An Expanded Set of SAS Macros for Calculating Confidence Limits and P-values Under Simon's Two-Stage Design Accounting for Actual and Planned Sample Sizes

Mikhail Melikov, Cytel;
Alex Karanevich, EMB Statistical Solutions;

ABSTRACT

Simon's two-stage designs are popular single-arm, binary-endpoint clinical trials that include a single interim analysis for futility. In the literature, it is widely acknowledged that the inferential statistics calculated (p-values, point estimates, and confidence limits) in Simon's designs typically do not account for when the actual sample size differs from the planned sample size at stage two. Koyama and Chen (2008) provided methods to calculate proper inferential statistics for studies where the planned sample size at stage two is either the same or different from the actual sample size.

Our previously published SAS macro implemented these methods for the case where the planned sample size matched the actual sample size. We acknowledge that the actual sample size for stage two is often different from what is planned: therefore, to expand the utility of our previous SAS macro, in the field of applied biostatistics among the audience with intermediate or advanced level of statistical skills, we have implemented methods for inference when there are deviations in the planned vs. actual sample size. The expanded SAS program now has macros to calculate inferential statistics with and without deviations in the planned vs. actual sample size at stage two, and also calculates two types of point estimates of the response rate at stage two - uniformly minimum variance unbiased estimator (UMVUE) as discussed by Jung and Kim (2004) and reasonable point estimate proposed by Koyama and Chen (2008).

We do not expect the SAS macro to be dependent on operating system or SAS software version.

INTRODUCTION

It is widely acknowledged that the inferential statistics calculated (p-values, point estimates, and confidence limits) in Simon's Two-Stage designs are often calculated naively, and typically do not account for when the actual sample size differs from the planned sample size at stage two. Koyama and Chen (2008) provided methods to calculate proper inferential statistics for studies where the planned sample size at stage two is either the same or different from the actual sample size.

Our previously published SAS macro implemented these methods to properly calculate confidence intervals for the case where the planned sample size matched the actual sample size. We acknowledge that the actual sample size for stage two is often different from what is planned: therefore, to expand the utility of our previous SAS macro, we have implemented methods for inference when there are deviations in the planned vs. actual sample size. The expanded SAS program now has macros to calculate inferential statistics with and without deviations in the planned vs. actual sample size at stage two. It also calculates the uniformly minimum variance unbiased estimator (UMVUE) for the point estimate of the response rate at stage two, for both matching and deviating stage two sample sizes, as discussed by Jung and Kim (2004), as well as reasonable point estimate of response rate at stage two proposed by Koyama and Chen (2008).

NOTATIONS AND WORKING EXAMPLES

A Simon's two-stage design is usually indexed by four numbers that represent the stage 1 sample size (n_1), stage 1 criterial value (r_1), final (total) sample size (n) and final critical value (r_t). In stage 1, a sample of size n_1 is taken. If the number of successes X_1 in stage 1 satisfies $X_1 \leq r_1$, then the trial is stopped for futility; otherwise, an additional sample of size $n_2 = n - n_1$ is taken. Let X_2 be the number of successes in stage 2 and let $X_t = X_1 + X_2$ be the total number of successes across both stages. If $X_t \leq r_t$, efficacy is not concluded by failing to reject H0; otherwise, efficacy is concluded by rejecting H0.

Let m be the stage where the trial stops (m=1 or 2) and s be the total number of successes accumulated up to stopping stage m. That is, $s = X_1$ when $m = 1$ and $s = X_t$ when $m = 2$. $If\ m = 2$ then let n_2^* be the actual stage 2 sample size and n_2 planned sample size.

Let $x_1, x_2,$ and x_t be the actual observed numbers of successes at stage 1, stage 2, and total respectively. Let π be the rate of success for $X_1, X_2,$ and X_t.

This paper uses three examples of study designs presented in Table 1 of which first two come from the (Koyama and Chen, 2008) paper and the third comes from the paper of (Zhao, Yu, and Feng, 2015).

