Software for Implementing the Bayesian Enhancement Two-stage (BET) Designs for Binary and Survival Endpoints

1 Introduction

The software used for implementing the Bayesian enhancement two-stage designs for binary and survival endpoints consists of two parts. For the binary endpoint, the calibration is performed via numerical enumeration to find the optimal design parameters; the file "BET binary.r" contains the corresponding R program for searching the optimal parameters. For the survival endpoint, as the design does not specify any fixed sample size, the software does not involve parameter calibration, and primarily implements decision rule based on the highest posterior density (HPD) interval and the posterior probability; the file "BET survival.r" contains the corresponding R program. It is worth emphasizing that the installation of JAGS is required for loading the R program for survival endpoint.

2 BET design for binary endpoint

The core function for calibrating the design parameters is `findparam()`. The function takes as input the minimum required response rate, the required HPD interval length, the posterior probability cutoff and the parameter for the prior beta distribution of the experimental response rate.

As an example, for the first stage, we set the parameters as follows.

```
# The uninteresting response rate in the null hypothesis
p0=0.2

# Required HPD interval length for stage 1
l1=0.4

# Posterior probability cutoff for stage 1
Pi1=0.8

# Parameters a and b for the beta prior distribution
a=1
b=1

# Minimum sample size for the first stage;
# for noninformative prior, typically set as 1;
# for informative prior typically set as 10 to 15
minn=1

# Maximum sample size, typically set as a large integer;
maxn=200
```

To calibrate the sample size and the required number of responses in the first stage, we run the function `findparam()`. The function output the sample size $n_1 = 8$ and the required number of responses $r_1 = 3$.

```
stage1nr=findparam(Pi1,a,b,p0,l1,maxn,minn)

#Output as follows:
#n
#[1] 8

#r
#[1] 3
```

For the second stage, we set the parameters as follows.

```
# The desirable target response rate in the alternative hypothesis
p1=0.4

# Required HPD interval length for stage 2
l2=0.2

# Posterior probability cutoff for stage 2
Pi2=0.9

# Parameters a and b for the beta prior distribution
a=1
b=1

# Minimum sample size for the first stage;
# for noninformative prior, typically set as 1;
# for informative prior typically set as 10 to 15
minn=1

# Maximum sample size, typically set as a large integer;
maxn=200
```

To calibrate the sample size and the required number of responses in the second stage, we run the function `findparam()`. The function output the sample size of $n = 65$ and the required number of responses $r = 32$.

```
stage2nr=findparam(Pi2,a,b,p1,l2,maxn,minn)

#Output as follows:
#n
#[1] 65

#r
#[1] 32
```

3 BET design for survival endpoint

The software for implementing the BET design for survival endpoint requires the installation of JAGS. In addition, the R packages `R2jags`, `HDInterval`, `invgamma`, and `TeachingDemos` need to be installed from CRAN.

During the trial, suppose investigator has observed a series of event times and the associated censoring indicators.

```
# list of event times
datat=c(0.1,0.4,0.2,0.6,0.7,0.7,0.1,0.9,0.1,1)

# list of censoring indicators
is.censored=c(0,1,0,0,0,0,1,0,0,0)
```

Suppose in the first stage, the cutoff for median survival time is $\theta_0 = 1$, the posterior probability cutoff is $\pi_1 = 0.8$, the required HPD interval length is $\ell_1 = 0.5$. To compute the HPD interval length and the posterior probability of $\theta > \theta_0$, we use the function `outputsurvival()`.

```
# Cutoff for median survival time in stage 1
theta0=1

# Posterior probability cutoff for stage 1
Pi1=0.8

# Value for k_HPD, default set as 1
kHPD=1

# Required HPD interval length for stage 1
l1=0.5

outputsurvival(Pi1,theta0,kHPD,datat,is.censored)

#Output as follows:
#hpd
#[1] 0.9543774

#prob
#[1] 0.0427
```

The function outputs an HPD interval length of 0.954, which is larger than the required length of $\ell_1 = 0.5$, hence we should continue to enroll subjects. The posterior probability of $\theta > \theta_0$ is 0.0427.

Another functionality in the software is to conduct simulation studies to find the design's average sample size, which can be implemented using the function `simu()`. As an example, consider the following setup.

```
# Cutoff for median survival time in stage 1 and stage 2
theta0=1
theta1=2
```

```
# Required HPD interval length for stage 1 and stage 2
l1=0.5
l2=0.8

# Posterior probability cutoff for stage 1 and stage 2
Pi1=0.7
Pi2=0.8

# Value for k_HPD, default set as 1
kHPD=1

# Shape parameter for Weibull distribution
weibullk=1.5

# Scale parameter for Weibull distribution, such that the median
# survival time is theta1+2
lambda=(theta1+2)^weibullk/log(2)

# Assumed rate of censoring
censoring=0.8

# Number of trial replications
numsim=1000

simu(censoring,lambda,weibullk,kHPD,theta0,theta1,l1,l2,Pi1,Pi2,numsim)

#Output as follows:
#n1
#[1] 27.7

#n
#[1] 66.6
```

Based on 1000 trial replications, the function outputs the average samples for the first stage and the whole trial as $\bar{n}_1 = 27.7$ and $\bar{n} = 66.6$.