Sequential clinical trials - Binomial example

 # Simulate the situation and draw a graph

```
> p = 0.1;              # The null value of the population proportion being tested
> N = 100;              # Sample size, in a non-sequential experiment
> alpha = 0.05;
```

 # Threshold for the number of positive responses sufficient to reject H_0...

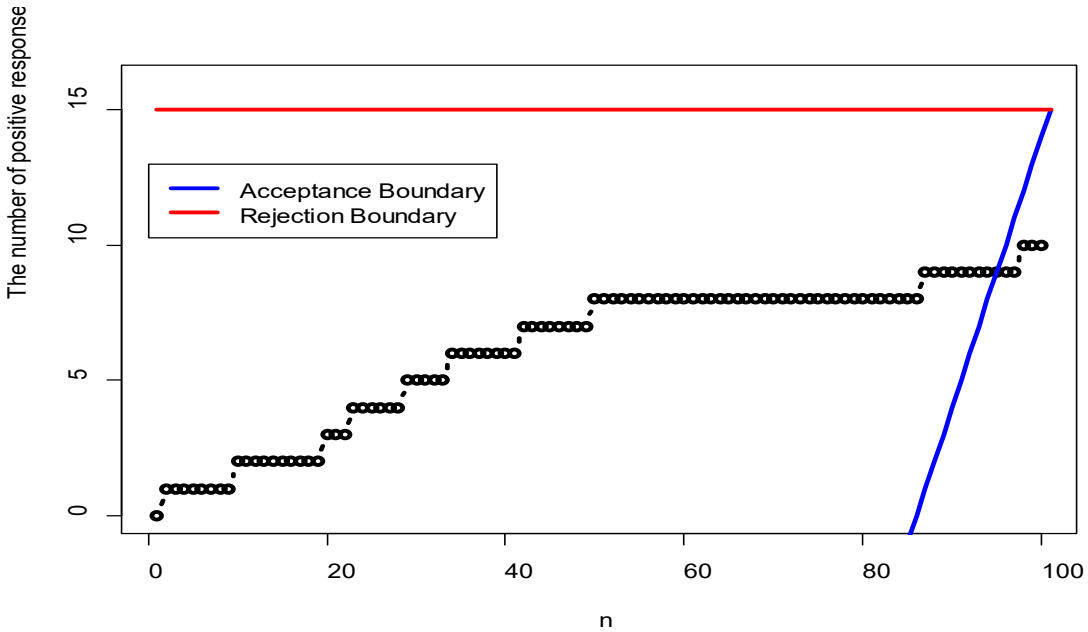
```
> Threshold = ceiling((qnorm(1-alpha)*sqrt(p*(1-p)/N) + p)*N);
```

 # Generate a random trajectory of positive responses, under H_0

```
> X = rbinom(N,1,p); n = 1:N;
> Nsuccesses = cumsum(X);
> BoundaryUpper = rep(Threshold,N);   # Rejection boundary
> BoundaryLower = n-N+Threshold-1;    # Acceptance boundary
```

 # Plot the trajectory along with the stopping boundaries

```
> plot(n,Nsuccesses,'b',lwd=3, ylim=c(0,Threshold+1),
+ ylab="The number of positive responses, among the first n participants")
> lines(c(n,N+1),c(BoundaryLower,Threshold),col="blue",lwd=3)
> lines(c(n,N+1),c(BoundaryUpper,Threshold),col="red",lwd=3)
> legend(0,Threshold-2,
+   legend=c("Acceptance Boundary","Rejection Boundary"),
+   col=c("blue", "red"), cex=1, lwd=3, bg='white')
```



 # Performance evaluation, by simulation

```
> Nruns = 10000;
> TypeIerror = rep(0,Nruns);
> SampleSize = rep(0,Nruns);

> for (i in 1:Nruns){
+ X = rbinom(N,1,p);
+ Nsuccesses = cumsum(X);
+ ContinueSampling = ((Nsuccesses < BoundaryUpper) & (Nsuccesses > BoundaryLower));
+ StoppingTime = sum(ContinueSampling)+1;
+ SampleSize[i] = StoppingTime;
+ TypeIerror[i] = Nsuccesses[StoppingTime] >= Threshold;
+                       }

> ASN = mean(SampleSize); ProbTypeIerror = mean(TypeIerror);
> print(data.frame(ASN,ProbTypeIerror))
```

ASN ProbTypeIerror
94.6292 0.0696

 # Exact performance evaluation
 # Survival probabilities
 # S[n] = P(need more than n units) = P(BoundaryLower < X[n] < BoundaryUpper)

```
> S = rep(0,N); P=S;
> for (k in 1:N){
+  S[k] = pbinom( BoundaryUpper[k]-1,k,p ) - pbinom( BoundaryLower[k],k,p ) }
```

 # Probability mass function of the stopping time = final sample size

```
>  P[2:N] = S[1:(N-1)] - S[2:N];

> # ASN = average sample number = expected sample size
> ASN = sum(P*(1:N));

# Probability of Type I error
> ProbTypeIerror = 1-pbinom( Threshold-1, N, p );
> print(data.frame(ASN,ProbTypeIerror))
      ASN ProbTypeIerror
 94.60438     0.07257297

# Why isn't P(Type I error) = α? That's the result of a Normal approximation. The Z statistic for
# testing a Binomial proportion is derived using the Normal distribution instead of the Binomial.
```