Package 'swCRTdesign'

July 9, 2022

Type Package

Title Stepped Wedge Cluster Randomized Trial (SW CRT) Design

Version 3.3

Date 2022-07-07

Maintainer Jim Hughes <jphughes@uw.edu>

Author Jim Hughes, Navneet R. Hakhu, Emily Voldal, and Fan Xia

Description A set of tools for examining the design and analysis aspects of stepped wedge cluster randomized trials (SW CRT) based on a repeated cross-sectional sampling scheme (Hussey MA and Hughes JP (2007) Contemporary Clinical Trials 28:182-191. <doi:10.1016/j.cct.2006.05.007>).

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2022-07-08 22:00:02 UTC

R topics documented:

swCRTdesign-package 2
blkDiag 3
swDsn 4
swGlmPwr 6
swPlot 9
swPwr 12
swSim 16
swSummary 19

Index **22**

swCRTdesign-package *Stepped Wedge Cluster Randomized Trial (SW CRT) Design*

Description

This package includes functions for the design and analysis of stepped wedge cluster randomized trials according to a repeated cross-sectional sampling scheme. For additional guidance, see (Voldal EC, Hakhu NR, Xia F, Heagerty PJ, Hughes JP. swCRTdesign: An R package for stepped wedge trial design and analysis. Computer Methods and Programs in Biomedicine 2020;196:105514. <doi:10.1016/j.cmpb.2020.105514>). Five primary functions - `swPwr`, `swPwrGlm`, `swSim`, `swSummary`, and `swPlot` - and two support functions - `blkDiag`, `swDsn` - are included. The `blkDiag` function creates a block diagonal matrix from a specified array or list of block-matrices. The `swDsn` function creates a stepped wedge (SW) design object based on specified information on clusters, time points, and the two arms of the cluster randomized trial (CRT). The `swPwr` function computes the (two-sided) power of treatment effect (θ) for the specified SW CRT design via weighted least squares (WLS), where the response/outcome of interest is assumed to come from a mixed effects model with linear link and random time effects and (possibly correlated) random intercepts and random treatment effects. The random time effects apply to all time points, and time is treated as categorical. `swPwrGlm` does power calculations using the generalized linear model framework (Xia et al, 2019). `swPwr` and `swPwrGlm` provide power calculations for both an immediate treatment (IT) model and an exposure time indicator (ETI) model (Kenny et al, 2022). The `swSim` function generates individual-level data consisting of response, treatment, time, and cluster variables based on a specified SW CRT design. The `swSummary` function computes the mean, sum, or number of non-missing response values for clusters separately or aggregated by wave at each time point from stepped wedge data that includes, at least, response, treatment, time, and cluster variables. The `swPlot` function plots mean response as a combined or separate plot, for waves and clusters. Some features of the package are also available as a shiny app, available online (https://swcrtdesign.shinyapps.io/stepped_wedge_power_calculation/) or to download and run locally (https://github.com/swCRTdesign/Stepped-wedge-power-calculation).

Details

Package:	swCRTdesign
Type:	Package
Version:	3.3
Date:	2022-07-7
License:	GPL (>= 2)

Author(s)

James P Hughes, Navneet R Hakhu, Emily C Voldal, and Fan Xia

Maintainer: James P Hughes <jphughes@uw.edu>

References

Hussey MA, Hughes JP. Design and analysis of stepped wedge cluster randomized trials. Contemporary Clinical Trials 2007;28:182-191.

Kenny A, Voldal E, Xia F, Heagerty PJ, Hughes JP. Analysis of stepped wedge cluster randomized trials in the presence of a time-varying treatment effect. Statistics in Medicine, in press, 2022.

Voldal EC, Hakhu NR, Xia F, Heagerty PJ, Hughes JP. swCRTdesign: An R package for stepped wedge trial design and analysis. Computer Methods and Programs in Biomedicine 2020;196:105514.

Xia F, Hughes JP, Voldal EC, Heagerty PJ. Power and sample size calculation for stepped-wedge designs with discrete outcomes. Trials. 2021 Dec;22(1):598.

`blkDiag` *Block Diagonal Matrix Generator*

Description

`blkDiag` returns block diagonal matrix based on specified square blocks (either as an array or a list).

Usage

```
blkDiag(z)
```

Arguments

`z`	numeric (array or list): User-specified matrices to be combined into one block diagonal matrix.

Value

numeric (matrix): `blkDiag` gives a block diagonal matrix.

Author(s)

James P Hughes and Navneet R Hakhu

Examples

```
library(swCRTdesign)
# Example 1 (input: array)
blkDiag.Ex1.array <- blkDiag( z=array(1:12, c(2,2,3)) )
blkDiag.Ex1.array

# Example 2 (input: list)
blkDiag.Ex2.list <- blkDiag( z=list(matrix(1,2,2), diag(2,3), matrix(3,4,4)) )
blkDiag.Ex2.list
```

swDsn *Study design of Stepped Wedge Cluster Randomized Trial (SW CRT)*

Description

swDsn returns a SW CRT study design object based on a repeated cross-sectional sampling scheme. All clusters that start the intervention at a given time point are collectively referred to as a wave or sequence. There many be a variable number of clusters in each wave. By default, all clusters are assumed to start in the control condition. Fractional treatment effect may be specified for each time after the intervention is introduced. Additional observations may be added to the end of the trial after the intervention has been introduced in all clusters. For incorporating transition periods where no data is collected, see swPwr. swDsn is used by other functions in this package.