	Design considerations	Design	Observed	Actual *vs* planned sample size
1	Minimax with H0: $\pi = 0.1$, H1: $\pi = 0.3$, $\alpha = 0.05, \beta = 0.2$	$n_1 = 10, r_1 = 1,$ $n = 29, r_t = 5$	$x_1 = 2,$ $x_t = 6$	$n_2^* = 19 = n_2$
2	Minimax with H0: $\pi = 0.3$, H1: $\pi = 0.5$, $\alpha = 0.05, \beta = 0.2$	$n_1 = 19, r_1 = 6,$ $n = 39, r_t = 16$	$x_1 = 7,$ $x_t = 17$	$n_2^* = 23 \neq n_2$
3	Optimal with H0: $\pi \leq 0.15$, H1: $\pi \geq 0.3$, $\alpha = 0.1, \beta = 0.2$	$n_1 = 19, r_1 = 3,$ $n = 39, r_t = 8$	$x_1 = 8,$ $x_t = 12$	$n_2^* = 6 \neq n_2$

Table 1: Examples of Simon's two-stage study designs

STATISTICAL METHODS IMPLEMENTED WITH SAS MACROS

In this section, you can find mathematical form of the equations used to describe applied statistical methods with their corresponding SAS Macros.

POINT ESTIMATE OF THE RESPONSE RATE

The MLE estimator (naïve) is unbiased when m=1. However, it is biased when m=2.

$$\hat{\pi}_{mle} = \begin{cases} \dfrac{x_1}{n_1}, & m = 1 \\ \dfrac{x_1 + x_2}{n_1 + n_2^*}, & m = 2 \end{cases}$$

Equation 1: Maximum likelihood estimator (MLE) - $\hat{\pi}_{mle}$

Contrarily, the uniformly minimum variance unbiased estimator (UMVUE) is unbiased and therefore a preferred estimator (Jung and Kim, 2004).

$$\hat{\pi}_{umvue} = \begin{cases} \dfrac{x_1}{n_1}, & m = 1 \\ \dfrac{\sum_{x_1=\max(r_1+1,\ s-n_2^*)}^{\min(s,n_1)} \binom{n_1-1}{x_1-1}\binom{n_2^*}{s-x_1}}{\sum_{x_1=\max(r_1+1,\ s-n_2^*)}^{\min(s,n_1)} \binom{n_1}{x_1}\binom{n_2^*}{s-x_1}}, & m = 2 \end{cases}$$

Equation 2: Uniformly minimum variance unbiased estimator (UMVUE): $\hat{\pi}_{umvue}$

The following SAS macro performs the calculation of UMVUE where num_success = s:

```
%macro UMVUE (num_success=, n1=, r1=, n2star=);
   data point_est (keep= umvue);
   num1=0;
   denom1=0;
   do X1=max((&r1+1),(&num_success-&n2star)) to min(&num_success,&n1);
        num0=comb(&n1-1, x1-1)*comb(&n2star, &num_success-x1);
```

```
            num1=num1+num0;
            denom0=comb(&n1, x1)*comb(&n2star, &num_success-x1);
            denom1=denom1+denom0;
            UMVUE=round(num1/denom1, 0.001);
            /* output; */
        end;
    run;
%mend;
```

The following macro call will produce the UMVUE estimate for design 2 from Table 1.

```
    %UMVUE(num_success=17, n1=19, r1=6, n2star=23);
```

P-VALUE ACCOUNTING FOR ACTUAL AND PLANNED SAMPLE SIZES

If $m = 1$, then standard exact binomial methods may be used, and the corresponding p-value is $p = P_{\pi_0}(X_1 \geq x_1)$. If $m = 2$ and the actual sample size in stage 2 is the same as the planned sample size ($n_2^* = n_2$), then the exact p-value is computed as

$$p = \sum_{x_1=r_1+1}^{n_1} P_{\pi_0}[X_1 = x_1]P_{\pi_0}[X_2 \geq x_t - x_1]$$

Equation 3: P-value when $n_2^* = n_2$

where $X_i \sim$ *Binomial* (n_i, π_0).

