

Package

September 16, 2019

Title Calculate Power and Sample Size with Beta Regression

Version 1.1-1

Date 2019-09-13

Description Power calculations are a critical component of any research study to determine the minimum sample size necessary to detect differences between multiple groups.

Researchers often work with data taking the form of proportions that can be modeled with a beta distribution. Here we present an R package, BetaPASS, and analogous SAS macro, that perform power and sample size calculations for data following a beta distribution with comparative nonparametric output. This package allows flexibility with multiple options for link functions to fit the data and graphing functionality for visual comparisons.

Depends R (≥ 3.5)

License GPL (≥ 2)

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Imports betareg, ggplot2, pbapply, stats

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

betapower	1
samplesize	3

Index	5
--------------	----------

betapower	<i>Find Power with Beta distribution</i>
-----------	--

Description

Find the power for a given sample size when testing the null hypothesis that the means for the control and treatment groups are equal against a two-sided alternative.

Usage

```
betapower(mu0, sd0, mu1.start, mu1.end = NULL, mu1.by = NULL,
ss.start, ss.end = NULL, ss.by = NULL, sig.level = 0.05,
trials = 100, seed = 1, link.type="logit",
equal.precision=TRUE, sd1 = NULL)
```

Arguments

<code>mu0</code>	mean for the control group
<code>sd0</code>	standard deviation for the control group
<code>mu1.start</code>	starting value of mean for the treatment group under the alternative <code>mu1</code>
<code>mu1.end</code>	ending value of mean for the treatment group under the alternative <code>mu1</code>
<code>mu1.by</code>	step length of mean for the treatment group under the alternative <code>mu1</code>
<code>ss.start</code>	starting value of sample size
<code>ss.end</code>	ending value of sample size
<code>ss.by</code>	step length of sample size
<code>sig.level</code>	significant level of test; default value is 0.05
<code>trials</code>	number of trials
<code>seed</code>	seed used in the simulation
<code>link.type</code>	type of link used in the beta regression. Default value is "logit", or you can use "all" or choose one or more of the following: "logit", "probit", "cloglog", "cauchit", "log", "loglog"
<code>equal.precision</code>	equal dispersion parameter assumption in simulation
<code>sd1</code>	standard deviation for the treatment group. Only applicable when <code>equal.precision = FALSE</code>

Details

`betapower` function allows you to control the number of trials in the simulation, the sample sizes used, and the alternative means. You can fix the alternative and vary sample size to match a desired power; You can fix the sample size and vary the alternative to see which will match a desired power; You can vary both; Start with a small number of trials (say 100) to determine the rough range of sample sizes or alternatives; Use a larger number of trials (say 1000) to get better estimates.

The plot function will return different plots depends on "by" statement. Type of link used in the beta regression. You can choose one or more of the following: "logit", "probit", "cloglog", "cauchit", "log", "loglog", "all"

`by = "linktype"`: return graphs that plot power against `mu1`, where `mu1` is the mean for the treatment group under the alternative. The number of plots will vary depending on the number of link types selected with the last plot showing power based on Wilcoxon Rank Sum Test. The first one or several plots show comparisons of power with different sample size, using GLM method with one or several link types. The last plot shows a comparison of the power with different sample size using Wilcoxon Rank Sum Test. Y-axis denotes power and X-axis denotes `mu1`, the mean for the treatment group under the alternative.

`by = "samplesize"`: return a number of plots equal to the number of sample sizes tested. Each plot compares power calculated with different link types and the Wilcoxon Rank Sum Test. Y-axis denotes power and X-axis denotes `mu1`, the mean for the treatment group under the alternative.

`by = "mu1"`: return a number of plots equal to the number of `mu1` used in the procedure. Each

plot compares power calculated with different link types and the Wilcoxon Rank Sum Test. Y-axis denotes power and X-axis denotes sample size.

Value

Return a betapower object including basic settings (mean and standard deviation for the control group, significant level, number of trials and link types), and a matrix of estimated power with given sample size and mu1.

power.of.GLM:	link name
	estimated power using beta regression method; it will return the power with every links if you use link.type = "all" statement.
power.of.Wilcoxon.test	
	estimated power from Wilcoxon Rank sum test.
sample size	sample size.
mu1	mean for the treatment group under the alternative.

Examples

```
betapower(mu0 = 0.56, sd0 = 0.255, mu1.start = .70, mu1.end = .75, mu1.by = .05,
ss.start = 30, ss.end = 50, ss.by = 20, trials = 100)
```

samplesize	<i>Find minimum sample size with Beta distribution</i>
------------	--

Description

Find minimum sample sizes with Beta distribution and given mu0,sd0,mu1 and target powers.

Usage

```
samplesize(mu0, sd0, mu1.start, mu1.end = NULL, mu1.by = NULL,
power.start, power.end = NULL, power.by = NULL, sig.level = 0.05,
trials = 100, delta = 1, seed = 1, link.type = "logit",
equal.precision = TRUE, sd1 = NULL)
```

Arguments

mu0	mean for the control group
sd0	standard deviation for the control group
mu1.start	starting value of mean for the treatment group under the alternative mu1
mu1.end	ending value of mean for the treatment group under the alternative mu1
mu1.by	step length of mean for the treatment group under the alternative mu1
power.start	starting value of target power
power.end	ending value of target power
power.by	step length of target power
sig.level	significant level; default value is 0.05
trials	number of trials; default value is 100

<code>delta</code>	accuracy of the result; must be integer
<code>seed</code>	seed used in the simulation
<code>link.type</code>	type of link used in the beta regression. Default link is "logit". Other link options include: "logit", "probit", "cloglog", "log", "loglog", "wilcoxon", or you can use "all" for all types of link
<code>equal.precision</code>	equal dispersion parameter assumption in simulation
<code>sd1</code>	standard deviation for the treatment group. Only applicable when <code>equal.precision</code> = FALSE

Details

The `samplesize` function allows you to control the number of trials in the simulation, the target power, delta, and the alternative means. You can fix the alternative and vary power to match a desired sample size; Use default values for the number of trials for a quick view; Use a larger number of trials (say 1000) and a smaller delta (say 1) to get better estimates.

The `plot` function will return a series of plots equal to the number of `mu1` used in the procedure. Type of link used in the beta regression. You can choose one or more of the following: "logit", "probit", "cloglog", "cauchit", "log", "loglog", "all". Y-axis denotes minimum sample size and X-axis denotes minimum power.

Value

Return a `samplesize` object including basic settings (mean and standard deviation for the control group, significant level, number of trials and link types), and a matrix of estimated power with given `mu1` and target power.

```

minimum sample size: link type:
                        minimum sample size for given given mu0, sd0, mu1, target power and type of
                        link.
minimum power: link type:
                        the minimum power greater than or equal to target power.

target power:         target power.

mu1:                  mean for the treatment group under the alternative.

```

Examples

```

samplesize(mu0=0.56, sd0=0.255, mu1.start = 0.8, power.start = 0.9, trials = 50,
link.type = c("logit","wilcoxon"))

```

```

SSmat <- samplesize(mu0=0.56, sd0=0.255, mu1.start = 0.75,
power.start = 0.8, power.end = 0.9, power.by = 0.1,
trials = 50, link.type = c("log","wilcoxon"))
## plot with parametric and nonparametric methods
plot(x = SSmat, link.type = c("log","wilcoxon"))

```

Index

betapower, [1](#)

samplesize, [3](#)