Usage

```
swDsn(clusters, tx.effect.frac = 1, extra.time = 0, all.ctl.time0 = TRUE)
```

Arguments

clusters	integer (vector): Number of clusters for each wave (e.g. c(6,6,6,6) specifies four waves with 6 clusters in each wave). A value of 0 in the vector means that no clusters introduce the intervention at a given time (see examples).
tx.effect.frac	numeric (scalar or vector): Fractional treatment effect upon crossing over from control. Note that this is not the treatment effect! If a scalar with value of 1, the standard SW CRT treatment effect will be presumed. If a scalar with a fractional value between 0 and 1, then only the first time point upon crossing over from control will have fractional treatment effect; the remaining time points in SW CRT design will have value of 1. If a vector of fractional treatment effect is specified, each element of the vector corresponds to the (fractional) treatment effect upon crossing over from control; if length of vector less than total number of time points after crossing over, the remaining time points will have treatment effect value of 1; if length of vector greater than total number of time points after crossing over, not all elements of vector will be used. The default value is (scalar) 1.
extra.time	integer (scalar): Number of additional time steps beyond the standard SW CRT design (standard + extended times corresponds to total time). The default value is 0.
all.ctl.time0	logical: If TRUE, all clusters receive control at the first time point. If FALSE, clusters in the first wave (i.e., the first element of clusters) receive intervention at the first time. The default is TRUE.

Value

numeric (list): Returns the following user-specified and computed objects

swDsn	numeric (matrix): schematic representation of the specified SW CRT design. Number of clusters is equal to the number of rows of the matrix and number of time intervals is equal to the number of columns of the matrix. swDsn[i,j] gives the intervention status for cluster i at time j and has possible values 0 (control), 1 (intervention) or a fractional value as specified by tx.effect.frac.
swDsn.unique.clusters	numeric (matrix): Truncated SW CRT design of interest, with one row for each wave.
n.waves	numeric (scalar): Number of waves for the SW CRT design of interest.
clusters	numeric (vector): Number of clusters changing from control to intervention at each wave for the SW CRT design of interest.
n.clusters	numeric (scalar): Total number of clusters for the SW CRT design of interest.
tx.effect.frac	numeric (scalar or vector): Fractional treatment effect for time points upon crossing over from control of SW CRT design of interest.
total.time	numeric (scalar): Total number of time points for the SW CRT design of interest.
extra.time	numeric (matrix): Number of time points added on to the standard SW CRT time points for the user-specified values (i.e., extending the trial).

Author(s)

James P Hughes and Navneet R Hakhu

References

Hussey MA, Hughes JP. Design and analysis of stepped wedge cluster randomized trials. Contemporary Clinical Trials 2007;28:182-191.

Examples

```
library(swCRTdesign)
# Example 1 (Equal clusters per wave, standard SW design)
swDsn.Ex1.std <- swDsn( clusters=c(3,3,3) )
swDsn.Ex1.std$swDsn

# Example 2 (Equal clusters per wave, extended SW design)
swDsn.Ex1.extend <- swDsn( clusters=c(3,3,3), extra.time=2 )
swDsn.Ex1.extend$swDsn

# Example 3 (Equal clusters per wave, not all ctl at time 0, "standard" for time SW design)
swDsn.Ex1.std.noAllctl <- swDsn( clusters=c(3,3,3), all.ctl.time0=FALSE )
swDsn.Ex1.std.noAllctl$swDsn

# Example 4 (Equal clusters per wave, not all ctl at time 0, extended SW design)
swDsn.Ex1.extend.noAllctl <- swDsn( clusters=c(3,3,3), extra.time=2, all.ctl.time0=FALSE )
swDsn.Ex1.extend.noAllctl$swDsn

# Example 5 (Unequal clusters per wave, standard SW design)
swDsn.Ex1.std.unequal <- swDsn( clusters=c(3,0,2) )
swDsn.Ex1.std.unequal$swDsn
```

```
# Example 6 (Unequal clusters per wave, extended SW design)
swDsn.Ex1.extend.unequal <- swDsn( clusters=c(3,0,2), extra.time=2 )
swDsn.Ex1.extend.unequal$swDsn

# Example 7 (Unequal clusters per wave, extended SW design)
swDsn.Ex1.extend.unequal.varyTxEffect <- swDsn( clusters=c(3,0,2), tx.effect.frac=c(0.8,0.9,1.0),
extra.time=2 )
swDsn.Ex1.extend.unequal.varyTxEffect$swDsn
```

`swGlmPwr` *Power of Stepped Wedge Cluster Randomized Trial with Discrete Outcomes*

Description

`swGlmPwr` returns (two-sided) power of the treatment effect for the specified SW CRT design in the context of generalized linear models by adopting the Laplace approximation detailed in Breslow and Clayton (1993) to obtain the covariance matrix of the estimated parameters. The response/outcome of interest can be binomial or Poisson distributed. The outcome is assumed to come from a model with fixed treatment effect (using an immediate treatment (IT) or exposure time indicator (ETI) model - see Kenny et al (2022)), fixed time effect, random intercepts, random treatment effects, and random cluster-specific time effects. The coefficients for fixed effects can be specified using `fixed.intercept`, `fixed.treatment.effect`, and `fixed.time.effect`. Variance components can be specified using `tau`, `eta`, `rho`, and `gamma`.

Usage

```
swGlmPwr(design, distn, n, fixed.intercept,
fixed.treatment.effect, fixed.time.effect, H = NULL,
tau = 0, eta = 0, rho = 0, gamma = 0, alpha=0.05, retDATA = FALSE)
```

Arguments

`design`	list: A stepped wedge design object, typically from swDsn, that includes at least the following components: swDsn, swDsn.unique.clusters, clusters, n.clusters, total.time
`distn`	character: Distribution assumed (binomial or Poisson). "binomial" implies binomial outcomes and "poisson" implies Poisson outcome.
`n`	integer (scalar, vector, or matrix): Number of observations: (scalar) for all clusters and all time points; (vector) for each cluster at all time points; and (matrix) for each cluster at each time point, where rows correspond to clusters and columns correspond to time. `n` can also be used to specify a design with transition periods (e.g. in the first time period that each sequence receives treatment, no observations are collected from that sequence). Simply define `n` as a matrix with a sample size of 0 during every transition period.

`fixed.intercept`
: numeric (scalar): Intercept for the fixed effect on canonical scales (logit for binomial outcomes and log for Poisson outcomes). It is the mean outcome under the control condition in the first time point transformed to the canonical scales.

