

Designing group-sequential trials with a binary endpoint with rpact

Marcel Wolbers, Gernot Wassmer, and Friedrich Pahlke

Last change: 20 Juni, 2019

This R markdown file provides examples for designing trials with binary endpoints using rpact.

Sample size calculation for a superiority trial with two groups without interim analyses

Sample size for a trial with binary endpoints can be calculated using the function `getSampleSizeRates`. This function is fully documented in the help page (`?getSampleSizeRates`). Hence, we only provide some examples below.

First, load the library `rpact`

```
# Load rpact
library(rpact)
packageVersion("rpact") # version should be version 2.0.1 or later
```

```
## [1] '2.0.1'
```

To get the **direction** of the effects correctly, note that in `rpact` the **index “2” in an argument name always refers to the control group, “1” to the intervention group, and treatment effects compare treatment versus control**. Specifically, for binary endpoints, the probabilities of an event in the control group and intervention group, respectively, are given by arguments `pi2` and `pi1`. The default treatment effect is the absolute risk difference `pi1-pi2` but the relative risk scale `pi1/pi2` is also supported if the argument `riskRatio` is set to `TRUE`.

```
# Example of a standard trial:
# - probability 25% in control (pi2=0.25) vs 40% (pi1=0.4) in intervention
# - one-sided test (sided=1)
# - type I error 0.025 (alpha=0.025) and power 80% (beta=0.2)
sampleSizeResult <- getSampleSizeRates(pi2 = 0.25, pi1 = 0.4,
  sided = 1,alpha = 0.025, beta = 0.2)
sampleSizeResult
```

```
## Design plan parameters and output for rates:
##
## Design parameters:
##   Significance level           : 0.0250
##   Type II error rate          : 0.2
##   Test                         : one-sided
##
## User defined parameters:
##   pi (1)                       : 0.400
##   pi (2)                       : 0.250
##
## Default parameters:
##   Normal approximation         : TRUE
##   Risk ratio                   : FALSE
```

```
## Theta H0 : 0
## Treatment groups : 2
## Planned allocation ratio : 1
##
## Sample size and output:
## Direction upper : TRUE
## Number of subjects fixed : 303.7
## Number of subjects fixed (1) : 151.9
## Number of subjects fixed (2) : 151.9
## Critical values (effect scale) : 0.103
## Local one-sided significance levels : 0.0250
##
## Legend:
## (i): values of treatment arm i
```

As per the output above, the required **total sample size** is 304 and the critical value corresponds to a minimal detectable difference (on the absolute risk difference scale, the default) of approximately 0.103. This calculation assumes that $\pi_2 = 0.25$ is the observed rate in treatment group 2.

You can change the randomization allocation between the treatment groups using `allocationRatioPlanned`:

```
# Example: Extension of standard trial
# - 2(intervention):1(control) randomization (allocationRatioPlanned = 2)
getSampleSizeRates(pi2 = 0.25, pi1 = 0.4,
  sided = 1, alpha = 0.025, beta = 0.2,
  allocationRatioPlanned = 2)
```

```
## Design plan parameters and output for rates:
##
## Design parameters:
## Significance level : 0.0250
## Type II error rate : 0.2
## Test : one-sided
##
## User defined parameters:
## pi (1) : 0.400
## pi (2) : 0.250
## Planned allocation ratio : 2
##
## Default parameters:
## Normal approximation : TRUE
## Risk ratio : FALSE
## Theta H0 : 0
## Treatment groups : 2
##
## Sample size and output:
## Direction upper : TRUE
## Number of subjects fixed : 346.3
## Number of subjects fixed (1) : 230.9
## Number of subjects fixed (2) : 115.4
## Critical values (effect scale) : 0.104
## Local one-sided significance levels : 0.0250
##
## Legend:
## (i): values of treatment arm i
```

