Package 'rdss'

January 9, 2025

Title Companion Datasets and Functions for Research Design in the
Social Sciences

Version 1.0.14

Description Helper functions to accompany the Blair, Coppock, and Humphreys (2022) ``Research Design in the Social Sciences: Declaration, Diagnosis, and Redesign'' <https://book.declaredesign.org>. 'rdss' includes datasets, helper functions, and plotting components to enable use and replication of the book.

Imports dplyr, rlang (>= 1.0.0), generics, ggplot2, tibble, tidyr,
dataverse, readr, broom, purrr, estimatr, randomizr

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Suggests testthat (>= 3.0.0), rdrobust, DIDmultiplegt, broom.mixed,
marginaleffects, grf, CausalQueries, metafor, cjoint, lme4,
rstanarm, spdep, DeclareDesign, curl

Depends R (>= 2.10)

Config/testthat/edition 3

NeedsCompilation no

Author Graeme Blair [aut, cre] (<https://orcid.org/0000-0001-9164-2102>),
Alexander Coppock [aut] (<https://orcid.org/0000-0002-5733-2386>),
Macartan Humphreys [aut] (<https://orcid.org/0000-0001-7029-2326>)

Maintainer Graeme Blair <graeme.blair@gmail.com>

Repository CRAN

Date/Publication 2025-01-09 12:10:02 UTC

Contents

1

---

add_parens   *Add parentheses around standard error estimates*

Description

Add parentheses around standard error estimates

Usage

```
add_parens(x, digits = 3)
```

Arguments

x	Numeric vector
digits	Number of digits to retain

Value

A character vector with enclosing parentheses

Examples

```
std.error <- c(0.12, 0.001, 1.2)
add_parens(std.error)
```

---

best_predictor   *Best predictor function from causal_forest*

Description

Best predictor function from causal_forest

Usage

```
best_predictor(data, covariate_names, cuts = 20)
```

Arguments

data	A data.frame of covariates
covariate_names	
	A character vector of covariates to assess
cuts	Either a numeric vector of two or more unique cut points or a single number (greater than or equal to 2) giving the number of intervals into which each covariate is to be cut.

Value

a data.frame of the best predictors

---

bonilla_tillery   *Replication data for Bonilla and Tillery (2020), American Political Science Review (obtained from Dataverse 10.7910/DVN/IUZDQI)*

Description

Replication data for Bonilla and Tillery (2020), American Political Science Review (obtained from Dataverse 10.7910/DVN/IUZDQI)

Usage

```
bonilla_tillery
```

Format

A data.frame

---

causal_forest_handler   *Tidy helper function for causal_forest function*

Description

Runs estimates estimation function from interference package and returns tidy data frame output

Usage

```
causal_forest_handler(data, covariate_names, share_train = 0.5, ...)
```

Arguments

data	A data.frame
covariate_names	
	Names of covariates
share_train	Share of units to be used for training
...	Options to causal_forest

Details

https://draft.declaredesign.org/complex-designs.html#discovery-using-causal-forests

See ?causal_forest for further details

Value

a data.frame of estimates

Examples

```
library(DeclareDesign)
library(ggplot2)

dat <- fabricate(
   N = 1000,
   A = rnorm(N),
   B = rnorm(N),
   Z = complete_rs(N),
   Y = A*Z + rnorm(N))

# note: remove num.threads = 1 to use more processors
estimates <- causal_forest_handler(data = dat, covariate_names = c("A", "B"), num.threads = 1)

ggplot(data = estimates, aes(A, pred)) + geom_point()
```

clingingsmith_etal	*Replication data for David Clingingsmith, Asim Ijaz Khwaja, Michael Kremer (2020): Estimating the Impact of The Hajj: Religion and Tolerance in Islam's Global Gathering. The Quarterly Journal of Economics, Volume 124, Issue 3, August 2009, Pages 1133-1170*

Description

Replication data for David Clingingsmith, Asim Ijaz Khwaja, Michael Kremer (2020): Estimating the Impact of The Hajj: Religion and Tolerance in Islam's Global Gathering. The Quarterly Journal of Economics, Volume 124, Issue 3, August 2009, Pages 1133-1170

Usage

```
clingingsmith_etal
```

Format

A data.frame

conjoint_assignment	*Conjoint experiment assignment handler: conducts complete random assignment of all attribute levels*

Description

See https://book.declaredesign.org/experimental-descriptive.html#conjoint-experiments

