

Package ‘gsearly’

January 23, 2026

Type Package

Title Creates Group Sequential Trial Designs when Early Outcomes are Available

Version 1.0.0

Maintainer Nick Parsons <nick.parsons@warwick.ac.uk>

Description Methods to construct and power group sequential clinical trial designs for outcomes at multiple times. Outcomes at earlier times provide information on the final (primary) outcome. A range of recruitment and correlation models are available as are methods to simulate data in order to explore design operating characteristics. For more details see Parsons (2024) <https://doi.org/10.1186/s12874-024-02174-w>.

Depends R (>= 4.1.2)

Imports graphics, grDevices, gsDesign (>= 3.2.1), methods, mvtnorm (>= 1.2-4), nlme (>= 3.1-160), stats, utils

License GPL (>= 2)

Encoding UTF-8

LazyData true

NeedsCompilation no

Author Nick Parsons [aut, cre] (ORCID: <https://orcid.org/0000-0001-9975-888X>)

Repository CRAN

Date/Publication 2026-01-23 21:20:12 UTC

Contents

gsearly-package	2
corrExp	3
corrUnif	4
dataOrder	5
expectSampsize	5
fixedSampsize	6

gsearlyFit	7
gsearlyModel	8
gsearlySimulate	11
gsearlyUser	13
modelParameters	15
plot.gsearly	16
plotBoundary	17
plotInform	18
plotPower	19
plotRecruit	21
print.gsearly	22
print.summary.gsearly	23
qol	24
roundInterims	24
simdataExtract	25
summary.gsearly	27
tfuStandard	27

Index	29
--------------	-----------

gsearly-package *Group sequential designs with early outcomes*

Description

Functions to implement group sequential clinical trial designs with early outcomes. Group sequential designs are one of the most widely used methodologies for adaptive design in randomized clinical trials. In such designs researchers collect data and undertake sequential analyses with the opportunity to either reject the null hypothesis, stop the study for futility or continue recruitment at an interim look, before reaching the planned sample size.

In situations where, for instance, outcomes are collected at long follow-up time-points, data at interim analyses are often available for not only the study primary (long-term) outcome time-point but also from early time-points for the same outcome (early outcomes); e.g. a primary outcome at 12 months and early outcomes at 3, 6 and 9 months. In settings where moderate to strong correlations exist between the sequence of such outcomes, information can be used from the early outcomes in addition to the final outcome at the interim analyses. The design, planning and sample size determination for such studies is more complex than for conventional group sequential designs and is generally achieved by simulating individual participant data for an assumed recruitment pattern as a means to determine information accrual during a proposed trial.

However, in practice, such simulations are complex and time-consuming to set-up and implement and provide a barrier to the use of group sequential designs. If we can assume approximate multivariate Normality for the distribution of the outcomes, and also make some assumptions about the expected correlation structure and recruitment patterns, then we can derive relatively simple analytic expressions for information accrual during a planned trial. Allowing a range of design options to be explored routinely without the burden of undertaking extensive simulation studies.

Details

The two main functions (i) `gsearlyModel` and (ii) `gsearlyUser` allow designs to be constructed based on a range of typical clinical trial recruitment patterns and correlation models. The function `gsearlySimulate` simulates multivariate Normal datasets based on a previously fitted `gsearly` model.

References

Parsons NR, Basu J, Stallard N. Group sequential designs for pragmatic clinical trials with early outcomes: methods and guidance for planning and implementation. *BMC Medical Research Methodology* 2024; 24:42. <https://wrap.warwick.ac.uk/id/eprint/183449/>

Parsons NR, Stallard N, Parsons H, Haque A, Underwood M, Mason J, Khan I, Costa ML, Griffin DR, Griffin J, Beard DJ, Cook JA, Davies L, Hudson J, Metcalfe A. Group sequential designs in pragmatic trials: feasibility and assessment of utility using data from a number of recent surgical RCTs. *BMC Medical Research Methodology* 2022; 22:256. <https://wrap.warwick.ac.uk/id/eprint/169801/>

See Also

`gsearlyModel`, `gsearlyUser`, `gsearlySimulate`

corrExp

Exponential correlation matrix

Description

Constructs an exponential correlation matrix.

Usage

```
corrExp(rho=0, tfu)
```

Arguments

<code>rho</code>	Correlation parameter.
<code>tfu</code>	Follow-up time points, in standardized format from function <code>tfuStandard</code> .

Value

Returns an $s \times s$ correlation matrix. Where s is the number of occasions (follow-up time points) at which the study outcome is observed.

Examples

```
# Exponential correlation model for outcomes at 3, 6, 12 and 24 months
# Based on correlation 0.5 between outcomes separated by 3 month
corrExp(0.5, tfuStandard(c(3,6,12,24), tref=c(1,2)))

# Based on correlation 0.25 between outcomes separated by 1 year
corrExp(0.25, tfuStandard(c(3,6,12,24), tref=c(3,4)))
```

corrUnif

Uniform correlation matrix

Description

Constructs a uniform correlation matrix.

Usage

```
corrUnif(rho=0, tfu)
```

Arguments

- | | |
|-----|---|
| rho | Correlation parameter. |
| tfu | Follow-up time points, in standardized format from function tfuStandard . |

Value

Returns an $s \times s$ correlation matrix. Where s is the number of occasions (follow-up time points) at which the study outcome is observed.

Examples

```
# Uniform correlation model for outcomes at 3, 6, 12 and 24 months
# Settings for tref argument of standardtfu do not change correlation matrix
corrUnif(0.5, tfuStandard(c(3,6,12,24), tref=c(1,2)))

# Based on correlation 0.5 between outcomes separated by 1 year
corrUnif(0.5, tfuStandard(c(3,6,12,24), tref=c(3,4)))
```

dataOrder*Order a data frame by subject and ordered time variable*

Description

Orders a user supplied data frame by subject and ordered time variable.

