Title

stata.com

gsdesign — Study design for group sequential trials

Description Quick start Menu Syntax
Options Remarks and examples Stored results Methods and formulas
References Also see

Description

`gsdesign` computes stopping boundaries and sample sizes for interim analyses of clinical trials using group sequential designs (GSDs). Stopping can be for efficacy, futility, or both. `gsdesign` can be used with sample-size calculations from a variety of [PSS-2] **power** methods, including user-defined methods. For stopping boundary calculations without sample sizes, see [ADAPT] **gsbounds**. For a software-free introduction to GSDs, see [ADAPT] **GSD intro**; for an introduction to Stata's gs suite of commands, see [ADAPT] **gs**.

Quick start

Sample sizes and stopping boundaries for a two-sided test of two sample means, with $H_0: \mu_1 = \mu_2$ versus $H_a: \mu_1 \neq \mu_2$ and a shared standard deviation of 9, with default power of 0.8 to detect the difference between control-group mean $m_1 = 8$ and experimental-group mean $m_2 = 12$ at default overall significance level $\alpha = 0.05$, using default group sequential specifications of O'Brien–Fleming efficacy boundaries with two analyses (one interim, one final)

```
gsdesign twomeans 8 12, sd(9)
```

Same as above, but with an overall significance level of 0.01 and using an O'Brien–Fleming design with three looks to calculate both efficacy and nonbinding futility boundaries

```
gsdesign twomeans 8 12, sd(9) alpha(0.01) efficacy(obfleming)     ///
         futility(obfleming) nlooks(3)
```

Same as above, but use Kim–DeMets boundaries with parameters $\rho_e = 4$ and $\rho_f = 2.5$, and assign twice as many participants to the experimental arm as to the control arm

```
gsdesign twomeans 8 12, sd(9) nratio(2) alpha(0.01)               ///
         efficacy(kdemets(4)) futility(kdemets(2.5)) nlooks(3)
```

Sample size and stopping boundaries for one-sample proportion test of $H_0: \pi = 0.2$ versus $H_a: \pi \neq 0.2$ with power of 0.9 to detect the difference between null proportion $p_0 = 0.2$ and alternative proportion $p_a = 0.3$ at overall significance level $\alpha = 0.1$, using Wang–Tsiatis efficacy boundaries with eight analyses and efficacy parameter $\Delta_e = 0.25$

```
gsdesign oneproportion 0.2 0.3, alpha(0.1) power(0.9)             ///
         efficacy(wtsiatis(0.25)) nlooks(8)
```

Same as above, but report fractional sample sizes and graph the boundaries without shading

```
gsdesign oneproportion 0.2 0.3, alpha(0.1) nfractional power(0.9) ///
         efficacy(wtsiatis(0.25)) nlooks(8) graphbounds(noshade)
```

Sample size and number of events for the log-rank test of $H_0: HR = 1$ versus $H_a: HR < 1$ with default significance level $\alpha = 0.05$ and power of 0.8 to detect the difference between a control-group survival probability of 0.3 and an experimental-group survival probability of 0.5, using error-spending O'Brien–Fleming-style efficacy boundaries with five analyses

```
gsdesign logrank 0.3 0.5, onesided efficacy(errobfleming) nlooks(5)
```

Same as above, but time the looks to occur with 40%, 60%, 80%, 90%, and 100% of the data, adjust the sample size for 5% withdrawal, and graph the boundaries

```
gsdesign logrank 0.3 0.5, wdprob(0.05) onesided                    ///
         efficacy(errobfleming) information(0.4 0.6 0.8 0.9 1)     ///
         graphbounds
```

Menu

Statistics > Power, precision, and sample size

Syntax

`gsdesign` *method* ... [, *designopts boundopts*]

where *method* ... refers to a `power` *method* that is used for sample-size calculation, *designopts* are options controlling the sample-size calculation, and *boundopts* are options controlling the calculation of the stopping boundaries.

method	Description
`onemean`	GSD for one-sample mean test
`twomeans`	GSD for two-sample means test
`oneproportion`	GSD for one-sample proportion test
`twoproportions`	GSD for two-sample proportions test
`logrank`	GSD for a log-rank test
usermethod	user-defined sample-size calculation

`gsdesign` supports the above methods when they are used to calculate sample size with simple random sampling. To use an unsupported method, specify option `methodok`; see *designopts* table below.

designopts	Description
Main	
methodopts	method-specific options
`alpha(#)`	overall significance level for all tests; default is `alpha(0.05)`
`power(#)`	overall power for all tests; default is `power(0.8)`
`beta(#)`	overall probability of type II error for all tests; default is `beta(0.2)`
`onesided`	request a one-sided test; default is two-sided
`nfractional`	report fractional sample size
`force`	allow calculation with unsupported *methodopts*
`methodok`	allow calculation with unsupported *method*
`poweriteration(`*powiteropts*`)`	iteration options for the calculation of fixed-study sample size; not available with *method* `logrank`; seldom used

`collect` is allowed; see [U] **11.1.10 Prefix commands**.

`force`, `methodok`, and `poweriteration()` do not appear in the dialog box.

methodopts	[ADAPT] entry
onemeanopts	[ADAPT] **gsdesign onemean**
twomeansopts	[ADAPT] **gsdesign twomeans**
onepropopts	[ADAPT] **gsdesign oneproportion**
twopropopts	[ADAPT] **gsdesign twoproportions**
logrankopts	[ADAPT] **gsdesign logrank**
usermethodopts	[ADAPT] ***gsdesign usermethod***

powiteropts	Description
`init(#)`	initial value for fixed-study sample size
`iterate(#)`	maximum number of iterations; default is `iterate(500)`
`tolerance(#)`	parameter tolerance; default is `tolerance(1e-12)`
`ftolerance(#)`	function tolerance; default is `ftolerance(1e-12)`

boundopts	Description
Bounds	
`efficacy(`*boundary*`)`	boundary for efficacy stopping; if neither `efficacy()` nor `futility()` is specified, the default is `efficacy(obfleming)`
`futility(`*boundary*`[ , binding ])`	boundary for futility stopping; use `binding` to request binding futility bounds (default is nonbinding)
`nlooks(#[ , equal ])`	total number of analyses (`nlooks()` − 1 interim analyses and one final analysis); use `equal` to enforce equal information increments; if neither `nlooks()` nor `information()` is specified, the default is `nlooks(2)`
`information(`*numlist*`)`	sequence of information levels for analyses; default is evenly spaced
`nopvalues`	suppress p-values
Graph	
`graphbounds[ (`*graphopts*`) ]`	graph boundaries
`matlistopts(`*general_options*`)`	control the display of boundaries and sample size; seldom used
optimopts	optimization options for boundary calculations; seldom used

`matlistopts()` and *optimopts* do not appear in the dialog box.

boundary	Description
`obfleming`	classical O'Brien–Fleming bound
`pocock`	classical Pocock bound
`wtsiatis(#)`	classical Wang–Tsiatis bound with specified parameter value
`errpocock`	error-spending Pocock-style bound
`errobfleming`	error-spending O'Brien–Fleming-style bound
`kdemets(#)`	error-spending Kim–DeMets bound with specified parameter value
`hsdecani(#)`	error-spending Hwang–Shih–de Cani bound with specified parameter value

graphopts	Description
xdimsampsize	label the x axis with the sample size collected (default)
xdiminformation	label the x axis with the information fraction; use information levels if information() specified
xdimlooks	label the x axis with the number of each look
noshade	do not shade the rejection, acceptance, and continuation regions
rejectopts(*area_options*)	change the appearance of the rejection region
acceptopts(*area_options*)	change the appearance of the acceptance region
continueopts(*area_options*)	change the appearance of the continuation region
efficacyopts(*connected_options*)	change the appearance of the efficacy bound
futilityopts(*connected_options*)	change the appearance of the futility bound
nolooklines	do not draw vertical reference lines at each look
looklinesopts(*added_line_suboptions*)	change the appearance of the reference lines marking each look
nofixed	do not label critical values from a fixed study design
fixedopts(*marker_options*)	change the appearance of the fixed-study critical values
twoway_options	any options other than by() documented in [G-3] ***twoway_options***

optimopts	Description
intpointsscale(#)	scaling factor for number of quadrature points; default is intpointsscale(20)
initinfo(*initinfo_spec*)	initial value(s) for maximum information
initscale(#)	initial value for scaling factor C of classical bounds
infotolerance(#)	tolerance for bisection search for maximum information of error-spending bounds with futility stopping; default is infotol(1e-6)
marquardt	use the Marquardt stepping algorithm in nonconcave regions; default is to use a mixture of steepest descent and Newton
technique(*algorithm_spec*)	maximization technique
iterate(#)	perform maximum of # iterations; default is iterate(300)
[no]log	display an iteration log; default is nolog
trace	display current parameter vector in iteration log
gradient	display current gradient vector in iteration log
showstep	report steps within an iteration in iteration log
hessian	display current negative Hessian matrix in iteration log
showtolerance	report the calculated result that is compared with the effective convergence criterion
tolerance(#)	tolerance for the parameter being optimized; default is tolerance(1e-12)
ftolerance(#)	tolerance for the objective function; default is ftolerance(1e-10)
nrtolerance(#)	tolerance for the scaled gradient; default is nrtolerance(1e-16)
nonrtolerance	ignore the nrtolerance() option

Options

Main

`alpha(#)` sets the overall significance level, which is the familywise type I error rate for all analyses (interim and final). `alpha()` must be in $(0, 0.5)$. The default is `alpha(0.05)`.

