```
*****************************************************************
DEVELOPER   : Takashi Kitahara (Novartis Pharma K.K)
DATE        : 2017/AUG/03
PLATFORM    : SAS 9.4
DESCRIPTION : Calculatin of 4 types of Probability of Success (PoS)
              1: Classical criteria/Classical power
              2: Assurance (Classical criteria/Bayesian power)
              3: Bayesian criteria/Classical power
              4: Bayesian criteria/Bayesian power
*****************************************************************;

%let p0 = 0.2  ;       * H0 ;
%let pstar = 0.4  ;    * H1 ;
%let alpha = 0.05 ;    * significant level ;
%let eta = 0.95 ;      * threshold ;
%let beta_alpha = 1 ; * parameter alpha for beta distribution ;
%let beta_beta = 1 ;  * parameter beta for beta distribution ;
%let max_n = 150 ;     * maximum sample size (up to 1000) ;


/* Calculatin of PoS */
%macro m_PoS(group,
             a_alpha = 1,  /* 1=Classical criteria, 2= Bayesian criteria */
             power = 1     /* 1=Classical PoS, 2= Bayesian PoS */
             ) ;
%if &a_alpha. = 1 %then %do ;
data DATA1 ;
  p0 = &p0. ;
  pstar = &pstar. ;
  alpha = &alpha. ;
  do n = 1 to &max_n. ;
    do u = 1 to n ;
      actual_alpha = round(1-probbnml(p0, n, u - 1), 0.00001) ;
      if actual_alpha <= alpha then output ;
    end ;
  end ;
run ;
%end ;

%if &a_alpha. = 2 %then %do ;
data DATA1 ;
  p0 = &p0. ;
  pstar = &pstar. ;
  eta = &eta.;
  a = &beta_alpha. ;
  b = &beta_beta. ;
  do n = 1 to &max_n. ;
    do u = 1 to n ;
      s0 = u ;
      f0 = n - s0 ;
      a0 = a + s0 ;
      b0 = b + f0 ;
      prob0 = 1-probbeta(p0, a0, b0) ;   * Prior distribution ;
      if prob0 >= eta then output ;
    end ;
  end ;
run ;
%end ;

proc sort data = DATA1 ;
  by n u ;
run ;
proc sort data = DATA1 nodupkey ;
```

```
  by n ;
run ;


/* Calculate power */
%if &power. = 1 %then %do ;
data POS&group.(keep = n PoS&group. u&group.) ;
  set DATA1 ;
  PoS&group. = round(1-probbnml(pstar, n, u - 1), 0.00001) ;
  u&group. = u ;
run ;
%end ;

%if &power. = 2 %then %do ;
data DATA2 ;
  set DATA1 ;
  do u_dash = 0 to u-1 ;
    output ;
  end ;
run ;

data POS&group.(keep = n PoS&group. u&group.) ;
  set DATA2 ;
  by n ;
  a = &beta_alpha. ;
  b = &beta_beta. ;
  retain PoS_ (0) ;
  if first.n then PoS_ = 0 ;
  comb = comb(n, u_dash) ;
  beta0 = beta(a, b) ;                  * denominator of beta-binomial distribution ;
  beta1 = beta(u_dash+a, n-u_dash+b) ; * numerator of beta-binomial distribution ;
  mk = comb*beta1/beta0 ;
  PoS_ = PoS_+mk ;
  PoS&group. = 1-PoS_ ;
  u&group. = u ;
  if last.n then output ;
run ;
%end ;
%mend m_PoS ;

%m_PoS(1) ;                           * Classical criteria/Classical PoS ;
%m_PoS(2, power = 2) ;                * Assurance (Classical criteria/Bayesian PoS) ;
%m_PoS(3, a_alpha = 2) ;              * Bayesian criteria/Classical PoS ;
%m_PoS(4, a_alpha = 2, power = 2) ;   * Bayesian criteria/Bayesian PoS ;


data FINAL ;
  merge POS1-POS4 ;
  by n ;
run ;

title "Probability of Success versus Sample size: p0=&p0., pstar=&pstar., Prior:
Beta(&beta_alpha., &beta_beta.)" ;
proc sgplot data = FINAL ;
  series x = n y=PoS1 / lineattrs = (pattern = 1 color = red) ;
  series x = n y=PoS2 / lineattrs = (pattern = 1 color = blue) ;
  series x = n y=PoS3 / lineattrs = (pattern = 1 color = green) ;
  series x = n y=PoS4 / lineattrs = (pattern = 1 color = purple) ;

  xaxis label = "Sample size";
  yaxis label = "Probability of Success";
  refline 0.8 / axis = y lineattrs = (pattern=2);
run ;
```