Usage

```
dataOrder(data, datanames=c("id", "atime", "catime",
                            "intervention", "outcome"))
```

Arguments

- | | |
|-----------|--|
| data | A data frame structured as those from function simdataExtract . |
| datanames | Names of the five required data variables; participant identifier, time-point, standardized (continuous) time-point (see tfuStandard), intervention arm and outcome, in that order; e.g. c("id", "atime", "catime", "intervention", "outcome"). |

Value

Returns an ordered data frame.

Examples

```
data(qol)
qol <- dataOrder(qol, datanames=c("Subject", "Weeks", "STime", "Treat", "QoL"))
head(qol, n=50)
```

expectSampszie*Expected sample size for a gsearly model*

Description

Calculates the expected sample size for a fitted gsearly model, from functions [gsearlyModel](#) or [gsearlyUser](#).

Usage

```
expectSampszie(mod, signif=3)
```

Arguments

- | | |
|--------|---|
| mod | A gsearly model. |
| signif | Rounds the sample size to the specified number of significant digits. |

Value

Returns the sample size for the control and treatment groups and total.

See Also

[gsearlyModel](#), [gsearlyUser](#)

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0010, 0.0100, 0.0250)
tn <- c(0.4800, 0.7200, 0.9750)
rctdesign <- gsearlyModel(rmodel="fix", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8, cmodel="exponential", sd=20,
                           rho=0.75, theta=8, fp=fp, tn=tn)
expectSampszie(rctdesign)
```

fixedSampszie

Fixed model sample size for a gsearly model

Description

Calculates the fixed model sample size, assuming no correlation between early and final outcomes, for a fitted gsearly model, from functions [gsearlyModel](#) or [gsearlyUser](#).

Usage

```
fixedSampszie(mod, pow=NULL, direct="u", signif=3)
```

Arguments

mod	A gsearly model.
pow	Power is set to a value between 0 and 1, or if unset is taken from mod.
direct	Rounds sample sizes to nearest integer, upwards "u" (using ceiling) or downwards "d" (using floor)
signif	Rounds power to the specified number of significant digits.

Value

Returns a list with the following items.

n	Sample size for the control and treatment groups and total.
power	Study power.

See Also

[gsearlyModel](#), [gsearlyUser](#)

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000,0.0010,0.0250)
tn <- c(0.4800,0.7200,0.9750)
rctdesign <- gsearlyModel(rmodel="fix", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8, cmodel="exponential", sd=20,
                           rho=0.75, theta=8, fp=fp, tn=tn)
fixedSampsize(rctdesign)
```

gsearlyFit

Fit a generalized least squares model

Description

Fits a generalized least squares model using function `gls` (from package `nlme`)

Usage

```
gsearlyFit(data, datanames = c("id","atime","catime",
                               "intervention","outcome"), cmodel=FALSE, vmodel=TRUE, full=FALSE)
```

Arguments

<code>data</code>	A data frame structured as function <code>simdataExtract</code> .
<code>datanames</code>	Names of the five required data variables; participant, time-point, standardized (continuous) time-point (see <code>tfuStandard</code>), intervention arm and outcome, in that order; e.g. <code>c("id","atime","catime","intervention","outcome")</code>).
<code>cmodel</code>	The correlation model is set to FALSE, for an unstructured model or to either "uniform" or "exponential", with default FALSE.
<code>vmodel</code>	The variance model is set to either FALSE or TRUE, where the latter allows variances to vary by assessment occasion; default is <code>vmodel=TRUE</code> .
<code>full</code>	Either FALSE, which provides model details and parameters only or TRUE which provides full <code>gls</code> model details.

Value

Either a `gls` model (`full=TRUE`) or a summary of the model fit (`full=FALSE`), which is a list with the components.

<code>parameters</code>	Estimates of variance of beta (<code>vbeta</code>), beta and z (<code>z=beta/sqrt(vbeta)</code>).
<code>model</code>	A list comprised of data sample size <code>N</code> , number of estimated <code>gls</code> model parameters, estimated covariance (<code>vcovmat</code>), correlation (<code>corrmat</code>) and standard deviation (<code>sdmat</code>) matrices.

See Also

[gsearlySimulate](#), [simdataExtract](#), [modelParameters](#)

Examples

```
# Model for full data set
data(qol)
mod.fulldata <- nlme::gls(QoL~(Weeks-1)+(Weeks-1):Treat, data=qol,
  correlation=nlme::corSymm(form=~1|Subject),
  weights=nlme::varIdent(form =~1|Weeks), method="ML")
summary(mod.fulldata)

# Use data available at day 150
data.interim1 <- qol[qol$Day<=150,]
mod.interim1 <- nlme::gls(QoL~(Weeks-1)+(Weeks-1):Treat, data=data.interim1,
  correlation=nlme::corSymm(form=~1|Subject),
  weights=nlme::varIdent(form=~1|Weeks), method="ML")
summary(mod.interim1)

# Use glsFit
gsearlyFit(qol, datanames=c("Subject", "Weeks", "STime", "Treat", "QoL"))
# vbeta
vcov(mod.fulldata)[6,6]
# beta
coef(mod.fulldata)[6]
```

gsearlyModel

Model based estimates of sample size and power for group sequential designs with early outcomes

Description

Provides sample sizes and power for group sequential designs with early outcomes defined by a recruitment and correlation model, recruitment period and interim analysis time-points.