`power(#)` sets the overall power for all analyses. `power()` must be in $(0.5, 1)$. The default is `power(0.8)`. If `beta()` is specified, `power()` is set to be $1 -$ `beta()`. Only one of `power()` or `beta()` may be specified.

`beta(#)` sets the overall probability of a type II error. `beta()` must be in $(0, 0.5)$. The default is `beta(0.2)`. If `power()` is specified, `beta()` is set to be $1 -$ `power()`. Only one of `beta()` or `power()` may be specified.

`onesided` requests a study design for a one-sided test. The direction of the test is inferred from the effect size.

`nfractional` specifies that fractional sample sizes be reported.

Bounds

`efficacy(`*boundary*`)` specifies the boundary for efficacy stopping. If neither `efficacy()` nor `futility()` is specified, the default is `efficacy(obfleming)`.

`futility(`*boundary*`[ , binding ])` specifies the boundary for futility stopping.

`binding` specifies binding futility bounds. With binding futility bounds, if the result of an interim analysis crosses the futility boundary and lies in the acceptance region, the trial must end or risk overrunning the specified type I error. With nonbinding futility bounds, the trial does not need to stop if the result of an interim analysis crosses the futility boundary; the familywise type I error rate is controlled even if the trial continues. By default, futility bounds are nonbinding.

`nlooks(# [ , equal ])` specifies the total number of analyses to be performed (`nlooks()` $-$ 1 interim analyses and one final analysis). If neither `nlooks()` nor `information()` is specified, the default is `nlooks(2)`.

`equal` indicates that equal information increments be enforced, which is to say that the same number of new observations will be collected at each look. The default behavior is to start by dividing information evenly among looks, then proceed by rounding up to a whole number of observations at each look. This can cause slight differences in the information collected at each look.

`information(`*numlist*`)` specifies a sequence of information levels for interim and final analyses. This must be a sequence of increasing positive numbers, but the scale is unimportant because the information sequence will be automatically rescaled to ensure the maximum information is reached at the final look. By default, analyses are evenly spaced.

`nopvalues` suppresses the p-values from being reported in the table of boundaries for each look.

Graph

`graphbounds` and `graphbounds(`*graphopts*`)` produce graphical output showing the stopping boundaries.

graphopts are the following:

`xdimsampsize` labels the x axis with the sample size collected (the default).

`xdiminformation` labels the x axis with the information fraction unless `information()` is specified, in which case information levels will be used.

`xdimlooks` labels the x axis with the number of each look.

`noshade` suppresses shading of the rejection, acceptance, and continuation regions of the graph.

`rejectopts(`*area_options*`)` affects the rendition of the rejection region. See [G-3] ***area_options***.

`acceptopts(`*area_options*`)` affects the rendition of the acceptance region. See [G-3] ***area_options***.

`continueopts(`*area_options*`)` affects the rendition of the continuation region. See [G-3] ***area_options***.

`efficacyopts(`*connected_options*`)` affects the rendition of the efficacy bound. See [G-3] ***cline_options*** and [G-3] ***marker_options***.

`futilityopts(`*connected_options*`)` affects the rendition of the futility bound. See [G-3] ***cline_options*** and [G-3] ***marker_options***.

`nolooklines` suppresses the vertical reference lines drawn at each look.

`looklinesopts(`*added_line_suboptions*`)` affects the rendition of reference lines marking each look. See *suboptions* in [G-3] ***added_line_options***.

`nofixed` suppresses the fixed-study critical values in the plot.

`fixedopts(`*marker_options*`)` affects the rendition of the fixed-study critical values. See [G-3] ***marker_options***.

twoway_options are any of the options documented in [G-3] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G-3] ***title_options***) and for saving the graph to disk (see [G-3] ***saving_option***).

The following options are available with `gsdesign` but are not shown in the dialog box:

`force` indicates that `gsdesign` should allow unsupported method options, such as options specifying a finite population correction or a cluster randomized design. Even with option `force`, the method options specified must be compatible with sample-size determination, not effect size or power calculation. In addition, *numlist*s are not supported in method options or in arguments as they are with `power`, even when `force` is specified.

`methodok` indicates that `gsdesign` should allow unsupported methods. Option `methodok` is not required to run `gsdesign` with user-defined methods, but it is required to use `power` methods other than those described in *method*. Option `methodok` implies option `force`.

`poweriteration(`*powiteropts*`)` controls the iterative algorithm used to calculate the fixed-study sample size. This is seldom used.

powiteropts are the following:

`init(#)` specifies an initial value for the sample size when iteration is used to compute the fixed-study sample size. The default is to use a closed-form normal approximation to compute an initial sample size.

`iterate(#)` specifies the maximum number of iterations for the Newton method during calculation of the fixed-study sample size. The default is `iterate(500)`.

`tolerance(#)` specifies the tolerance used to determine whether successive parameter estimates have converged when calculating the fixed-study sample size. The default is `tolerance(1e-12)`. See *Convergence criteria* in [M-5] **solvenl( )** for details.

`ftolerance(#)` specifies the tolerance used when calculating the fixed-study sample size to determine whether the proposed solution of a nonlinear equation is sufficiently close

to 0 based on the squared Euclidean distance. The default is `ftolerance(1e-12)`. See *Convergence criteria* in [M-5] **solvenl()** for details.

`matlistopts(`*general_options*`)` affects the display of the matrix of boundaries and sample sizes. *general_options* are `title()`, `tindent()`, `rowtitle()`, `showcoleq()`, `coleqonly`, `colorcoleq()`, `aligncolnames()`, and `linesize()`; see *general_options* in [P] **matlist**. This option is seldom used.

optimopts control the iterative algorithm used to calculate stopping boundaries:

`intpointsscale(#)` specifies the scaling factor for the number of quadrature points used during the numerical evaluation of stopping probabilities at each look. The default is `intpointsscale(20)`. See *Methods and formulas* in [ADAPT] **gsbounds**.

`initinfo(`*initinfo_spec*`)` specifies either one or two initial values to be used in the iterative calculation of the maximum information.

The syntax `initinfo(#)` is applicable when using classical group sequential boundaries (Pocock bounds, O'Brien–Fleming bounds, and Wang–Tsiatis bounds), as well as with efficacy-only stopping when using error-spending boundaries (error-spending Pocock-style efficacy bounds, error-spending O'Brien–Fleming-style efficacy bounds, Kim–DeMets efficacy bounds, and Hwang–Shih–de Cani efficacy bounds). The default is to use the information from a fixed study design; see *Methods and formulas* in [ADAPT] **gsbounds**.

The syntax `initinfo(# #)` is applicable when using error-spending group sequential boundaries with futility stopping (error-spending Pocock-style bounds, error-spending O'Brien–Fleming-style bounds, Kim–DeMets bounds, and Hwang–Shih–de Cani bounds). With this syntax, the first and second numbers specify the lower and upper starting values, respectively, for the bisection algorithm estimating the maximum information. The default is to use the information from a fixed study design for the lower initial value and the information corresponding to a Bonferroni correction for the upper initial value; see *Methods and formulas* in [ADAPT] **gsbounds**. To specify just the lower starting value, use `initinfo(# .)`, and to specify just the upper starting value, use `initinfo(. #)`.

`initscale(#)` specifies the initial value to be used during the iterative calculation of scaling factor C for classical group sequential boundaries (Pocock bounds, O'Brien–Fleming bounds, and Wang–Tsiatis bounds). The default is to use the z-value corresponding to the specified value of `alpha()`. See *Methods and formulas* in [ADAPT] **gsbounds**.

`infotolerance(#)` specifies the tolerance for the bisection algorithm used in the iterative calculation of the maximum information of error-spending group sequential boundaries with futility stopping (error-spending Pocock-style bounds, error-spending O'Brien–Fleming-style bounds, Kim–DeMets bounds, and Hwang–Shih–de Cani bounds). The default is `infotolerance(1e-6)`. See *Methods and formulas* in [ADAPT] **gsbounds**.

`marquardt` specifies that the optimizer should use the modified Marquardt algorithm when, at an iteration step, it finds that H is singular. The default is to use a mixture of steepest descent and Newton, which is equivalent to the `difficult` option in [R] **ml**.

`technique(`*algorithm_spec*`)` specifies how the objective function is to be maximized. The following algorithms are allowed. For details, see Pitblado, Poi, and Gould (2024).

`technique(bfgs)` specifies the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm.

`technique(nr)` specifies Stata's modified Newton–Raphson (NR) algorithm.

