

# Package ‘retrodesign’

October 14, 2022

**Type** Package

**Title** Tools for Type S (Sign) and Type M (Magnitude) Errors

**Version** 0.1.0

**Description** Provides tools for working with Type S (Sign) and Type M (Magnitude) errors, as proposed in Gelman and Tuerlinckx (2000) <[doi.org/10.1007/s001800000040](https://doi.org/10.1007/s001800000040)> and Gelman & Carlin (2014) <[doi.org/10.1177/1745691614551642](https://doi.org/10.1177/1745691614551642)>. In addition to simply calculating the probability of Type S/M error, the package includes functions for calculating these errors across a variety of effect sizes for comparison, and recommended sample size given “tolerances” for Type S/M errors. To improve the speed of these calculations, closed forms solutions for the probability of a Type S/M error from Lu, Qiu, and Deng (2018) <[doi.org/10.1111/bmsp.12132](https://doi.org/10.1111/bmsp.12132)> are implemented. As of 1.0.0, this includes support only for simple research designs. See the package vignette for a fuller exposition on how Type S/M errors arise in research, and how to analyze them using the type of design analysis proposed in the above papers.

**Depends** R (>= 3.1.0)

**License** MIT + file LICENSE

**URL** <https://github.com/andytimmm/retrodesign>

**BugReports** <https://github.com/andytimmm/retrodesign/issues>

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 6.1.1

**Imports** graphics

**Suggests** ggplot2, knitr, rmarkdown, gridExtra, testthat

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Andrew Timm [cre, aut],  
Andrew Gelman [ctb, cph],  
John Carlin [ctb, cph]

**Maintainer** Andrew Timm <timmandrew1@gmail.com>

**Repository** CRAN

**Date/Publication** 2019-03-08 14:00:02 UTC

## R topics documented:

retrodesign . . . . .	2
retrodesign.list . . . . .	3
retrodesign.numeric . . . . .	4
retro_design . . . . .	4
retro_design.list . . . . .	5
retro_design.numeric . . . . .	6
sim_plot . . . . .	6
type_m . . . . .	7
type_m.list . . . . .	8
type_m.numeric . . . . .	8
type_s . . . . .	9
type_s.list . . . . .	10
type_s.numeric . . . . .	10

**Index** **12**

---

retrodesign	<i>retrodesign: Calculates Power, Type S, and Type M error</i>
-------------	--

---

### Description

Calculates Power, Type S, and Type M error and returns them in a list or df, depending on whether a single true effect size or range is provided. `retro_design()` is faster as it uses the closed form solution from Lu et al. (2018), but this function can be used for t distributions, whereas `retro_design()` cannot. Function originally provided in Gelman and Carlin (2014), reused with permission.

### Usage

```
retrodesign(A, s, alpha = 0.05, df = Inf, n.sims = 10000)
```

### Arguments

A	a numeric or list, an estimate of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold
df	a numeric, the degrees of freedom. <code>df=Inf</code> is equivalent to a normal distribution.
n.sims	a numeric, how many times to simulate when calculating Type M error.

**Value**

either a list of length 3 containing the power, type s, and type M error, or if A is a list, a df that is 4 by length(A), with an effect size and it's corresponding power, type s, and type m errors in each row.

**Examples**

```
retrodesign(1,3.28)
retrodesign(list(.2,2,20),8.1)
retrodesign(.5,1,df=10)
```

---

retrodesign.list	<i>List retrodesign</i>
------------------	-------------------------

---

**Description**

retrodesign.list is the S3 method of the generic retrodesign() function, used when a list is passed for A.

**Usage**

```
## S3 method for class 'list'
retrodesign(A, s, alpha = 0.05, df = Inf,
  n.sims = 10000)
```

**Arguments**

A	a list, estimates of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold
df	a numeric, the degrees of freedom. df=Inf is equivalent to a normal distribution.
n.sims	a numeric, how many times to simulate when calculating Type M error

**Value**

A df that is 4 by length(A), with an effect size and it's corresponding power, type s, and type m errors in each row.

**Examples**

```
retrodesign(list(.2,2,20), 8.1)
retrodesign(list(.2,2,20), 8.1,df = 10)
```

---

retrodesign.numeric    *Numeric retrodesign*

---

### Description

retrodesign.numeric is the S3 method of the generic retrodesign() function, used when a single numeric is passed for A.

### Usage

```
## S3 method for class 'numeric'
retrodesign(A, s, alpha = 0.05, df = Inf,
  n.sims = 10000)
```

### Arguments

A	a numeric, an estimate of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold
df	a numeric, the degrees of freedom. df=Inf is equivalent to a normal distribution.
n.sims	a numeric, how many times to simulate when calculating Type M error

### Value

A list of length 3 containing the power, type s, and type M error.

### Examples

```
retrodesign(1,3.28)
retrodesign(2,8.1)
retrodesign(.5,1,df=10)
```

---

retro\_design    *retro\_design: Calculates Power, Type S, and Type M error*

---

### Description

Calculates Power, Type S, and Type M error and returns them in a list or df, depending on whether a single true effect size or range is provided. Uses the closed form solution found for the Type-M error found by Lu et al. (2018), and thus is faster than retrodesign. For t distributions, use retrodesign() instead; the closed form solution only applies in the normal case.

### Usage

```
retro_design(A, s, alpha = 0.05)
```

**Arguments**

A	a numeric or list, an estimate of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold

**Value**

either a list of length 3 containing the power, type s, and type M error, or if A is a list, a df that is 4 by length(A), with an effect size and its corresponding power, type s, and type m errors in each row.

**Examples**

```
retrodesign(1,3.28)
retrodesign(list(.2,2,20),8.1)
```

---

retro\_design.list      *List retro\_design*

---

**Description**

retro\_design.list is the S3 method of the generic retro\_design() function, used when a list is passed for A.