However, when the stage 2 sample size deviates from the planned sample size $(n_2^* \neq n_2)$, then equation 3 is not applicable (Koyama and Chen, 2008). They propose the following approach for calculating the p-value when $n_2^* \neq n_2$:
1. Calculate the conditional p-value using the observed n_2^*, denoted as $cpval = P_{\pi_0}[X_2^* \geq x_2|n_2^*]$, where $X_2^* \sim Binomial(n_2^*, \pi_0)$. Note that once the data is observed, $cpval$ is a fixed constant.
2. For the conditional power function, $A(x_1, \pi^*) = P_{\pi^*}[X_2 \geq r_t - x_1 |n_2]$, determine the value of π^* such that $cpval = A(x_1, \pi^*)$. Note that $A(X_1, \pi^*)$ is calculated using the *planned* n_2, meaning $X_2 \sim Binomial(n_2, \pi^*)$.
    a. π^* can be determined by numerical methods, or by exploiting the relationship between the CDF of a binomial random variable and the CDF of a Beta distribution: if $Z \sim Binom(n, \pi)$, then $P_\pi(Z \geq z) = P(B \leq \pi)$, where $B \sim Beta(z, n - z + 1)$.
3. Compute the final unconditional p-value by summing across all possible values of x_1:

$$p = \sum_{x_1=r_1+1}^{n_1} P_{\pi_0}[X_1 = x_1]A(x_1, \pi^*)$$

Equation 4: P-value when $n_2^* \neq n_2$

Note that Equation 4 reduces to Equation 3 if $n_2^* = n_2$.

The following SAS macro calculates the p-value, accounting for possible deviations between the actual and planned sample sizes:

```
    %macro Simon_p_KC (num_success=, n1=, n=, r1=, rt=, p0=, n2star=, x1=);
        data pval (keep=cpval calpha pstar p_val);
        /*conditional p-value of stage 2 */
        cpval = round(1-CDF('BINOMIAL', (&num_success-&x1)-1, &p0, &n2star),
        0.0001);
        /*conditional type I error of stage 2 */
```

```
    calpha = round(1-CDF('BINOMIAL', (&rt+1)-(&x1)-1, &p0, &n-&n1),
    0.0001);
    /*finding pstar*/
    pstar=round(betainv(cpval, (&rt+1)-&x1, (&n-&n1)-((&rt+1)-&x1)+1),
    0.0001);
    /* function calculating p-value using KC equation (7) */
    p_val=0;
    do X1=&r1+1 to &n1;
        X2 = (&rt+1) - X1; /* possible values of X2 should be based on
        rt*/
        pr_x = PDF('BINOMIAL', X1, &p0, &n1)*(1-CDF('BINOMIAL', X2-1,
        pstar, &n-&n1));
        p_val =p_val+pr_x;
        /* output; */
    end;
    proc print data=pval label;
    label cpval='Conditional p-value'
    calpha='Conditional alpha'
    p_val='Unconditional p-value'
    ;
    run;
%mend;
```

The following macro calls will produce conditional p-value *cpval*, the conditional type I error rate, and (most importantly) the unconditional final p-value for the 3 designs described in Table 1.

```
%Simon_p_KC(num_success=6, n1=10, n=29, r1=1, rt=5, p0=0.1, n2star=19,
x1=2);
%Simon_p_KC(num_success=17, n1=19, n=39, r1=6, rt=16, p0=0.3,n2star=23,
x1=7);
%Simon_p_KC(num_success=12, n1=19, n=39, r1=3, rt=8, p0=0.15, n2star=6,
x1=8);
```