`technique(dfp)` specifies the Davidon–Fletcher–Powell (DFP) algorithm.

The default is `technique(bfgs)` when using classical group sequential boundaries (Pocock bounds, O'Brien–Fleming bounds, and Wang–Tsiatis bounds) and also for the second optimization step used to estimate the maximum information with efficacy-only stopping when using error-spending boundaries (error-spending Pocock-style efficacy bounds, error-spending O'Brien–Fleming-style efficacy bounds, Kim–DeMets efficacy bounds, and Hwang–Shih–de Cani efficacy bounds). The default is `technique(nr)` for the sequential optimization steps used to estimate critical values for error-spending boundaries. You can also switch between two algorithms by specifying the technique name followed by the number of iterations. For example, specifying `technique(nr 10 bfgs 20)` requests 10 iterations with the NR algorithm followed by 20 iterations with the BFGS algorithm, and then back to NR for 10 iterations, and so on. The process continues until convergence or until the maximum number of iterations is reached.

`iterate(#)` specifies the maximum number of iterations. If convergence is not declared by the time the number of iterations equals `iterate()`, an error message is issued. The default value of `iterate(#)` is the number set using `set maxiter`, which is 300 by default.

[`no`]`log` requests an iteration log showing the progress of the optimization. The default is `nolog`.

`trace` adds to the iteration log a display of the current parameter vector.

`gradient` adds to the iteration log a display of the current gradient vector.

`showstep` adds to the iteration log a report on the steps within an iteration. This option was added so that developers at StataCorp could view the stepping when they were improving the `ml` optimizer code. At this point, it mainly provides entertainment.

`hessian` adds to the iteration log a display of the current negative Hessian matrix.

`showtolerance` adds to the iteration log the calculated value that is compared with the effective convergence criterion at the end of each iteration. Until convergence is achieved, the smallest calculated value is reported. `shownrtolerance` is a synonym of `showtolerance`.

Below, we describe the three convergence tolerances. Convergence is declared when the `nrtolerance()` criterion is met and either the `tolerance()` or the `ftolerance()` criterion is also met.

`tolerance(#)` specifies the tolerance for the parameter vector. When the relative change in the parameter vector from one iteration to the next is less than or equal to `tolerance()`, the `tolerance()` convergence criterion is satisfied. The default is `tolerance(1e-12)`.

`ftolerance(#)` specifies the tolerance for the objective function. When the relative change in the objective function from one iteration to the next is less than or equal to `ftolerance()`, the `ftolerance()` convergence is satisfied. The default is `ftolerance(1e-10)`.

`nrtolerance(#)` specifies the tolerance for the scaled gradient. Convergence is declared when $\mathbf{g}\mathbf{H}^{-1}\mathbf{g}' <$ `nrtolerance()`. The default is `nrtolerance(1e-16)`.

`nonrtolerance` specifies that the default `nrtolerance()` criterion be turned off.

boundary

`obfleming` specifies a classical O'Brien–Fleming design for efficacy or futility bounds (O'Brien and Fleming 1979). O'Brien–Fleming efficacy bounds are characterized by being extremely conservative at early looks. The O'Brien–Fleming design is a member of the Wang–Tsiatis family and is equivalent to specifying a *boundary* of `wtsiatis(0)`.

`pocock` specifies a classical Pocock design for efficacy or futility bounds (Pocock 1977). Pocock efficacy bounds are characterized by using the same critical value at all looks. The Pocock design is a member of the Wang–Tsiatis family and is equivalent to specifying a *boundary* of `wtsiatis(0.5)`.

`wtsiatis(#)` specifies a classical Wang–Tsiatis design for efficacy or futility bounds (Wang and Tsiatis 1987). The shape of Wang–Tsiatis bounds is determined by parameter $\Delta \in [-10, 0.7]$, where smaller values of Δ yield bounds that are more conservative at early looks.

`errpocock` specifies an error-spending Pocock-style design for efficacy or futility bounds (Lan and DeMets 1983). The critical values from error-spending Pocock-style bounds are very similar to those of classic Pocock bounds, but they are obtained using an error-spending function.

`errobfleming` specifies an error-spending O'Brien–Fleming-style design for efficacy or futility bounds (Lan and DeMets 1983). The critical values from error-spending O'Brien–Fleming-style bounds are very similar to those of classic O'Brien–Fleming bounds, but they are obtained using an error-spending function.

`kdemets(#)` specifies an error-spending Kim–DeMets design for efficacy or futility bounds (Kim and DeMets 1987). The shape of Kim–DeMets bounds is determined by power parameter $\rho \in (0, 10]$, where larger values of ρ yield bounds that are more conservative at early looks.

`hsdecani(#)` specifies an error-spending Hwang–Shih–de Cani design for efficacy or futility bounds (Hwang, Shih, and de Cani 1990). The shape of Hwang–Shih–de Cani bounds is determined by parameter $\gamma \in [-30, 3]$, where smaller values of γ yield bounds that are more conservative at early looks.

For a design with both efficacy and futility stopping boundaries, if you specify a classical boundary (that is, in the Wang–Tsiatis family) for one, then you must specify a classical boundary for the other. So, you could not specify a boundary in the Wang–Tsiatis family for one boundary and an error-spending boundary for the other. When specifying efficacy and futility boundaries from the same family, the efficacy parameter does not need to be the same as the futility parameter.

Boundaries that are conservative at early looks, such as the O'Brien–Fleming bound, offer little chance of early stopping unless the true effect size is quite large (in the case of efficacy bounds) or quite small (in the case of futility bounds). A trial employing a conservative bound is more likely to continue to the final look, yielding an expected sample size that is not dramatically smaller than the sample size required by an equivalent fixed-sample trial. However, the maximum sample size (that is, the sample size at the final look) of a trial with a conservative bound is generally not much greater than the sample size required by an equivalent fixed trial. Another direct result of specifying conservative bounds is that the critical value at the final look tends to be close to the critical value employed by an equivalent fixed design. In contrast, anticonservative boundaries such as the Pocock bound offer a much better shot at early stopping (often yielding a small expected sample size) at the cost of a larger maximum sample size and final critical values that are considerably larger than the critical value of an equivalent fixed design.

Remarks and examples

stata.com

Remarks are presented under the following headings:

Introduction
Examples
Design for GSD with tests of two means
Background on the BHAT study
Design for GSD with survival analysis

This entry describes the gsdesign command and the methodology for calculating stopping boundaries and sample sizes for group sequential designs, or GSDs. For a software-free introduction to GSDs, see [ADAPT] **GSD intro**; for an introduction to Stata's gs suite of commands, see [ADAPT] **gs**; to calculate stopping boundaries without sample sizes, see [ADAPT] **gsbounds**; and to calculate sample sizes for fixed study designs, see [PSS-2] **power**.

Introduction

Clinical trials are studies investigating the effects of a treatment on human participants, and sponsors of clinical trials have both ethical and economic motivations for making trials as efficient as possible. One way of accomplishing this is to analyze trial data while the study is still underway. A positive result at an interim analysis can lead to early termination of the study due to treatment efficacy, sparing future participants from being assigned to the control group and receiving an inferior treatment. If the interim analysis demonstrates that the new treatment is ineffective, the trial can stop early and resources can be allocated to testing more promising treatments.

When done naïvely, conducting multiple analyses at a nominal significance level will inflate type I error. Group sequential experimental designs provide a protocol for the interim analysis of clinical trial data and a framework in which the trial can be stopped early for efficacy or futility while maintaining control of familywise type I and type II errors.

A GSD lays out a sequence of looks, or analyses of the clinical trial data. Interim analyses, which take place before the trial is scheduled to end, provide the ability to terminate the trial early if the result at the interim look is sufficiently unambiguous. Efficacy stopping occurs when the null hypothesis, H_0, is rejected at an interim look and the clinical trial is terminated early due to treatment efficacy. The complement to efficacy stopping is futility stopping, in which the trial is terminated because H_0 has been accepted during an interim look. The concept of accepting the null hypothesis runs counter to the prevailing modern interpretation of null hypothesis significance testing, but accepting H_0 has a long history in the context of sequential trials and is commonly performed in the literature about sequential clinical trials. See *Origins of GSD* in [ADAPT] **GSD intro** for a history of GSDs.

The decision to terminate a clinical trial is frequently made by an independent monitoring group, often called a Data Monitoring Committee. The committee may decide to terminate the trial early because of demonstrated treatment efficacy or futility at an interim analysis. The Data Monitoring Committee can also stop a clinical trial for reasons such as safety and the prevalence of adverse events, which are harmful side effects of the treatment and negative medical outcomes not associated with an underlying disease. When determining whether to terminate a trial because of efficacy or futility, the committee can compare the test statistic from the interim analysis against the critical values of the efficacy or futility bounds. Test statistics with asymptotically standard normal distributions under H_0 can be compared directly with the boundary critical values, and statistics that follow other distributions under H_0 may be evaluated using the significance level approach.

