters such as the degree of compliance of the study, the recruitment rate, the variance of the outcome variable, etc. So we can modify the designed protocol and assign the accurate parameters for the second stage before it starts. When the second stage has...
been finished, we make hypothesis test by analyzing the data both from recruited patients in the internal pilot and those from the second stage. In all the adjustment for the second stage, the most important and attractive one is sample size adjustment in which the whole sample size needs to be recalculated. Several different points of view have been given out about the sample size recalculation. Different types of sample size adjustment procedures can be referenced to Betensky and Tierney [2] who gave an excellent description about them. Although simulation studies [1,3] showed that internal pilot study has a high potential to protect from an inappropriate sample size even if the variance was misspecified in the planning phase, they suggested that the actual type I error rate may exceed nominal significance when the sample size is adjusted [1,3]. On one hand, controlling the type I error rate not being beyond the specified value is one of the important biostatistical and regulatory issues according to ICH guideline E9 [4] and a number of methods have been developed as for this problem [5–9]. On the other hand, keeping data of internal pilot blinded is becoming more and more attractive while recalculating the sample size. Gould and Shih [10] firstly proposed to use the standard pooled estimate of variance from the pilot data to recalculate the sample size without unblinding the data, that is, remaining internal pilot masked as to treatment assignment. They suggested treating the treatment effect as missing at random and using an EM algorithm. However, Kieser and Friede [9,11] thought the EM algorithm to estimate the blind variance was inappropriate for sample size recalculation. They gave simple procedures for blinded sample size adjustment that do not affect the type I error rate and mentioned that if a randomization test was used for the analysis, control of the type I error probability would be guaranteed for any arbitrary blind sample size adjustment strategy. But they did not give the specific procedure to make randomization test. Based on the procedure proposed by Kieser and Friede, we designed a SAS simulation program which is in fact a SAS macro to calculate the estimate of variance without unblinding the treatment status of internal pilot, re-estimate the sample size at the interim stage, combine the data both from the internal pilot study and the second stage at the end of the trial and make a randomization test to evaluate the efficacy. By means of Monte Carlo simulation, the SAS macro works well in simulating two-stage adaptive designs, recalculating blinded sample size and implementing randomization test. The results show us that significance level was kept below the specified level successfully.

2. Methods

2.1. Recalculating the sample size with blinded data from internal pilot study

Consider a clinical trial with two groups, each with unknown mean and common unknown variance. Suppose that \( X_{11}, X_{12}, \ldots, X_{1N} \) and \( X_{21}, X_{22}, \ldots, X_{2N} \) are series of independent normally distributed observations with unknown means \( \mu_{1X} \) and \( \mu_{2X} \), respectively, and common unknown variance \( \sigma^2 \). Furthermore, suppose that we wish to perform the two-sample problem of testing the two-sided hypothesis \( H_0: \Delta = 0 \) (\( \mu_{1X} = \mu_{2X} \)) against alternative \( H_1: \Delta \neq 0 \) (\( \mu_{1X} \neq \mu_{2X} \)). For simplicity of presentation, we assume that there are equal sample sizes in both groups. Prior to the study, the first thing is to determine the required sample size denoted as \( N \). If \( \sigma^2 \) were known, the required total sample size \( N \) per group with type I error rate \( \alpha \) and power \( 1 - \beta \) at the alternative \( \Delta = \delta \) could be obtained from

\[
N = \frac{2(Z_{1-\alpha/2} + Z_{1-\beta})^2\sigma^2}{\delta^2}
\]  

(1)

where \( Z_\gamma \) is the \( \gamma \)-percentile of the standard normal distribution, and \( \delta \) is the specified relevant treatment difference. However, the true value of \( \sigma^2 \) is generally unknown, so it is often a guessed value denoted as \( \sigma_0^2 \) obtained by looking at previous trials, reviewing literatures, and so on. \( \sigma_0^2 \), the so-called planned variance, is commonly used to replace \( \sigma^2 \) in formula (1) to calculate the original sample size. However, \( \sigma_0^2 \) based on data from previous trials may be subject to bias if those studies took place under different conditions, with different patient populations from the current study. In order to get the most approximate value of \( \sigma^2 \) of current study, Kieser and Friede [9] proposed a method for determining \( \sigma^2 \) through internal pilot on the basis of the method given by Gould and Shih [10]. Recruit \( n_1 < N \) observations per group and make internal pilot trial. After the trial is finished, the data collected are used to estimate the pooled variance denoted \( S_0^2 \). Then recalculate sample size by replacing a variance \( \sigma^2 \) for \( S_0^2 \) in the sample size formula [9], i.e.

\[
N_{\text{recal}}(S_0^2) = \frac{2(Z_{1-\alpha/2} + Z_{1-\beta})^2S_0^2}{\delta^2}
\]

(2)

where \( \alpha, \beta, Z_\gamma, \gamma \) and \( \delta \) have the same meaning as that of formula (1). \( S_0^2 \) will be easily obtained if the internal pilot study is unblinded. But the ICH E9-Guideline ‘Statistical Principles for Clinical Trials’ [12] stresses that the variance estimation should be ‘conducted on the blinded data’. So here we get \( S_0^2 \) without unblinding the internal pilot. Assume that we do not distinguish the observations \( X_{11}, X_{12}, \ldots, X_{1N_1} \) of treatment group from \( X_{21}, X_{22}, \ldots, X_{2N_2} \) of control group which means that we suppose they are in the same group. Then the simplest one-sample variance \( S_{0\text{os}}^2 \) can be calculated from the subsample \( n_1 \) without unblinding.

\[
S_{0\text{os}}^2 = \frac{1}{2n_1 - 1} \left\{ \sum_{i=1}^{n_1} \sum_{j=1}^{n_1} (X_{ij} - \bar{X})^2 \right\}
\]

