Continuous Toxicity Monitoring in Phase II trials in Oncology
Sequential Boundaries for Binary Outcomes (version 1.1)
(c) Bahjat Qaqish (2003-2005)
Department of Biostatistics, CB 7420, McGavran Greenberg Hall
University of North Carolina, Chapel Hill, NC 27599-7420
email address: bahjat_qaqish@unc.edu

Introduction

The files in this directory include programs and sample input and output files for implementing the methods described in the paper

> Ivanova A., Qaqish, B. F. & Schell, M. J. (2005). Continuous Toxicity Monitoring in Phase II trials in Oncology. *Biometrics* **61**, 540–545.

A copy of the manuscript is available upon request. While the manuscript discusses only upper boundaries, the software available here allows both upper and lower boundaries.

For users who do not wish to download or run our software, we have a long list of pre-computed upper boundaries available as plain (ASCII) text files. These files can be viewed using virtually any text editor or word processor. The files `bndry1.lst` and `bndry2.lst` contain lists of pre-computed and ready to use boundaries for a wide range of scenarios. These files can be inspected and the desired information extracted without running any of our software tools. However, the programs described below offer more power and flexibility: 1) They can compute Pocock-type upper boundaries for virtually any scenario, 2) They can analyze any given boundary regardless of how it was constructed.

Notation and the Basic Model

The setup is as follows. The responses (outcomes) are independent Bernoulli (binary) random variables with mean θ. The maximum sample size is denoted by K; the trial either stops early or continues until all K patients have been treated.

Boundaries can be based on effectiveness or on toxicity. An example is clinical trials to estimate toxicity of new drugs. In such trials, it is desirable to allow stopping the trial early if there are too many toxicities. That is achieved by implementing an *upper boundary*.

Another example is estimation of effectiveness of new drugs. It may be desirable to stop the trial early if the drug does not seem effective. That is achieved by implementing a *lower boundary*. It may even be desirable to stop early if the drug seems to be very effective, in which case there will be an upper boundary as well.

Generally, there are two boundaries; an upper boundary and a lower boundary. The upper boundary sets the maximum number of events that will be allowed in the course of the trial. On the other hand, the lower boundary sets the minimum number of events that will be allowed. The trial will be stopped if either boundary is reached at any point.

It is assumed that the trial continues even if it becomes clear at any point that neither boundary will be hit. This is the situation when a toxicity-based upper boundary is implemented in a phase II trial. Phase II trials are carried out primarily to estimate efficacy, not toxicity.

For convenience, define the *right boundary* as that reached after all K patients have been treated and neither the lower nor the upper boundary has been reached.

The lower boundary will be denoted $l_1, l_2, l_3, \cdots, l_K$, the upper $u_1, u_2, u_3, \cdots, u_K$. If a trial does not implement a lower boundary, the lower boundary is set to -1, that is $l_1 = l_2 = \cdots = l_K = -1$. If a trial does not implement an upper boundary, the upper boundary is set to $K + 1$, that is $u_1 = u_2 = \cdots = u_K = K + 1$.

The trial will be stopped if there are:
$\leq l_1$ or $\geq u_1$ events in the first patient,
$\leq l_2$ or $\geq u_2$ events in the first 2 patients,
$\leq l_3$ or $\geq u_3$ events in the first 3 patients,
...
$\leq l_K$ or $\geq u_K$ events in the first K patients.
Such stopping is called *early stopping* (even if it happens with the K-th patient). Otherwise, the study is considered to have reached the right boundary (thinking of the trial as moving from left to right).

Note that the boundaries must be nondecreasing ($l_{i+1} \geq l_i$ and $u_{i+1} \geq u_i$) and the upper must be greater than the lower ($u_i > l_i$).

Software Installation

To run the programs (files that end with .exe) they must be downloaded to a local directory. The programs are designed to run in a DOS window. To open a DOS window in Windows/XP, click on Start, then Programs, Accessories and Command Prompt. At the command prompt, you can run a program by typing its full path name, or just its name if the program file is in the *path*. Ask your system administrator or a local guru for help if you have difficulty with this step.

Finding a Boundary Using CP20 and CP20S

The files `cp20.exe` and `cp20s.exe` are DOS/Windows programs for finding Pocock-type upper boundaries for continuous monitoring of clinical trials involving binary (Bernoulli) outcomes. Program `cp20.exe` performs exact computations while `cp20s.exe` uses simulation (one million sequences). These programs are interactive and self-explanatory. They generate output directly useful for the user, and also generate a file named `_bndry_.dat` that can be used (as is, or possibly after user modification) as input for programs described below.