The critical values of the group sequential efficacy and futility bounds depend on several factors: the overall power $(1 - \beta)$ and significance level (α) of the design, the type of boundary (gsdesign supports seven types of boundaries), whether the test has a one- or two-sided alternative hypothesis, and the information fraction at which the analyses occur. Technically, the information fraction is the proportion of the maximum possible Fisher information that has been collected about the parameter being estimated as part of the test, but this definition is too abstract to be useful. In most cases, the information fraction is the proportion of the maximum sample size that has been collected. For survival data, the information fraction is the proportion of the total number of events (failures) that have been observed, not the total number of participants. To calculate the maximum sample size of a GSD, gsdesign scales up the sample size of an equivalently powered fixed-sample design by a factor known as the information ratio.

Examples

Design for GSD with tests of two means

▷ Example 1: Pocock efficacy bounds for a test of two sample means

Jennison and Turnbull (2000, 27) demonstrate the use of Pocock efficacy bounds by considering a test of two means: μ_1 and μ_2. The null hypothesis is $H_0: \mu_1 = \mu_2$, and the two-sided alternative hypothesis is $H_a: \mu_1 \neq \mu_2$. They assume a known standard deviation of 2 for both groups and desire a test with 90% power to detect a difference in means of one unit, while maintaining an overall significance level of $\alpha = 0.05$ over five evenly spaced looks.

Given these specifications, we use `gsdesign twomeans` with a control group mean, m_1, of 0 and a difference in means of 1, specified with the `diff(1)` option. The `efficacy(pocock)` and `nlooks(5)` options request the efficacy boundaries and sample size for a Pocock design with five evenly spaced looks. `alpha()` is omitted because it is left at its default value of 0.05, and `beta()` is omitted because `power()`, defined as $(1-\beta)$, is specified instead. The `graphbounds` option instructs Stata to draw a graph of the boundaries and sample size at each look. The `sd()` option specifies the shared standard deviation of both groups, and the `knownsd` option indicates that the population standard deviation is known for both control and treatment groups.

```
. gsdesign twomeans 0, diff(1) sd(2) knownsds power(0.9) efficacy(pocock)
> nlooks(5) graphbounds

Group sequential design for a two-sample means test
z test assuming sd1 = sd2 = sd
H0: m2 = m1 versus Ha: m2 != m1

Efficacy: Pocock

Study parameters:
      alpha = 0.0500  (two-sided)
      power = 0.9000
      delta = 1.0000
         m1 = 0.0000
         m2 = 1.0000
       diff = 1.0000
         sd = 2.0000

Expected sample size:
         H0 = 199.00
         Ha = 115.43

Info. ratio = 1.2066
    N fixed =    170
      N max =    204
     N1 max =    102
     N2 max =    102

Fixed-study crit. values = ±1.9600

Critical values, p-values, and sample sizes for a group sequential design
```

Look	Info. frac.	Efficacy Lower	Efficacy Upper	p-value	Sample size N1	Sample size N2	Sample size N
1	0.20	-2.4132	2.4132	0.0158	21	21	42
2	0.40	-2.4132	2.4132	0.0158	41	41	82
3	0.60	-2.4132	2.4132	0.0158	61	61	122
4	0.80	-2.4132	2.4132	0.0158	82	82	164
5	1.00	-2.4132	2.4132	0.0158	102	102	204

```
Notes: Critical values are for z statistics; otherwise, use p-value
       boundaries.
       Requested information fraction not attained.
```

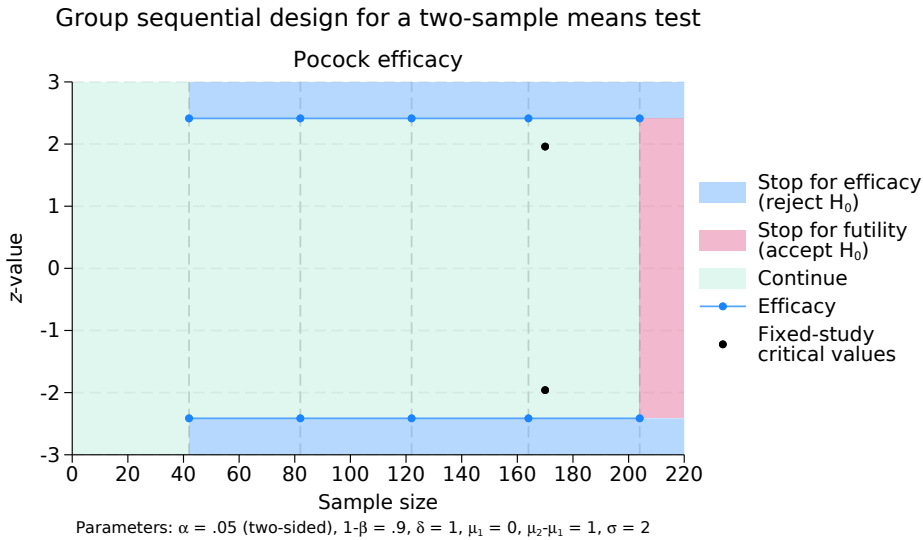

Figure 1. Pocock efficacy bounds for a test of the equality of two means

According to this design, the first look will occur when results have been collected from 21 participants in the control group and 21 participants in the experimental group. A z test of the two means will be conducted, and if the z statistic from that test, z_1, lies in the rejection region ($z_1 \geq 2.413$ or $z_1 \leq -2.413$), then H_0 will be rejected and the trial will end due to treatment efficacy. The efficacy boundary separates the rejection region from the continuation region; if $|z_1| < 2.413$, the test statistic lies within the continuation region and the trial will continue to the second look.

At each successive look, the same procedure is repeated. A defining characteristic of Pocock efficacy bounds is that the same critical value is used at all looks, so at each look the test statistic is compared with ± 2.413. At the fifth and final look, there is no continuation region: if $|z_5| \geq 2.413$, then the null hypothesis is rejected, and if $|z_5| < 2.413$, then the null hypothesis is accepted.

The graph displays the bounds visually, dividing the range of possible z-values into continuation, rejection, and acceptance regions. The vertical axis is the value of the z statistic, and the horizontal axis is the sample size. We progress from left to right in the graph as samples are collected during the course of the trial. The efficacy bounds, which separate the continuation and rejection regions, are drawn in blue and marked with a dot at each look. Before the first look (that is, when fewer than 42 samples have been collected), it is impossible to reject H_0 because the data have not yet been analyzed, so all z-values fall within the continuation region. Beginning with the first look, the range of z-values is divided into continuation and rejection regions. Because we are conducting a two-sided test, the rejection region is made up of two areas: z-values ≥ 2.413 and z-values ≤ -2.413. At the final look, there is no continuation region; it has been replaced by the acceptance region because the trial cannot be continued beyond the fifth look.

To facilitate comparison with a fixed-sample study design, `gsdesign` displays the estimated sample size and critical values for a fixed study along with the information ratio, the ratio of the maximum sample size from a GSD to the sample size of a fixed design. The Pocock design allows the trial to end after collecting data from as few as 42 participants, but if the trial continues to completion, it will require 20% more participants to attain the same power and significance level as a fixed-sample trial.

When comparing the efficiency of a GSD versus a fixed-sample design, it is useful to examine the expected sample size of the GSD. The expected sample size, which is calculated relative to a given effect size, is the average sample size that a group sequential trial would need if the experiment were to be repeated many times. In the output above, we see that the expected sample size under H_0 is 199. This means that if the true difference between group means is 0 and the trial is repeated many times, the average sample size will be 199. The expected sample size under H_a of 115.43 means that if the true difference between group means is 1, the average sample size over repeated experiments will be 115.43, a substantial savings over the 170 subjects required by the fixed-sample design.

When designing this study, Jennison and Turnbull (2000) reported the maximum sample size as 210 participants, slightly more than the 204 calculated by `gsdesign`. The difference is due to the fact that Jennison and Turnbull forced the spacing of the looks to be exactly equal by requiring each arm of the study to collect data from 21 new participants between each look. By default, `gsdesign` begins by dividing information evenly among looks, and then `gsdesign` rounds the sample sizes up to whole numbers (which can cause slight differences in the spacing between looks). To match the calculation of Jennison and Turnbull (2000), we add the `equal` suboption in the `nlooks()` option.

```
. gsdesign twomeans 0, diff(1) sd(2) knownsds power(0.9) efficacy(pocock)
> nlooks(5, equal)
Group sequential design for a two-sample means test
z test assuming sd1 = sd2 = sd
H0: m2 = m1 versus Ha: m2 != m1

Efficacy: Pocock

Study parameters:
      alpha = 0.0500  (two-sided)
      power = 0.9000
      delta = 1.0000
         m1 = 0.0000
         m2 = 1.0000
       diff = 1.0000
         sd = 2.0000

Expected sample size:
         H0 = 204.80
         Ha = 116.94

Info. ratio = 1.2066
    N fixed =    170
      N max =    210
     N1 max =    105
     N2 max =    105

Fixed-study crit. values = ±1.9600

Critical values, p-values, and sample sizes for a group sequential design
```