(3)

where \( \bar{X} \) is the grand mean of the pooled sample. So we can replace \( S_0^2 \) in (2) with \( S_{0\text{os}}^2 \) and recalculate the sample size per group by

\[
N_{\text{recal}}(S_{0\text{os}}^2) = \frac{2(Z_{1-\alpha/2} + Z_{1-\beta})^2S_{0\text{os}}^2}{\delta^2}
\]

(4)

When \( H_1 \) holds true, the estimate may be biased. So we can adjust the one-sample variance \( S_{0\text{os}}^2 \) under the assumed
alternative \( H_1 \) and get \( S_{adj}^2 \) by [13]:

\[
S_{adj}^2 = S_{OS}^2 - \frac{n_1}{2(N_1 - 1)} \sigma^2
\]

(5)

And again replace \( S_{adj}^2 \) in (4) with \( S_{adj}^2 \), then the sample size per group can be recalculated by

\[
N_{\text{recal}}(S_{adj}^2) = \frac{2(Z_{1-\alpha/2} + Z_{1-\beta})^2 S_{adj}^2}{\delta^2}
\]

(6)

In formulae (3) and (5), \( n_1 \), the subsample entering the internal pilot from \( N \), plays an important role in the sample size recalculation. How much proportion should we take from \( N \) as \( n_1 \)? As for this question, various proposals [1,3,5,13,14] have been given since Wittes and Brittain [1] and Birkett and Day [2] firstly proposed their different opinions respectively. We can specify \( n_1 = nN \) observations as the sample size per group for internal pilot to compute the one-sample variance \( S_{OS}^2 \). \( \pi \) is the proportion of subjects from the originally planned sample size and can be determined according to specific situation. There seems no optimal value of \( \pi \) and we often use default choice of \( \pi = 0.5 \). At the end of the pilot phase, the data are kept blinded and these \( 2n\pi \) observations produce an estimate of one-sample variance \( S_{OS}^2 \) under the assumed null H0 and adjusted \( S_{adj}^2 \) under the assumed alternative H1. Because the bias of the recalculated sample size using \( S_{OS}^2 \) instead of \( S_{adj}^2 \) is negligible in most situation of clinical trials [15,16], we use only \( S_{OS}^2 \) to recalculate the sample size.

Thus, after \( n_1 = nN \) observations are recruited, a reliable estimate of the one-sample variance \( S_{OS}^2 \) is obtained without unblinding the treatment group at the interim analysis. We continue the trial by recruiting further \( N_{\text{recal}} - n_1 \) observations until it is finished and make analysis with all data both from the internal pilot design and the second stage. Here we adopt restricted design proposed by Birkett and Day [3], that is, the final sample size \( N_{\text{final}} \) per group is the smallest integer greater than or equal to \( \max(n_1, N_{\text{recal}}) \), which can avoid an unnecessarily large sample size when the one-sample variance \( S_{OS}^2 \) from the internal pilot is less than the planned \( \sigma_0^2 \) or extravagant power beyond requirement when the true variance \( \sigma^2 \) is small relative to the planned \( \sigma_0^2 \).

2.2. Analyzing the trial data through randomization test

Once the two-stage clinical trial with internal pilot design has been finished, the most important thing we should do is to analyze the data and make inference. Various procedures have been proposed to make the hypothesis test [1,3–6,9]. Since type I error probability may inflate with \( t \)-test applied in this situation [1,3], Kieser and Friede [9] and Zucker et al. [13] have suggested using the randomization test for the \( t \)-test situation. They claimed that this approach can keep the probability of type I error not to exceed nominal test level as long as the sample size adjustment rule uses only information from data pooled over the treatment groups.

A randomization test, also called a permutation test, is a type of statistical significance test in which a reference distribution is obtained by calculating almost all possible values of the test statistic under rearranging the labels on the observed data points. In two-stage adaptive design with internal pilot study, it can be used to test the null hypothesis (\( H_0 \)) to find out whether there is significant difference in treatment outcome between two treatment groups with observations \( X_{11}, X_{12}, \ldots, X_{1N_{\text{final}}} \) and \( X_{21}, X_{22}, \ldots, X_{2N_{\text{final}}} \) \( (H_0): \Delta = 0 \ (\mu_{1X} = \mu_{2X}); \ H_1: \Delta \neq 0 \ (\mu_{1X} \neq \mu_{2X}) \). The randomization test’s algorithm for the differences of means in this situation is described as follows:

Original data from the adaptive clinical trial is called observed set of data. It should be permuted many times in randomization test. Let \( \Gamma \) denote the collection of a great number of permutations \( x \) with \( N_{\text{final}} \) subjects in group 1 and same subjects in group 2, which include the observed set of data \( X \) from internal pilot and second stage, that is \( \Gamma = \{ \text{permutations of } X \} \) and \( x \in \Gamma \). \( \Gamma \) is also called the conditional reference set. Here we define \( d_m(x) \) and \( D_m(X) \) as the mean difference of the permuted and observed data, respectively.

Under the null hypothesis of no difference in the means of two groups, the labels that indicate the subjects assigned to group 1 or 2 should not be important. Thus, we could randomly assign the \( N_{\text{final}} \) subjects to group 1 and others to group 2. Therefore, a procedure as the following could be used:

1. Before permuting the data, compute a test statistic of interest on the original data that measures the discrepancy between group 1 and group 2 samples. In our simulations mean difference \( D_m(X) = (\bar{X}_{1N_{\text{final}}} - \bar{X}_{2N_{\text{final}}}) \) is chosen as the test statistic for the observed data, where \( \bar{X}_{1N_{\text{final}}} \) and \( \bar{X}_{2N_{\text{final}}} \) are the sample means of group 1 and 2, respectively.
2. Randomly permute the data a large number of times so that the total observations are randomly assigned to group 1 and group 2 by conditioned on \( N_{\text{group1}} = N_{\text{group2}} = N_{\text{final}} \) and a lot of permuted data sets are obtained. Take the mean difference \( d_m(x) = (\bar{X}_{1N_{\text{final}}} - \bar{X}_{2N_{\text{final}}}) \) as the test statistic and compute it for each of the resulting permuted data set, where \( \bar{X}_{1N_{\text{final}}} \) and \( \bar{X}_{2N_{\text{final}}} \) are the sample means of group 1 and 2, respectively.
3. Compute the distribution of the test statistic of \( d_m(x) = (\bar{X}_{1N_{\text{final}}} - \bar{X}_{2N_{\text{final}}}) \) under the null hypothesis and find out the probability of the null hypothesis, i.e., the proportion of the number of resamples that give a result of test statistic \( d_m(x) \) at least as great as that of the observed test statistic \( D_m(X) \) with the total number of permutations,

\[
P(d_m(x) \geq D_m(X) | N_{\text{group1}} = N_{\text{group2}} = N_{\text{final}}) = \sum_{i=1}^{k} I_i \frac{1}{k}
\]