`fixed.treatment.effect`
: numeric (scalar, vector): If H = NULL then an IT model is assumed and and fixed.treatment.effect is the scalar coefficient for the treatment in the fixed effect model on canonical scales (logit for binomial outcomes and log for Poisson outcomes). If H is non-NULL then an ETI model is assumed and fixed.treatment.effect is a vector as long as the longest treatment effect lag (typically, number of time periods minus one) giving the coefficient for the treatment effect on the canonical scale.

`fixed.time.effect`
: numeric(scalar, vector): Coefficients for the time (as dummy variables) in the fixed effect model on canonical scales (logit for binomial outcomes and log for Poisson outcomes). The first time point is always used as reference. Specify a common time effect for all time points after the first (scalar) or differnt time effects for each time point after the first (vector of length (total time-1)).

`H`
: numeric (vector): If NULL, then swGlmPwr assumes an immediate, constant treatment effect (IT) model. If not NULL, then an exposure time indicator (ETI) model is assumed and H should be a vector as long as the longest treatment effect lag (typically, number of time periods minus one). H specifies the desired linear combination of fixed.treatment.effect. For example, in a stepped wedge trial with 5 time periods and four exposure times, H = rep(.25,4) gives the average treatment effect over the four exposure times; H = c(0,0,.5,.5) ignores the first two periods after the intervention is introduced and averages the remaining periods. Typically, the sum of H is 1.0.

`tau`
: numeric (scalar): Standard deviation of random intercepts on canonical scales (logit for binomial outcomes and log for Poisson outcomes).

`eta`
: numeric (scalar): Standard deviation of random treatment effects on canonical scales (logit for binomial outcomes and log for Poisson outcomes).

`rho`
: numeric (scalar): Correlation between random intercepts and random treatment effects on canonical scales (logit for binomial outcomes and log for Poisson outcomes).

`gamma`
: numeric (scalar): Standard deviation of random time effects on canonical scales (logit for binomial outcomes and log for Poisson outcomes).

`alpha`
: numeric (scalar): Statistical significance level. Default is 0.05.

`retDATA`
: logical: if `TRUE`, all stored (input, intermediate, and output) values of `swGlmPwr` are returned. Default value is `FALSE`.

Details

The two-sided statistical power of treatment effect θ (equal to H%*%fixed.treatment.effect if H is non-NULL) is

$$Pwr(\theta) = \Phi(\frac{Z - z_{1-\alpha/2}\sqrt{V_0(\hat{\theta})}}{\sqrt{V_\alpha(\hat{\theta})}}) + 1 - \Phi(\frac{Z + z_{1-\alpha/2}\sqrt{V_0(\hat{\theta})}}{\sqrt{V_\alpha(\hat{\theta})}})$$

, where Φ is the cumulative distribution function of the standard normal distribution.

The variance of $\hat{\theta}$ under the null is denoted as $V_0(\hat{\theta})$, and the variance of $\hat{\theta}$ under the alternative is denoted as $V_\alpha(\hat{\theta})$). Both variances are approximated by simplifying the Laplace approximation that marginalizes the random effects in the generalized linear mixed models. For more details, see Xia et al. (2020).

When the outcome is Gaussian, the method adopted by `swGlmPwr` coincides with that of `swPwr`, so power calculation for Gaussian outcomes is not included in `swGlmPwr` to avoid repetition. When the outcome is binomial, `swGlmPwr` performs power calculation on the natural scale (logit), while `swPwr` performs power calculation on the linear scale.

Value

numeric (scalar): `swGlmPwr` returns the power of treatment effect if retDATA = FALSE.

numeric (list): `swGlmPwr` returns all specified and computed items as objects of a list if retDATA = TRUE.

`$design` list: A stepped wedge design object, typically from swDsn, that includes at least the following components: swDsn, swDsn.unique.clusters, clusters, n.clusters, total.time

`$distn` character: Distribution assumed (binomial or Poisson). "binomial" implies binomial outcomes and "poisson" implies Poisson outcome.

`$n` integer (scalar, vector, or matrix): Number of observations: (scalar) for all clusters and all time points; (vector) for each cluster at all time points; and (matrix) for each cluster at each time point, where rows correspond to clusters and columns correspond to time. `n` can also be used to specify a design with transition periods (e.g. in the first time period that each sequence receives treatment, no observations are collected from that sequence). Simply define `n` as a matrix with a sample size of 0 during every transition period.

`$fixed.intercept`
numeric (scalar): Intercept for the fixed effect on canonical scales (logit for binomial outcomes and log for Poisson outcomes).

`$fixed.treatment.effect`
numeric (scalar): Coefficient for the treatment in the fixed effect model on canonical scales (logit for binomial outcomes and log for Poisson outcomes).

`$fixed.time.effect`
numeric(scalar, vector): Coefficients for the time (as dummy variables) in the fixed effect model on canonical scales (logit for binomial outcomes and log for Poisson outcomes). The first time point is always used as reference. A common time effect for all time points after the first (scalar) or differnt time effects for each time point after the first (vector of length (total time-1)).

`$tau` numeric (scalar): Standard deviation of random intercepts on canonical scales (logit for binomial outcomes and log for Poisson outcomes).

`$eta` numeric (scalar): Standard deviation of random treatment effects on canonical scales (logit for binomial outcomes and log for Poisson outcomes).

`$rho` numeric (scalar): Correlation between random intercepts and random treatment effects on canonical scales (logit for binomial outcomes and log for Poisson outcomes).

$gamma	numeric (scalar): Standard deviation of random time effects on canonical scales (logit for binomial outcomes and log for Poisson outcomes).
$alpha	numeric (scalar): Statistical significance level. Default is 0.05.
$var.theta.null	numeric (scalar): Variance estimate of the estimated treatment effect under the null for this SW CRT design.
$var.theta.alt	numeric (scalar): Variance estimate of the estimated treatment effect under the alternative for this SW CRT design.
$pwrGLM	numeric (scalar): Power of treatment effect using a simplified Laplace approximation.

Author(s)

Fan Xia, James P Hughes, and Emily C Voldal

References

Breslow NE and Clayton DG (1993). Approximate inference in generalized linear mixed models. Journal of the American Statistical Association, 88(421):9-25.