allocationRatioPlanned = 0 can be defined in order to obtain the optimum allocation ratio minimizing the overall sample size (the optimum ample size is only slightly smaller than sample size with equal allocation; practically, this has no effect):

```
# Example: Extension of standard trial
# optimum randomization ratio
getSampleSizeRates(pi2 = 0.25, pi1 = 0.4,
  sided = 1,alpha = 0.025,beta = 0.2,
  allocationRatioPlanned = 0)
```

```
## Design plan parameters and output for rates:
##
## Design parameters:
##   Significance level           : 0.0250
##   Type II error rate         : 0.2
##   Test                        : one-sided
##
## User defined parameters:
##   pi (1)                     : 0.400
##   pi (2)                     : 0.250
##
## Default parameters:
##   Normal approximation       : TRUE
##   Risk ratio                 : FALSE
##   Theta H0                  : 0
##   Treatment groups          : 2
##
## Sample size and output:
##   Planned allocation ratio    : 0.953
##   Direction upper            : TRUE
##   Number of subjects fixed   : 303.6
##   Number of subjects fixed (1) : 148.1
##   Number of subjects fixed (2) : 155.5
##   Critical values (effect scale) : 0.103
##   Local one-sided significance levels : 0.0250
##
## Legend:
##   (i): values of treatment arm i
```

Power can be calculated using the function `getPowerRates`. This function has the same arguments as `getSampleSizeRates` except that the maximum total sample size needs to be defined (`maxNumberOfSubjects`) and the type II error `beta` is no longer needed. For one-sided tests, the direction of the test is also required (`directionUpper = TRUE` indicates that the alternative is that the probability in the intervention group `pi1` is larger than the probability in the control group `pi2` (`directionUpper = FALSE` is the other direction)):

```
# Example: Calculate power for a simple trial with total sample size 304
# as in the example above in case of pi2 = 0.25 (control) and
# pi1 = 0.37 (intervention)
powerResult <- getPowerRates(pi2 = 0.25, pi1 = 0.37,
  maxNumberOfSubjects = 304, sided = 1,alpha = 0.025)
powerResult
```

```
## Design plan parameters and output for rates:
##
## Design parameters:
##   Significance level           : 0.0250
```

```
## Test : one-sided
##
## User defined parameters:
## pi (1) : 0.370
## pi (2) : 0.250
## Maximum number of subjects : 304.0
##
## Default parameters:
## Normal approximation : TRUE
## Risk ratio : FALSE
## Theta H0 : 0
## Treatment groups : 2
## Planned allocation ratio : 1
## Direction upper : TRUE
##
## Sample size and output:
## Number of subjects fixed : 304.0
## Number of subjects fixed (1) : 152.0
## Number of subjects fixed (2) : 152.0
## Effect : 0.12
## Overall reject : 0.62
## Critical values (effect scale) : 0.103
## Local one-sided significance levels : 0.0250
##
## Legend:
## (i): values of treatment arm i
```

The calculated **power** is provided in the output as “**Overall reject**” and is 0.620 for the example.

The `getPowerRates` (as well as `getSampleSizeRates`) functions can also be called with a vector argument for the probability `pi1` in the intervention group. This is illustrated below via a plot of power depending on this probability. For examples of all available plots, see the R markdown file `How to create admirable plots with rpact`.

```
# Example: Calculate power for simple design (with sample size 304 as above)
# for probabilities in intervention ranging from 0.3 to 0.5
powerResult <- getPowerRates(pi2 = 0.25, pi1 = seq(0.3, 0.5, by = 0.01),
  maximumNumberOfSubjects = 304, sided = 1, alpha = 0.025)

# one of several possible plots, this one plotting true effect size vs power
plot(powerResult, type = 7)
```

Sample size calculation for a non-inferiority trial with two groups without interim analyses

Sample size calculation proceeds in the same fashion as for superiority trials except that the role of the null and the alternative hypothesis are reversed. I.e., in this case, the non-inferiority margin Δ corresponds to the treatment effect under the null hypothesis (`thetaH0`) which one aims to reject. Testing in non-inferiority trials is always one-sided.

```
# Example: Sample size for a non-inferiority trial
# Assume pi2(control) = pi1(intervention) = 0.2
# Test H0: pi1-pi2 = 0.1 (risk increase in intervention >= Delta = 0.1)
# vs. HA: pi1-pi2 < 0.1
sampleSizeNoninf <- getSampleSizeRates(pi2 = 0.2, pi1 = 0.2,
```