Usage

```
gsearlyModel(rmodel="fix", trecruit, s, tfu, tinterims, pow=0.9, n=NULL,
  theta, tref=c(1,2), vphi=0.5, m=2, cmodel="uniform", sd=1, rho=0.5,
  fp, tn, treatnames=c("control", "treat"),
  sopt=list(r=18, bisect=list(min=20, max=10000, niter=1000, tol=0.001)))
```

Arguments

rmodel	Recruitment model, options are c("fix", "dlin", "ilin", "dquad", "iquad", "dilin", "idlin", "diquad", "idquad", "filin", "fdlin", "ilinf", "dlinf") with default="fix".
--------	---

trecruit	The recruitment period (an integer >1) which is used to define the base units for all analyses and plotting functions (which are on a scale of 1:trecruit) and is usually the smallest practical study time unit; e.g. if trecruit=30 months, then tinterims and tfu should also be set in units of months.
s	The study outcome is observed at a sequence of s occasions, with s - 1 early outcomes. An integer value, 2 <= s <= 10, the upper limit is purely for reasons of practical implementation.
tfu	A vector of length s, of follow-up time points, in the same units as trecruit.
tinterims	A vector of ordered numeric interim analysis time-points, in the same units as trecruit.
pow	The study target power. If n is set, then this is ignored and the study power is reported.
n	The total study sample size (control + treatment groups). If this is set to NULL then the bisection algorithm is used to find n to give power pow.
theta	The treatment effect, in the same units as sd. Current implementation forces a positive value for theta.
tref	A vector of integers of length 2 that defines the correlation parameter rho for the exponential model; the parameter rho is the correlation between data at times tfu[tref[1]] and tfu[tref[2]], with default c(1, 2); see tfuStandard .
vphi	A weight for unequal group sizes $0 < vphi < 1$ where the ratio of control to treatment group participants is vphi / (1 - vphi), with default of 0.5 (equal group sizes).
m	A parameter ($1 \leq m < \text{Inf}$) that sets the recruitment model breakpoint for those models that have two phases (e.g. "diquad" or "dilin"). Ignored for models that do not have two phases.
cmodel	The correlation model is set to either "uniform" or "exponential"; see corrExp , corrUnif .
sd	The standard deviation ($0 < sd < \text{Inf}$) of the outcome, that is assumed to be the same in both arms and at each time-point.
rho	The correlation parameter ($0 \leq rho < 1$) associated with correlation model cmodel.
fp	A vector (of length tinterims + 1) of false positive rates; see Examples.
tn	A vector (of length tinterims + 1) of true negative rates; see Examples.
treatnames	Names used for the two study groups, referred to elsewhere as the first (or 0) and second (or 1) groups, with default c("control", "treat").
sopt	List of settings for the gsBound and gsProbability functions in the gsDesign package and settings for the bisection algorithm with default list(r=18, bisect=list(min=20, max=10000, niter=1000, tol=0.001)). The min, max, niter and tol settings represent the minimum and maximum starting points, maximum number of iterations and tolerance for the bisection algorithm.

Value

An object of class `gsearly` is a list containing the following components.

<code>title</code>	Package title and version number.
<code>call</code>	Call to <code>gsearlyModel</code> .
<code>rdata</code>	A list of the recruitment model (<code>rmodel</code>), the recruitment period (<code>trecruit</code>), the number of follow-up time-points (<code>s</code>), the follow-up time points (<code>tfu</code>), on the scale of <code>trecruit</code> (<code>tfu\$tfu</code>) and on the standardized scale (<code>tfu\$stfu</code>), the study sample size (<code>n</code>) for the control and treatment groups and the total (control, treat, total), the weight for the unequal group sizes (<code>vphi</code>), the recruitment model parameter (<code>m</code>), the interims analyses time-points (<code>tinterims</code>), in the units of <code>trecruit</code> , and the numbers of study participants recruited and with outcome data at each of the interims (<code>interims</code>).
<code>idata</code>	A list of the correlation model (<code>cmodel</code>), consisting of the correlation model type (<code>cmodel\$type</code>), parameter (<code>cmodel\$rho</code>) and correlation matrix (<code>cmodel\$corrrmat</code>), the standard deviation (<code>sd</code>), interim and final analysis times (<code>tinterims</code>) and the information fraction (<code>tau</code>) and variance of the treatment effect estimate (<code>vbeta</code>) at times <code>tinterims</code> .
<code>power</code>	A list of the total number of the looks (<code>nlooks</code>), treatment effect (<code>theta</code>), the target power (<code>setpow</code>), the false positive (<code>fp</code>) and true negative rates (<code>tn</code>), the lower (<code>loweror</code>) and upper (<code>upperor</code>) boundaries and stopping probabilities (<code>bound</code> , <code>prob</code>) for each look and the total stopping probabilities (<code>totalerror</code>) summed across looks (<code>futility</code> , <code>efficacy</code>).

See Also

`expectSampsize`, `fixedSampsize`, `roundInterims`, `corrExp`, `corrUnif`

Examples

```
# RCT planning:

# Recruitment is expected to take 36 months (trecruit) with three
# follow-up times (s) for outcomes at 3, 6 and 12 months (tfu) and
# interim analyses (tinterims) at 18 and 30 months.
# Study procedures and experience suggest that "dilin" (m=2) is likely
# to be a good approximation to temporal patterns of recruitment and data
# accrual (rmodel). Correlations between outcomes at 3, 6 and 12 are
# assumed to follow a uniform model (cmodel) with parameter (rho) set to 0.5.
# Randomisation to the two study arms will be on a 1:1 basis (vphi=0.5).
# The standard deviation of the final 12m outcome is expected to be 20,
# and the anticipated treatment effect (theta) 8.

# Primary interest for the RCT is in stopping early for futility.
# Therefore the following values for the false positive and true
# negative rates were selected.
# Vectors of cumulative probabilities for crossing boundaries under the
# null hypothesis (of no treatment difference).
fp <- c(0.0000, 0.0010, 0.0250)
```

```

tn <- c(0.2400,0.7200,0.9750)

# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
rctdesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(18,30), pow=0.9, vphi=0.5, m=2,
                           cmodel="uniform", sd=20, rho=0.5, theta=8, fp=fp, tn=tn)
rctdesign

# If n is set, rather than pow, then we can explore power for lower n
update(rctdesign, n=136)

# Expected numbers of participants at interim analyses
rctdesign$rdata$interims

# Information at these interims and final analysis
rctdesign$idata$interims

# Upper and lower stopping boundaries and probabilities
rctdesign$power$lowererror
rctdesign$power$uppererror

```

gsearlySimulate*Simulates data for a previously fitted gsearly model***Description**

Simulates multivariate Normal datasets based on a previously fitted gsearly model (see [gsearlyModel](#)).

