

# Package ‘radiant.design’

September 7, 2023

**Type** Package

**Title** Design Menu for Radiant: Business Analytics using R and Shiny

**Version** 1.6.1

**Date** 2023-9-5

**Description** The Radiant Design menu includes interfaces for design of experiments, sampling, and sample size calculation. The application extends the functionality in 'radiant.data'.

**Depends** R ( $\geq 4.0.0$ ), radiant.data ( $\geq 1.5.0$ ),

**Imports** dplyr ( $\geq 1.0.7$ ), magrittr ( $\geq 1.5$ ), shiny ( $\geq 1.7.1$ ),  
AlgDesign ( $\geq 1.1.7.3$ ), import ( $\geq 1.1.0$ ), pwr ( $\geq 1.1.2$ ),  
randomizr ( $\geq 0.20.0$ ), mvtnorm ( $\geq 1.2.0$ ), polycor

**Suggests** testthat ( $\geq 2.0.0$ ), pkgdown ( $\geq 1.1.0$ )

**URL** <https://github.com/radiant-rstats/radiant.design/>,  
<https://radiant-rstats.github.io/radiant.design/>,  
<https://radiant-rstats.github.io/docs/>

**BugReports** <https://github.com/radiant-rstats/radiant.design/issues/>

**License** AGPL-3 | file LICENSE

**LazyData** true

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**NeedsCompilation** no

**Author** Vincent Nijs [aut, cre]

**Maintainer** Vincent Nijs <radiant@rady.ucsd.edu>

**Repository** CRAN

**Date/Publication** 2023-09-07 02:30:02 UTC

## R topics documented:

doe . . . . .	2
estimable . . . . .	3
plot.sample_size_comp . . . . .	3
radiant.design . . . . .	4
radiant.design_viewer . . . . .	5
radiant.design_window . . . . .	5
randomizer . . . . .	6
rndnames . . . . .	7
sample_size . . . . .	8
sample_size_comp . . . . .	9
sampling . . . . .	10
summary.doe . . . . .	11
summary.randomizer . . . . .	12
summary.sample_size . . . . .	13
summary.sample_size_comp . . . . .	13
summary.sampling . . . . .	14
<b>Index</b>	<b>16</b>

---

doe	<i>Create (partial) factorial design</i>
-----	--

---

### Description

Create (partial) factorial design

### Usage

```
doe(factors, int = "", trials = NA, seed = NA)
```

### Arguments

factors	Categorical variables used as input for design
int	Vector of interaction terms to consider when generating design
trials	Number of trials to create. If NA then all feasible designs will be considered until a design with perfect D-efficiency is found
seed	Random seed to use as the starting point

### Details

See <https://radiant-rstats.github.io/docs/design/doe.html> for an example in Radiant

### Value

A list with all variables defined in the function as an object of class doe

**See Also**

[summary.doe](#) to summarize results

**Examples**

```
doe(c("price; $10; $13; $16", "food; popcorn; gourmet; no food"))
doe(
  c("price; $10; $13; $16", "food; popcorn; gourmet; no food"),
  int = "price:food", trials = 9, seed = 1234
)
```

---

 estimable

*Determine coefficients that can be estimated based on a partial factorial design*

---

**Description**

A function to determine which coefficients can be estimated based on a partial factorial design.

Adapted from a function written by Blakeley McShane at <https://github.com/fzettelmeyer/mktg482/blob/master/R/expdesign>.

**Usage**

```
estimable(design)
```

**Arguments**

`design` An experimental design generated by the `doe` function that includes a partial and full factorial design

**Examples**

```
design <- doe(c("price; $10; $13; $16", "food; popcorn; gourmet; no food"), trials = 6)
estimable(design)
```

---

 plot.sample\_size\_comp *Plot method for the sample\_size\_comp function*


---

**Description**

Plot method for the `sample_size_comp` function

**Usage**

```
## S3 method for class 'sample_size_comp'
plot(x, ...)
```

## Arguments

x                    Return value from `sample_size_comp`  
...                   further arguments passed to or from other methods