Look	Info. frac.	Efficacy Lower	Efficacy Upper	p-value	Sample size N1	Sample size N2	Sample size N
1	0.20	-2.4132	2.4132	0.0158	21	21	42
2	0.40	-2.4132	2.4132	0.0158	42	42	84
3	0.60	-2.4132	2.4132	0.0158	63	63	126
4	0.80	-2.4132	2.4132	0.0158	84	84	168
5	1.00	-2.4132	2.4132	0.0158	105	105	210

```
Note: Critical values are for z statistics; otherwise, use p-value
      boundaries.
```

If we enforce equal information increments, we arrive at a maximum sample size of 210. The increased sample size causes a slight increase in attained power, stored as `r(power_a)`.

```
. display "Power attained at final analysis: " r(power_a) * 100
Power attained at final analysis: 91.020745
```

We see that the additional observations yield an attained power of 91%. To understand why the information increments were not exactly equal in the original design, it is informative to view the fractional sample-size calculations by specifying the `nfractional` option.

```
. gsdesign twomeans 0, diff(1) sd(2) knownsds nfractional power(0.9)
> efficacy(pocock) nlooks(5)

Group sequential design for a two-sample means test
z test assuming sd1 = sd2 = sd
H0: m2 = m1 versus Ha: m2 != m1

Efficacy: Pocock

Study parameters:
      alpha = 0.0500  (two-sided)
      power = 0.9000
      delta = 1.0000
         m1 = 0.0000
         m2 = 1.0000
       diff = 1.0000
         sd = 2.0000

Expected sample size:
         H0 = 197.83
         Ha = 115.15

Info. ratio = 1.2066
    N fixed = 168.12
      N max = 202.85
     N1 max = 101.43
     N2 max = 101.43

Fixed-study crit. values = ±1.9600

Critical values, p-values, and sample sizes for a group sequential design
```

Look	Info. frac.	Efficacy Lower	Efficacy Upper	p-value	Sample size N1	Sample size N2	Sample size N
1	0.20	-2.4132	2.4132	0.0158	20.285	20.285	40.571
2	0.40	-2.4132	2.4132	0.0158	40.571	40.571	81.141
3	0.60	-2.4132	2.4132	0.0158	60.856	60.856	121.71
4	0.80	-2.4132	2.4132	0.0158	81.141	81.141	162.28
5	1.00	-2.4132	2.4132	0.0158	101.43	101.43	202.85

```
Note: Critical values are for z statistics; otherwise, use p-value
      boundaries.

. display "Power attained at final analysis: " r(power_a) * 100
Power attained at final analysis: 90.003222
```

Option `nfractional` instructs `gsdesign` not to round sample sizes up to the nearest whole number. We can see that the first look occurs with 20.285 observations per arm, and the second occurs with 40.571 observations per arm. Rounding up to whole numbers of participants, this gives us 21 observations per arm for the first look, and an additional 20 observations (for a total of 41) at the second look. If this trial were to continue to the fifth look, it would require 202.85 participants to attain 90% power to detect a difference in means of one unit. As the sample size increases, the relative impact of rounding up to a whole number of observations diminishes.

◁

Example 2: Pocock bounds with efficacy and futility stopping

In example 1, we saw that the GSD resulted in a substantially smaller expected sample size than an equivalent fixed study design if the alternative hypothesis was true but not if the null hypothesis was true. To increase the potential to stop the trial early if the treatment is ineffective, we now add futility bounds to the experimental design. Futility bounds separate the continuation region from the acceptance region and allow early acceptance of H_0 when there is evidence that the treatment is not meaningfully different from the control.

We use the same design as in example 1, this time adding the `futility(pocock)` option to add nonbinding Pocock futility bounds.

```
. gsdesign twomeans 0, diff(1) sd(2) knownsds power(0.9) efficacy(pocock)
> futility(pocock) nlooks(5) graphbounds

Group sequential design for a two-sample means test
z test assuming sd1 = sd2 = sd
H0: m2 = m1 versus Ha: m2 != m1

Efficacy: Pocock
Futility: Pocock, nonbinding

Study parameters:
      alpha = 0.0500  (two-sided)
      power = 0.9000
      delta = 1.0000
         m1 = 0.0000
         m2 = 1.0000
       diff = 1.0000
         sd = 2.0000

Expected sample size:
         H0 = 124.55
         Ha = 132.66

Info. ratio = 1.5966
    N fixed =    170
      N max =    270
     N1 max =    135
     N2 max =    135

Fixed-study crit. values = ±1.9600

Critical values, p-values, and sample sizes for a group sequential design
```

Look	Info. frac.	Efficacy Lower	Efficacy Upper	Efficacy p-value	Futility Lower	Futility Upper	Futility p-value
1	0.20	-2.4132	2.4132	0.0158	-0.1490	0.1490	0.8815
2	0.40	-2.4132	2.4132	0.0158	-0.9078	0.9078	0.3640
3	0.60	-2.4132	2.4132	0.0158	-1.4900	1.4900	0.1362
4	0.80	-2.4132	2.4132	0.0158	-1.9808	1.9808	0.0476
5	1.00	-2.4132	2.4132	0.0158	-2.4132	2.4132	0.0158

```
Note: Critical values are for z statistics; otherwise, use p-value
      boundaries.
```

Look	Sample size N1	N2	N
1	27	27	54
2	54	54	108
3	81	81	162
4	108	108	216
5	135	135	270

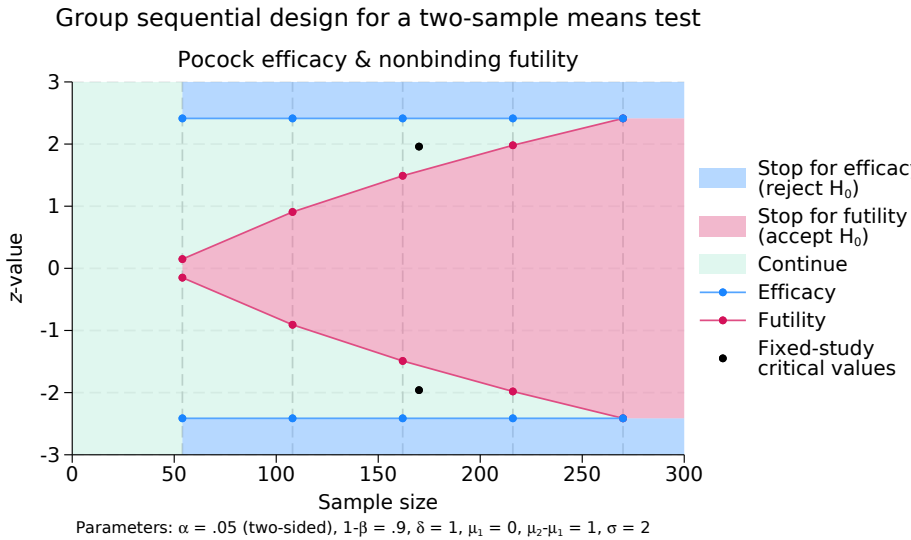

Figure 2. Pocock efficacy and futility bounds for a test of the equality of two means

The maximum sample size required by this design is even larger than that of the efficacy-only design, but the ability to end the trial early for futility can result in a considerably smaller sample size if H_0 is true. The efficacy bounds for this design are the same as they were in example 1; this is because adding nonbinding futility bounds to a group sequential trial does not affect the calculation of efficacy bound critical values.

As before, if $|z_1| \geq 2.413$, we reject H_0 and end the trial early for efficacy. With the addition of the futility bounds, we have the option of ending the trial early for futility if $|z_1| < 0.149$. If $|z_1| \in [0.149, 2.413)$, the trial must continue. While the Pocock efficacy bounds use the same critical values for all looks, the futility bounds do not; they grow from ± 0.149 at the first look to ± 1.981 by the fourth look, coinciding with the efficacy bounds at the fifth look.

As we move from left to right on the graph by collecting additional samples, we see the futility region grow and the continuation region shrink. The narrowing continuation region means that the trial is increasingly likely to stop due to futility or efficacy as more samples are collected. But if the test statistics do not cross the boundaries and the trial continues to the fifth look, the group sequential trial will require about 60% more participants than an equivalently powered fixed study.

One way to reduce the maximum sample size would be to use a boundary that is more conservative at early looks, such as an O'Brien–Fleming boundary. Another option is to use binding futility bounds instead of nonbinding bounds. While nonbinding futility bounds offer the option to stop the trial for efficacy if they are crossed, binding futility bounds require the termination of the trial if they are crossed. Continuing a trial that has crossed a binding futility bound can inflate the type I error, and any conclusions reached by the trial will be viewed with suspicion.