Usage

```
gsearlySimulate(mod, nsim=1, minsamp=c(1,1), mean=NULL, cmodel=NULL,
                 sd=NULL, rho=NULL, full=FALSE)
```

Arguments

<code>mod</code>	A previously fitted model from function gsearlyModel .
<code>nsim</code>	Number of required simulated datasets.
<code>minsamp</code>	Simulations where numbers at the first interim are less than <code>minsamp</code> , a vector of length 2 giving the minimum study sample size for the control and treatment groups, are excluded.
<code>mean</code>	A matrix of dimension $2 \times s$ giving mean values for the control (first row) and treatment (second row) groups for each of the s outcome time-points. If unset, these are taken directly from <code>mod</code> , where the control group mean is set to zero and the treatment group mean to <code>theta</code> for the primary outcome time-point with early means all set to zero (see gsearlyModel).

<code>cmodel</code>	Either a correlation model, <code>c("uniform", "exponential")</code> , defined by correlation parameter <code>rho</code> , or a correlation matrix of dimensions <code>s x s</code> . If unset, taken from <code>mod</code> .
<code>sd</code>	The standard deviation of the outcome, that is assumed to be the same in both arms. Either a single value or a vector of length <code>s</code> , allowing <code>sd</code> to differ at each time-point. If unset, taken from <code>mod</code> .
<code>rho</code>	Correlation parameter if <code>cmodel</code> is set to <code>"uniform"</code> or <code>"exponential"</code> , otherwise ignored. If unset, taken from <code>mod</code> .
<code>full</code>	Either <code>FALSE</code> , which provides model details and parameters only or <code>TRUE</code> which provides full details of numbers and datasets in addition to the model details.

Value

A list containing the following components.

<code>model</code>	A list of model settings <code>nsim</code> , <code>s</code> , <code>tinterims</code> , <code>tfu</code> and <code>mean</code> , covariance and correlation matrices.
<code>parameters</code>	A list of two matrices, comprising variances of beta (<code>vbeta</code>) and beta at each interim and final analysis.
<code>n</code>	A list of length 2 (control and treatment), each comprising of lists of matrices (for recruitment and follow-up times) showing sample sizes at each interim and final analysis for the <code>nsim</code> simulations.
<code>data</code>	A list of length 2 (control and treatment), each comprising of a list of length <code>nsim</code> comprising a list of matrices of outcome data (if <code>full=TRUE</code>) at interims and final analyses.

See Also

[gsearlyModel](#), [simdataExtract](#), [modelParameters](#), [gsearlyFit](#)

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000, 0.0010, 0.0250)
tn <- c(0.2400, 0.7200, 0.9750)
modeldesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(18,30), pow=0.9, vphi=0.5, m=2,
                           cmodel="uniform", sd=20, rho=0.5, theta=8, fp=fp, tn=tn)

# Simulate data from this model
simdata <- gsearlySimulate(mod=modeldesign, nsim=10, full=FALSE)
# Model parameters
simdata$parameters

# Change correlation model
newsimdata <- gsearlySimulate(mod=modeldesign, nsim=10,
                               cmodel="exponential", rho=0.75, full=FALSE)
# Model parameters
```

```
newsimdata$parameters
```

gsearlyUser

User input estimates of sample size and power for group sequential designs with early outcomes

Description

Provides sample sizes and power for group sequential designs with early outcomes defined by a matrix of fixed numbers of participants with data for all outcomes (early and primary) at each interim time-point or a function that gives a similarly structured matrix for all sample sizes in a pre-set range and a correlation model or correlation matrix, recruitment period and interim analysis time-points.

Usage

```
gsearlyUser(trecruit, s, tfu, tinterims, ninterims, pow=0.9, n=NULL,
            tref=c(1,2), vphi=0.5, cmodel="uniform", sd=1, rho=0.5, theta, fp,
            tn, treatnames=c("control","treat"),
            sopt=list(r=18, bisect=list(min=20, max=10000, niter=1000, tol=0.001)))
```

Arguments

trecruit	As for gsearlyModel .
s	As for gsearlyModel .
tfu	As for gsearlyModel .
tinterims	As for gsearlyModel .
ninterims	A matrix with $s + 1$ columns and $\text{length}(\text{interims})$ rows giving the numbers of participants providing data or present at each interim analysis (row) at recruitment (first column) and at each follow-up time point (tfu) in columns 2 to $s + 1$. Alternatively, a function $fn(x)$ that gives a matrix of these dimensions for sample sizes x in the range ($\text{sopt}\$bisect\min , $\text{sopt}\$bisect\max). Currently this is only implemented and tested for functions based on fixed proportions.
pow	As for gsearlyModel .
n	As for gsearlyModel .
tref	As for gsearlyModel .
vphi	As for gsearlyModel .
cmodel	Either a correlation model, <code>c("uniform", "exponential")</code> , defined by correlation parameter ρ , or a user supplied correlation matrix of dimensions $s \times s$.
sd	As for gsearlyModel .
rho	As for gsearlyModel if $cmodel$ is set to "uniform" or "exponential", otherwise ignored.

theta	As for gsearlyModel .
fp	As for gsearlyModel .
tn	As for gsearlyModel .
treatnames	As for gsearlyModel .
sopt	As for gsearlyModel .

Value

An object of class *gsearly* is a list containing the following components

title	Package title and version number.
call	Call to <i>gsearlyUser</i> .
rdata	A list of same structure as for gsearlyModel with the recruitment model set to "none", the recruitment model parameter (<i>m</i>) set to NA and the numbers of study participants recruited and with outcome data at each of the interims set to <i>ninterims</i> .
idata	A list of same structure as for gsearlyModel with the correlation model type (<i>cmodel\$type</i>) set to "none" and the parameter (<i>cmodel\$rho</i>) set to NA, if either "uniform" or "exponential" are not selected.
power	A list of same structure as for gsearlyModel . In addition for designs based on fixed numbers of participants at interim, where <i>n</i> is unset, the minimum and maximum attainable power (<i>rangepow</i>).

See Also

[gsearlyModel](#)

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000, 0.0010, 0.0250)
tn <- c(0.2400, 0.7200, 0.9750)
modeldesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(18,30), pow=0.9, vphi=0.5, m=2,
                           cmodel="uniform", sd=20, rho=0.5, theta=8, fp=fp, tn=tn)
modeldesign

# This design can be replicated using gsearlyUser
n <- modeldesign$rdata$n["total"]
ninterims <- modeldesign$rdata$interims
cmodel <- modeldesign$idata$cmodel$corrrmat
userdesign <- gsearlyUser(trecruit=36, s=3, tfu=c(3,6,12), tinterims=c(18,30),
                           ninterims=ninterims, n=n, vphi=0.5, cmodel=cmodel,
                           sd=20, theta=8, fp=fp, tn=tn)
userdesign

# Expected numbers of participants at interim analyses
modeldesign$rdata$interims
userdesign$rdata$interims
```

```

# Information at these interims and final analysis
modeldesign$idata$interims
userdesign$idata$interims

# Upper and lower stopping boundaries and probabilities
rbind(modeldesign$power$lowererror, modeldesign$power$uppererror)
rbind(userdesign$power$lowererror, userdesign$power$uppererror)

# Change correlation matrix and interim numbers
cmodel <- matrix(c(1,0.2,0.1, 0.2,1,0.1, 0.1,0.1,1), nrow=3, byrow=TRUE)
ninterims <- matrix(c(130,110,90,45, 200,175,160,120), nrow=2, byrow=TRUE)
# For 90 percent power (pow), a call to gsearlyUser provides a feasible design
nuserdesign <- gsearlyUser(trecruit=36, s=3, tfu=c(3,6,12), tinterims=c(18,30),
                           ninterims=ninterims, vphi=0.5, pow=0.9, cmodel=cmodel,
                           sd=20, theta=8, fp=fp, tn=tn)
nuserdesign