Usage

```
conjoint_assignment(data, levels_list)
```

Arguments

data	A data.frame
levels_list	List of conjoint levels to assign

Value

a data.frame with random assignment added

---

conjoint_inquiries   *Conjoint experiment inquiries handler*

Description

See https://book.declaredesign.org/experimental-descriptive.html#conjoint-experiments

Usage

```
conjoint_inquiries(data, levels_list, utility_fn)
```

Arguments

data	A data.frame
levels_list	List of conjoint levels
utility_fn	a function that takes data and returns an additional column called U, which represents the utility of the choice

Value

a data.frame of estimand values

---

conjoint_measurement   *Conjoint experiment assignment handler: conducts complete random assignment of all attribute levels*

Description

See https://book.declaredesign.org/experimental-descriptive.html#conjoint-experiments

Usage

```
conjoint_measurement(data, utility_fn)
```

Arguments

data	A data.frame
utility_fn	a function that takes data and returns an additional column called U, which represents the utility of the choice

Value

a data.frame

dd_palette	*Access color palette used in the book "Research Design: Declare, Diagnose, Redesign" (Blair, Coppock, Humphreys)*

Description

Based on Karthik Ram's wesanderson package (https://github.com/karthik/wesanderson)

Usage

```
dd_palette(name, n)
```

Arguments

name	Color palette name (character)
n	Number of colors

Details

Available color palettes:

color_palette = c("#72B4F3", "#F38672", "#C6227F")

grey_palette = c("#72B4F3", "#F38672", "#C6227F", gray(0.8))

dd_dark_blue = "#3564ED"

dd_light_blue = "#72B4F3"

dd_orange = "#F38672"

dd_purple = "#7E43B6"

dd_gray = gray(0.2)

dd_pink = "#C6227F"

dd_light_gray = gray(0.8)

dd_dark_blue_alpha = "#3564EDA0"

dd_light_blue_alpha = "#72B4F3A0"

Value

character vector of colors

---

did_multiplegt_tidy   *Tidy helper function for did_multiplegt*

Description

Runs did_multiplegt estimation function and returns tidy data frame output

Usage

```
did_multiplegt_tidy(data, ...)
```

Arguments

data	a data.frame
...	options passed to did_multiplegt

Details

See https://book.declaredesign.org/observational-causal.html#difference-in-differences

Value

a data.frame of estimates

---

estimator_AS_tidy   *Tidy helper function for estimator_AS function*

Description

Runs estimates estimation function from interference package and returns tidy data frame output

Usage

```
estimator_AS_tidy(data, permutatation_matrix, adj_matrix)
```

Arguments

data	a data.frame
permutatation_matrix	
	a permutation matrix of random assignments
adj_matrix	an adjacency matrix

Details

The estimator_AS_tidy function requires the 'interference' package, which is not yet available on CRAN.

To use this function:

1. install the developer version of interference via remotes::install_github('szonszein/interference') and

2. install the developer version of rdss via remotes::install_github('DeclareDesign/rdss@remotes')

See https://book.declaredesign.org/experimental-causal.html#experiments-over-networks

Value

a data.frame of estimates

---

fairfax	*Shapefile of Fairfax County, Virginia, voting precincts*

Description

An sf object containing the boundaries of voting precincts for Fairfax County, Virginia as well as precinct ID, name, district, polling place name, address, city, zip code, area, length, and geometry (polygons)

Usage

```
fairfax
```

Format

An sf object with 236 rows and 10 variables:

---

foos_etal	*Replication data for Foos, John, Muller, and Cunningham (2021), Journal of Politics (derived from from Dataverse 10.7910/DVN/NDPXND)*

Description

Replication data for Foos, John, Muller, and Cunningham (2021), Journal of Politics (derived from from Dataverse 10.7910/DVN/NDPXND)

Usage

```
foos_etal
```

Format

A data.frame

---

format_num	*Round and pad a number to a specific decimal place*

Description

Round and pad a number to a specific decimal place

Usage

```
format_num(x, digits = 3)
```

Arguments

x	Numeric vector
digits	Number of digits to retain

Value

a character vector of formatted numbers

Examples

```
std.error <- c(0.12, 0.001, 1.2)
format_num(std.error)
```

---

get_exposure_AS	*Helper function to obtain the observed exposure for the Aronow and Samii estimator*

Description

See https://book.declaredesign.org/experimental-causal.html#experiments-over-networks

Usage

```
get_exposure_AS(obs_exposure)
```

Arguments

obs_exposure	A numeric vector

Value

a data.frame of observed exposure to a treatment created using the interference package

---

get_rdss_file	*Download a replication file from the dataverse archive for Research Design in the Social Sciences: Declaration, Diagnosis, and Redesign*

Description

See https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/HYVPO5 for further details and the code used to create these files.