CONFIDENCE LIMITS ACCOUNTING FOR ACTUAL AND PLANNED SAMPLE SIZES

If $m = 1$, the exact 2-sided 95% confidence interval can be computed using Clopper-Pearson method. If $m = 2$, we can compute a p-value using the approach described above for testing $H_0: \pi \leq \pi_0'$ for any π_0'. Using a similar approach as the Clopper-Pearson method of "inverting the hypothesis testing", a 2-sided 95% confidence interval is a collection of π_0' such that the corresponding p-value is within [0.025, 0.975]. Of special note, π^* changes with different π_0'. The macro below computes π^* for every incrementally increasing π_0' until criteria for lower or upper bound is met. The boundaries are constructed using 2 one-sided 95% confidence levels, so that the decision to reject the null hypothesis corresponds to the p-value (and the original Simon's Two-Stage design, if the sample size remains as planned). Additionally, if one wants to report the associated point estimate according to (Koyama and Chen, 2008), which is the $\hat{\pi}$ which corresponds to a p-value of 0.5, this can be performed through this macro by simply setting alpha parameter in the macro call to 0.5.

```
%macro Simon_CI_KC (num_success=, n1=, n=, r1=, rt=, n2star=, x1=, al-
pha=);
data confint (keep=p0prime);
p_val=0;
p0prime=0;
do while (p_val < &alpha);
    p0prime = p0prime + 0.0001;
    /*conditional p-value of stage 2 for each incrementally increas-
    ing p0prime */
```

```
              cpval = round(1-CDF('BINOMIAL', (&num_success-&x1)-1, p0prime,
              &n2star), 0.0001);
              /*pstar for each incrementally increasing p0prime */
              pstar=round(betainv(cpval, (&rt+1)-&x1, (&n-&n1)-((&rt+1)-
              &x1)+1), 0.0001);
              /* function calculating p-value using KC equation (7) */
              /* the function below is simultaneously evaluated with p0prime
              and corresponding pstar */
              do X1= &r1+1 to &n1;
                    X2 = (&rt+1) - X1;
                    pr_x = PDF('BINOMIAL', X1, p0prime, &n1)*(1-CDF('BINOMIAL',
                    X2-1, pstar, &n-&n1));
                    p_val =p_val+pr_x;
                    if x1=&r1+1 then p_val=pr_x;
                    /* output; */
              end;
          end;
      run;
  %mend;
```

The following macro calls will produce median response rate with associated one-sided lower and upper confidence limits for the 2nd design described in Table 1. Depending on the value of alpha parameter *P0prime* can be one-sided confidence limit or median response rate.

```
      %Simon_CI_KC(num_success=17, n1=19, n=39, r1=6, rt=16, n2star=23, x1=7,
      alpha=0.05);
      %Simon_CI_KC(num_success=17, n1=19, n=39, r1=6, rt=16, n2star=23, x1=7,
      alpha=0.5);
      %Simon_CI_KC(num_success=17, n1=19, n=39, r1=6, rt=16, n2star=23, x1=7,
      alpha=0.95);
```

COLLATE INDIVIDUAL SAS MACROS TOGETHER

A complete macro puts together all individual SAS macros to provide the user a SAS report with the following statistics: UMVUE - point estimate, MEDIAN - point estimate based on p-value mid-point, unconditional p-value, and (one-sided) lower and upper confidence limits.

```
  %macro Simon_KC (num_success=, n1=, n=,  r1=, rt=, p0=, n2star=, x1=, al-
  pha=);
  data pval (keep= umvue cpval calpha p_val ord);
  num1=0;
  denom1=0;
  do X1=max((&r1+1),(&num_success-&n2star)) to min(&num_success,&n1);
        num0=comb(&n1-1, x1-1)*comb(&n2star, &num_success-x1);
        num1=num1+num0;
        denom0=comb(&n1, x1)*comb(&n2star, &num_success-x1);
        denom1=denom1+denom0;
        UMVUE=round(num1/denom1, 0.0001);
  /*    output; */
   end;

  /*conditional p-value of stage 2 */
  cpval = round(1-CDF('BINOMIAL', (&num_success-&x1)-1, &p0, &n2star),
  0.0001);
```

```sas
/*conditional type I error of stage 2 */
calpha = round(1-CDF('BINOMIAL', (&rt+1)-(&x1)-1, &p0, &n-&n1),
0.0001);

/*finding pstar*/
pstar=round(betainv(cpval, (&rt+1)-&x1, (&n-&n1)-((&rt+1)-&x1)+1),
0.0001);

/* function calculating p-value using KC equation (7)  */
p_val=0;
      do X1=&r1+1 to &n1;
            X2 = (&rt+1) - X1; /*To generalize the macro possible val-
ues of X2 should be based on rt*/
            pr_x = PDF('BINOMIAL', X1, &p0, &n1)*(1-CDF('BINOMIAL', X2-
            1, pstar, &n-&n1));
            p_val =round(p_val+pr_x, 0.0001);
/*          output; */
      end;
ord=1;
run;


/* A confidence interval and a reasonable point estimate */
data confint (keep= ord p0prime );
p_val=0;
p0prime=0;

do clevel = &alpha, 0.5, 1-&alpha;
 do while (p_val < clevel) ;
 p0prime = p0prime + 0.0001;

/*conditional p-value of stage 2 for each incrementally increasing
p0prime */
cpval = round(1-CDF('BINOMIAL', (&num_success-&x1)-1, p0prime,
&n2star), 0.0001);

/*pstar for each incrementally increasing p0prime */
pstar=round(betainv(cpval, (&rt+1)-&x1, (&n-&n1)-((&rt+1)-&x1)+1),
0.0001);

/* function calculating p-value using KC equation (7)*/
/* KC noted that pstar changes with different p0prime*/
/* the function below is simultaneously evaluated with p0prime and cor-
responding pstar */
      do X1= &r1+1 to &n1;
            X2 = (&rt+1) - X1;
            pr_x = PDF('BINOMIAL', X1, p0prime, &n1)*(1-CDF('BINOMIAL',
            X2-1, pstar, &n-&n1));
            p_val =p_val+pr_x;
            if x1=&r1+1 then p_val=pr_x;
/*          output; */
      end;
 end;
 ord=1;
 output;
end;
run;
```

```
    proc transpose data=confint out=confint1 (rename=(col1=LCL col2=MEDIAN
    col3=UCL) drop=_name_) prefix=col;
    var p0prime;
    by ord;
    run;

    data final;
    merge pval confint1;
    by ord;
    run;

    proc print data=final (drop=ord) label;
        title "Proper inference for Simon's Two Stage Design";
        label umvue='UMVUE (Point Estimate)'
              median='MEDIAN (Point Estimate)'
              cpval='Conditional P-value'
              calpha='Conditional Alpha'
              p_val='Unconditional P-value'
        ;
        var umvue median p_val lcl ucl;
    run;
%mend;
```

