Comparing Sample Size and Power Calculation Results for a Group Sequential Trial with a Survival Endpoint: rpact vs. gsDesign

Gernot Wassmer, Friedrich Pahlke, and Marcel Wolbers

Last change: 15 November, 2022

Contents

Summary **1**

1 The design **1**

2 Calculation with gsDesign **2**

3 Calculation with rpact **3**
3.1 Design 3
3.2 Sample size / timing of interim analyses 3

4 Comparison: Analysis time of rpact vs. gsDesign **5**

5 Remark **5**

Summary

This R Markdown document provides an example that illustrates how to compare sample size and power calculation results of the two different R packages rpact and gsDesign.

1 The design

- 1:1 randomized
- Two-sided log-rank test; 80% power at the 5% significance level (or one-sided at 2.5%)
- Target HR for primary endpoint (PFS) is 0.75
- PFS in the control arm follows a piece-wise exponential distribution, with the hazard rate h(t) estimated using historical controls as follows:
  - h(t) = 0.025 for t between 0 and 6 months;
  - h(t) = 0.04 for t between 6 and 9 months;
  - h(t) = 0.015 for t between 9 and 15 months;
  - h(t) = 0.01 for t between 15 and 21 months;
  - h(t) = 0.007 for t beyond 21 months.
- An annual dropout probability of 20%
- Interim analyses at 33% and 70% of total information
- Alpha-spending version of O'Brien-Fleming boundary for efficacy
- No futility interim
- 1405 subjects recruited in total
- Staggered recruitment:
  - 15 pt/month during first 12 months;

- subsequently, increase of # of sites and ramp up of recruitment by +6 pt/month each month until a maximum of 45 pt/month

2 Calculation with gsDesign

```
# Load the package `gsDesign`
library(gsDesign)
options(warn = -1) # avoid warnings generated by gsDesign
x <- gsSurv(
    k = 3, test.type = 1, alpha = 0.025, beta = 0.2,
    timing = c(0.33, 0.7), sfu = sfLDOF, # boundary
    hr = 0.75,
    lambdaC = c(0.025, 0.04, 0.015, 0.01, 0.007), # piecewise lambdas
    S = c(6, 3, 6, 6), # piecewise survival times
    eta = -log(1 - 0.2) / 12, # dropout
    gamma = c(15, 21, 27, 33, 39, 45), # recruitment, pt no
    R = c(12, 1, 1, 1, 1, (1405 - 300) / 45), # recruitment duration
    minfup = NULL
)
print(x, digits = 5)
```

```
## Time to event group sequential design with HR= 0.75
## Equal randomization:          ratio=1
## One-sided group sequential design with
## 80 % power and 2.5 % Type I Error.
##
##   Analysis  N   Z   Nominal p  Spend
##          1 128 3.73    0.0001 0.0001
##          2 271 2.44    0.0074 0.0073
##          3 386 2.00    0.0227 0.0176
##      Total                   0.0250
##
## ++ alpha spending:
##  Lan-DeMets O'Brien-Fleming approximation spending function with none = 1.
##
## Boundary crossing probabilities and expected sample size
## assume any cross stops the trial
##
## Upper boundary (power or Type I Error)
##           Analysis
##    Theta      1      2      3 Total  E{N}
##   0.0000 0.0001 0.0073 0.0176 0.025 385.0
##   0.1437 0.0175 0.4517 0.3309 0.800 329.1
##              T         n   Events HR efficacy
## IA 1  26.78703  785.4162 127.3407       0.516
## IA 2  38.62360 1318.0620 270.1171       0.743
## Final 50.80093 1405.0000 385.8810       0.816
## Accrual rates:
##          Stratum 1
## 0-12            15
## 12-13           21
## 13-14           27
## 14-15           33
```

```
## 15-16           39
## 16-40.56        45
## Control event rates (H1):
##        Stratum 1
## 0-6        0.025
## 6-9        0.040
## 9-15       0.015
## 15-21      0.010
## 21-Inf     0.007
## Censoring rates:
##        Stratum 1
## 0-6       0.0186
## 6-9       0.0186
## 9-15      0.0186
## 15-21     0.0186
## 21-Inf    0.0186
```

3 Calculation with rpact

3.1 Design

```
# Load the package `rpact`
library(rpact)
packageVersion("rpact")
```

```
design <- getDesignGroupSequential(
    sided = 1, alpha = 0.025, beta = 0.2,
    informationRates = c(0.33, 0.7, 1),
    typeOfDesign = "asOF"
)
kable(summary(design))
```

Sequential analysis with a maximum of 3 looks (group sequential design)

O'Brien & Fleming type alpha spending design, one-sided overall significance level 2.5%, power 80%, undefined endpoint, inflation factor 1.015.