Kenny A, Voldal E, Xia F, Heagerty PJ, Hughes JP. Analysis of stepped wedge cluster randomized trials in the presence of a time-varying treatment effect. Statistics in Medicine, in press, 2022.

Xia F, Hughes JP, Voldal EC, Heagerty PJ. Power and sample size calculation for stepped-wedge designs with discrete outcomes. Trials. 2021 Dec;22(1):598.

Examples

```
##test-case large clusters
library(swCRTdesign)
#specify large cluster sizes
size = c(35219,53535,63785,456132,128670,96673,
51454,156667,127440,68615,56502,17719,75931,58655,52874,75936)
#calculate power
swGlmPwr(design=swDsn(c(4,3,5,4)),distn="binomial",n=size,
fixed.intercept=log(28.62/(2*100000)),fixed.time.effect = 1,fixed.treatment.effect = log(.6),
tau=.31,eta=abs(0.4*log(.6)),rho=0,gamma=.15,alpha=.05,retDATA = FALSE)
```

---

swPlot *Plot of Mean Response/Outcome for Stepped Wedge Cluster Randomized Trial (SW CRT)*

Description

swPlot returns plot of the mean response versus time based on waves and/or clusters from a SW CRT.

Usage

```
swPlot(response.var, tx.var, time.var, cluster.var, data, choose.mfrow=NULL,
by.wave=TRUE, combined.plot=TRUE, choose.xlab="Time", choose.main=NULL,
choose.pch=NULL, choose.cex=1, choose.tx.col=NULL, choose.tx.lty = c(2,1),
choose.ncol=2, choose.tx.pos="topright", choose.legend.pos="right")
```

Arguments

response.var	numeric(vector): Response (Outcome) variable.
tx.var	numeric(vector): Treatment (Predictor of Interest) variable. Typically, 0=control, 1=intervention, values between 0 and 1 correspond to fractional treatment/intervention effects, and values greater than 1 correspond to other treatment options.
time.var	integer(vector): Time (points) variable, corresponding to the time points when data were collected during the SW CRT.
cluster.var	integer(vector): Cluster (identification) variable, corresponding to the cluster where an observation is from.
data	An optional data frame containing (at least) the response, treatment (tx), time, and cluster variables.
choose.mfrow	numeric (vector): Choose mfrow for plot. If NULL, mfrow is automatically assigned based on the plot created. The default is NULL.
by.wave	logical: If TRUE, plot mean response according to each wave. If FALSE, plot mean response according to each cluster. The default is TRUE.
combined.plot	logical: If TRUE, plot mean response on same plot (what is plotted depends on by.wave). If FALSE, plot mean response on separate plots for each wave (what is plotted depends on by.wave). The default is TRUE.
choose.xlab	Choose xlab for plot. The default is "Time".
choose.main	Choose main for plot. If NULL, main is chosen for the user; which is highly recommended. The default is NULL.
choose.pch	Choose pch for plot. If NULL, pch are chosen for the user; which is highly recommended. The default is NULL.
choose.cex	Choose cex for choose.pch. Standard cex option in points() applies. The default is 1.
choose.tx.col	Choose colors for different treatment options. Vector of two colors, corresponding to control and treatment groups, respectively. If NULL, colors are chosen for the unique treatment options in the data supplied. The default is NULL.
choose.tx.lty	Choose line types for different treatment options. Vector of two numbers for lty, corresponding to control and treatment groups, respectively. The default is c(2,1).
choose.ncol	Choose number of columns for non-treatment legend. Standard ncol option in legend() applies. The default is 2.
choose.tx.pos	Choose where to place treatment colors "legend". Standard legend() positions apply. The default is "topright"; if this covers points, it should be changed.

choose.legend.pos
: Choose where to place the non-treatment legend. Standard legend() positions apply. The default is "right"; if this covers points, it should be changed. If "mouseclick", user specifies location with mouse/trackpad by clicking on the plot that appears. Specify NULL to remove legend.

Details

Returns a plot of the mean response versus time with a combination of by wave (TRUE / FALSE) and combined plot (TRUE / FALSE) from a SW CRT.

Author(s)

James P Hughes, Navneet R Hakhu, and Emily C Voldal

References

Hussey MA, Hughes JP. Design and analysis of stepped wedge cluster randomized trials. Contemporary Clinical Trials 2007;28:182-191.

Examples

```
library(swCRTdesign)
# Example 1 (Generating SW CRT data)
# (binary response with 1 missing value, 5 clusters, 4 time points)

n.Ex1 <- 120
p0.Ex1 <- 0.05
clusters.Ex1 <- c(2,2,1)
dsn.Ex1 <- swDsn( clusters=clusters.Ex1 )
time.Ex1 <- c(1:dsn.Ex1$total.time)*4 - 4
response.Ex1 <- rbinom(n.Ex1 * dsn.Ex1$n.clusters * dsn.Ex1$total.time, 1, p0.Ex1)
response.Ex1[1] <- NA
tx.Ex1 <- as.vector( apply( dsn.Ex1$swDsn, 1, function(z){rep(z, n.Ex1)}) )
time.Ex1 <- rep( time.Ex1, n.Ex1 * dsn.Ex1$n.clusters )
cluster.Ex1 <- rep( 1:dsn.Ex1$n.clusters, each=n.Ex1 * dsn.Ex1$total.time )
data.Ex1 <- data.frame(response.Ex1, tx.Ex1, time.Ex1, cluster.Ex1)

# Example 1 (Mean Response vs Time, by.wave=TRUE, combined.plot=TRUE)
swPlot(response.Ex1, tx.Ex1, time.Ex1, cluster.Ex1, data.Ex1, by.wave=TRUE,
combined.plot=TRUE, choose.tx.pos="bottomright", choose.legend.pos="bottom")

# Example 2 (Mean Response vs Time, by.wave=TRUE, combined.plot=FALSE)
swPlot(response.Ex1, tx.Ex1, time.Ex1, cluster.Ex1, data.Ex1, by.wave=TRUE,
combined.plot=FALSE, choose.tx.pos="bottomright", choose.legend.pos="bottom")

# Example 3 (Mean Response vs Time, by.wave=FALSE, combined.plot=TRUE)
swPlot(response.Ex1, tx.Ex1, time.Ex1, cluster.Ex1, data.Ex1, by.wave=FALSE,
combined.plot=TRUE, choose.tx.pos="bottomright", choose.legend.pos="bottom")

# Example 4 (Mean Response vs Time, by.wave=FALSE, combined.plot=FALSE)
swPlot(response.Ex1, tx.Ex1, time.Ex1, cluster.Ex1, data.Ex1, by.wave=FALSE,
```

```
combined.plot=FALSE, choose.tx.pos="bottomright", choose.legend.pos="bottom")
```