Suppose we want to design a study with 10 patients such that the study terminates early with probability 5% if the true toxicity rate is 20%. We run the program CP20. The program generates the following output and prompts for input:

```
Continuous monitoring of clinical trials involving binary outcomes
```

```
   using Pocock-type boundaries (version 1.1)
          (c) Bahjat Qaqish (2004)

(Exact calculations)
Definitions:
  K     = the maximum planned sample size
  THETA = the event probability
  PHI   = the desired probability of early stopping (hitting the boundary)
Input  K  THETA  PHI:
```

After entering the design parameters, the last line above will look as follows

```
Input  K  THETA  PHI: 10 0.2 0.05
```

and will be followed by the following output:

```
K     = 10
THETA = 0.2
PHI   = 0.05
PHI*  = 0.0421258 = the actual probability of hitting the upper boundary
ALPHA = 0.0272 = the pointwise stopping probability

The upper boundary is:
b[1] = 2
b[2] = 3
b[3] = 3
b[4] = 3
b[5] = 4
b[6] = 4
b[7] = 5
b[8] = 5
b[9] = 5
b[10] = 6

THETA   PHI*    E[Y]    SD[Y]   E[N]    SD[N]   E[Y/N]  SD[Y/N]

0.20    0.042   1.96    1.19    9.79    1.07    0.21    0.16
0.30    0.148   2.80    1.20    9.32    1.83    0.33    0.21
0.40    0.332   3.40    1.07    8.51    2.48    0.46    0.23
0.50    0.561   3.70    0.93    7.41    2.83    0.58    0.23
0.60    0.773   3.71    0.86    6.18    2.78    0.69    0.21
0.70    0.918   3.51    0.78    5.02    2.35    0.78    0.18
0.80    0.984   3.26    0.59    4.07    1.66    0.86    0.16
0.90    0.999   3.07    0.29    3.41    0.88    0.93    0.12
```

```
Definitions:
  Y    = the number of events, random, between 0 and N
  N    = the number of patients, random, between 1 and K
  PHI* = the actual probability of early stopping (hitting the boundary)
  E[ ]   denotes the expected value (mean)
  SD[ ]  denotes the standard deviation


The boundary has been written to file _bndry_.dat in a format
appropriate for reading into other supplied programs (e.g. cp3).
Date and time: Mon Feb 7 11:48:07 2005
```

Here is an explanation of the output shown above. Due to discreteness, an exact early stopping probability of 5% is not achievable. The largest achievable probability under 5% is `PHI* = 0.0421258`. The Pocock-type boundary is constructed using a constant pointwise stopping probability, `ALPHA = 0.0272`. The boundary is `b[1]` up to `b[10]`. The boundary `b[1] = 2` means that the study stops if there are 2 toxicities in the first patient, which simply means that the study will never stop after just one patient. Similarly, because `b[2] = 3` the study will never stop after two patients. The boundary `b[3] = 3` means that the study stops if there are 3 toxicities in the first 3 patients, `b[4] = 3` means that the study stops if there are 3 toxicities in the first 4 patients, `b[5] = 4` means that the study stops if there are 4 toxicities in the first 5 patients, $\cdots$, `b[10] = 6` means that the study stops if there are 6 toxicities in the first 10 patients (although this will never happen because the study would have stopped at 5 toxicities in the first 9 patients). Note that the lower boundary is implicitly fixed at -1, and is not printed.

The table above shows some aspects of performance of the design. The first line says that if toxicity is actually 20%, the number of toxicities has mean 1.96 and standard deviation 1.19, sample size (number of patients actually treated) has mean 9.79 and standard deviation 1.07, the observed toxicity rate has mean 0.21 and standard deviation 0.16. These are very close to what would happen without an early stopping rule. However as true toxicity increases, the effect of the design becomes more appreciable. For example, if true toxicity is 50%, the chance of early stopping is 56.1% and the expected sample size is 7.41.

The program generates a file `_bndry_.dat`:

```
0.2
10
-1 2
-1 3
-1 3
-1 3
-1 4
-1 4
-1 5
-1 5
-1 5
-1 6
```

```
This file was generated by program cp20
PHI   = 0.05
pr(Hitting the upper boundary) = 0.0421258
ALPHA = 0.0272
```

This file can be used as is, or edited manually, and used as input to the programs described below.

Computing the Performance of a Given Boundary Using CP3 and CP3S

The files `cp3.exe` and `cp3s.exe` are DOS/Windows programs for computing boundary crossing probabilities and other quantities for a sequential design. `cp3.exe` performs exact computations while `cp3s.exe` uses simulation (one million sequences) to obtain approximate answers. `cp3.exe` is meant for small sample sizes while `cp3s.exe` is meant for large sample sizes. What is "small" or "large" is to be determined by the user. If `cp3.exe` takes too long a time to run, one can say that sample size is large and try to use `cp3s.exe` instead.

The input to the programs is a sequence of numbers separated by white space. The numbers are:
θ = the event probability
K = the planned maximum sample size (number of patients)
$l_1 \quad u_1 \quad l_2 \quad u_2 \quad \cdots \quad l_K \quad u_K$ = lower and upper boundaries

Note that the boundaries must be nondecreasing ($l_{i+1} \geq l_i$ and $u_{i+1} \geq u_i$) and the upper must be greater than the lower ($u_i > l_i$). The programs quit if that is not the case, if $K < 1$, if $\theta \leq 0$, if $\theta \geq 1$, or if there is any sort of input error.