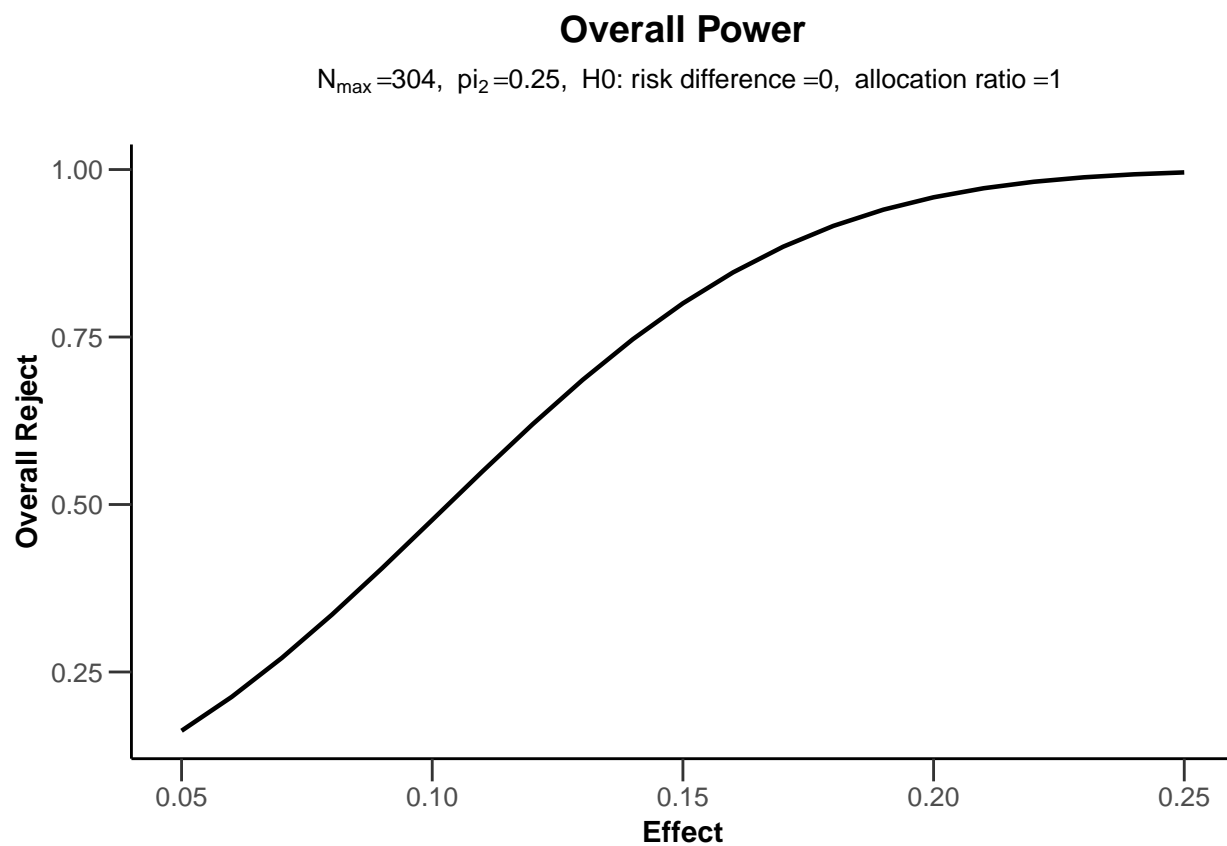


Figure 1: Figure: example for an overall power plot

```

thetaH0 = 0.1, sided = 1, alpha = 0.025, beta = 0.2)
sampleSizeNoninf

## Design plan parameters and output for rates:
##
## Design parameters:
##   Significance level           : 0.0250
##   Type II error rate         : 0.2
##   Test                        : one-sided
##
## User defined parameters:
##   Theta H0                   : 0.1
##   pi (1)                     : 0.200
##
## Default parameters:
##   Normal approximation       : TRUE
##   Risk ratio                 : FALSE
##   pi (2)                     : 0.200
##   Treatment groups          : 2
##   Planned allocation ratio   : 1
##
## Sample size and output:
##   Direction upper            : FALSE
##   Number of subjects fixed   : 508.4
##   Number of subjects fixed (1) : 254.2
##   Number of subjects fixed (2) : 254.2
##   Critical values (effect scale) : 0.0285
##   Local one-sided significance levels : 0.0250
##
## Legend:
##   (i): values of treatment arm i