Usage

```
get_rdss_file(name, verbose = TRUE)
```

Arguments

name	quoted name of the file on the dataverse archive
verbose	print declaration code if requesting a declaration

Details

The available names include:

Design declaration objects:

declaration_9.5
declaration_2.1
declaration_2.2
declaration_4.1
declaration_5.1
declaration_7.1
declaration_9.1
declaration_9.2
declaration_9.3
declaration_9.4
declaration_9.6
declaration_9.7
declaration_10.1
declaration_10.2
declaration_10.3
declaration_10.4
declaration_10a
declaration_11.1
declaration_11.2
declaration_11.3
declaration_11.4
declaration_11.5
declaration_12.1a
declaration_12.1b

declaration_12.1c
declaration_12.1d
declaration_13.1
declaration_13.2
declaration_15.1
declaration_15.2
declaration_15.3a
declaration_15.3b
declaration_15.3c
declaration_15.4
declaration_15.5
declaration_15.6
declaration_16.1a
declaration_16.1b
declaration_16.2
declaration_16.3
declaration_16.4
declaration_16.5
declaration_16.6
declaration_17.1
declaration_17.2
declaration_17.3
declaration_17.4
declaration_17.5
declaration_17.6_a
declaration_17.6_b
declaration_18.1
declaration_18.2
declaration_18.3
declaration_18.4
declaration_18.5
declaration_18.6
declaration_18.7
declaration_18.8
declaration_18.9a
declaration_18.9b
declaration_18.9c
declaration_18.10
declaration_18.11
declaration_18.12
declaration_18.13
declaration_19.1
declaration_19.2
declaration_19.3
declaration_19.4
declaration_23.1a
declaration_23.1b
declaration_23.1c

declaration_23.1d

Diagnosis objects:

diagnosis_2.1
diagnosis_4.1
diagnosis_9.1
diagnosis_9.2
diagnosis_9.3
diagnosis_9.4
diagnosis_9.5
diagnosis_9.6
diagnosis_9.7
simulation_10.1
diagnosis_10.1
diagnosis_10.2
diagnosis_10.3
diagnosis_10.4
diagnosis_10.5
diagnosis_10a
diagnosis_11.1
diagnosis_11.2
diagnosis_11.3
diagnosis_11.4
diagnosis_11.5
diagnosis_12.1
diagnosis_12.2
diagnosis_13.1
diagnosis_15.1
diagnosis_15.2
diagnosis_15.3
diagnosis_15.4
diagnosis_15.5
diagnosis_16.1
diagnosis_16.2
diagnosis_16.3
diagnosis_16.4
diagnosis_16.5
diagnosis_17.1
diagnosis_17.2
diagnosis_17.3
diagnosis_17.4
diagnosis_17.5
diagnosis_18.1
diagnosis_18.10_encouragment
diagnosis_18.10_placebo
diagnosis_18.11
diagnosis_18.12
diagnosis_18.13

diagnosis_18.2
diagnosis_18.3
diagnosis_18.4
diagnosis_18.5
diagnosis_18.6
diagnosis_18.7
diagnosis_18.8
diagnosis_18.9
diagnosis_19.1
diagnosis_19.2
diagnosis_19.3
diagnosis_19.4
diagnosis_19a
diagnosis_21a
diagnosis_21b
diagnosis_23.1
diagnosis_23a

Value

an r object

Examples

```
## Not run:
# Requires internet access
if(curl::has_internet()) {
  diagnosis_2.1 <- get_rdss_file("diagnosis_2.1")
  diagnosis_2.1
}

## End(Not run)
```

---

hex_add_alpha   *Add alpha transparency to a color defined in hexadecimal*

Description

Add alpha transparency to a color defined in hexadecimal

Usage

```
hex_add_alpha(col, alpha)
```

Arguments

col   Original color code in hex
alpha   Level of alpha transparency to add

Value

color codes with alpha added

| lag_by_group | *Generate lags in grouped data* |

Description

See https://book.declaredesign.org/observational-causal.html#difference-in-differences

Usage

```
lag_by_group(x, groups, n = 1, order_by, default = NA)
```

Arguments

x	Vector of values
groups	Grouping variable
n	Positive integer of length 1, giving the number of positions to lead or lag by
order_by	Ordering variable withing group (e.g., time)
default	Value used for non-existent rows. Defaults to NA.