## Details

See [https://radiant-rstats.github.io/docs/design/sample\\_size\\_comp.html](https://radiant-rstats.github.io/docs/design/sample_size_comp.html) for an example in Radiant

## See Also

`sample_size_comp` to generate the results

## Examples

```
sample_size_comp(  
  type = "proportion", p1 = 0.1, p2 = 0.15,  
  conf_lev = 0.95, power = 0.8  
) %>% plot()
```

---

radiant.design

*radiant.design*

---

## Description

Launch `radiant.design` in the default web browser

## Usage

```
radiant.design(state, ...)
```

## Arguments

state                Path to state file to load  
...                   additional arguments to pass to `shiny::runApp` (e.g, port = 8080)

## Details

See <https://radiant-rstats.github.io/docs/> for documentation and tutorials

## Examples

```
## Not run:  
radiant.design()  
  
## End(Not run)
```

---

`radiant.design_viewer` *Launch radiant.design in the Rstudio viewer*

---

### Description

Launch `radiant.design` in the Rstudio viewer

### Usage

```
radiant.design_viewer(state, ...)
```

### Arguments

<code>state</code>	Path to state file to load
<code>...</code>	additional arguments to pass to <code>shiny::runApp</code> (e.g, port = 8080)

### Details

See <https://radiant-rstats.github.io/docs/> for documentation and tutorials

### Examples

```
## Not run:  
radiant.design_viewer()  
  
## End(Not run)
```

---

`radiant.design_window` *Launch radiant.design in an Rstudio window*

---

### Description

Launch `radiant.design` in an Rstudio window

### Usage

```
radiant.design_window(state, ...)
```

### Arguments

<code>state</code>	Path to state file to load
<code>...</code>	additional arguments to pass to <code>shiny::runApp</code> (e.g, port = 8080)

### Details

See <https://radiant-rstats.github.io/docs/> for documentation and tutorials

**Examples**

```
## Not run:
radiant.design_window()

## End(Not run)
```

---

randomizer

*Randomize cases into experimental conditions*


---

**Description**

Randomize cases into experimental conditions

**Usage**

```
randomizer(
  dataset,
  vars,
  conditions = c("A", "B"),
  blocks = NULL,
  probs = NULL,
  label = ".conditions",
  seed = 1234,
  data_filter = "",
  arr = "",
  rows = NULL,
  na.rm = FALSE,
  envir = parent.frame()
)
```

**Arguments**

dataset	Dataset to sample from
vars	The variables to sample
conditions	Conditions to assign to
blocks	A vector to use for blocking or a data.frame from which to construct a blocking vector
probs	A vector of assignment probabilities for each treatment conditions. By default each condition is assigned with equal probability
label	Name to use for the generated condition variable
seed	Random seed to use as the starting point
data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")
arr	Expression to arrange (sort) the data on (e.g., "color, desc(price)")

rows	Rows to select from the specified dataset
na.rm	Remove rows with missing values (FALSE or TRUE)
envir	Environment to extract data from

### Details

Wrapper for the `complete_ra` and `block_ra` from the `randomizr` package. See <https://radiant-rstats.github.io/docs/design/randomizer.html> for an example in Radiant

### Value

A list of variables defined in `randomizer` as an object of class `randomizer`

### See Also

[summary.sampling](#) to summarize results

### Examples

```
randomizer(rndnames, "Names", conditions = c("test", "control")) %>% str()
```

---

<code>rndnames</code>	<i>100 random names</i>
-----------------------	-------------------------

---

### Description

100 random names

### Usage

```
data(rndnames)
```

### Format

A data frame with 100 rows and 2 variables

### Details

A list of 100 random names. Description provided in `attr(rndnames,"description")`

---

`sample_size`*Sample size calculation*

---

**Description**

Sample size calculation

**Usage**

```
sample_size(  
  type,  
  err_mean = 2,  
  sd_mean = 10,  
  err_prop = 0.1,  
  p_prop = 0.5,  
  conf_lev = 0.95,  
  incidence = 1,  
  response = 1,  
  pop_correction = "no",  
  pop_size = 1e+06  
)
```