(7)

where \( I_i \) is the indicator function and can be defined as:

\[
I_i = \begin{cases} 
1 & d_m(x) \geq D_m(X) \\
0 & d_m(x) < D_m(X) 
\end{cases}
\]

and \( k \) denotes the collection of all resamples in permutations, that is, the number of permuted data sets, including the observed data set.
4. Compare the observed test statistic \( D_m(X) \) to the distribution of test statistics \( d_m(x) \). If the observed test statistic is extremely small or large in the distribution, that is,
which $H_0$ is rejected, needs to perform the calculation of the test power, as the fraction of simulated datasets for which $H_0$ is accepted. Each $P$ value's calculation itself requires many simulations in which the original data is permuted many times. Loops should be taken to study the tests' type I error if the total number to carry out randomization tests is $N_S$, that is, $N_S$ loops are adopted, we will obtain $N_S$ $P$ values which are virtually the Monte Carlo $P$ values. The proportion of the number that $P$ values are at least as small as the nominal level under $H_0$ is type I error rate. So the true size of type I error probability of the permutation procedure is

$$\alpha_e = \frac{\sum_{i=1}^{N_S} I_i(P \leq \alpha)}{N_S}$$

where $I_i$ is the indicator function which takes value 1 if $P \leq \alpha$ and 0 if $P > \alpha$, and $\alpha$ is the specified nominal significance level.

On the other hand, the test’s performance is also studied through Monte Carlo simulations. It is known that the power is the probability that we can reject null hypothesis correctly when $H_0$ does not hold true. Just as mentioned in the calculation of type I error in the previous paragraph, the estimation of the test power, as the fraction of simulated datasets for which $H_0$ is rejected, needs to perform the calculation of $P$ value of a test many times. In Monte Carlo simulations it is the proportion of the number that $P$ values are not larger than the nominal level $\alpha$ under $H_1$, which can be calculated by the following formula:

$$P_{\text{power}} = \frac{\sum_{i=1}^{N_S} I_i(P \leq \alpha)}{N_S}$$

where the meaning of $I_i$, $P$, $\alpha$ and $N_S$ is just the same as that of formula (8). It can be easily seen that formula (9) is just the same as formula (8). But the condition of them is totally different. We use formula (8) to calculate the type I error of the randomization test on the condition that $H_0$ is accepted, but we use formula (9) to calculate power on the condition that $H_0$ is rejected. In Section 5, we will give Fig. 1 to show the power curves of the randomization tests as a function of mean differences between two treatment groups when the assumed true value of $\sigma$ is fixed and Fig. 2 to show the power curves as a function of assumed true standard deviation.

### 2.3. Type I error rate and power

The type I error probability can be guaranteed if randomization test is used to make hypothesis of adaptive design with blinded internal pilot. In principle, randomization tests are based on the randomization model that in a randomized experiment the observed values of the outcome variable are fixed numbers and the null hypothesis states that the treatments have no effect on these numbers [17]. Under the randomization model, the labels indicating variables identifying individual's group assignments are the only random elements. If the labels as to group 1 or 2 are exchangeable under the null hypothesis, then the resulting tests yield exact significance levels. In practice of internal pilot design, the observed outcomes are considered as fixed constants and the only quantities considered random are the labels identifying the assignment to group of each individual. Because the sample size adjustment is implemented with only merged data from blinded internal pilot and depends only upon fixed quantities in the randomization's point of view, the final sample size is a fixed, non-random quantity and independent of the final test statistic. Thus the maximum of type I error rate will not be beyond nominal level. Type I error rate of randomization test can be obtained by Monte Carlo simulations. In its calculations the $P$ value of a test must be calculated many times to estimate the probability of type I error as the fraction of simulated datasets for which $H_0$ is accepted. Each $P$ value's calculation itself requires many simulations in which the original data is permuted many times. Loops should be taken to study the tests' type I error. If the total number to carry out randomization tests is $N_S$, that is, $N_S$ loops are adopted, we will obtain $N_S$ $P$ values which are virtually the Monte Carlo $P$ values. The proportion of the number that $P$ values are at least as small as the nominal level under $H_0$ is type I error rate. So the true size of type I error probability of the permutation procedure is

$$\alpha_e = \frac{\sum_{i=1}^{N_S} I_i(P \leq \alpha)}{N_S}$$

where $I_i$ is the indicator function which takes value 1 if $P \leq \alpha$ and 0 if $P > \alpha$, and $\alpha$ is the specified nominal significance level.

On the other hand, the test’s performance is also studied through Monte Carlo simulations. It is known that the power is the probability that we can reject null hypothesis correctly when $H_0$ does not hold true. Just as mentioned in the calculation of type I error in the previous paragraph, the estimation of the test power, as the fraction of simulated datasets for which $H_0$ is rejected, needs to perform the calculation of $P$ value of a test many times. In Monte Carlo simulations it is the proportion of the number that $P$ values are not larger than the nominal level $\alpha$ under $H_1$, which can be calculated by the following formula:

$$P_{\text{power}} = \frac{\sum_{i=1}^{N_S} I_i(P \leq \alpha)}{N_S}$$

where the meaning of $I_i$, $P$, $\alpha$ and $N_S$ is just the same as that of formula (8). It can be easily seen that formula (9) is just the same as formula (8). But the condition of them is totally different. We use formula (8) to calculate the type I error of the randomization test on the condition that $H_0$ is accepted, but we use formula (9) to calculate power on the condition that $H_0$ is rejected. In Section 5, we will give Fig. 1 to show the power curves of the randomization tests as a function of mean differences between two treatment groups when the assumed true value of $\sigma$ is fixed and Fig. 2 to show the power curves as a function of assumed true standard deviation.