```

modelParameters *Estimates model parameters from raw data*

Description

Provides estimates of the variance of beta (vbeta) and beta, the treatment effect, directly from raw data and a user supplied covariance matrix.

Usage

```
modelParameters(data, datanames=c("id", "atime", "intervention", "outcome"),
                vcovmat)
```

Arguments

data	A data frame structured as those from function simdataExtract , with full=TRUE, but without the requirement for a standardized (continuous) time-point (see tfuStandard).
datanames	Names of the four variables in the data frame data that are respectively participant id, time-point, intervention arm and outcome, in that order.
vcovmat	A covariance matrix of dimensions $s \times s$.

Value

Returns the variance of beta (vbeta), beta and test statistic z.

See Also

[gsearlySimulate](#), [simdataExtract](#)

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000,0.0010,0.0250)
tn <- c(0.2400,0.7200,0.9750)
modeldesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(18,30), pow=0.9, vphi=0.5, m=2,
                           cmodel="uniform", sd=20, rho=0.5, theta=8, fp=fp, tn=tn)

# Simulate data from this model with raw data using full=TRUE
simdata <- gsearlySimulate(mod=modeldesign, nsim=10, full=TRUE)

# Extract raw data for a single simulation
simdat1 <- simdataExtract(simdata, simn=1, tinterims=18, full=TRUE)
# Get model parameters
modelParameters(data=simdat1$data, vcovmat=simdat1$model$covariance)

# Try alternative covariance model
varmat <- diag(c(18,22,24))
vcovmat <- tcrossprod(crossprod(varmat,corrExp(rho=0.8,
                                                tfu=simdat1$model$tfu)),varmat)
modelParameters(data=simdat1$data, vcovmat=vcovmat)
```

plot.gsearly

Plot a gsearly model

Description

Plots data from a gsearly model constructed using either `gsearlyModel` or `gsearlyUser`.

Usage

```
## S3 method for class 'gsearly'
plot(x, plottype=1, ...)
```

Arguments

- x A gsearly model from function either `gsearlyModel` or `gsearlyUser`.
- plottype A value which is one of either 1 or "recruit", 2 or "inform", 3 or "plotBoundary" and 4 or "plotPower".
- ... Further graphical parameters for lines and points.

Value

A plot of the selected type.

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000,0.0010,0.0250)
tn <- c(0.4800,0.7200,0.9750)
rctdesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8,
                           cmodel="exponential", sd=20, rho=0.75, theta=8, fp=fp, tn=tn)
plot(rctdesign, "recruit")
```

plotBoundary

A boundary plot for a gsearly model

Description

A boundary plot for a gsearly model.

Usage

```
plotBoundary(mod, xlim=c(0,1), ylim=NA, xlab=NA, ylab=NA, title=NULL, col=1,
            lty=c(3, 3), pch=3, las=1, concol=grey(0.9), reglab=TRUE, signif=3,
            bounds=TRUE, pcol=1, labpos=c(2,2), ...)
```

Arguments

mod	A gsearly model from function gsearlyModel or gsearlyUser functions.
xlim	The x limits $c(x_1, x_2)$ of the plot; default NA.
ylim	The y limits of the plot; default NA.
xlab	A label for the x axis; defaults to "Information".
ylab	A label for the y axis; defaults to "Normal Critical Values".
title	A main title for the plot; default NULL.
col	Default plotting color for lines and points; default 1.
lty	A vector of line types for upper and lower boundaries; default $c(2, 2)$.
pch	Plotting character or symbol for points; default 3.
las	A numeric value which is one of either 0, 1, 2 or 3; the style of axis labels, 0: always parallel to the axis, 1: always horizontal (default), 2: always perpendicular to the axis, 3: always vertical.
concol	Color for the continue region, default grey(0.9). Set to grey(1) for no color.
reglab	Logical for plotting of region (Continue, Reject H0 and Accept H0) labels; default TRUE.
signif	Number of decimal places for the boundaries; default 3.
bounds	Logical for plotting boundary values; default TRUE.
pcol	Color for region labels and points; default 1.
labpos	Position for upper and lower boundary values (if bounds=TRUE); default $c(2, 2)$.
...	Further arguments to plot function.