Value

vector of lagged values

| lapop_brazil | *Data used in student exercises for RDSS based on LAPOP survey of Brazil in 2018* |

Description

These data were resampled with replacement from LAPOP data (to 10,000 rows) for a subset of variables. These data cannot be used for scientific inferences, and are only useful for teaching purposes. ID numbers were scrambled so that individuals and municipalities cannot easily be identified.

Usage

```
lapop_brazil
```

Format

A data.frame

Details

Download the original data from https://www.vanderbilt.edu/lapop/raw-data.php

See https://www.vanderbilt.edu/lapop/core-surveys.php for survey questionnaire

---

la_voter_file   *Voter file sample for Los Angeles County*

Description

A dataset containing the party registration, age, census tract number, and voter turnout in 2012 for 1,000 randomly-sampled registered voters in Los Angeles County, California.

Usage

```
la_voter_file
```

Format

A data frame with 1000 rows and 4 variables:

party political party registration

age age of voter in years

census_tract US Census tract number

voted_2012 voter turnout in 2012 election

Source

California Secretary of State.

---

make_interval_entry   *Format confidence intervals for nice printing*

Description

Format confidence intervals for nice printing

Usage

```
make_interval_entry(conf.low, conf.high, digits = 2)
```

Arguments

conf.low	a numeric vector of lower bounds
conf.high	a numeric vector of upper bounds
digits	number of digits to retain

Value

a character vector of intervals

Examples

```
conf.low <- c(-0.1652, 0.00304, -6.352)
conf.high <- c(0.3052, 0.00696, -1.648)

make_interval_entry(conf.low, conf.high)
```

---

make_se_entry   *Format estimates and standard errors for nice printing*

Description

Format estimates and standard errors for nice printing

Usage

```
make_se_entry(estimate, std.error, digits = 2)
```

Arguments

estimate	a numeric vector of parameter estimates
std.error	a numeric vector of standard error estimates
digits	number of digits to retain

Value

a character vector of formatted estimates and standard errors

Examples

```
estimate <- c(0.07, 0.005, -4)
std.error <- c(0.12, 0.001, 1.2)

make_se_entry(estimate, std.error)
```

`post_stratification_helper`
Post stratification estimator helper

Description

Calculates predicted values from a multilevel regression and the post-stratified state-level estimates

Usage

```
post_stratification_helper(model_fit, data, group, weights)
```

Arguments

`model_fit`	a model fit object from, e.g., glmer or lm_robust
`data`	a data.frame
`group`	unquoted name of the group variable to construct estimates for
`weights`	unquoted name of post-stratification weights variable

Details

Please see https://book.declaredesign.org/observational-descriptive.html#multi-level-regression-and-poststratification

Value

data.frame of post-stratified group-level estimates

`process_tracing_estimator`
Process tracing estimator

Description

Draw conclusions from a model given a query, data, and process tracing strategies

Usage

```
process_tracing_estimator(causal_model, query, data, strategies)
```

Arguments

`causal_model`	a model generated by `CausalQueries`
`query`	a causal query of interest
`data`	a single row dataset with data on nodes in the model
`strategies`	a vector describing sets of nodes to be examined e.g. c("X", "X-Y")

Details

See https://book.declaredesign.org/observational-causal.html#process-tracing

Value

a data.frame of estimates

Examples

```
# Simple example showing ambiguity in attribution
process_tracing_estimator(
  causal_model = CausalQueries::make_model("X -> Y"),
  query = "Y[X=1] > Y[X=0]",
  data = data.frame(X=1, Y = 1),
  strategies = "X-Y")

# Example where M=1 acts as a hoop test
process_tracing_estimator(
  causal_model = CausalQueries::make_model("X -> M -> Y") |>
   CausalQueries::set_restrictions("Y[M=1] < Y[M=0]") |>
   CausalQueries::set_restrictions("M[X=1] < M[X=0]"),
  query = "Y[X=1] > Y[X=0]",
  data = data.frame(X=1, Y = 1, M = 0),
  strategies = c("Y", "X-Y", "X-M-Y"))
```

---

rdrobust_helper	*Helper function for using rdrobust as a model in* declare_estimator

---

Description

Helper function for using rdrobust as a model in declare_estimator

Usage

```
rdrobust_helper(data, y, x, subset = NULL, ...)
```

Arguments

data	a data.frame
y	unquoted name of the outcome variable
x	unquoted name of the running variable
subset	an optional vector specifying a subset of observations to be used in the fitting process
...	Other arguments to rdrobust