### 3. Description of simulation program and SAS code

A SAS macro program using Monte Carlo simulations has been written to simulate the process of adaptive design with internal pilot study, implement the recalculation of sample size, perform randomization test and calculate the type I error and power of it. The macro includes three main parts corresponding to three tasks of simulations.
3.1. Simulations of two-stage design and sample size adjustment

To simulate the two-stage adaptive design with internal pilot study, the macro first calculates the original sample size of clinical trial according to formula (1) with the parameter specified based on previous trials before the internal pilot study is simulated. Then it determines the sample size of internal pilot design and simulates it with the help of SAS/IML (PROC IML). After that an interim analysis is made to calculate the one-sample variance of pooled data without unblinding the internal pilot at the end of it. Next, the macro uses bootstrap method to estimate the blindable one-sample variance so as to get the value with less variability. Bootstrap allows us to take the original random sample as if it were a population, and then resample from it with replacement to compute estimators and test statistics [18]. Resampling number can be input by the users through macro variable Num_Bootstrap. For every resampling, an estimate of variance can be obtained. The larger the input number, the more invariable the estimate is. When the resampling number reaches to 2000(Num_Bootstrap = 2000), the median of the variances is invariable enough and can be used to calculate the sample size from formula (4). When the recalculated sample size of the whole trial is obtained, the macro makes sample size adjustment toredetermine the sample size of the second stage and simulates it by PROC IML. Finally the data from the first and second stage are combined into one dataset to be analyzed by a randomization test.

3.2. Simulations of randomization test

In the macro program a randomization test is performed for the subjects both from the first and second stage after the simulations of two stages are finished. In order to implement the randomization successfully and accurately, the macro program first generates the resample data with the help of SAS PROC SURVEYSELECT which makes us take a 100% sample many times without replacement. The resample number can be input by macro variable Num_Repls. Next, the macro program combines the resample data with the observed one to get permuted data. The permuted data are stored in a single data set but labeled based on a variable Replicate. And then the macro calculates the means differences between the two treatment groups for the permuted data set. The calculation is also programmed by SAS/IML (PROC IML) so as to save real running time. Finally, the macro estimates the probability value of the randomization test by comparing the mean differences of the permuted data with that of the observed one. The number of permuted mean differences greater than or equal to the observed mean difference divided by the total number of permuted data is exactly the P value of randomization test.

3.3. Simulations of power and type I error of randomization test

The macro uses loops containing Np replications to calculate the type I error and the power of randomization test. The replication number Np is actually the number to perform randomization tests as described in Section 2.3 and embodied as a macro variable Num_Simu. When every loop is executed, the simulated observed data for the first and second stage of the trial are generated under the same specific hypothesis with same means following normal distribution, but with different errors, that is,

\[ X_{ijk} = \mu_{ij} + \varepsilon_{ijk} \quad i = 1, 2 \quad j = 1, 2 \quad k = 1, \ldots, N_p, \]

where \( X_{ijk} \) and \( \varepsilon_{ijk} \) represent the observed value and error of group j in ith stage of kth loop, respectively, and \( \mu_{ij} \) is the group mean of group j in ith stage. Here the group means are identical for the first and second stage, so \( \mu_{1j} = \mu_{2j} \). For each replication of randomization test, the data are permuted many times and a P value is obtained. Once the Np loops are finished, the macro calculates the power of the test. When H0 is accepted such as macro variable U1x = U2x in the macro, the type I error is the number of P ≤ α divided by the total loop number Np. When H0 is rejected, the power is obtained in the same way. Section 5 will demonstrate the use of the macro in example and give the power curve in Figs. 1 and 2.

3.4. Description of the parameter input

Twelve macro variables are chosen as the positional parameters in the program. Sigma0, Delta, Alpha and Beta correspond to planned standard deviation \( \sigma_0 \), specified relevant treatment difference \( \delta \), specified probability of type I error \( \alpha \) and type II error \( \beta \) in formula (1), respectively, which are mainly used to calculate the planned sample size \( N \). \( \alpha \) is also used as the significance level of the randomization test. Sigma is the assumed true standard deviation \( \sigma \), mainly for generating simulating data of two-stage adaptive design. Pi is the proportion of subjects used in internal pilot from the originally planned sample size. It is in fact the value of \( \pi \) in Section 2.1 that usually takes the traditional value of 0.5. U1x and U2x are the assumed means for treatment group 1 and 2, respectively. Num_bootstrap is the number for bootstrap to resample the original data to get the accurate one-sample variance. Num_reps is the number of permuting the original data to generate randomization datasets and Num_Simu represents the number of simulations for calculating the type I error rate and power of randomization test. Examples about the specification of the parameters are given in Section 5 and Appendices B and C.

4. Computational time requirement and computer resources

Although a randomization test is powerful and distribution-free, a disadvantage of it known to us is that it requires extensive computation and may be computer-intensive. It usually needs more time to be executed than classical parametric test. However, this issue becomes less relevant now due to faster algorithms and the advent of high-speed computer.

The real strength of our macro in saving running time lies in its ability to generate datasets as less as possible and use PROC IML as much as possible. Generally speaking, implementing of a randomization test needs to resample the observed data many times to generate a great many of permuted datasets. Too many datasets stored in the diskettes will not only occupy...
much disk space, but also reduce the operating speed of the program. In our macro, the resample data used to perform randomization test are generated by PROC SURVEYSELECT and stored in a single data set instead of many separate data sets. This means that procedures only need to be called a single time when carrying out randomization test, which saves time greatly. Furthermore, most of the calculations such as the means of the permuted data, the mean differences between treatment groups, etc. are performed by PROC IML. So the main time to perform randomization test in our macro is the time to execute PROC SURVEYSELECT and PROC IML. Both of the PROCs can be carried out in relatively short time so that the computations can be done in a reasonable amount of time. At the same time, SAS ODS is adopted to prevent the results from being shown in the output window, which also makes CPU and real time decrease more predictably and the program become more efficient to some extent.