Value

A plot of the selected type or a list consisting of interim information fractions and lower and upper boundaries.

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000,0.0010,0.0250)
tn <- c(0.4800,0.7200,0.9750)
rctdesign <- gsearlyModel(rmodel="fix", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8,
                           cmodel="exponential", sd=20, rho=0.75, theta=8, fp=fp, tn=tn)
plotBoundary(rctdesign)
```

plotInform

*An information plot for a gsearly model***Description**

An information or variance plot for a gsearly model.

Usage

```
plotInform(mod, xlim=NA, ylim=NA, xlab=NA, ylab=NA, title=NULL, col=1,
           lty=1, pch=3, las=1, tlag=c(0,0), wopcol=grey(0.9), intlab=NA, labpos=1,
           legpos="bottomleft", ptype="var", pcol=1, ...)
```

Arguments

<code>mod</code>	A gsearly model from function <code>gsearlyModel</code> or <code>gsearlyUser</code> functions.
<code>xlim</code>	The x limits <code>c(x1, x2)</code> of the plot; default NA.
<code>ylim</code>	The y limits of the plot; default NA.
<code>xlab</code>	A label for the x axis; defaults to "Time".
<code>ylab</code>	A label for the y axis; defaults to "Information fraction" if <code>ptype</code> is "inform" or "SE" (standard error) if "var".
<code>title</code>	A main title for the plot; default NULL.
<code>col</code>	Default plotting color for lines and points; default 1.
<code>lty</code>	Line type; default 1.
<code>pch</code>	Plotting character or symbol for points; 3
<code>las</code>	A numeric value which is one of either 0, 1, 2 or 3; the style of axis labels, 0: always parallel to the axis, 1: always horizontal (default), 2: always perpendicular to the axis, 3: always vertical.
<code>tlag</code>	A vector of length 2, with positive values for time lags, pre-recruitment and post-recruitment, over which to extend the plotted lines; default <code>c(0, 0)</code> .

wopcol	Color for the window of opportunity region, default grey(0.9). Set to grey(1) for no color.
intlab	A vector of labels for the interim analyses; defaults to t[1], t[2], etc. Set to NULL, for no labels.
labpos	Position of interim labels; default 1
legpos	Legend position is one of "bottomright", "bottom", "bottomleft", "left", "topleft", "top", "topright", "right" and "center"; default "topleft".
ptype	Plot type, which is one of "var" or "inform", for a variance or information plot, with the former plotting the square root of the variance of the treatment effect (\sqrt{var}) and the latter the information ($1/var$); default "var".
pcol	Color for points; default 1.
...	Further arguments to plot function.

Value

A plot of the selected type or a list consisting of the plotted line and interim point data.

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000, 0.0010, 0.0250)
tn <- c(0.4800, 0.7200, 0.9750)
rctdesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8,
                           cmode1="exponential", sd=20, rho=0.75, theta=8, fp=fp, tn=tn)
plotInform(rctdesign)
```

Description

A power plot for a gsearly model.

Usage

```
plotPower(mod, xlim=NA, ylim=NA, xlab=NA, ylab=NA, title=NULL,
          col=c(1,2), lty=NA, pch=3, las=1, legpos=c("bottomright","topright"),
          xtype="theta", delta=seq(0,1,0.05), legstudy=NA, ...)
```

Arguments

<code>mod</code>	A gsearly model from function <code>gsearlyModel</code> or <code>gsearlyUser</code> functions.
<code>xlim</code>	The x limits <code>c(x1, x2)</code> of the plot; default NA.
<code>ylim</code>	The y limits of the plot; default NA.
<code>xlab</code>	A label for the x axis; default depends on <code>xtype</code> setting such that if "delta", then label is "Standardized effect size" or if "theta", then label is "Effect size".
<code>ylab</code>	A label for the y axis; default "Cumulative boundary crossing probability".
<code>title</code>	A main title for the plot; default NULL.
<code>col</code>	Plotting colours for the lower and upper boundaries; default <code>c(1, 2)</code> .
<code>lty</code>	Line types, which should be a numeric vector of length equal to the number of looks (<code>mod\$power\$nlooks</code>); default <code>1:nlooks</code> .
<code>pch</code>	Plotting character or symbol for points; default 3.
<code>las</code>	A numeric value which is one of either 0, 1, 2 or 3; the style of axis labels, 0: always parallel to the axis, 1: always horizontal (default), 2: always perpendicular to the axis, 3: always vertical.
<code>legpos</code>	Legend position is a vector of length 2 giving the positions of the legends (interims and probabilities), which are one of "bottomright", "bottom", "bottomleft", "left", "topleft", "top", "topright", "right" and "center"; default "topleft".
<code>xtype</code>	X-axis options are either "delta" or "theta", the former uses theta/sd (standardized effect size) and the latter theta; default "theta".
<code>delta</code>	An ordered vector of values at which the probabilities are calculated. Smoother curves are obtained if the intervals between values are smaller; default <code>seq(0, 1, 0.05)</code> .
<code>legstudy</code>	A label for the study name in the legend; default is "study".
<code>...</code>	Further arguments to plot function.

Value

A plot of the selected type or a list consisting of the plotted line and interim point data.