---

swPwr *Power of Stepped Wedge Cluster Randomized Trial (SW CRT)*

Description

swPwr returns (two-sided) power of treatment effect (θ) for the specified SW CRT design using a linear models weighted least squares (WLS) approach for an immediate treatment effect (IT) model or an exposure time indicator (ETI) model (Kenny et al, 2022). The response/outcome of interest can be binomial or Gaussian distributed and is assumed to come from a model with random intercepts, random treatment effects, and random cluster-specific time effects. Variance components can be specified using either tau, eta, rho, and gamma, or icc and cac (see details). If a random intercepts only model is used (i.e., eta and gamma are 0 and n is constant over clusters and time), then the power calculation is comparable to the closed-form formula of [Hussey and Hughes, 2007]. See [Voldal et al., 2020] for more guidance. This function is also available as a Shiny app at https://swcrtdesign.shinyapps.io/stepped_wedge_power_calculation/.

Usage

```
swPwr(design, distn, n, mu0, mu1, H=NULL, sigma, tau, eta, rho, gamma,
icc, cac, alpha=0.05, retDATA=FALSE, silent=FALSE)
```

Arguments

design	list: A stepped wedge design object, typically from swDsn, that includes at least the following components: ## swDsn, swDsn.unique.clusters, clusters, n.clusters, total.time
distn	character: Distribution assumed (gaussian or binomial). Currently, 'Binomial' implies Bernoulli.
n	integer (scalar, vector, or matrix): Number of observations: (scalar) for all clusters and all time points; (vector) for each cluster at all time points; and (matrix) for each cluster at each time point, where rows correspond to clusters and columns correspond to time. n can also be used to specify a design with transition periods (e.g. in the first time period that each sequence receives treatment, no observations are collected from that sequence). Simply define n as a matrix with a sample size of 0 during every transition period.
mu0	numeric (scalar): Mean outcome in the control group. See also documentation for H, below.
mu1	numeric (scalar): Mean outcome in the treatment group. Note: Treatment effect is the difference in means $\theta = \mu_1 - \mu_0$. See also documentation for H, below.

H	numeric (vector): If NULL, then swPwr assumes an immediate, constant treatment effect (IT) model. If not NULL, then an exposure time indicator (ETI) model is assumed and H should be a vector as long as the longest treatment effect lag (typically, number of time periods minus one). H specifies the desired linear combination of exposure time treatment effects. For example, in a stepped wedge trial with 5 time periods and four exposure times, H = rep(.25,4) gives the average treatment effect over the four exposure times; H = c(0,0,.5,.5) ignores the first two periods after the intervention is introduced and averages the remaining periods. mu0 and mu1 give expected values of the linear combination of exposure time treatment effects under the null and alternative hypotheses, respectively.
sigma	numeric (scalar): Standard deviation when assuming Gaussian distribution (distn=gaussian). For binomial distribution σ^2 is automatically set to $\bar{\mu}(1-\bar{\mu})$ where $\bar{\mu} = (\mu_1 + \mu_0)/2$
tau	numeric (scalar): Standard deviation of random intercepts.
eta	numeric (scalar): Standard deviation of random treatment effects.
rho	numeric (scalar): Correlation between random intercepts and random treatment effects.
gamma	numeric (scalar): Standard deviation of random time effects.
icc	numeric (scalar): Within-period intra-cluster correlation. Can be used with CAC instead of tau, eta, rho, and gamma; see details.
cac	numeric (scalar): Cluster auto-correlation. Can be used with ICC instead of tau, eta, rho, and gamma; see details.
alpha	numeric (scalar): Two-sided statistical significance level.
retDATA	logical: if TRUE, all stored (input, intermediate, and output) values of swPwr are returned. Default value is FALSE.
silent	logical: if TRUE, hides a warning about differences in argument order between version 3.0 and prior versions. When n is not a scalar, also hides reminder about order of entries in n. Default value is FALSE.

Details

The two-sided statistical power of treatment effect ($\theta = \mu_1 - \mu_0$) is

$$Pwr(\theta) = \Phi(Z - z_{1-\alpha/2}) + \Phi(-Z - z_{1-\alpha/2})$$

where

$$Z = \frac{|\theta|}{\sqrt{Var(\hat{\theta}_{WLS})}}$$

and Φ is the cumulative distribution function of the standard normal $N(0,1)$ distribution. If H is non-NULL then the μ are assumed to be equal to Hδ where δ is a vector of exposure time treatment effects.

When eta (and rho) are 0, instead of using tau, eta, rho, and gamma, the icc and cac can be used to define the variability of the random intercepts and time effects. In this model,

$$ICC = \frac{\tau^2 + \gamma^2}{\tau^2 + \gamma^2 + \sigma^2}$$

$$CAC = \frac{\tau^2}{\tau^2 + \gamma^2}$$

Value

numeric (matrix): `swPwr` returns the power of treatment effect (θ), where the variance of treatment effect is computed by WLS.

numeric (list): `swPwr` returns all specified and computed items as objects of a list if `retDATA = TRUE`.

