dalGenE: borrow control arm only

Yi Li

27/11/2022

```
setwd("/Users/yili/Desktop/EPIB 635/Final Project")
```

Assume we would have 117 events in treatment arm and 147 events in the control arm. Total no. of events is 264.

Interim analysis is planned at the time when there are ~$264/2 = 132$ events in total in both arms.

Because

$$\theta = \frac{1 - TE}{2 - TE} = \frac{P_t}{P_t + P_c},$$
$$TE = \frac{P_c - P_t}{P_c} = \frac{n_c - n_t}{n_c}.$$

Assume we want treatment efficacy (TE) to be 70%,

$$TE = \frac{n_c - n_t}{n_c} = 0.7,$$

so

$$0.3n_c = n_t,$$
$$n_t = 30, n_c = 102.$$

```
# install.packages("VGAM")
library(VGAM) # package for the beta-binomial distribution
```

```
## Warning: package 'VGAM' was built under R version 4.1.2
```

```
## Loading required package: stats4
```

```
## Loading required package: splines
```

```
pbetabinom.ab(117, # no. of cases in trx group
              size = 264, # no. of cases in trx group + control group
              shape1 = 1 + 30,
              shape2 = 1 + 102,
              log = FALSE)
```

```
## [1] 0.9999907
```

1. Trial simulation: without MAP

The efficacy thresholds are specified as $U_n = 0.995$ for interim stopping and $U = 0.986$ for final analysis.

```
trial = function(TE,
                 ni = c(132, 264), # total no. of cases at interim and final analyses
                 a0 = 1, b0 = 1) {
  n = c(ni[1], diff(ni, 1))
  theta = (1 - TE)/(2 - TE)
  p_eff = NULL
  p_success = NULL
  y_i = 0
  i = 0
  repeat {
    i = i + 1
    y_i = y_i + rbinom(1, n[i], theta) # y_i ~ Binom(n_i, theta)
    ai = a0 + y_i
    bi = b0 + ni[i] - y_i
    peff_new = pbeta((1 - 0)/(2 - 0), ai, bi)
    p_eff = c(p_eff, peff_new)
    p_success = c(p_success, pbetabinom.ab(q = 117,
                                           size = 264,
                                           shape1 = ai,
                                           shape2 = bi,
                                           log = FALSE))
    if (i<=1) {
      if (peff_new > 0.995 | p_success[length(p_success)] < 0.05) break
    } else break
  }
    out = list(p_eff, p_success)
  return(out)
}

trial(TE = 0.7)
```

```
## [[1]]
## [1] 1
##
## [[2]]
## [1] 0.9999979
```

1.1 FPR

```
fp = NULL
for (m in 1:10000) {
  p_eff = trial(TE = 0)
  p_eff_unlist = unlist(p_eff[[1]])
  check = ifelse(length(p_eff_unlist) < 2,
                 p_eff_unlist[length(p_eff_unlist)] > 0.995,
                 p_eff_unlist[length(p_eff_unlist)] > 0.986)
```

```
  fp = c(fp, check)
}
head(fp)
```

```
## [1] FALSE FALSE FALSE FALSE FALSE FALSE
```

```
mean(fp)
```

```
## [1] 0.0185
```

1.2 TPR

```
tp = NULL
for (m in 1:10000) {
  p_eff = trial(TE = 0.7)
  p_eff_unlist = unlist(p_eff[[1]])
  check = ifelse(length(p_eff_unlist) < 2,
                 p_eff_unlist[length(p_eff_unlist)] > 0.995,
                 p_eff_unlist[length(p_eff_unlist)] > 0.986)
  tp = c(tp, check)
}
mean(tp)
```

```
## [1] 1
```

1.3 Early stopping

The probability that the trial stops for efficacy at the interim analysis:

```
early = NULL
check1 = NULL
for (m in 1:10000) {
    p_eff = trial(TE = 0.7)
    p_eff_unlist = unlist(p_eff[[1]])
    check = length(p_eff_unlist) < 2
    early = c(early, check)
    check1 = c(check1, length(p_eff_unlist)==1)
}
mean(early)
```

```
## [1] 1
```

```
mean(check1)
```

```
## [1] 1
```

2. Trial simulation: add MAP (borrow historical control arm only)

Using Schmidli et al. (2014), p.1024, top right corner.