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0010, 0.0100, 0.0250)
tn <- c(0.4800, 0.7200, 0.9750)
rctdesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8,
                           cmodel="exponential", sd=20, rho=0.75, theta=8, fp=fp, tn=tn)
plotPower(rctdesign)
```

<code>plotRecruit</code>	<i>A recruitment plot for a gsearly model</i>
--------------------------	---

Description

A recruitment plot for a gsearly model.

Usage

```
plotRecruit(mod, xlim=NA, ylim=NA, xlab=NA, ylab=NA, title=NULL, col=NA,
            lty=c(1,2), pch=3, las=1, wopcol=grey(0.9), intlab=NA, labpos=3,
            legpos="topleft", tlag=c(0,0), pcol=1, ...)
```

Arguments

<code>mod</code>	A gsearly model from function <code>gsearlyModel</code> or <code>gsearlyUser</code> functions.
<code>xlim</code>	The x limits <code>c(x1, x2)</code> of the plot; default NA.
<code>ylim</code>	The y limits of the plot; default NA.
<code>xlab</code>	A label for the x axis; defaults to "Time".
<code>ylab</code>	A label for the y axis; defaults to "Number Recruited".
<code>title</code>	A main title for the plot; default NULL.
<code>col</code>	Plotting colors for the lines showing recruited numbers; defaults to <code>1:mod\$rdata\$s</code> .
<code>lty</code>	A vector of length 2 indicating the line types for the lines showing numbers recruited and the interim analyses times; default <code>c(1, 2)</code> .
<code>pch</code>	Plotting character or symbol for points; default 3.
<code>las</code>	A numeric value which is one of either 0, 1, 2 or 3; the style of axis labels, 0: always parallel to the axis, 1: always horizontal (default), 2: always perpendicular to the axis, 3: always vertical.
<code>wopcol</code>	Color for the window of opportunity region, default <code>grey(0.9)</code> . Set to <code>grey(1)</code> for no color.
<code>intlab</code>	A vector of labels for the interim analyses; defaults to <code>t[1], t[2]</code> , etc. Set to NULL, for no labels.
<code>labpos</code>	Position of interim labels; default 3.
<code>legpos</code>	Legend position is one of "bottomright", "bottom", "bottomleft", "left", "topleft", "top", "topright", "right" and "center"; default "topleft".
<code>tlag</code>	A vector of length 2, with positive values for time lags, pre-recruitment and post-recruitment, over which to extend the plotted lines; default <code>c(0, 0)</code> .
<code>pcol</code>	Color for points and interim lines; default 1.
...	Further arguments to plot function.

Value

A plot of the selected type or a list consisting of the plotted line and interim point data and bounds for the window of opportunity.

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000,0.0010,0.0250)
tn <- c(0.4800,0.7200,0.9750)
rctdesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8,
                           cmodel="exponential", sd=20, rho=0.75, theta=8, fp=fp, tn=tn)
plotRecruit(rctdesign)
```

print.gsearly *Print a gsearly model*

Description

Prints gsearly model details.

Usage

```
## S3 method for class 'gsearly'
print(x, digits=4, ...)
```

Arguments

x	A fitted gsearly object from gsearlyModel or gsearlyUser .
digits	The number of digits required for numeric output.
...	Further arguments passed to print (not currently implemented).

Value

A list containing the following components.

title	Package name and version number.
call	Call used to create mod.
rdata	See details in gsearlyModel or gsearlyUser .
idata	See details in gsearlyModel or gsearlyUser .
power	See details in gsearlyModel or gsearlyUser .

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000,0.0010,0.0250)
tn <- c(0.4800,0.7200,0.9750)
rctdesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8,
                           cmodel="exponential", sd=20, rho=0.75, theta=8, fp=fp, tn=tn)
rctdesign
```

```
print(rctdesign)
str(print(rctdesign))
```

`print.summary.gsearly` *Print a gsearly model summary*

Description

Prints a gsearly model summary.

Usage

```
## S3 method for class 'gsearly'
print.summary(x, digits=4, ...)
```

Arguments

<code>x</code>	A fitted gsearly object from function <code>gsearlyModel</code> or <code>gsearlyUser</code> .
<code>digits</code>	The number of digits required for numeric output.
<code>...</code>	Further arguments passed to print (not currently implemented).

Value

A list containing the following components

<code>title</code>	Package name and version number.
<code>call</code>	Call used to create mod.
<code>rdata</code>	See details in <code>gsearlyModel</code> or <code>gsearlyUser</code> .
<code>idata</code>	See details in <code>gsearlyModel</code> or <code>gsearlyUser</code> .
<code>power</code>	See details in <code>gsearlyModel</code> or <code>gsearlyUser</code> .

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000,0.0010,0.0250)
tn <- c(0.4800,0.7200,0.9750)
rctdesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8,
                           cmodel="exponential", sd=20, rho=0.75, theta=8, fp=fp, tn=tn)
summary(rctdesign)
str(summary(rctdesign))
```

qol	<i>Quality of life index for patients with knee pain</i>
-----	--

Description

Quality of life scores for 175 participants, from a randomized controlled trial, at 2, 4 and 8 weeks after treatment.

Usage

```
data(qol)
```

Format

A data frame with 455 observations on the following four variables.

Subject A participant identifier variable.

Weeks Recorded at follow-up times of 2, 4 and 8 weeks.

STime Standardized follow-up time variable; see [tfuStandard](#).

Treat Random treatment allocation (A or B).

QoL Quality of life index (0-100), where a higher value indicates a higher QoL.

Day Day number (starting at 1), when QoL was assessed.

Examples

```
data(qol)
head(qol)
```

roundInterims	<i>Round a gsearly design interim sample size to integer values</i>
---------------	---

Description

Rounds the interim sample size to integer values.

Usage

```
roundInterims(mod, direct="u", full=FALSE)
```

Arguments

- | | |
|---------------|---|
| mod | A fitted gsearly object from function gsearlyModel or gsearlyUser . |
| direct | Rounds interim sample sizes to nearest integer, upwards "u" (using <code>ceiling</code>) or downwards "d" (using <code>floor</code>). |
| full | Either FALSE, which provides total numbers only or TRUE which provides full details of numbers by groups. |

Value

Returns a matrix (or matrices, if `full=TRUE`) of the total sample sizes (and control and treatment groups, if `full=TRUE`) for each outcome at interims analyses.