`design`	list: The stepped wedge design object as input.
`distn`	character: Distribution assumed (`gaussian` or `binomial`).
`n`	integer (scalar, vector, or matrix): Number of observations: (scalar) for all clusters and all time points; (vector) for each cluster at all time points; and (matrix) for each cluster at each time point, where rows correspond to clusters and columns correspond to time.
`mu0`	numeric (scalar): Mean outcome in the control group.
`mu1`	numeric (scalar): Mean outcome in intervention group. Note: treatment effect is difference in means $\theta = \mu_1 - \mu_0$.
`sigma`	numeric (scalar): Standard deviation input. For binomial distribution, sigma = NA
`tau`	numeric (scalar): Standard deviation of random intercepts.
`eta`	numeric (scalar): Standard deviation of random treatment effects.
`rho`	numeric (scalar): Correlation between random intercepts and random treatment effects.
`gamma`	numeric (scalar): Standard deviation of random time effects.
`icc`	numeric (scalar): Within-period intra-cluster correlation. Can be used with CAC instead of tau, eta, rho, and gamma; see details.
`cac`	numeric (scalar): Cluster auto-correlation. Can be used with ICC instead of tau, eta, rho, and gamma; see details.
`alpha`	numeric (scalar): Statistical significance level.
`Xmat`	numeric (matrix): Design matrix for this SW CRT design.
`Wmat`	numeric (matrix): Covariance matrix for this SW CRT design.
`var.theta.WLS`	numeric (scalar): Variance estimate of θ using weighted least squares (WLS) for this SW CRT design.
`pwrWLS`	numeric (scalar): Power of treatment effect (θ) using weighted least squares (WLS) for this SW CRT design.
`pwrCLOSED`	numeric (scalar): Power of treatment effect (θ) using closed-form formula from Hughes, et al. (2003) for this SW CRT design. Returned if eta and gamma are 0 and n is constant over clusters and time.

Author(s)

James P Hughes, Navneet R Hakhu, and Emily C Voldal

References

Hussey MA, Hughes JP. Design and analysis of stepped wedge cluster randomized trials. Contemporary Clinical Trials 2007;28:182-191.

Kenny A, Voldal E, Xia F, Heagerty PJ, Hughes JP. Analysis of stepped wedge cluster randomized trials in the presence of a time-varying treatment effect. Statistics in Medicine, in press, 2022.

Voldal EC, Hakhu NR, Xia F, Heagerty PJ, Hughes JP. swCRTdesign: An R package for stepped wedge trial design and analysis. Computer Methods and Programs in Biomedicine 2020;196:105514.

Examples

```
library(swCRTdesign)
# Example 1 (Random Intercepts Only, standard Stepped Wedge (SW) design)
swPwr.Ex1.RIO.std <- swPwr(swDsn(c(6,6,6,6)), distn="binomial",
n=120, mu0=0.05, mu1=0.035, tau=0.01, eta=0, rho=0, gamma=0, alpha=0.05, retDATA=FALSE)
swPwr.Ex1.RIO.std

# Example 2 (Random Intercepts Only, extended SW design)
swPwr.Ex1.RIO.extend <- swPwr(swDsn(c(6,6,6,6), extra.time=3), distn="binomial",
n=120, mu0=0.05, mu1=0.035, tau=0.01, eta=0, rho=0, gamma=0,
alpha=0.05, retDATA=FALSE)
swPwr.Ex1.RIO.extend

# Example 3 (Independent Random Intercepts and Treatment effects, standard SW design)
swPwr.Ex1.IRIS <- swPwr(swDsn(c(6,6,6,6)), distn="binomial",
n=120, mu0=0.05, mu1=0.035, tau=0.01, eta=0.0045, rho=0, gamma=0,
alpha=0.05, retDATA=FALSE)
swPwr.Ex1.IRIS

# Example 4 (Correlated Random Intercepts and Slopes, standard SW design)
swPwr.Ex1.CRIS <- swPwr(swDsn(c(6,6,6,6)), distn="binomial",
n=120, mu0=0.05, mu1=0.035, tau=0.01, eta=0.0045, rho=0.4, gamma=0,
alpha=0.05, retDATA=FALSE)
swPwr.Ex1.CRIS

# Example 5 (Random time effect and correlated Random Intercepts and Slopes, standard SW design)
swPwr.Ex1.RTCRIS <- swPwr(swDsn(c(6,6,6,6)), distn="binomial",
n=120, mu0=0.05, mu1=0.035, tau=0.01, eta=0.0045, rho=0.4, gamma = 0.1,
alpha=0.05, retDATA=FALSE)
swPwr.Ex1.RTCRIS

#Example 6 (Sample size varying by cluster)
sample.size.vector <- c(35219,53535,63785,456132,128670,96673,
           51454,156667,127440,68615,56502,17719,
           75931,58655,52874,75936)
swPwr.Ex1.vector <- swPwr(swDsn(c(4,3,5,4)), distn="gaussian",
n=sample.size.vector, mu0=2.66, mu1=2.15,
sigma=sqrt(1/2.66), tau=0.31, eta=0.2, rho=0, gamma = 0.15,
alpha=0.05, retDATA=FALSE)
swPwr.Ex1.vector

#Example 7 (Sample size varying by cluster and time)
```

```
sample.size.matrix <- matrix(c(26, 493, 64, 45, 48, 231, 117, 17, 49, 36, 19, 77, 67, 590,
261, 212, 67, 318, 132, 58, 44, 57, 59, 78, 115, 532, 176, 199, 73, 293, 129, 79, 51,
62, 109, 94, 174, 785, 133, 79, 120, 305, 224, 99, 83, 79, 122, 122, 94, 961, 90, 131, 166,
352, 316, 59, 54, 131, 101, 133),nrow=12,ncol=5, byrow=FALSE)
swPwr.Ex1.matrix <- swPwr(swDsn(c(3,3,3,3)), distn="binomial",
n=sample.size.matrix, mu0=0.08, mu1=0.06, tau=0.017, eta=0.006, rho=-0.5, gamma = 0,
alpha=0.05, retDATA=FALSE)
swPwr.Ex1.matrix

#Example 8 (Using ICC and CAC)
swPwr.Ex1.icccac <- swPwr(swDsn(c(6,6,6,6)), distn="gaussian",
n=120, mu0=0.05, mu1=0.035, sigma=0.1, icc=0.02, cac=0.125, alpha=0.05, retDATA=FALSE)
swPwr.Ex1.icccac
```