On the other hand, the real run time depends strongly upon the computing hardware used to execute the macro. Hardware with modern developments, such as high-speed cache memory, will provide more efficient performance. On our desktop platform, an Intel® Pentium IV processor of 2.8 GHz speed, 160 GB Hard Drive and 512 MB of Ram, were used to run this macro in SAS Version 8.2. If we only make randomization simulation without calculating the power or the type I error, the total CPU and real time carrying out the macro is about 7.13 s. The longest time is the calculating of the mean differences between two treatment groups by PROC IML, about 0.73 and 0.51 s for the CPU and real time, respectively.

The calculation of the power and the type I error is time-consuming and lengthy procedure, because it needs loops to calculate many P values from many times of randomization tests. Even when using the fastest microcomputer, the time may be very long, especially when the sample size of the two groups is large. When the macro is running on our desktop platform with configuration described in the previous paragraph, the longest time needed to carry out power calculation is about 58 min and the shortest one is about 26 min, which correspond to power calculation at the biggest and smallest fixed mean difference, respectively shown in Fig. 1. The average time to perform the power calculation for a fixed mean difference is about 38 min. The total time need to run all of the programs to generate Fig. 1 is about 13.3 h. If the users care more about the result of the randomization test than the power or type I error, they can let the macro execute only once so as to save the real time. In fact randomization test are more accurate and powerful than those from parametric tests when testing the null hypothesis such as t-test, ANOVA, etc., and has less probability to commit a type I error. The user can ignore the calculation of power and type I error probability by letting the simulation number be one (Num_Simu = 1) in the macro.

5. Example

In a randomized placebo-controlled clinical trial, a new drug for curing recurrent oral ulcer was evaluated (The efficacy of the new drug was investigated). Reduction in ulcer size was measured after a week of treatment and was approximately normally distributed. We want to compare two treatment groups on the continuous efficacy variable. A difference in reductions of ulcer size between the treatment and placebo groups of δ = 1.3 was of clinical interest. According to previous trials we specify α = 0.05, 1 − β = 0.85 and the standard deviation of the ulcer size reduction approximately 1. We now specify Sigma0 = 1, Delta = 1.3, Alpha = 0.05, Beta = 0.15 in the macro and get an initial total sample size of 22 patients, with N =11 patients in each treatment group. Assuming further that π = 0.5, that is Pi = 0.5 in the macro, and therefore, a total sample size of the internal pilot study is 22 × 0.5 = 11. For simplicity the macro will simulate to recruit 12 patients with n1 = 6 per group automatically and then output “Initial sample size of internal pilot study is 6.000” in the log window. In order to simulate data for the 12 patients in the internal pilot study, we assume 𝜇1 = 1.3, 𝜇2 = 0, σ = 1.2 (𝜎2 = 1.44) to (generate normally distributed simulating data x1 ∼ N(𝑈1, 𝜎2) and x2 ∼ N(𝑈2, 𝜎2). We should let Sigma = 1.2, U1x = 1.3 and U2x = 0 in the macro. The users may notice the difference between macro variable Sigma0 and Sigma. Sigma0, the specified n0 according to previous trials, is used to calculate the planned sample size of the whole trial, but Sigma is the supposed true standard deviation of the current trial and designed specifically to simulate the trial. From these parameters, we obtain an estimate of one-sample variance $S^2_{OS}$ from the SAS macro without unblinding the treatment status. The messages in the log window “The one-sample variance of internal pilot study is 1.52777219” tells us that the estimated one-sample variance is $S^2_{OS}$ = 1.52777219. With $S^2_{OS}$ and the specified difference $\delta$ = 1.3, the macro recalculates the sample size and shows us “The recalculated sample size per group is 17.000” in the log window, which reminds us that the total required sample size is 34 patients, i.e. $N_{recalc} = 17$ per group. Next, the number of patients in each treatment group is chosen by restricted design: max($n_1$, $N_{recalc}$) = 17, which means that we should take 11 further subjects in each group instead of 6 and continue the second stage of the trial. After the two stages are finished, randomization test is used to find out treatment difference and make hypothesis test at 5% level. The number of iterations of the randomizations procedure is determined according to Manly [18] who suggested that 1000 iterations of the data are sufficient to test hypothesis at a 5% level of significance and 5000 iterations are sufficient to test at a 1% level. After 5000 times of permutations(Num_Reps = 5000), the message “The randomization test has significance level of 0.0092” will appear in the log window, showing us that the P value of randomization test is 0.0092. Here we intentionally let macro variable Num_Simu = 1 to speed up the macro without calculating the power. $H_0$ is thus rejected at the significance level 0.05. Specification of parameters for this example is given in Appendix B.

To study the simulation power of randomization test, we first let the assumed true standard deviation $\sigma$ fixed and the mean differences between treatment group 1 and 2 increase gradually, from 0 to 2.0 by increment 0.1. Every increment is corresponding to 1000 loops of simulations (Num_Simu = 1000). A power value is obtained at the end of the loops. Many increments require calling the macro many times to output a group of power values. With these values we get Fig. 1 showing the power curves of randomization test as a function of mean differences with other parameters fixed as...
described in above example. When the mean difference is 0, for example \( U_{1x} = 0, U_{2x} = 0 \), the resulting value is actually the type I error rate, which is the point of the power curve intersecting with the vertical axis. We can see that type I error rate is not beyond 0.05. In fact, the value of type I error rate calculated from 1000 loops is about 0.0450. With \( \sigma \) fixed, the power is gradually increasing with the growing of the mean difference. The inputs of the macro parameters to study power are shown in Appendix C.