**Arguments**

<code>type</code>	Choose "mean" or "proportion"
<code>err_mean</code>	Acceptable Error for Mean
<code>sd_mean</code>	Standard deviation for Mean
<code>err_prop</code>	Acceptable Error for Proportion
<code>p_prop</code>	Initial proportion estimate for Proportion
<code>conf_lev</code>	Confidence level
<code>incidence</code>	Incidence rate (i.e., fraction of valid respondents)
<code>response</code>	Response rate
<code>pop_correction</code>	Apply correction for population size ("yes","no")
<code>pop_size</code>	Population size

**Details**

See [https://radiant-rstats.github.io/docs/design/sample\\_size.html](https://radiant-rstats.github.io/docs/design/sample_size.html) for an example in Radiant

**Value**

A list of variables defined in `sample_size` as an object of class `sample_size`



**See Also**

[summary.sample\\_size](#) to summarize results

**Examples**

```
sample_size(type = "mean", err_mean = 2, sd_mean = 10)
```

---

sample\_size\_comp      *Sample size calculation for comparisons*

---

**Description**

Sample size calculation for comparisons

**Usage**

```
sample_size_comp(  
  type,  
  n1 = NULL,  
  n2 = NULL,  
  p1 = NULL,  
  p2 = NULL,  
  delta = NULL,  
  sd = NULL,  
  conf_lev = NULL,  
  power = NULL,  
  ratio = 1,  
  alternative = "two.sided"  
)
```

**Arguments**

type	Choose "mean" or "proportion"
n1	Sample size for group 1
n2	Sample size for group 2
p1	Proportion 1 (only used when "proportion" is selected)
p2	Proportion 2 (only used when "proportion" is selected)
delta	Difference in means between two groups (only used when "mean" is selected)
sd	Standard deviation (only used when "mean" is selected)
conf_lev	Confidence level
power	Power
ratio	Sampling ratio (n1 / n2)
alternative	Two or one sided test

**Details**

See [https://radiant-rstats.github.io/docs/design/sample\\_size\\_comp.html](https://radiant-rstats.github.io/docs/design/sample_size_comp.html) for an example in Radiant

**Value**

A list of variables defined in `sample_size_comp` as an object of class `sample_size_comp`

**See Also**

[summary.sample\\_size\\_comp](#) to summarize results

**Examples**

```
sample_size_comp(  
  type = "proportion", p1 = 0.1, p2 = 0.15,  
  conf_lev = 0.95, power = 0.8  
)
```

---

sampling

*Simple random sampling*

---

**Description**

Simple random sampling

**Usage**

```
sampling(  
  dataset,  
  vars,  
  sample_size,  
  seed = 1234,  
  data_filter = "",  
  arr = "",  
  rows = NULL,  
  na.rm = FALSE,  
  envir = parent.frame()  
)
```

**Arguments**

<code>dataset</code>	Dataset to sample from
<code>vars</code>	The variables to sample
<code>sample_size</code>	Number of units to select
<code>seed</code>	Random seed to use as the starting point

data_filter	Expression entered in, e.g., Data > View to filter the dataset in Radiant. The expression should be a string (e.g., "price > 10000")
arr	Expression to arrange (sort) the data on (e.g., "color, desc(price)")
rows	Rows to select from the specified dataset
na.rm	Remove rows with missing values (FALSE or TRUE)
envir	Environment to extract data from

### Details

See <https://radiant-rstats.github.io/docs/design/sampling.html> for an example in Radiant

### Value

A list of class 'sampling' with all variables defined in the sampling function

### See Also

[summary.sampling](#) to summarize results

### Examples

```
sampling(rndnames, "Names", 10)
```

---

summary.doe	<i>Summary method for doe function</i>
-------------	--

---

### Description

Summary method for doe function

### Usage

```
## S3 method for class 'doe'
summary(object, eff = TRUE, part = TRUE, full = TRUE, est = TRUE, dec = 3, ...)
```

### Arguments

object	Return value from <a href="#">doe</a>
eff	If TRUE print efficiency output
part	If TRUE print partial factorial
full	If TRUE print full factorial
est	If TRUE print number of effects that will be estimable using the partial factorial design
dec	Number of decimals to show
...	further arguments passed to or from other methods.