We rerun the previous example with `futility()` suboption `binding` to specify binding futility bounds, omitting the `graphbounds` option.

```
. gsdesign twomeans 0, diff(1) sd(2) knownsds power(0.9) efficacy(pocock)
> futility(pocock, binding) nlooks(5)

Group sequential design for a two-sample means test
z test assuming sd1 = sd2 = sd
H0: m2 = m1 versus Ha: m2 != m1

Efficacy: Pocock
Futility: Pocock, binding

Study parameters:
      alpha = 0.0500  (two-sided)
      power = 0.9000
      delta = 1.0000
         m1 = 0.0000
         m2 = 1.0000
       diff = 1.0000
         sd = 2.0000

Expected sample size:
         H0 = 120.18
         Ha = 113.00

Info. ratio = 1.5453
    N fixed =    170
      N max =    260
     N1 max =    130
     N2 max =    130

Fixed-study crit. values = ±1.9600

Critical values, p-values, and sample sizes for a group sequential design
```

Look	Info. frac.	Efficacy Lower	Efficacy Upper	Efficacy p-value	Futility Lower	Futility Upper	Futility p-value
1	0.20	-2.3564	2.3564	0.0185	-0.1290	0.1290	0.8974
2	0.40	-2.3564	2.3564	0.0185	-0.8754	0.8754	0.3813
3	0.60	-2.3564	2.3564	0.0185	-1.4482	1.4482	0.1476
4	0.80	-2.3564	2.3564	0.0185	-1.9310	1.9310	0.0535
5	1.00	-2.3564	2.3564	0.0185	-2.3564	2.3564	0.0185

```
Note: Critical values are for z statistics; otherwise, use p-value
      boundaries.
```

Look	Sample size N1	N2	N
1	26	26	52
2	52	52	104
3	78	78	156
4	104	104	208
5	130	130	260

The binding futility bounds give a modest reduction in maximum sample size, down from 270 to 260. Compared with the nonbinding design, the binding design uses slightly smaller futility critical values. Also, while the efficacy-only design and the design with nonbinding futility bounds used efficacy critical values of ± 2.413, here the efficacy critical values have shrunk to ± 2.356.

To understand why, consider what happens when the null hypothesis is true. In this case, the correct action is to accept H_0, and it is a type I error to reject H_0. In the efficacy-only design of example 1, each interim look presents the opportunity to continue the trial or to commit a type I

error and mistakenly reject H_0; only at the final look do we have the option to correctly accept H_0. With binding futility bounds, every look offers the possibility of crossing the futility boundary and correctly accepting H_0, making it less likely that the trial will continue to later looks. If we were to use the same efficacy critical values as in the efficacy-only design, the actual probability of committing a type I error would be lower than the specified significance level, and the test would be conservative. By relaxing the efficacy critical values, the desired significance level is achieved. We do not relax the efficacy critical values when nonbinding futility boundaries are used because there is no guarantee that the trial will be stopped after crossing a futility boundary.

See [ADAPT] **gsdesign twomeans** for more examples of GSDs for tests of two sample means.

◁

Background on the BHAT study

The Beta-Blocker Heart Attack Trial (BHAT) was one of the first large-scale clinical trials to adopt a group sequential monitoring plan (Cook and DeMets 2008). This was a double-blind study in which participants who had experienced a heart attack were randomized to one of two groups: the control group (which received a placebo) and the intervention group (which received the beta blocker propranolol). The endpoint, or outcome of interest, was total mortality, and survival analysis was conducted using a log-rank test with a two-sided alternative hypothesis.

Recruitment ran from June 1978 to October 1980, with follow-up scheduled to continue until June 1982. Oversight was provided by an independent Policy and Data Monitoring Board (PDMB), which contained physicians, biostatisticians, and an ethicist. While the BHAT's study protocol did not set strict rules for early termination, the PDMB adopted the then-recently published O'Brien–Fleming method early on (DeMets et al. 1984).

Based on a combination of factors, including a log-rank test statistic that crossed the O'Brien–Fleming boundary at the sixth of seven looks, the PDMB stopped the BHAT for treatment efficacy in October of 1981, eight months before follow-up was scheduled to end in June 1982. Lan and DeMets (1989) report the values of the log-rank test statistic at each of the interim looks:

	May 1979	October 1979	March 1980	October 1980	April 1981	October 1981
test statistic	1.68	2.24	2.37	2.30	2.34	2.82

DeMets, Furberg, and Friedman (2006, Case 2) report that the BHAT was designed with a two-tailed alpha level of 0.05 and 90% power to detect the difference between nonadherence-adjusted three-year survival probabilities of 82.54% for the control group and 86.25% for the intervention group. A total of seven biannual analyses were planned, and O'Brien–Fleming efficacy bounds were calculated assuming seven evenly spaced looks.

Design for GSD with survival analysis

▷ Example 3: BHAT study

To re-create the design of the BHAT, we run `gsdesign logrank` with survival probabilities 0.8254 and 0.8625 for the control and intervention arms, respectively. We specify a power of 90% and O'Brien–Fleming efficacy bounds with seven evenly spaced looks.

```
. gsdesign logrank 0.8254 0.8625, power(0.9) efficacy(obfleming) nlooks(7)
> graphbounds
Group sequential design for two-sample comparison of survivor functions
Log-rank test, Freedman method
H0: HR = 1 versus Ha: HR != 1

Efficacy: O'Brien-Fleming

Study parameters:
      alpha = 0.0500  (two-sided)
      power = 0.9000
      delta = 0.7709  (hazard ratio)
     hratio = 0.7709

Censoring:
         s1 = 0.8254
         s2 = 0.8625
       Pr_E = 0.1560

Expected number of events:
         H0 = 642.71
         Ha = 459.40

Info. ratio = 1.0323
    E fixed =    628
    N fixed =  4,024
      N max =  4,152
     N1 max =  2,076
     N2 max =  2,076

Fixed-study crit. values = ±1.9600

Critical values, p-values, and sample sizes for a group sequential design
```

Look	Info. frac.	Efficacy Lower	Efficacy Upper	p-value	Events E
1	0.14	-5.4590	5.4590	0.0000	93
2	0.29	-3.8601	3.8601	0.0001	186
3	0.43	-3.1518	3.1518	0.0016	278
4	0.57	-2.7295	2.7295	0.0063	371
5	0.71	-2.4413	2.4413	0.0146	463
6	0.86	-2.2286	2.2286	0.0258	556
7	1.00	-2.0633	2.0633	0.0391	648

```
Note: Critical values are for z statistics; otherwise, use
      p-value boundaries.
```

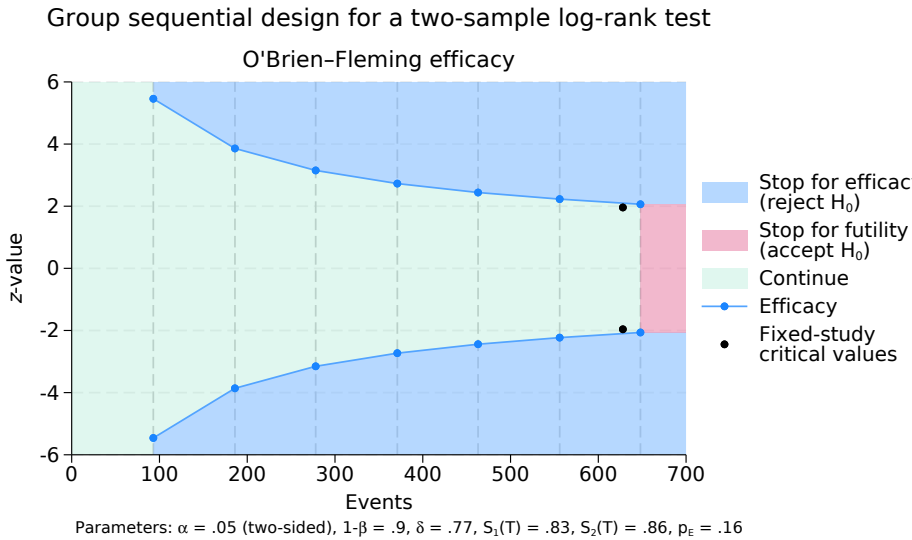

Figure 3. BHAT trial with O'Brien–Fleming efficacy bounds

At the top of the output, `gsdesign` displays a description of the trial with null and alternative hypotheses as well as study parameters. We see that the survival probabilities 0.8254 and 0.8625 correspond to a hazard ratio of 0.7709, which is the effect size used when calculating the number of events necessary to achieve 90% power. A fixed study would require 628 events (deaths) to detect a hazard ratio of 0.7709 with 90% power, and with the specified survival probabilities, this corresponds to a sample size of 4,024.

The GSD requires a maximum of 648 events (corresponding to a sample of size 4,152) if it continues to the final look. If the null hypothesis is correct (the hazard ratio is 1) and the BHAT were to be repeated many times using this design, we would expect to observe an average of 642.71 events per trial. This is near the maximum because if the null hypothesis is true, in most replications the trial will continue to the final look; only rarely will the trial be stopped early for efficacy (which would be a type I error). If the hazard ratio is truly 0.7709 (the value under the alternative hypothesis) and the trial were to be repeated many times, we would expect an average of 459.4 events per trial. The substantial sample-size savings (try saying that five times fast) is due to the fact that many replications of the trial will correctly be stopped early for efficacy.