`swSim` *Simulating individual-level data for specified study design of Stepped Wedge Cluster Randomized Trial (SW CRT)*

Description

`swSim` returns individual-level data set of a SW CRT study design for the specified number of clusters per wave, fractional treatment effect at each time after crossing over from control, time (standard SW CRT time computed; extending trial beyond standard time needs to be specified by the user), family (and link function), number of individuals per cluster per wave, mean in control arm, mean in treatment arm, time effect, pooled standard deviation for both arms, standard deviation of random intercepts, standard deviation of random treatment effects, correlation between random intercepts and random treatment effects, standard deviation of random time effects, time point label, and option to simulate data with time on treatment lag. Alternatively, for a Gaussian family standard deviations of random effects may be specified using ICC and CAC; see `swPwr` details.

Usage

```
swSim(design, family, log.gaussian = FALSE, n, mu0, mu1, time.effect, sigma, tau, eta,
rho, gamma, icc, cac, time.lab = NULL, retTimeOnTx = FALSE, silent = FALSE)
```

Arguments

`design`	list: A stepped wedge design object, typically from swDsn, that includes at least the following components: ## swDsn, clusters, n.clusters, total.time
`family`	character: Used in typical way. However, only Gaussian, Binomial, and Poisson families accepted. Also, only identity, logit, and log links accepted. Logit link is only available for Binomial family, and log link is only available for Binomial and Poisson. Currently, 'Binomial' implies Bernoulli. ***NOTE: It is the users responsibility to make sure specified parameters (mu0, mu1, time.effect, tau, eta, rho, gamma) are ALL on SAME scale as specified link function; see example.***

log.gaussian	character: When TRUE with a Gaussian family, simulates data whose log follows a Gaussian distribution; all parameters (mu0, mu1, time.effect, variance parameters) refer to the log scale. Default is FALSE.
n	integer (scalar, vector, or matrix): Number of observations: (scalar) for all clusters and all time points; (vector) for each cluster at all time points; and (matrix) for each cluster at each time point, where rows correspond to clusters and columns correspond to time. n can also be used to specify a design with transition periods were no data is collected; see swPwr.
mu0	numeric (scalar): Mean outcome in the control group on the appropriate scale.
mu1	numeric (scalar): Mean outcome in the treatment group on the appropriate scale.
time.effect	integer (scalar or vector): Time effect at each time point on the appropriate scale (added to mean at each time).
sigma	numeric (scalar): Pooled treatment and control arm standard deviation on the appropriate scale. Ignored if family != Gaussian.
tau	numeric (scalar): Standard deviation of random intercepts on the appropriate scale.
eta	numeric (scalar): Standard deviation of random treatment effects on the appropriate scale.
rho	numeric (scalar): Correlation between random intercepts and random treatment effects on the appropriate scale.
gamma	numeric (scalar): Standard deviation of random time effects on the appropriate scale.
icc	numeric (scalar): Within-period intra-cluster correlation on the appropriate scale. Can be used with CAC instead of tau, eta, rho, and gamma when the outcome is Gaussian.
cac	numeric (scalar): Cluster auto-correlation on the appropriate scale. Can be used with ICC instead of tau, eta, rho, and gamma when the outcome is Gaussian.
time.lab	character (vector): Labeling for time points when output is display; choice of labeling does not affect results.
retTimeOnTx	logical: If TRUE, outputs time on treatment variable (timeOnTx.var) in addition to the usual treatment variable (tx.var). To simulate data with a time on treatment lag effect, the fractional treatment effect can be specified through design. The default is FALSE.
silent	logical: if TRUE, hides reminder about order of entries in n when n is not a scalar. Default value is FALSE.

Value

numeric (data frame): Returns the following (individual-level) variables corresponding to the specified SW CRT design:

$response.var	numeric (vector): Response variable based on specified SW CRT design of interest (including family and link function) for each observation in the data frame/set.

`$tx.var`	numeric (vector): Predictor of interest. (Fractional) treatment effect corresponding to 0=control, 1=treatment, and value between 0 and 1 corresponding to treatment arm with fractional treatment effect (for each observation in the data frame/set).
`$timeOnTx.var`	numeric (vector): Predictor of interest when interested in time on treatment lag effect. Total time spent on treatment for each observation in the data frame/set, with 0=control, 1=first time period on treatment, 2=second time period on treatment, etc.
`$time.var`	numeric (vector): Time point id for each observation in the data frame/set.
`$cluster.var`	numeric (vector): Grouping variable. Cluster id for each observation in the data frame/set.

Author(s)

James P Hughes, Navneet R Hakhu, and Emily C Voldal

References

Hussey MA, Hughes JP. Design and analysis of stepped wedge cluster randomized trials. Contemporary Clinical Trials 2007;28:182-191.

Examples

```
library(swCRTdesign)
# Example 1 [ n = scalar; can be vector (for different n for each cluster,
# n=rep(120,22)) or matrix (different n for each cluster at each time point,
# n=matrix(120,22,5)) ]

# generate SW data (fractional treatment effect)
design <- swDsn(clusters=c(6,6,6,4), tx.effect.frac=c(0.8,0.9,1.0),
extra.time=0, all.ctl.time0=TRUE)
set.seed(5)
swGenData.nScalar <- swSim( design,
family=binomial(link="logit"), n=120,
mu0=log(0.1/0.9), mu1=log(0.9) + log(0.1/0.9),
time.effect=0, tau=0.2, eta=0,
rho=0, gamma=0, time.lab=seq(0,12,3), retTimeOnTx=FALSE)

# summarize SW data by wave
swSummary(response.var, tx.var, time.var, cluster.var, swGenData.nScalar,
type="mean", digits=3)$response.wave

swSummary(response.var, tx.var, time.var, cluster.var, swGenData.nScalar,
type="mean", digits=3)$swDsn
```

swSummary *Summary of Response/Outcome for Stepped Wedge Cluster Randomized Trial (SW CRT)*

Description

swSummary returns the mean, sum, and number of non-missing values for the response/outcome variable of interest for each cluster at each time point from a SW CRT.

