Sequential clinical trials – two-stage designs

 # Calculating optimal designs "by hand" is tedious but feasible, for small samples.
 # First, create a function that calculates error probabilities α and β, probabilities of
 # early termination PET, and the average sample number ASN.

```
> TwoStageDesign = function(n1,n,r1,r,p0,p1){
+ k = (r1+1):n1;
+ n2=n-n1;
+ alpha = sum( dbinom(k,n1,p0)*(1-pbinom(r-k,n2,p0)) );
+ beta = pbinom(r1,n1,p1) + sum( dbinom(k,n1,p1)*pbinom(r-k,n2,p1) );
+ PET0 = pbinom(r1,n1,p0)
+ PET1 = pbinom(r1,n1,p1)
+ ASN0 = n1 + (1-PET0)*n2
+ ASN1 = n1 + (1-PET1)*n2
+ return(data.frame(alpha,beta,PET0,PET1,ASN0,ASN1))}
```

 # For example:
```
> TwoStageDesign(5, 10, 3, 7, 0.5, 0.8)
       alpha       beta   PET0    PET1    ASN0    ASN1
1 0.04492187 0.3893093 0.8125 0.26272 5.9375 8.6864
```

 # Then, we'll search for the optimal design, minimizing ASN0 = $E_0(T)$ in nested loops.
 # If a new design yields a smaller ASN0, we update the design parameters.

```
> OptimalDesign=function(p0,p1,alpha,beta,Nmax){
+ BestComb = data.frame(n1=0,r1=0,r=0,ASN0=Nmax+1)
+ for (n1 in 1:Nmax){ for (r1 in 1:n1){ for (r in (r1+1):n){
+ Params = TwoStageDesign(n1,Nmax,r1,r,p0,p1)
+ if ( Params$alpha <= alpha & Params$beta <= beta & Params$ASN0 < BestComb$ASN0 ){
+ BestComb$n1 = n1; BestComb$r1=r1; BestComb$r=r; BestComb$ASN0=Params$ASN0;
}}}}
+ if (BestComb$ASN0 <= Nmax){ return(BestComb) } else {return("The optimal design cannot
be found. Increase Nmax.")}}
```

 # For any α and β, this function calculates the optimal design or state that it is impossible.
 # For example:
```
> OptimalDesign(0.5,0.8,0.1,0.33,10)
  n1 r1 r  ASN0
```

```
1  6  3 7 7.375
> OptimalDesign(0.5,0.8,0.05,0.1,10)
[1] "The optimal design cannot be found. Increase Nmax."
```

R package "clinfun" (Clinical Trial Design and Data Analysis Functions) contains designs for
standard two-stage and group sequential clinical trials

```
> install.packages("clinfun")
> library(clinfun)
```

Calculate parameters of a two-stage design for the Binomial one-sample testing of
$H_0: p = 0.25$ (unacceptable response rate) vs $H_A: p = 0.40$ (acceptable response rate)
with the significance level 0.05 and power 0.90.

```
> ph2simon(0.25, 0.40, 0.05, 0.10)

Simon 2-stage Phase II design

Unacceptable response rate:  0.25
Desirable response rate:  0.4
Error rates: alpha =  0.05 ; beta =  0.1


        r1 n1  r   n EN(p0)  PET(p0)
Optimal 10 37 31 99  56.16  0.6909
Minimax 13 57 27 83  72.11  0.4190


Warning message:
In print.ph2simon(x) :   Optimal sample size too close to nmax.
  Try increasing nmax (current value = 100)
```

By default, Nmax = 100. We can set the maximum sample size, but notice that it may be
insufficient to attain the required significance level and power.

```
> ph2simon(0.25, 0.40, 0.05, 0.10, nmax=90)

 Simon 2-stage Phase II design

Unacceptable response rate:  0.25
Desirable response rate:  0.4
Error rates: alpha =  0.05 ; beta =  0.1


        r1 n1  r   n EN(p0)  PET(p0)
```

```
Optimal   9 37 28 87   59.48   0.5503
Minimax  13 57 27 83   72.11   0.4190

Warning message:
In print.ph2simon(x) :   Optimal sample size too close to nmax.
  Try increasing nmax (current value = 90)
```

 # So, for Nmax = 90, solution can still be found. But not for Nmax = 50…

```
> ph2simon(0.25, 0.40, 0.05, 0.10, nmax=50)

Error in ph2simon(0.25, 0.4, 0.05, 0.1, nmax = 50) :
    No feasible solution found.
        Increase maximum sample size.  Current nmax value = 50.
```