```

Sample size calculation for a single arm trial without interim analyses

The function `getSampleSizeRates` allows to set the number of groups (which is =2 by defaults) to 1 for the design of single arm trials. The probability under the null hypothesis can be specified with argument `thetaH0` and the specific alternative hypothesis which is used for the sample size calculation with the argument `pi2`. The sample size calculation can be based either on a normal approximation (`normalApproximation = TRUE`, the default) or on exact binomial probabilities (`normalApproximation = FALSE`).

```

# Example: Sample size for a single arm trial which tests
# H0: pi2=0.1 vs. HA: pi2=0.25
# (use conservative exact binomial calculation)
getSampleSizeRates(groups=1, thetaH0 = 0.1, pi1 = 0.25,
  normalApproximation = FALSE, sided = 1, alpha = 0.025, beta = 0.2)

```

```

## Design plan parameters and output for rates:
##
## Design parameters:
##   Significance level           : 0.0250
##   Type II error rate         : 0.2
##   Test                        : one-sided

```

```
##
## User defined parameters:
##   Normal approximation           : FALSE
##   Theta H0                       : 0.1
##   pi (1)                         : 0.250
##   Treatment groups               : 1
##
## Default parameters: not available
##
## Sample size and output:
##   Direction upper                : TRUE
##   Number of subjects fixed       : 53.0
##   Critical values (effect scale)  : 0.181
##   Local one-sided significance levels : 0.0250
```

Sample size calculation for group-sequential designs

Sample size calculation for a group-sequential trial is performed in **two steps**:

1. **Define the (abstract) group-sequential design** using the function `getDesignGroupSequential`. For details regarding this step, see the R markdown file “Defining group-sequential boundaries with `rpact`”.
2. **Calculate sample size** for the binary endpoint by feeding the abstract design into function `getSampleSizeRates`. Note that the power $1 - \beta$ needs to be defined in the design function, and not in `getSampleSizeRates`.

In general, `rpact` supports both one-sided and two-sided group-sequential designs. However, if futility boundaries are specified, only one-sided tests are permitted.

R code for a simple example is provided below:

```
# Example: Group-sequential design with O'Brien&Fleming type alpha-spending and
# one interim at 60% information
design <- getDesignGroupSequential(sided = 1, alpha = 0.025, beta = 0.2,
  informationRates = c(0.6, 1), typeOfDesign = "asOF")

# Sample size calculation assuming event probabilities are 25% in control
# (pi2=0.25) vs 40% (pi1=0.4) in intervention
sampleSizeResultGs <- getSampleSizeRates(design, pi2 = 0.25, pi1 = 0.4)
# Standard rpact output (sample size object only, not design object)
sampleSizeResultGs
```

```
## Design plan parameters and output for rates:
##
## Design parameters:
##   Significance level           : 0.0250
##   Type II error rate          : 0.2
##   Test                        : one-sided
##
## User defined parameters:
##   pi (1)                      : 0.400
##   pi (2)                      : 0.250
##
## Default parameters:
##   Normal approximation        : TRUE
##   Risk ratio                  : FALSE
```

```
## Theta H0 : 0
## Treatment groups : 2
## Planned allocation ratio : 1
##
## Sample size and output:
## Direction upper : TRUE
## Information rates [1] : 0.600
## Information rates [2] : 1.000
## Maximum number of subjects : 306.3
## Maximum number of subjects (1) : 153.2
## Maximum number of subjects (2) : 153.2
## Number of subjects [1] : 183.8
## Number of subjects [2] : 306.3
## Expected number of subjects under H0 : 305.9
## Expected number of subjects under H0/H1 : 299.3
## Expected number of subjects under H1 : 268.1
## Reject per stage [1] : 0.312
## Reject per stage [2] : 0.488
## Early stop : 0.312
## Critical values (effect scale) [1] : 0.187
## Critical values (effect scale) [2] : 0.104
## Local one-sided significance levels [1] : 0.003808
## Local one-sided significance levels [2] : 0.023798
##
## Legend:
## (i): values of treatment arm i
## [k]: values at stage k
```

System: rpact 2.0.1, R version 3.5.2 (2018-12-20), platform: x86_64-w64-mingw32

To cite package 'rpact' in publications use:

Gernot Wassmer and Friedrich Pahlke (2019). rpact: Confirmatory Adaptive Clinical Trial Design and Analysis. R package version 2.0.1. <https://CRAN.R-project.org/package=rpact>

License

This work is licensed under the Creative Commons Attribution-ShareAlike 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/4.0/>.