Usage

```
swSummary(response.var, tx.var, time.var, cluster.var, data,
type="mean", digits=16, fcn.Call=FALSE)
```

Arguments

response.var	numeric(vector): Response (Outcome) variable.
tx.var	numeric(vector): Treatment (Predictor of Interest) variable. Typically, 0=placebo, 1=intervention, values between 0 and 1 correspond to fractional treatment/intervention effects, and values greater than 1 correspond to other treatment options.
time.var	integer(vector): Time (points) variable, corresponding to the time points when data were collected during the SW CRT.
cluster.var	integer(vector): Cluster (identification) variable, corresponding to the cluster where an individual is from.
data	An optional data frame containing (at least) the response, treatment (tx), time, and cluster variables.
type	character (scalar): Specify which summary measure is of interest from "mean", "sum", and "n". (Note: The default returns "mean" as the summary measure of response. Note that all summary measures may not be scientifically relevant in every situation.)
digits	integer (scalar): Number of places right of the decimal. The default value is 16.
fcn.Call	logical: Only TRUE when calling swSummary from within swPlot. The default is FALSE.

Details

Returns a list containing a matrix of dimension length(unique(data$cluster)) by length(unique(data$time)) that summarizes data$response for specified type. Either the mean, sum, or the number of non-missing data$response values may be requested using type. dimnames of the matrix correspond to the unique values of cluster and time. Note that the stepping pattern in the data may be obtained by specifying the treatment variable name as the response and type = "mean".

Value

numeric (list): swSummary returns a list containing the following

type	One of user-specified options "mean", "sum", or "n".
swDsn	The SW design.
swDsn.unique.clusters	The unique clusters (i.e., rows) SW design.
n.waves	Number of waves.
clusters	Clusters per wave.
n.clusters	Total number of clusters.
time.at.each.wave	Time at each wave.
total.time	Total time points.
response.cluster	numeric (matrix): Response variable summarized according to type for all clusters, with dimension length(data$cluster) by length(unique(data$time)).
response.wave	numeric (matrix): Response variable summarized according to type, for all waves (all clusters of a particular wave are combined), with dimension length(unique(data$cluster)) by length(unique(data$time)).

Author(s)

James P Hughes and Navneet R Hakhu

References

Hussey MA, Hughes JP. Design and analysis of stepped wedge cluster randomized trials. Contemporary Clinical Trials 2007;28:182-191.

Examples

```
library(swCRTdesign)
# Example 1 (Generating SW CRT data)
# (binary response with 1 missing value, 5 clusters, 4 time points)

n.Ex1 <- 120
p0.Ex1 <- 0.05
clusters.Ex1 <- c(2,2,1)
dsn.Ex1 <- swDsn( clusters=clusters.Ex1 )
time.Ex1 <- c(1:dsn.Ex1$total.time)*4 - 4
response.Ex1 <- rbinom(n.Ex1 * dsn.Ex1$n.clusters * dsn.Ex1$total.time, 1, p0.Ex1)
response.Ex1[1] <- NA
tx.Ex1 <- as.vector( apply( dsn.Ex1$swDsn, 1, function(z){rep(z, n.Ex1)}) )
time.Ex1 <- rep( time.Ex1, n.Ex1 * dsn.Ex1$n.clusters )
cluster.Ex1 <- rep( 1:dsn.Ex1$n.clusters, each=n.Ex1 * dsn.Ex1$total.time )
data.Ex1 <- data.frame(response.Ex1, tx.Ex1, time.Ex1, cluster.Ex1)

## Example 1 (type="mean", by cluster and by wave)
```

```
swSummary.Ex1.mean <- swSummary(response.Ex1, tx.Ex1, time.Ex1, cluster.Ex1,
data=data.Ex1, type="mean", digits=3)

swSummary.Ex1.mean$response.cluster

swSummary.Ex1.mean$response.wave

# Example 1 (type="sum", by cluster and by wave)
swSummary.Ex1.sum <- swSummary(response.Ex1, tx.Ex1, time.Ex1, cluster.Ex1,
data=data.Ex1, type="sum")

swSummary.Ex1.sum$response.cluster

swSummary.Ex1.sum$response.wave

## Example 1 (type="n", by cluster and by wave)
swSummary.Ex1.n <- swSummary(response.Ex1, tx.Ex1, time.Ex1, cluster.Ex1,
data=data.Ex1, type="n")

swSummary.Ex1.n$response.cluster

swSummary.Ex1.n$response.wave
```

Index

∗ **block diagonal matrix**
  blkDiag, 3
∗ **block diagonal**
  swCRTdesign-package, 2
∗ **cluster randomized trial**
  swCRTdesign-package, 2
  swDsn, 4
  swGlmPwr, 6
  swPlot, 9
  swPwr, 12
  swSim, 16
  swSummary, 19
∗ **design**
  swCRTdesign-package, 2
  swDsn, 4
  swGlmPwr, 6
  swSim, 16
∗ **discrete outcomes**
  swGlmPwr, 6
∗ **plot**
  swCRTdesign-package, 2
  swPlot, 9
∗ **power**
  swCRTdesign-package, 2
  swGlmPwr, 6
  swPwr, 12
∗ **repeated cross-sectional sampling**
  swCRTdesign-package, 2
∗ **simulated data set**
  swCRTdesign-package, 2
∗ **stepped wedge**
  swCRTdesign-package, 2
  swDsn, 4
  swGlmPwr, 6
  swPlot, 9
  swPwr, 12
  swSim, 16
  swSummary, 19
∗ **summary**
  swCRTdesign-package, 2
  swSummary, 19

blkDiag, 3

swCRTdesign *(*swCRTdesign-package*)*, 2
swCRTdesign-package, 2
swDsn, 4
swGlmPwr, 6
swPlot, 9
swPwr, 12, *16*
swSim, 16
swSummary, 19