## Details

See <https://radiant-rstats.github.io/docs/design/does.html> for an example in Radiant

## See Also

[doe](#) to calculate results

## Examples

```
c("price; $10; $13; $16", "food; popcorn; gourmet; no food") %>%
  doe() %>%
  summary()
```

---

summary.randomizer	<i>Summary method for the randomizer function</i>
--------------------	---

---

## Description

Summary method for the randomizer function

## Usage

```
## S3 method for class 'randomizer'
summary(object, dec = 3, ...)
```

## Arguments

object	Return value from <a href="#">randomizer</a>
dec	Number of decimals to show
...	further arguments passed to or from other methods

## Details

See <https://radiant-rstats.github.io/docs/design/randomizer.html> for an example in Radiant

## See Also

[randomizer](#) to generate the results

## Examples

```
randomizer(rndnames, "Names", conditions = c("test", "control")) %>% summary()
```

---

summary.sample\_size    *Summary method for the sample\_size function*

---

**Description**

Summary method for the sample\_size function

**Usage**

```
## S3 method for class 'sample_size'  
summary(object, ...)
```

**Arguments**

object	Return value from <a href="#">sample_size</a>
...	further arguments passed to or from other methods

**Details**

See [https://radiant-rstats.github.io/docs/design/sample\\_size.html](https://radiant-rstats.github.io/docs/design/sample_size.html) for an example in Radiant

**See Also**

[sample\\_size](#) to generate the results

**Examples**

```
sample_size(type = "mean", err_mean = 2, sd_mean = 10) %>%  
summary()
```

---

summary.sample\_size\_comp    *Summary method for the sample\_size\_comp function*

---

**Description**

Summary method for the sample\_size\_comp function

**Usage**

```
## S3 method for class 'sample_size_comp'  
summary(object, ...)
```

### Arguments

object            Return value from [sample\\_size\\_comp](#)  
...               further arguments passed to or from other methods

### Details

See [https://radiant-rstats.github.io/docs/design/sample\\_size\\_comp.html](https://radiant-rstats.github.io/docs/design/sample_size_comp.html) for an example in Radiant

### See Also

[sample\\_size\\_comp](#) to generate the results

### Examples

```
sample_size_comp(  
  type = "proportion", p1 = 0.1, p2 = 0.15,  
  conf_lev = 0.95, power = 0.8  
) %>% summary()
```

---

summary.samplng

*Summary method for the sampling function*

---

### Description

Summary method for the sampling function

### Usage

```
## S3 method for class 'samplng'  
summary(object, dec = 3, ...)
```

### Arguments

object            Return value from [samplng](#)  
dec               Number of decimals to show  
...               further arguments passed to or from other methods

### Details

See <https://radiant-rstats.github.io/docs/design/samplng.html> for an example in Radiant

### See Also

[samplng](#) to generate the results

**Examples**

```
sampling(rndnames, "Names", 10) %>% summary()
```

# Index

## \* datasets

- [rndnames](#), [7](#)
- [doe](#), [2](#), [11](#), [12](#)
- [estimable](#), [3](#)
- [plot.sample\\_size\\_comp](#), [3](#)
- [radiant.design](#), [4](#)
- [radiant.design\\_viewer](#), [5](#)
- [radiant.design\\_window](#), [5](#)
- [randomizer](#), [6](#), [12](#)
- [rndnames](#), [7](#)
- [sample\\_size](#), [8](#), [13](#)
- [sample\\_size\\_comp](#), [4](#), [9](#), [14](#)
- [sampling](#), [10](#), [14](#)
- [summary.doe](#), [3](#), [11](#)
- [summary.randomizer](#), [12](#)
- [summary.sample\\_size](#), [9](#), [13](#)
- [summary.sample\\_size\\_comp](#), [10](#), [13](#)
- [summary.sampling](#), [7](#), [11](#), [14](#)