The following macro calls produces SAS report for each design described in Table 1.

```
%Simon_KC(num_success=6, n1=10, n=29,  r1=1, rt=5, p0=0.1, n2star=19, x1=2,
alpha=0.05);
```

Proper inference for Simon's Two Stage Design

Obs	UMVUE (Point Estimate)	MEDIAN (Point Estimate)	Unconditional P-value	LCL	UCL
1	0.2613	0.2147	0.0471	0.1016	0.4008

Output 1. SAS report for design 1


```
%Simon_KC(num_success=17, n1=19, n=39,  r1=6, rt=16, p0=0.3, n2star=23,x1=7,
alpha=0.05);
```

Proper inference for Simon's Two Stage Design

Obs	UMVUE (Point Estimate)	MEDIAN (Point Estimate)	Unconditional P-value	LCL	UCL
1	0.4381	0.4046	0.0827	0.2821	0.5461

Output 2. SAS report for design 2


```
%Simon_KC(num_success=12, n1=19, n=39,  r1=3, rt=8, p0=0.15, n2star=6, x1=8,
alpha=0.05);
```

Proper inference for Simon's Two Stage Design

Obs	UMVUE (Point Estimate)	MEDIAN (Point Estimate)	Unconditional P-value	LCL	UCL
1	0.48	0.4352	.0008	0.2707	0.6046

Output 3. SAS report for design 3

CONCLUSION

The expanded SAS program now has macros to calculate inferential statistics with and without deviations in the planned vs. actual sample size at stage two, and also calculates two types of point estimates of the response rate at stage two - uniformly minimum variance unbiased estimator (UMVUE) and p-value mid-point based point estimate. We verified the macros provide equivalent results for an example in the literature.

REFERENCES

Koyama, T. Chen, H. 2008. "Proper inference from Simon's two-stage designs." S*tat Med.* 27(16): 3145-54.

Sin-Ho, J. 2020. *Randomized Phase II Cancer Clinical Trials*. Chapman & Hall/CRC Biostatistics Series.

Zhao, J. Yu, M. Feng, XP. 2015 Statistical inference for extended or shortened phase II studies based on Simon's two-stage designs. BMC Med Res Methodol 15, 48

Karanevich, A. Ames, M. 2020. "A SAS Macro for Calculating Confidence Limits and P-values Under Simon's Two-Stage Design." *PharmaSUG Proceedings of the Quick Tips*, Paper QT- 035.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Mikhail Melikov
mikhail.melikov@cytel.com

Alex Karanevich
akaranevich@embstats.com

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