Similarly, we set the mean difference fixed at 1.3, for example, \( U_{1x} = 1.3, U_{2x} = 0 \); the assumed true value of standard deviation, Sigma \( (\sigma) \), increases from 0.6 to 2.8 by increment 0.2; and Sigma0 \( (\sigma_0) \) fixed at 1 to observe the variability of the simulation power of randomization test. Fig. 2 is the power curves of randomization test as a function of Sigma. The setting of parameters for generating Fig. 2 is shown in Appendix C.

When the prior estimate of variance \( \sigma_0^2 \) is equal to the true study variance \( \sigma^2 \), that is \( \sigma_0^2 = \sigma^2 = 1 \), the power is close to the desired 85% because the sample size is variable in this situation and the adaptive design is equal to the fixed sample size design. When \( \sigma_0^2 > \sigma^2 \) the power is higher than the design required. On the contrary, \( \sigma_0^2 < \sigma^2 \) results in the loss of power. But with \( \sigma \) becoming bigger and bigger, the loss of the power is not growing, showing us that the misspecification of the true variance has small effect on the power when \( \sigma_0^2 < \sigma^2 \).

6. Discussion

Inflation of type I error rate and the loss of power has long been an intractable problem for sample size adjustment in adaptive design with internal pilot study. If t-test is used to make analysis for the data and sample size is adjusted with the information from unblinding internal pilot study, type I error rate may be inflated especially when small sample sizes are used. In the simulation program described in this paper, we use the blinded variance of internal pilot study to make sample size adjustment and randomization test to perform analysis, the achieved type I error rate is well controlled below the nominal level, particularly in the case of small studies. Furthermore, the results for the expected power demonstrate that these procedures are also effective in ensuring the desired power. The power of 82.95% at \( \delta = 1.3 \) is a little bit below the desired 85%, which may have resulted from the very small sample size used in internal pilot design. The loss of power as a cost for using small sample size and controlling type I error rate is acceptable.

For the calculation of blinded one-sample variance from internal pilot study, bootstrap method is used to compute the estimate. It is well known that the variance estimate obtained from a small pilot trial may be subject to substantial sampling error. Resample many times through bootstrap method can simulate the most approximate distribution of the internal pilot study; and so the variance can suffer from less bias. Bootstrap method uses the original sample of internal pilot study to carry out resample and does not need extra sample size. It is easy to be performed and feasible in practice. And its use can be recommended in estimating the blinded variance of internal pilot study in practice.

7. Conclusion

In clinical trials, blinded one-sample variance estimation is an effective way for sample size adjustment for internal pilot study. With the internal pilot being kept in blinded situation, we get the required one-sample variance easily and use it to recalculate the sample size. So there is no reason for unveiling the treatment groups in the interim analysis. Adjusting the sample size without unblinding internal pilot not only complies well with the regulations for drug approval which require performing the recalculation without breaking the blind and controlling the type I error rate at the nominal level [12]; but also saves cost and maintains the credibility of the trial.

Randomization tests are also ideal for studying efficacy of adaptive design with internal pilot study. It is an important and a much more elegant alternative for testing the null hypotheses about treatment mean. It works well in situation where the initial sample size is small and guarantees the control of the type I error probability for any arbitrary blind sample size adjustment strategy. Combining the blinded sample size adjustment with randomization test in the internal pilot design, type I errors, or probabilities of falsely rejecting the null hypothesis, can be controlled effectively and the loss of power is acceptable. SAS macro is programmed to finish the process. It runs successfully on desktop platform under OS system WindowsXP with advanced hardware configuration: an Intel Pentium IV processor of 2.8 GHz speed, 160 GB Hard Drive and 512 MB of Ram. As microcomputers become faster and more powerful, the runtime consumption of the macro will not become a main problem and the randomization methods will be used more and more frequently in the efficacy study in various designs of clinical trials.

Acknowledgements

Support for this research was provided by Research Grant 30271157 from National Natural Science Foundation of China. The authors gratefully acknowledge the Editor and the two reviewers whose valuable comments and helpful remarks and suggestions led to a remarkable improvement of this paper.

Appendix A

SAS code for re-calculating the required sample and making randomization test.
/* MACRO:SampleAdj_RandT */

/* FUNCTIONALITY: */

/* SAS Macro for performing the estimation of the one-sample variance without unblinding */

/* the treatment at the end of the internal pilot, for re-calculating the required sample size of the*/

/* trial and making randomization test. */

/* */

/* USE/CALL OF PROGRAM: */

/* %SampleAdj_RandT: */

/* */

/*Main DATASETS OUTPUT: */

/* Recalsample: Storing the recalculated sample size */

/* */

/* Meaning of macro variables */

/* Sigma0: Supposed deviation according to previous trials */

/* Sigma: Assumed true deviation for generating simulating data of two stages */

/* Delta: Treatment difference specified according to previous trials */

/* Alpha: Probability of type I error specified for determining percentile of the standard normal */

/* distribution used to calculate initial sample */

/* Beta: Probability of type II error specified for determining percentile of the standard normal */
distribution used to calculate initial sample

/* Pi: The proportion of subjects used in internal pilot from the originally planned sample size. */

/* U1x: Mean specified for treatment group (group 1) to generate simulating data */

/* U2x: Mean specified for control group (group 2) to generate simulating data */

/* Seed: Specifies the seed of SAS random number */

/* Num_Bootstrap: Number of bootstrapping the original data to get the one-sample variance */

/* Num_Repls: Number of permuting the original data and generating randomization datasets */

/* Num_Simu: Number of simulations for calculating the type I error rate and power of */

/* randomization test */

*******************************************************************************/

options nonotes nosource nosource2;

%Macro SampleAdj_RandT(Sigma0,Sigma,Delta,Alpha,Beta,pi,U1x,U2x,Seed,
Num_Bootstrap,Num_Repls,Num_Simu);