The log-rank statistic is asymptotically normally distributed with independent information increments, and can be compared directly against group sequential critical values (Tsiatis 1982). The critical values we calculate match those used by the PDMB Cook and DeMets (2008, 306).

At the first look, the test statistic $z_1 = 1.68 < 5.459$, so the trial continued. The test statistics at the following four looks are also in the continuation region ($z_2 = 2.24 < 3.86$, $z_3 = 2.37 < 3.152$, $z_4 = 2.30 < 2.73$, and $z_5 = 2.34 < 2.441$), bringing the trial to the sixth of seven planned looks. At the sixth look, the test statistic crosses the efficacy bound, $z_6 = 2.82 > 2.229$, which supports the PDMB's decision to stop the trial for treatment efficacy.

Two aspects of the O'Brien–Fleming bound that the PDMB found appealing were the conservative critical values early in the trial and the final critical value that is only marginally larger than the fixed-study critical value (DeMets et al. 1984). An additional advantage is that even if the trial were to continue to the final look, the O'Brien–Fleming design requires only 3% more information (deaths, in this case) than a fixed study.

While the BHAT was a success story for the use of group sequential clinical trials, it was not without its challenges (DeMets, Furberg, and Friedman 2006). The number of participants recruited was nearly equal to the desired sample size, so the power would have been almost 90% to detect the difference between the anticipated survival probabilities of 82.54% and 86.25%, but survival was higher than anticipated for both the control and intervention groups. At the sixth look, only 318 of the anticipated 556 events had been observed, and a smaller-than-anticipated number of events can reduce the power of the test. Fortunately, adherence was also better than anticipated, and the effect size was larger than anticipated. The reduced number of events observed impacted the power of the test but did not influence the probability of committing a type I error.

A potentially more vexing issue is that the efficacy critical values were calculated under the assumption of equal information increments, but the interim analyses were scheduled based on calendar time, making it impossible to enforce an evenly spaced information sequence. Severe violations of this assumption can cause excessive type I error, but the number of deaths between looks was approximately equal, and type I error control is robust to minor violations of this assumption (DeMets et al. 1984).

◁

▷ Example 4: Error-spending bounds

One of the members of the PDMB from the BHAT, David DeMets, was inspired by the experience to find a more flexible method of calculating group sequential boundaries. Lan and DeMets (1983) developed error-spending methods, which depend on the total information to be collected and the interim analyses already conducted but not on the critical values of future looks. This flexibility allows error-spending bounds to adjust to scenarios such as the BHAT, where the precise information fraction at each look is not known a priori. This framework was further extended by Lan and DeMets (1989), who introduced methods for calculating stopping boundaries based on calendar time.

Here we reimagine the BHAT trial using an error-spending approximation to the classical O'Brien–Fleming boundary (Lan and DeMets 1983). Instead of specifying evenly spaced looks, we use Method 2 from Lan and DeMets (1989, 1195) to specify the timing of interim looks based on calendar time. To do this, we use the `information()` option instead of the `nlooks()` option, and we specify the timing of each look as the number of months since June 1979, when the study began accruing participants. We graph the bounds and label the x axis with the number of months since June 1979.

```
. gsdesign logrank 0.8254 0.8625, power(0.9) efficacy(errobfleming)
> information(11 16 21 28 34 40 48)
> graphbounds(xdiminformation xtitle("Months"))

Group sequential design for two-sample comparison of survivor functions
Log-rank test, Freedman method
H0: HR = 1 versus Ha: HR != 1

Efficacy: Error-spending O'Brien-Fleming style

Study parameters:
      alpha = 0.0500  (two-sided)
      power = 0.9000
      delta = 0.7709  (hazard ratio)
     hratio = 0.7709

Censoring:
         s1 = 0.8254
         s2 = 0.8625
       Pr_E = 0.1560

Expected number of events:
         H0 = 641.04
         Ha = 461.13

Info. ratio = 1.0280
    E fixed =    628
    N fixed =  4,024
      N max =  4,136
     N1 max =  2,068
     N2 max =  2,068

Fixed-study crit. values = ±1.9600

Critical values, p-values, and sample sizes for a group sequential design
```

Look	Info. frac.	Efficacy Lower	Efficacy Upper	p-value	Events E
1	0.23	-4.5380	4.5380	0.0000	148
2	0.33	-3.7128	3.7128	0.0002	216
3	0.44	-3.2081	3.2081	0.0013	283
4	0.58	-2.7361	2.7361	0.0062	377
5	0.71	-2.4739	2.4739	0.0134	458
6	0.83	-2.2717	2.2717	0.0231	538
7	1.00	-2.0473	2.0473	0.0406	646

```
Note: Critical values are for z statistics; otherwise, use
      p-value boundaries.
```

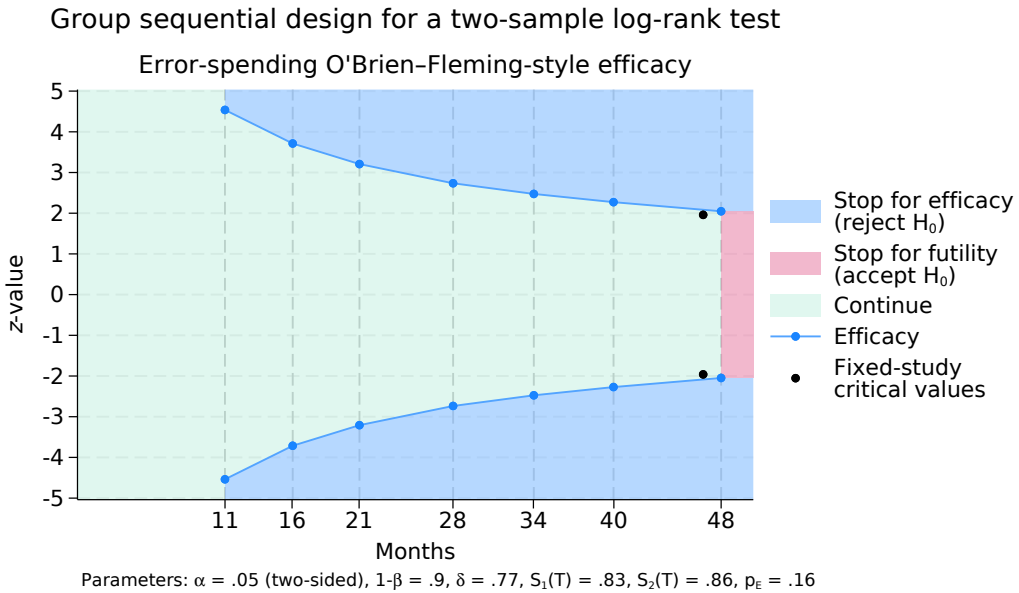

Figure 4. BHAT trial with error-spending bounds

The new design maintains the same familywise significance level, power, and effect size as the original BHAT design, so the fixed-study equivalent of the new design requires the same 628 events as the fixed equivalent of the original BHAT. Comparing the stopping boundaries of the new error-spending design against those of the original design, we see that the new critical values are quite close to those calculated using classical O'Brien–Fleming bounds with evenly spaced looks. The maximum number of events remains nearly the same, with the new design calling for 646 events at the final analysis versus 648 for the classical O'Brien–Fleming design.

More importantly, when the new error-spending boundaries are used to determine stopping for the BHAT trial, they support the same conclusion as the classical O'Brien–Fleming boundaries: to terminate the trial for efficacy at the sixth look. The first five tests statistics lie in the continuation region of the new design, but at the sixth look, $z_6 = 2.82 > 2.272$.

◁

Stored results

To calculate the fixed-study sample size, `gsdesign` *method* runs `power` *method* and returns all the method-specific stored results as well as the following common results in `r()`:

Scalars

`r(alpha)`	overall significance level (familywise type I error)
`r(beta)`	overall probability of a type II error
`r(binding)`	`1` for binding futility bounds, `0` for nonbinding
`r(E_fixed)`	total number of events (failures) in a fixed study design (survival analysis only)
`r(E_max)`	maximum observed events if the study continues to completion (survival analysis only)
`r(effparam)`	efficacy parameter (if `wtsiatis()`, `kdemets()`, or `hsdecani()` specified)
`r(Efrac_fixed)`	fractional total number of events (failures) in a fixed study design (survival analysis only)
`r(ESS0)`	expected sample size under null hypothesis
`r(ESS1)`	expected sample size under alternative hypothesis
`r(futparam)`	futility parameter (if `wtsiatis()`, `kdemets()`, or `hsdecani()` specified)
`r(info_ratio)`	ratio of maximum information required to that of a fixed study design
`r(N_fixed)`	sample size of a fixed study design
`r(N_fixedfrac)`	fractional sample size of a fixed study design
`r(N_max)`	maximum sample size if the study continues to completion
`r(N1_fixed)`	sample size of the control group in a fixed study design (multiarm trials only)
`r(N1_fixedfrac)`	fractional sample size of the control group in a fixed study design (multiarm trials only)
`r(N1_max)`	maximum sample size of the control group if the study continues to completion (multiarm trials only)
`r(N2_fixed)`	sample size of the experimental group in a fixed study design (multiarm trials only)
`r(N2_fixedfrac)`	fractional sample size of the experimental group in a fixed study design (multiarm trials only)
`r(N2_max)`	maximum sample size of the experimental group if the study continues to completion (multiarm trials only)
`r(nfractional)`	`1` if `nfractional` is specified, `0` otherwise
`r(nlooks)`	number of analyses
`r(onesided)`	`1` for a one-sided test, `0` otherwise
`r(power)`	specified overall power
`r(power_a)`	attained overall power
`r(stop)`	`0` for futility bounds, `1` for efficacy bounds, `2` for both
`r(z_fixed)`	critical value for an equivalent fixed study design