$$\tilde{\omega}_R \propto \frac{\omega_R f_0}{\omega_R f_0 + (1-\omega_R)\sum_k \omega_k f_k},$$

$$\tilde{\omega}_k \propto \frac{\omega_k f_k}{\sum_k \omega_k f_k},$$

$$f_k = \frac{B(a_k + y_*, b_k + n_* - y_*)}{B(a_k, b_k)}.$$

```
# posterior weight: corresponds to w_R in Eq. (7) in Schmidli et al. (2014)
w_bar = function(w, y, n, a, b, a0 = 1, b0 = 1) {
  f = beta(a + y, b + n - y)/beta(a, b)
  f0 = beta(a0 + y, b0 + n - y)/beta(a0, b0)
  w_bar = w*f/(w*f + (1 - w)*f0)
  return(w_bar)
}

w_bar(w = 0.9, y = 20, n = 100, a = 5, b = 5)
```

```
## [1] 0.8138288
```

```
options(warn = -1)
trial1 = function(TE, w,
                  ni = c(132, 264), # total no. of cases at interim and final analyses
                  a0 = 1, b0 = 1) {
  n = c(ni[1], diff(ni, 1))
  theta = (1 - TE)/(2 - TE)
  p_eff = NULL
  p_success = NULL
  y_i = 0
  i = 0

  repeat {
    i = i + 1
    y_i = y_i + rbinom(1, n[i], theta) # y_i ~ Binom(n_i, theta)
    ai = 1 + y_i
    bi = 1 + ni[i] - y_i

    ### Add robust MAP prior
    # posterior weight:
    wb = w_bar(w, y_i, n[i], ai, bi)

    # P(TE > 0|data):
    peff_new = wb*pbeta(q = 0.5,
                        shape1 = ai + 226,
                        shape2 = bi + (226 + 180) - 226) +
                   (1-wb)*pbeta(q = 0.5,
                                shape1 = ai,
                                shape2 = bi)
    p_eff = c(p_eff, peff_new)
```

```
    p_success = c(p_success,
                  wb*pbetabinom.ab(q = 117,
                                   size = 264,
                                   shape1 = ai + 226, # yi = 226 = no. of cases in ctrl arm
                                   shape2 = bi + (226 + 180) - 226,  # ni - yi
                                   log = FALSE) +
                    (1-wb)*pbetabinom.ab(q = 117,
                                         size = 264,
                                         shape1 = ai,
                                         shape2 = bi,
                                         log = FALSE)
                  )



    if (i <= 1) {
      # P(TE > 0|data) > 0.995;
      if (peff_new > 0.995 | p_success[length(p_success)] < 0.05) break
    } else break
  }
  out = list(round(p_eff, 2), round(p_success, 3))
  return(out)
}

trial1(TE = 0.7, w = 0.7)
```

```
## [[1]]
## [1] 0.89 1.00
##
## [[2]]
## [1] 0.26 1.00
```

2.1 FPR

```
options(warn = -1)
fp = NULL
p_eff <- p_eff_unlist <- check <- list()
for (m in 1:10000) {
  p_eff = trial1(TE = 0, w = 0.7)
  p_eff_unlist = unlist(p_eff[[1]])
  check = ifelse(length(p_eff_unlist) < 2,
                 p_eff_unlist[length(p_eff_unlist)] > 0.995,
                 p_eff_unlist[length(p_eff_unlist)] > 0.986)
  fp = c(fp, check)
}
head(fp)
```

```
## [1] FALSE FALSE FALSE FALSE FALSE FALSE
```

```r
mean(fp, na.rm = T)
```

```
## [1] 0.01181536
```

```r
sum(is.na(fp))
```

```
## [1] 13
```

2.2 TPR

```r
options(warn = -1)
tp = NULL
for (m in 1:10000) {
  p_eff = trial1(TE = 0.7, w = 0.7)
  p_eff_unlist = unlist(p_eff[[1]])
  check = ifelse(length(p_eff_unlist) < 2,
                 p_eff_unlist[length(p_eff_unlist)] > 0.995,
                 p_eff_unlist[length(p_eff_unlist)] > 0.986)
  tp = c(tp, check)
}
mean(tp, na.rm = T)
```

```
## [1] 1
```

```r
sum(is.na(tp))
```

```
## [1] 0
```

2.3 Early stopping

The probability that the trial stops for efficacy at the interim analysis:

```r
options(warn = -1)
early = NULL
check1 = NULL
for (m in 1:10000) {
    p_eff = trial1(TE = 0.7, w = 0.7)
    p_eff_unlist = unlist(p_eff[[1]])
    check = length(p_eff_unlist) < 2
    early = c(early, check)
    check1 = c(check1, length(p_eff_unlist)==1)
}
mean(early)
```

```
## [1] 2e-04
```

```
mean(check1)
```

```
## [1] 2e-04
```