对中国计算的总体规划的样本大小（每组）

\[ N_{init-cal} = 2 \times (\text{probit}(1 - \&\text{Alpha}/2) + \text{probit}(1 - \&\text{Beta}/2)) \times \&\text{Sigma0}^2 / \&\text{Delta}^2 \]

对中国计算的样本大小根据内部试点组

\[ N_{internal} = \&\text{pi} \times N_{init-cal} \]

\[ N_{internal} = \text{ceil}(N_{internal}) \]

call symput('N_ips',N_internal); /* N_ips is the sample size of internal pilot study */
Put "Initial sample size of internal pilot study is" N内部 6.3;

Run;

%Do nra=1 %To &Num_Simu;

/* Generating simulation sample size of internal pilot study */

Proc iml;

N_ips=&N_ips;

y=j(2*N_ips,1,);

group=j(2*N_ips,1,);

Seed=&Seed-&nra;

Do j=1 To N_ips;

y[j,1]=&U1x+&Sigma*rannor(Seed);

group[j,1]=1;

End;

Do j=N_ips+1 To 2*N_ips;

y[j,1]=&U2x+&Sigma*rannor(Seed);

group[j,1]=2;

End;

/* Creating the data set of the internal pilot study */

Create firstS var{group,y};

Append;

Quit; /* quit proc iml */

/* Use resampling of bootstrap method to determine the one-sample variance of the internal pilot
study without unblinding */

Proc SurveySelect Data=firstS Out=BootSample Method=URS Rate=1 Rep=&Num_Bootstrap;

/* URS requests unrestricted random sampling, which is selection with equal probability and with
replacement */

Id y;
Run;

/*Calculate the one-sample variance for each resampling data set */

Proc Means Data=BootSample NoPrint;

By Replicate NotSorted;
Var y;
Freq NumberHits; /*Specifies a numeric variable that contains the frequency of each observation;*/
Output Out=BootStrap Var=SSos;
Run;

/*Take the median of one-sample variance for bootstrap data and put it into the data set SSos */

Data SSos;
Set BootStrap;

Proc sort;
By SSos;
Run;
Data SSos;
Set SSos;
If n=int(&Num_Bootstrap/2)
Then Do;

    Call symput("SSos",SSos);

    Put "The one-sample variance of internal pilot study is " SSos 10.8;

    End;

Run;

/*Recalculate the sample size according to the one-sample variance SSos*/

Data RecalSample;

    Nrecalc=ceil(2*(probit(1-&Alpha/2)+probit(1-&Beta))**2*SSos/&Delta**2);

    Nrecalc=max(Nrecalc,&N_ips);*Here use only restricted design;

    Call symput('N_recalc',Nrecalc);

    Put "The recalculated sample size per group is " Nrecalc 6.3;

Run;

/*If the recalculated sample size is equal to that of internal pilot study, we will only pick up the data of the first stage*/

    %If &N_recalc=&N_ips %Then %Do; Data TwoStages;

        Set firstS;

        Run;

        %End;

    %Else %Do;

/* Generating sample size for the second stage according to recalculated sample size*/

    Proc iml;

        Seed=&Seed-&nra-2007;
N_recalc=N_recalc-N_ips;
y=j(2*N_recalc,1,);
group=j(2*N_recalc,1,);
Do j=1 To N_recalc;
    y[j,1]=&U1x+&Sigma*rannor(Seed);
    group[j,1]=1;
End;

Do j=N_recalc+1 To 2*N_recalc;
    y[j,1]=&U2x+&Sigma*rannor(Seed);
    group[j,1]=2;
End;

/*Creating data set of the second stage in the clinical trial*/
Create secondS var{group y};
Append;
Quit; /* quit proc iml */

/*Combining the data of the first stage with that of the second one */
Data Twostages;
Set firstS secondS;
Run;
%End;

Data Twostages;
Set Twostages;
Proc sort;
By group;
Run;

/****************************Randomization test***************************/

/* Generate all of the data sets first. These are stored in a single data set but labelled according to a
variable Replicate. Then subsequent analyses can be performed BY Replicate. Make the
Num_Reps randomzied Reps. We use SURVEYSELECT to take a 100 percent sample without
replacement. */

Proc SurveySelect Data=TwoStages Method=SRS Rate=1 Out=Randomization
Rep=&Num_Reps;
Run;

/* Randomly order these observations by assigning each observation a uniform random number
and sorting by that. */

Data Randomization;
Set Randomization;
RN=Ranuni(0);
Proc Sort;
By Replicate RN;
Run;

/* Make the first Nfinal observations within a Replicate the group 1 and the last Nfinal the group
2. */

Data Randomization;
Set Randomization;

By Replicate NotSorted;

If First.Replicate Then i=0;

i+1;

If i<=&N_recale Then group=1;

Else group=2;

Run;

/*/ Put the original data back with the permuted data. These original data will be Replicate=0. */

Data Randomization;

Set TwoStages(In=In1) Randomization;

If In1 Then Replicate=0;

Run;

/*/ Compute the mean of two group respectively by Replicate. Save the results in data sets for
processing. */

Proc means nocprint data = Randomization;

Var y;

Class group;

By Replicate NotSorted;

Output out=outrand mean=mean;

Run;

Data outrand;

Set outrand(where=(._type_=''0'')) end = cof;
Keep Replicate group mean;

Run;

/*Compute the mean difference of two treatment groups through Proc iml and store it in the
dataset "outrandif". */

Proc iml;

Use outrand;

Read all into outrand1 where (group=1);

Read all into outrand2 where (group=2);

Create outrand1 var{Replicate group mean1};

Append from outrand1;

Create outrand2 var{Replicate group mean2};

Append from outrand2;

Quit;

Data outrandif;

Merge outrand1 outrand2 by Replicate;

meandif=mean1-mean2;

Keep Replicate meandif;

Run;

/**************Calculating the power and type I error rate of randomization test**************/

Proc sql;

Select meandif into :meansta from outrandif where Replicate=0;

Select count(*) into:summean_h0 from outrandif where meandif=abs(&meansta) or
meandiF<-abs(meansta);