Macros

`r(cmd)`	`gsdesign`
`r(cmdline)`	command as typed
`r(direction)`	`upper`, `lower`, or `two-sided`
`r(effbnd)`	`pocock`, `obfleming`, `wtsiatis`, `errpocock`, `errobfleming`, `kdemets`, or `hsdecani`
`r(futbnd)`	`pocock`, `obfleming`, `wtsiatis`, `errpocock`, `errobfleming`, `kdemets`, or `hsdecani`
`r(method)`	method name

Matrices

`r(aspent)`	cumulative alpha spent per look (stored with efficacy-only stopping or when futility bounds are binding)
`r(aspent_fstop)`	cumulative alpha spent per look if futility stopping does occur (stored when futility bounds are nonbinding)
`r(aspent_nofstop)`	cumulative alpha spent per look if futility stopping does not occur (stored when futility bounds are nonbinding)
`r(bounds)`	stopping boundaries
`r(bspent)`	cumulative beta spent per look (when futility bounds are specified)
`r(bspent_a)`	attained cumulative beta spent per look (when futility bounds are specified)
`r(design)`	sample size and stopping boundaries at interim looks
`r(info_frac)`	specified information fraction
`r(info_frac_a)`	fraction of attained information
`r(info_level)`	specified information level
`r(p_crit)`	p-values corresponding to boundary critical values
`r(sampsize)`	sample size at interim looks

Methods and formulas

See *Methods and formulas* in [ADAPT] **gsbounds** for the formulas used to calculate the stopping boundaries, information fraction, and information ratio. See *Methods and formulas* in [PSS-2] **power** for the formulas used to calculate sample size of a fixed study design.

Methods and formulas are presented under the following headings:

Sample sizes at interim analyses
Expected sample size

Sample sizes at interim analyses

When planning a study using a GSD with K looks, we must specify the information fraction at each look, denoted as $(\mathcal{I}_1, \ldots, \mathcal{I}_K)$. For any k in $(1, \ldots, K)$, let $\mathcal{I}_k$ represent the proportion of trial data that has been collected by look k. In most cases, the information fraction is the proportion of the maximum sample size that has been collected, but for time-to-event data, the information fraction is the proportion of the total number of failure events that have been observed, not the total number of participants.

With `gsdesign`, the `information(`*numlist*`)` option can be used to specify the information fraction as a strictly increasing sequence, which is then scaled so that $\mathcal{I}_K = 1$. Alternatively, the `nlooks()` option can be used to specify the number of evenly spaced looks, and the information fraction is calculated automatically.

To determine the sample size required at each look of a GSD, we begin by calculating n_{fix}, the sample size of a fixed study design with equivalent type I and type II error. Next we calculate the information ratio, R, which is the ratio of the maximum sample size of the GSD to n_{fix}. Regardless of the properties of the study, R is always greater than 1 (see *Methods and formulas* in [ADAPT] **gsbounds** for more information).

Let $(n_1, \ldots, n_K)$ be the cumulative sample sizes at looks 1 through K, with the maximum sample size of n_K attained at the final look. For any look k in $(1, \ldots, K)$, the sample size $n_k = \mathcal{I}_k \times n_{\text{fix}} \times R$. In practice, sample sizes must be rounded up to whole numbers of participants, so `gsdesign` rounds up sample sizes unless the `nfractional` option is specified.

Expected sample size

After each group of observations is collected, an analysis is performed and the test statistic Z is calculated. In the description that follows, we assume that Z follows a standard normal distribution under H_0. For test statistics that follow other distributions, the normal model is used to calculate boundary critical values, and then p-values for the test statistics are compared with p-values corresponding to the boundary critical values. The p-value comparison is known as the significance level approach and is described in [ADAPT] **gsbounds**.

Without loss of generality, consider a GSD for an upper one-sided test with both efficacy and binding futility bounds. Denote critical values for efficacy stopping as $(e_1, \ldots, e_K)$ and critical values for futility stopping as $(f_1, \ldots, f_K)$. At interim look $k < K$, if test statistic $Z_k \geq e_k$, the trial is stopped for efficacy; if $Z_k < f_k$, the trial is stopped for futility; and if $f_k \leq Z_k < e_k$, the trial continues. At the final look, there is no continuation region because $f_K = e_K$.

The probability of stopping the trial at look k is a function of the effect size δ and is denoted as $\omega_k(\delta)$, where $\omega_1(\delta) = \Pr_\delta(Z_1 < f_1) + \Pr_\delta(Z_1 \geq e_1)$ and

$$\omega_k(\delta) = \Pr_\delta\left\{\left(Z_k < f_k \cup Z_k \geq e_k\right) \cap \bigcap_{j=1}^{k-1} f_j \leq Z_j < e_j\right\} \quad \text{for } k \in (2, \ldots, K)$$

For trials with efficacy stopping only, replace $(f_1, \ldots, f_{K-1})$ with $-\infty$ and let $f_K = e_K$. For trials with nonbinding futility bounds, replace $(f_1, \ldots, f_{K-1})$ with $-\infty$ when $\delta = 0$ but not when $\delta \neq 0$. For trials with futility stopping only, replace $(e_1, \ldots, e_{K-1})$ with ∞ and let $e_K = f_K$. For two-sided trials, replace Z_k with $|Z_k|$.

The expected sample size is a function of effect size δ and is calculated as

$$\text{ESS}(\delta) = \sum_{k=1}^{K} n_k * \omega_k(\delta)$$

References

Cook, T. D., and D. L. DeMets. 2008. *Introduction to Statistical Methods for Clinical Trials*. Boca Raton, FL: Chapman and Hall/CRC.

DeMets, D. L., C. D. Furberg, and L. M. Friedman, ed. 2006. *Data Monitoring in Clinical Trials: A Case Studies Approach*. New York: Springer.

DeMets, D. L., R. J. Hardy, L. W. Friedman, and K. K. G. Lan. 1984. Statistical aspects of early termination in the beta-blocker heart attack trial. *Controlled Clinical Trials* 5: 362–372. https://doi.org/10.1016/S0197-2456(84)80015-X.

Hwang, I. K., W. J. Shih, and J. S. de Cani. 1990. Group sequential designs using a family of type I error probability spending functions. *Statistics in Medicine* 9: 1439–1445. https://doi.org/10.1002/sim.4780091207.

Jennison, C., and B. W. Turnbull. 2000. *Group Sequential Methods with Applications to Clinical Trials*. Boca Raton, FL: Chapman and Hall/CRC.

Kim, K., and D. L. DeMets. 1987. Design and analysis of group sequential tests based on the type I error spending rate function. *Biometrika* 74: 149–154. https://doi.org/10.1093/biomet/74.1.149.

Lan, K. K. G., and D. L. DeMets. 1983. Discrete sequential boundaries for clinical trials. *Biometrika* 70: 659–663. https://doi.org/10.1093/biomet/70.3.659.

——. 1989. Group sequential procedures: Calendar versus information time. *Statistics in Medicine* 8: 1191–1198. https://doi.org/10.1002/sim.4780081003.

O'Brien, P. C., and T. R. Fleming. 1979. A multiple testing procedure for clinical trials. *Biometrics* 35: 549–556. https://doi.org/10.2307/2530245.

Pitblado, J. S., B. P. Poi, and W. W. Gould. 2024. *Maximum Likelihood Estimation with Stata*. 5th ed. College Station, TX: Stata Press.

Pocock, S. J. 1977. Group sequential methods in the design and analysis of clinical trials. *Biometrika* 64: 191–199. https://doi.org/10.1093/biomet/64.2.191.

Tsiatis, A. A. 1982. Repeated significance testing for a general class of statistics used in censored survival analysis. *Journal of the American Statistical Association* 77: 855–861. https://doi.org/10.1080/01621459.1982.10477898.

Wang, S. K., and A. A. Tsiatis. 1987. Approximately optimal one-parameter boundaries for group sequential trials. *Biometrics* 43: 193–199. https://doi.org/10.2307/2531959.

Also see

[ADAPT] **GSD intro** — Introduction to group sequential designs

[ADAPT] **gs** — Introduction to commands for group sequential design

[ADAPT] **gsbounds** — Boundaries for group sequential trials

[ADAPT] **Glossary**

[PSS-2] **power** — Power and sample-size analysis for hypothesis tests