**Usage**

```
## S3 method for class 'list'
retro_design(A, s, alpha = 0.05)
```

**Arguments**

A	a list, estimates of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold

**Value**

A df that is 4 by length(A), with an effect size and its corresponding power, type s, and type m errors in each row.

**Examples**

```
retro_design(list(.2,2,20),8.1)
```

---

retro\_design.numeric    *Numeric retro\_design*

---

### Description

retro\_design.numeric is the S3 method of the generic retro\_design() function, used when a single numeric is passed for A.

### Usage

```
## S3 method for class 'numeric'
retro_design(A, s, alpha = 0.05)
```

### Arguments

A	a numeric, an estimate of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold

### Value

A list of length 3 containing the power, type s, and type M error.

### Examples

```
retrodesign(1,3.28)
retrodesign(2,8.1)
```

---

sim\_plot                    *sim\_plot: visualize type S/M errors*

---

### Description

Graphs type S/M errors resulting from a simulation using the provided parameters (using the same simulation method as retrodesign()). Can optionally display using ggplot.

### Usage

```
sim_plot(A, s, alpha = 0.05, df = Inf, n.sims = 5000, gg = TRUE)
```

**Arguments**

A	a numeric, an estimate of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold
df	a numeric, the degrees of freedom
n.sims	a numeric, how many times to simulate when calculating Type M error
gg	If TRUE and ggplot2 is installed, uses ggplot2 for graphic

**Value**

A list of length 3 containing the power, type s, and type M error.

**Examples**

```
sim_plot(1,3.28)
sim_plot(.5,1)
```

---

type_m	<i>type_m</i>
--------	---------------

---

**Description**

Calculates type m error. Is calculated using simulation, and thus supports t distributions through the df parameter.

**Usage**

```
type_m(A, s, alpha = 0.05, df = Inf, n.sims = 10000)
```

**Arguments**

A	a numeric or list, estimate(s) of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold
df	a numeric, the number of degrees of freedom. df=Inf is equivalent to a normal distribution.
n.sims	a numeric, how many times to simulate when calculating Type M error

**Value**

either the type m error, a numeric if a single A is provided, or a df of length 2 by A, with the effect size and corresponding type m error in each row.

**Examples**

```
type_m(1,3.28)
type_m(list(.2,2,20),8.1)
```

---

type_m.list	<i>List type_m</i>
-------------	--------------------

---

**Description**

type\_m.list is the S3 method of the generic type\_m() function, used when a list is passed for A.

**Usage**

```
## S3 method for class 'list'
type_m(A, s, alpha = 0.05, df = Inf, n.sims = 10000)
```

**Arguments**

A	a list, estimates of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold
df	a numeric, the number of degrees of freedom. df=Inf is equivalent to a normal distribution.
n.sims	a numeric, how many times to simulate when calculating Type M error

**Value**

A df that is 2 by length(A), with an effect size and it's corresponding type m errors in each row.

**Examples**

```
type_s(list(.2,2,20),8.1)
```

---

type_m.numeric	<i>Numeric type_m</i>
----------------	-----------------------

---

**Description**

this is the S3 method of the generic type\_m() function, used when a numeric is passed for A.

**Usage**

```
## S3 method for class 'numeric'
type_m(A, s, alpha = 0.05, df = Inf,
      n.sims = 10000)
```



**Arguments**

A	a numeric, estimate of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold
df	a numeric, the number of degrees of freedom. df=Inf is equivalent to a normal distribution.
n.sims	a numeric, how many times to simulate when calculating Type M error

**Value**

either the type m, a numeric if a single A is provided, or a df of length 2 by A, with the effect size and corresponding type m error in each row.

**Examples**

```
type_m(1, 3.28)
```

---

type_s	<i>type_s</i>
--------	---------------

---

**Description**

Calculates type s error.

**Usage**

```
type_s(A, s, alpha = 0.05)
```

**Arguments**

A	a numeric or list, estimate(s) of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold

**Value**

either the type S, a numeric if a single A is provided, or a df of length 2 by A, with the effect size and corresponding type S error in each row.

**Examples**

```
type_s(1, 3.28)
type_s(list(.2, 2, 20), 8.1)
```

---

type_s.list	<i>List type_s</i>
-------------	--------------------

---

**Description**

type\_s.list is the S3 method of the generic type\_s() function, used when a list is passed for A.

**Usage**

```
## S3 method for class 'list'
type_s(A, s, alpha = 0.05)
```

**Arguments**

A	a list, estimates of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold

**Value**

A df that is 2 by length(A), with an effect size and it's corresponding type s errors in each row.

**Examples**

```
type_s(list(.2,2,20),8.1)
```

---

type_s.numeric	<i>Numeric type_s</i>
----------------	-----------------------

---

**Description**

this is the S3 method of the generic type\_s() function, used when a numeric is passed for A.

**Usage**

```
## S3 method for class 'numeric'
type_s(A, s, alpha = 0.05)
```

**Arguments**

A	a numeric, estimate of the true effect size
s	a numeric, standard error of the estimate
alpha	a numeric, the statistical significance threshold

**Value**

either the type S, a numeric if a single A is provided, or a df of length 2 by A, with the effect size and corresponding type S error in each row.

**Examples**

`type_s(1, 3.28)`

# Index

retro\_design, 4  
retro\_design.list, 5  
retro\_design.numeric, 6  
retrodesign, 2  
retrodesign.list, 3  
retrodesign.numeric, 4  
  
sim\_plot, 6  
  
type\_m, 7  
type\_m.list, 8  
type\_m.numeric, 8  
type\_s, 9  
type\_s.list, 10  
type\_s.numeric, 10