See Also

[gsearlyModel](#), [gsearlyUser](#)

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000,0.0010,0.0250)
tn <- c(0.2400,0.7200,0.9750)
rctdesign <- gsearlyModel(rmodel="fix", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8,
                           cmodel="exponential", sd=20, rho=0.75, theta=10, fp=fp, tn=tn)
rctdesign

# Expected numbers of participants at interim analyses
rctdesign$rdata$interims

# Round design up to integer values
round_rctdesign <- roundInterims(rctdesign, direct="u")
round_rctdesign

# Power for rounded design
n <- rctdesign$rdata$n["total"]
ninterims <- round_rctdesign
cmodel <- rctdesign$idata$cmodel$corrmat
userdesign <- gsearlyUser(trecruit=36, s=3, tfu=c(3,6,12), tinterims=c(16,31),
                           ninterims=ninterims, n=n, cmodel=cmodel,
                           sd=20, theta=10, fp=fp, tn=tn)
userdesign
userdesign$rdata$interims
```

simdataExtract

Extract simulated data for a single trial

Description

Extract data created using function `gsearlySimulate`.

Usage

```
simdataExtract (simmod, simn, tinterims,
                datanames=c("id", "atime", "catime", "intervention", "outcome"),
                full=FALSE)
```

Arguments

<code>simmod</code>	A simulation model created using function gsearlySimulate , where full data are available i.e. where <code>full=TRUE</code> .
<code>simn</code>	Simulation number.
<code>tinterims</code>	A vector of ordered numeric valid (i.e. a subset of those in <code>simmod</code>) interim analysis time-points. If unset, these are taken from <code>simmod</code> .
<code>datanames</code>	Names of the five required data variables; participant identifier, time-point, standardized (continuous) time-point (see tfuStandard), intervention arm and outcome, in that order; e.g. <code>c("id", "atime", "catime", "intervention", "outcome")</code> .
<code>full</code>	Either FALSE, which provides a data frame only or TRUE which provides model details in addition to the data frame.

Value

A list containing the following components.

<code>model</code>	A list of <code>s</code> , <code>tinterims</code> , <code>tfu</code> and <code>mean</code> , covariance and correlation matrices and parameters and sample sizes.
<code>data</code>	A data frame consisting of five columns that are respectively the participant id, time-point, standardized time-point, intervention arm and outcome, in that order.

See Also

[gsearlySimulate](#)

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000, 0.0010, 0.0250)
tn <- c(0.2400, 0.7200, 0.9750)
modeldesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(18,30), pow=0.9, vphi=0.5, m=2,
                           cmodel="uniform", sd=20, rho=0.5, theta=8, fp=fp, tn=tn)

# Simulate data from this model with raw data using full=TRUE
simdata <- gsearlySimulate(mod=modeldesign, nsim=10, full=TRUE)

# Extract raw data for a single simulation
simdat1 <- simdataExtract(simdata, simn=1, tinterims=18,
                           full=TRUE, datanames=c("ID", "Time", "cTime", "Treat", "Outcome"))
head(simdat1$data, n=20)
```

<code>summary.gsearly</code>	<i>Summarise a gsearly model</i>
------------------------------	----------------------------------

Description

Provides a summary of a gsearly model.

Usage

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

Arguments

<code>object</code>	A fitted gsearly object from function gsearlyModel or gsearlyUser .
...	Further arguments passed to function (not currently implemented).

Value

A gsearly model summary.

Examples

```
# For 90 percent power (pow), a call to gsearlyModel provides a feasible design
fp <- c(0.0000,0.0010,0.0250)
tn <- c(0.4800,0.7200,0.9750)
rctdesign <- gsearlyModel(rmodel="dilin", trecruit=36, s=3, tfu=c(3,6,12),
                           tinterims=c(16,31), pow=0.8,
                           cmodel="exponential", sd=20, rho=0.75, theta=8, fp=fp, tn=tn)
summary(rctdesign)
str(summary(rctdesign))
```

<code>tfuStandard</code>	<i>Standardize follow-up times</i>
--------------------------	------------------------------------

Description

Standardize the follow-up (early) time-points.

Usage

```
tfuStandard(tfu, tref=c(1,2))
```

Arguments

- tfu** A vector of length s , of follow-up time points, in the same units as the recruitment time (`trecruit`) from function `gsearlyModel` or `gsearlyUser`.
- tref** A vector of integers of length 2 that defines the correlation parameter `rho` for the exponential model `corrExp`; the parameter `rho` is the correlation between data at times `tfu[tref[1]]` and `tfu[tref[2]]`, with default `c(1, 2)`.

Value

A list containing the following components.

- tfu** Follow-up time points.
- stfu** Standardized follow-up time-points, depending on the selected reference categories `tref`.

Examples

```
# Early outcomes at 3, 6 and 12 months, and final at 24 months
# Standardized to units of 3 months if tref is c(1,2)
tfuStandard(c(3,6,12,24), tref=c(1,2))

# Standardized to yearly units if tref is c(1,2)
tfuStandard(c(3,6,12,24), tref=c(3,4))
```

Index

* **datasets**
 qol, 24

corrExp, 3, 9, 10, 28
corrUnif, 4, 9, 10

dataOrder, 5

expectSampszie, 5, 10

fixedSampszie, 6, 10

gsearly (gsearly-package), 2
gsearly-package, 2
gsearlyFit, 7, 12
gsearlyModel, 3, 6, 8, 11–14, 16–18, 20–25,
 27, 28
gsearlySimulate, 3, 8, 11, 15, 26
gsearlyUser, 3, 6, 13, 16–18, 20–25, 27, 28

modelParameters, 8, 12, 15

plot.gsearly, 16
plotBoundary, 17
plotInform, 18
plotPower, 19
plotRecruit, 21
print.gsearly, 22
print.summary.gsearly, 23

qol, 24

roundInterims, 10, 24

simdataExtract, 5, 7, 8, 12, 15, 25
summary.gsearly, 27

tfuStandard, 3–5, 7, 9, 15, 24, 26, 27