Stage	1	2	3
Information rate	33%	70%	100%
Efficacy boundary (z-value scale)	3.731	2.440	2.000
Stage Levels	<0.0001	0.0074	0.0227
Cumulative alpha spent	<0.0001	0.0074	0.0250
Overall power	0.0175	0.4691	0.8000

3.2 Sample size / timing of interim analyses

```
piecewiseSurvivalTime <- list(
    "0 - <6" = 0.025,
    "6 - <9" = 0.04,
    "9 - <15" = 0.015,
    "15 - <21" = 0.01,
    ">= 21" = 0.007
)
```

```
accrualTime <- list(
    "0  - <12" = 15,
    "12 - <13" = 21,
    "13 - <14" = 27,
    "14 - <15" = 33,
    "15 - <16" = 39,
    ">= 16" = 45
)

y <- getPowerSurvival(
    design = design, typeOfComputation = "Schoenfeld",
    thetaH0 = 1, directionUpper = FALSE,
    dropoutRate1 = 0.2, dropoutRate2 = 0.2, dropoutTime = 12,
    allocationRatioPlanned = 1,
    accrualTime = accrualTime,
    piecewiseSurvivalTime = piecewiseSurvivalTime,
    hazardRatio = 0.75,
    maxNumberOfEvents = x$n.I[3],
    maxNumberOfSubjects = 1405
)
kable(summary(y))
```

Power calculation for a survival endpoint

Sequential analysis with a maximum of 3 looks (group sequential design), overall significance level 2.5% (one-sided). The results were calculated for a two-sample logrank test, H0: hazard ratio = 1, power directed towards smaller values, H1: hazard ratio = 0.75, piecewise survival distribution, piecewise survival time = c(0, 6, 9, 15, 21), control lambda(2) = c(0.025, 0.04, 0.015, 0.01, 0.007), maximum number of subjects = 1405, maximum number of events = 386, accrual time = c(12, 13, 14, 15, 16, 40.556), accrual intensity = c(15, 21, 27, 33, 39, 45), dropout rate(1) = 0.2, dropout rate(2) = 0.2, dropout time = 12.

Stage	1	2	3
Information rate	33%	70%	100%
Efficacy boundary (z-value scale)	3.731	2.440	2.000
Overall power	0.0175	0.4702	0.8009
Expected number of subjects	1354.8		
Number of subjects	785.4	1318.1	1405.0
Expected number of events	328.9		
Cumulative number of events	127.3	270.1	385.9
Expected study duration	44.9		
Cumulative alpha spent	<0.0001	0.0074	0.0250
One-sided local significance level	<0.0001	0.0074	0.0227
Efficacy boundary (t)	0.516	0.743	0.816
Exit probability for efficacy (under H0)	<0.0001	0.0073	
Exit probability for efficacy (under H1)	0.0175	0.4526	

Legend:

- *(t)*: treatment effect scale

4 Comparison: Analysis time of rpact vs. gsDesign

Absolute differences:

```
timeDiff <- as.data.frame(sprintf("%.5f", (x$T - y$analysisTime)))
rownames(timeDiff) <- c("Stage 1", "Stage 2", "Stage 3")
colnames(timeDiff) <- "Difference analysis time"
kable(timeDiff)
```

	Difference analysis time
Stage 1	-0.00000
Stage 2	0.00004
Stage 3	-0.00011

5 Remark

Obviously, there is a difference in the calculation of the necessary number of events which are, in rpact, calculated as

```
(qnorm(0.975) + qnorm(0.8))^2 / log(0.75)^2 * 4 *
    getDesignCharacteristics(getDesignGroupSequential(
        sided = 1, alpha = 0.025,
        kMax = 3, typeOfDesign = "asOF", informationRates = c(0.33, 0.7, 1)
    ))$inflationFactor
```

```
## [1] 385.0479
```

which is slightly different to the maximum number of events in gsDesign which is

```
x$n.I[3]
```

```
## [1] 385.881
```

Therefore, running

```
getSampleSizeSurvival(
    design = design, typeOfComputation = "Schoenfeld",
    thetaH0 = 1,
    dropoutRate1 = 0.2, dropoutRate2 = 0.2, dropoutTime = 12,
    allocationRatioPlanned = 1,
    accrualTime = accrualTime,
    piecewiseSurvivalTime = piecewiseSurvivalTime,
    hazardRatio = 0.75,
    maxNumberOfSubjects = 1405
)$analysisTime
```

```
##          [,1]
## [1,] 26.76183
## [2,] 38.57834
## [3,] 50.63114
```

is not exactly equal to getPowerSurvival from above. This, however, has definitely no consequences in practice but explains the slight differences in rpact and gsDesign.

System: rpact 3.3.2, R version 4.2.1 (2022-06-23 ucrt), platform: x86_64-w64-mingw32

To cite R in publications use:

R Core Team (2022). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Um Paket 'rpact' in Publikationen zu zitieren, nutzen Sie bitte:

Wassmer G, Pahlke F (2022). *rpact: Confirmatory Adaptive Clinical Trial Design and Analysis*. https://www.rpact.org, https://www.rpact.com, https://github.com/rpact-com/rpact.

 file