The programs have identical input and output formats, so one can switch between them with no change to the input file. The output is labeled appropriately as either "Exact" or "Simulation-based".

The programs print a table of all possible outcomes (number of events, Y, and number of patients, N) and their probabilities. The programs also print summary measures such as the mean, $E[]$, and standard deviation, $S[]$, of the observed proportion Y/N, number of events, Y, and number of patients, N.

The programs accept input from the standard input file (stdin) and write output to the standard output file (stdout). This means that they can read input from the keyboard or from a file and can send output to the screen or to a file.

To find the exact distribution of all possible outcomes for the boundary in file `_bndry_.dat` and send the output to file `_bndry_.out`, we run CP3 with the following command:

```
cp3  < _bndry_.dat  > _bndry_.out
```

The output (file `_bndry_.out`) is:

```
Exact Boundary Crossing Probability, version 1.1
```

```
(c) Bahjat Qaqish, 2003

The distribution of all possible outcomes (events/subjects):

    Events   Subjects   Boundary    Probability
         3          3      upper          0.008
         3          4      upper         0.0192
         4          6      upper       0.006144
         5          8      upper     0.00262144
         5          9      upper    0.006160384
         0         10      right   0.1073741824
         1         10      right    0.268435456
         2         10      right    0.301989888
         3         10      right   0.1946157056
         4         10      right   0.0750780416
         5         10      right   0.0103809024

P(event) = 0.2
Maximum sample size = 10
P(Hitting the lower boundary) = 0
P(Hitting the upper boundary) = 0.0421258
P(Hitting the right boundary) = 0.957874
E[#events/#subjects] = 0.212405
S[#events/#subjects] = 0.161949
E[#events]     = 1.95856
S[#events]     = 1.1897
E[#subjects]   = 9.79282
S[#subjects]   = 1.07481
E[] denotes the mean, S[] denotes the standard deviation
Program: cp3
CPU time: 0 milliseconds
Real time: 0 seconds
Date and time: Mon Feb 14 14:48:27 2005
```

Suppose we want to modify the above boundary so that the study never stops after only 3 patients (Naturally, another approach is to go back an rerun CP20 with a smaller value for PHI). To compute the performance characteristics of the modified design we edit `_bndry_.dat` so that it looks as follows:

```
0.2
10
-1 2
-1 3
-1 4
-1 4
```

```
-1 4
-1 4
-1 5
-1 5
-1 5
-1 6
```

and then run CP3 with the following command:

```
cp3  < _bndry_.dat  > _bndry_.out
```

The output (file `_bndry_.out`) is:

```
Exact Boundary Crossing Probability, version 1.1
(c) Bahjat Qaqish, 2003
The distribution of all possible outcomes (events/subjects):

    Events   Subjects   Boundary    Probability
         4          4      upper         0.0016
         4          5      upper        0.00512
         4          6      upper        0.01024
         5          8      upper      0.0032768
         5          9      upper     0.00720896
         0         10      right   0.1073741824
         1         10      right    0.268435456
         2         10      right    0.301989888
         3         10      right    0.201326592
         4         10      right    0.081788928
         5         10      right   0.0116391936

P(event) = 0.2
Maximum sample size = 10
P(Hitting the lower boundary) = 0
P(Hitting the upper boundary) = 0.0274458
P(Hitting the right boundary) = 0.972554
E[#events/#subjects] = 0.20475
S[#events/#subjects] = 0.14103
E[#events]     = 1.98202
S[#events]     = 1.22191
E[#subjects]   = 9.91008
S[#subjects]   = 0.60139
E[] denotes the mean, S[] denotes the standard deviation
Program: cp3
CPU time: 0 milliseconds
Real time: 0 seconds
Date and time: Mon Feb 14 14:44:50 2005
```

Other reasons to run CP3 include:

1. Computing the performance characteristics for other values of θ.
2. Computing the performance characteristics of a sequential phase-II design that incorporates both a lower boundary (for futility) and an and upper boundary (for effectiveness); the study stops early if the treatment seems futile or seems very effective.

The output files in this directory were generated by the commands:
cp3 < example1.dat > example1.out
cp3 < example2.dat > example2.out
cp3 < example3.dat > example3.out
cp3 < example4.dat > example4.out
cp3s < example4.dat > example4s.out
cp3s < example5.dat > example5s.out
cp3 < _bndry_.dat > _bndry_.out

The effectiveness of the simulation approach for large sample size is illustrated by comparing the results and the run times in `example4.out` and `example4s.out`

Note: Some programs print the CPU time. For technical reasons, if the actual CPU time exceeds about 35 minutes the value printed will not be correct (and may be negative). The real time printed is always correct.