    Select count(*) into summean from outrandif;

Quit;

    Data random_temp;

    randp0=&summean_h0/&summean;

Run;

%If &nra=1 %Then %Do;

    Data Randoma;

    Set random_temp;

    Run;

%End;

%Else %Do;

    Data Randoma;

    Set Randoma random_temp;

    Run;

%End;

%End; /*End of loop for Num_Simu */

/*If the user does not want to calculate the power or the type I error rate, the macro will only
output the significant p value */

%If &Num_Simu=1 %Then %Do;

    Data randoma;

    Set randoma;
Put "The randomization test has significance level of" randp0 6.4;

Run;

%End;

%Else %DO;

Data type1ran&Num_Simu;

Retain numtype1 numtype2 numtot 0;

Set Randoma end=endfile;

numtype1+(randp0<=&Alpha);

numtot+1;

If endfile Then Do;

    typel=numtype1/numtot;

    power=numtype1/numtot;

    If &U1x<>&U2x Then Put "power of randomization test is " power 6.4;

Else Put "type I error of randomization test is " typel 6.4;

End;

If power=. Then delete;

Sinn=&Num_Simu;

Dmean=&U1x-&U2x;

Keep Sinn Dmean power;

Run;

%End;
Appendix B

In order to facilitate users, we give the following examples to explain how to call the macro. The example described in Section 5 can be simulated as:

\%SampleAdj\_RandT(1,1.2,1.3,0.05,0.15,0.5,1.3,0,20071018,2000,5000,1);

And the output in the log window will be:

Initial sample size of internal pilot study is 6.000.
The one-sample variance of internal pilot study is 1.527777219.
The recalculated sample size per group is 17.000.
The randomization test has significance level of 0.0092.

The users may notice that the output of one-sample variance and significance level in the log window may be a little bit different from the above result. There are two reasons: one is that the one-sample variance from internal pilot study is computed by bootstrap methods with randomly resampling; the other is that the permuted data sets in the randomization test are generated randomly. It is the difference of the randomly resampling and permuted data sets that result in the variability of the P value in randomization test.

Appendix C

In order to calculate the power of randomization test corresponding to different mean differences, we need to call the macro many times, with mean difference between two treatment groups increasing from 0 to 2.0 by increment 0.1. We wrote another macro named CallSampleAdj as following shown to call the macro SampleAdj\_RandT. Every calling of SampleAdj\_RandT outputs the power corresponding to specific mean difference. With 20 different power values we get the power curve as a function of mean differences in Fig. 1. It should be emphasized that the power of randomization test is time-consuming because it has to loop many times. When the mean difference is zero, for example, both U1 and U2 are zero, the output is type I error rate.

SAS macro for calling the macro SampleAdj\_RandT to perform power calculation
%**MACRO** CallSampleAdj

/* **FUNCTIONALITY:** */

/* SAS Macro for calling the macro SampleAdj_RandT */

/* **USE/CALL OF PROGRAM:** */

/* %**CallSampleAdj:** */

/* **Meaning of macro variables:** */

/* &minU1x: the minimum of the assumed true mean for treatment group (group 1) */

/* &maxU1x: the max of the assumed true mean for control group (group 2) */

/* &min_Num_Simu: the minimum number to replicate the macro SampleAdj_RandT for power */

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Macro CallSampleAdj(minU1x,maxU1x,min_Num_Simu);

%Let Setting_U1x=&minU1x;

%Let Setting_Num_Simu=&min_Num_Simu;

%Do %While (&Setting_U1x<=&maxU1x);

%SampleAdj_RandT(1,1.2,1.3,0.05,0.15,0.5,&Setting_U1x,0,20071026,2000,5000,&Setting_Num_Simu);

%Let Setting_U1x=%sysevalf(&Setting_U1x+0.1);

%Let Setting_Num_Simu=%sysevalf(&Setting_Num_Simu+1);

%End;
%Mend;

%CallsSampleAdj(0.2,0.1000);

Executing the above macro(% CallsSampleAdj(0.2,0.1000);) is equal to executing the following macros.

%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0,0,20071026,2000,5000,1000);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0.1,0,20071026,2000,5000,1001);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0.2,0,20071026,2000,5000,1002);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0.3,0,20071026,2000,5000,1003);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0.4,0,20071026,2000,5000,1004);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0.5,0,20071026,2000,5000,1005);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0.6,0,20071026,2000,5000,1006);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0.7,0,20071026,2000,5000,1007);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0.8,0,20071026,2000,5000,1008);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,0.9,0,20071026,2000,5000,1009);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,1.0,0,20071026,2000,5000,1010);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,1.1,0,20071026,2000,5000,1011);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,1.2,0,20071026,2000,5000,1012);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,1.3,0,20071026,2000,5000,1013);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,1.4,0,20071026,2000,5000,1014);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,1.5,0,20071026,2000,5000,1015);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,1.6,0,20071026,2000,5000,1016);
%SampleAdj_RandT(1.2,1.3,0.05,0.15,0.5,1.7,0,20071026,2000,5000,1017);
In the same way, we wrote a macro to call the macro SampleAdj_RandT for getting the power of randomization test as a function of Sigma(σ). The macro for getting the power of Fig. 2 is as following:

**%Macro** CallSampleAdj_fixedDelta(minSigma,maxSigma,min_Num_Simu);

**%Let** Setting_Sigma=&minSigma;

**%Let** Setting_Num_Simu=&min_Num_Simu;

**%Do %While** (&Setting_Sigma<&maxSigma);

**%SampleAdj_RandT**(1,
&Setting_Sigma,1.3,0.05,0.15,0.5,1.3,0,20071026,2000,5000,&Setting_Num_Simu);

**%Let** Setting_Sigma=&Sysevalf(&Setting_Sigma+0.2);

**%Let** Setting_Num_Simu=&Sysevalf(&Setting_Num_Simu+1);

**%End**;

**%Mend**;

**%CallSampleAdj_fixedDelta**(0.6,2.8,1000);

---

**REFERENCES**