Value

rdrobust model fit object

| rdss | *rdss package* |

Description

Companion datasets and functions for the book "Research Design in the Social Sciences: Declaration, Diagnosis, and Redesign" (book.declaredesign.org)

Author(s)

Maintainer: Graeme Blair <graeme.blair@gmail.com> (ORCID)

Authors:

- Alexander Coppock <acoppock@gmail.com> (ORCID)
- Macartan Humphreys <macartan@gmail.com> (ORCID)

| rma_helper | *Helper function for rma function in metafor package* |

Description

See https://book.declaredesign.org/complex-designs.html#meta-analysis

Usage

```
rma_helper(data, yi, sei, method = "REML", ...)
```

Arguments

data	a data.frame
yi	unquoted variable name of estimates used in meta-analysis
sei	unquoted variable name of standard errors used in meta-analysis
method	character string to specify whether a fixed- or a random/mixed-effects model should be fitted. A fixed-effects model (with or without moderators) is fitted when using method = "FE". Random/mixed-effects models are fitted by setting method equal to one of the following: "DL", "HE", "SJ", "ML", "REML", "EB", "HS", "HSk", or "GENQ". Default is "REML".
...	Further options to be passed to rma

Details

See ?rma for further details

Value

a data.frame of estimates

---

rma_mu_tau *Extract mu and tau parameters from rma model fit*

Description

See https://book.declaredesign.org/complex-designs.html#meta-analysis

Usage

```
rma_mu_tau(fit)
```

Arguments

fit   Fit object from the rma function in the metafor package

Value

a data.frame of estimates

---

theme_dd *ggplot Theme used in the book "Research Design: Declare, Diagnose, Redesign" (Blair, Coppock, Humphreys)*

Description

ggplot Theme used in the book "Research Design: Declare, Diagnose, Redesign" (Blair, Coppock, Humphreys)

Usage

```
theme_dd()
```

Value

ggplot theme

| tidy.amce | *Tidy estimates from the amce estimator* |

Description

Runs amce estimation function and returns tidy data frame output

Usage

```
## S3 method for class 'amce'
tidy(x, alpha = 0.05, ...)
```

Arguments

x	an amce fit object from cjoint::amce
alpha	Confidence level
...	Extra arguments to pass to tidy

Details

See https://book.declaredesign.org/experimental-descriptive.html#conjoint-experiments

Value

a data.frame of estimates

Examples

```
library(cjoint)

data(immigrationconjoint)
data(immigrationdesign)

# Run AMCE estimator using all attributes in the design
results <- amce(Chosen_Immigrant ~  Gender + Education + `Language Skills` +
                 `Country of Origin` + Job + `Job Experience` + `Job Plans` +
                 `Reason for Application` + `Prior Entry`, data = immigrationconjoint,
             cluster = TRUE, respondent.id = "CaseID", design = immigrationdesign)

# Print summary
# tidy(results)
```

---

tidy.rdrobust   *Tidy helper function for rdrobust function*

Description

Runs rdrobust estimation function and returns tidy data frame output

Usage

```
## S3 method for class 'rdrobust'
tidy(x, ...)
```

Arguments

x	Model fit object from rdrobust
...	Other arguments (not used)

Details

See https://book.declaredesign.org/observational-causal.html#regression-discontinuity-designs

Value

a data.frame of estimates

---

tidy_stan   *Tidy results from a stanreg regresion and exponentiate the estimated coefficient*

Description

Note no standard errors or other summary statistics are provided

This function is deprecated. Please use the 'tidy' function from the 'broom.mixed' package.

Usage

```
tidy_stan(x, conf.int = FALSE, conf.level = 0.95, exponentiate = FALSE, ...)

tidy_stan(x, conf.int = FALSE, conf.level = 0.95, exponentiate = FALSE, ...)
```

Arguments

`x`	A stanreg fit from stan_glm
`conf.int`	Logical indicating whether or not to include a confidence interval in the tidied output. Defaults to FALSE.
`conf.level`	The confidence level to use for the confidence interval if conf.int = TRUE. Must be strictly greater than 0 and less than 1. Defaults to 0.95, which corresponds to a 95 percent confidence interval.
`exponentiate`	Logical indicating whether or not to exponentiate the the coefficient estimates. Defaults to FALSE.
`...`	Other arguments to broom.mixed::tidy

Details

See https://book.declaredesign.org/choosing-an-answer-strategy.html#bayesian-formalizations

See https://book.declaredesign.org/choosing-an-answer-strategy.html#bayesian-formalizations

Value

data.frame of results

data.frame of results

Index