Package 'Sobol4R'

May 7, 2026

Type Package

Title Sobol Indices for Models with Fixed and Stochastic Parameters

Version 0.4.0

Date 2025-11-26

Depends R (>= 3.5.0)

Imports ggplot2, Rcpp (>= 1.0.0), rlang, sensitivity, stats

Suggests knitr, markdown, rmarkdown, simmer, testthat (>= 3.0.0)

LinkingTo Rcpp

Author Frederic Bertrand [cre, aut] (ORCID:
<https://orcid.org/0000-0002-0837-8281>),
Elizaveta Logosha [aut],
Myriam Maumy-Bertrand [aut] (ORCID:
<https://orcid.org/0000-0002-4615-1512>)

Maintainer Frederic Bertrand <frederic.bertrand@lecnam.net>

Description Tools to design experiments, compute Sobol sensitivity indices,
and summarise stochastic responses inspired by the strategy described by
Zhu and Sudret (2021) <doi:10.1016/j.ress.2021.107815>. Includes helpers
to optimise toy models implemented in C++, visualise indices with
uncertainty quantification, and derive reliability-oriented sensitivity
measures based on failure probabilities.
It is further detailed in Logosha, Maumy and Bertrand (2022)
<doi:10.1063/5.0246026> and (2023) <doi:10.1063/5.0246024> or in Bertrand,
Logosha and Maumy (2024) <https://hal.science/hal-05371803>,
<https://hal.science/hal-05371795> and <https://hal.science/hal-05371798>.

LazyLoad yes

VignetteBuilder knitr

License GPL-3

Encoding UTF-8

URL https://fbertran.github.io/Sobol4R/,
https://github.com/fbertran/Sobol4R

BugReports https://github.com/fbertran/Sobol4R/issues

RoxygenNote 7.3.3

NeedsCompilation yes

Repository CRAN

Date/Publication 2025-12-02 15:00:02 UTC

Contents

`Sobol4R-package` *Sobol4R-package*

Description

Tools to design experiments, compute Sobol sensitivity indices, and summarise stochastic responses inspired by the strategy described by Zhu and Sudret (2021) doi:10.1016/j.ress.2021.107815. Includes helpers to optimise toy models implemented in C++, visualise indices with uncertainty quantification, and derive reliability-oriented sensitivity measures based on failure probabilities. It is further detailed in Logosha, Maumy and Bertrand (2022) doi:10.1063/5.0246026 and (2023) doi:10.1063/5.0246024 or in Bertrand, Logosha and Maumy (2024) https://hal.science/hal-05371803, https://hal.science/hal-05371795 and https://hal.science/hal-05371798.

Author(s)

Maintainer: Frederic Bertrand <frederic.bertrand@lecnam.net> (ORCID)

Authors:

- Elizaveta Logosha <elizaveta.logosha@utt.fr>
- Myriam Maumy-Bertrand <myriam.maumy@ehesp.fr> (ORCID)

References

Elizaveta Logosha, Myriam Maumy, Frederic Bertrand; Confidence interval determination using discrete event simulations for real estate sales case. AIP Conf. Proc. 31 March 2025; 3182 (1): 100008. doi:10.1063/5.0246026.

Elizaveta Logosha, Myriam Maumy, Frédéric Bertrand; Sensitivity analysis of stochastic simulator in the case of sales date prediction. AIP Conf. Proc. 31 March 2025; 3182 (1): 100001. doi:10.1063/5.0246024.

Frédéric Bertrand, Elizaveta Logosha, Myriam Maumy-Bertrand. Extension of sensitivity analysis to uncertainties in distribution parameters. 32nd Conference on Intelligent Systems for Molecular Biology, International Society for Computational Biology, Jul 2024, Montreal (QC), Canada. https://hal.science/hal-05371795.

Frédéric Bertrand, Elizaveta Logosha, Myriam Maumy-Bertrand. Sobol4RV: Global Sensitivity Analysis in Several Random Settings. BioC 2024, BioConductor, Jul 2024, Grand Rapids, MI, United States. https://hal.science/hal-05371803

Frédéric Bertrand, Elizaveta Logosha, Myriam Maumy-Bertrand. Global Sensitivity Analysis in Several Random Settings. 2024 Joint Statistical Meetings, American Statistical Association, Aug 2024, Portland (OR), United States. https://hal.science/hal-05371798.

See Also

`sobol4r_design()`, `sobol4r_qoi_indices()`, `vignette("Sobol_RV_five_examples", package = "Sobol4R")`, `vignette("Sobol4R_vignette_stochastic", package = "Sobol4R")`, `vignette("Sobol4R_vignette_`
`package = "Sobol4R")` and `vignette("simmer_MM1_Sobol_example", package = "Sobol4R")`.

Examples

```
ex1_results <- sobol_example_g_deterministic(n=100, nboot=10)
print(ex1_results)
autoplot(ex1_results, ncol = 1)
rm(ex1_results)
```

Autoplot implementations

Autoplot implementations

Description

Provide a ggplot visualisation when ggplot2 is available, otherwise fallback to a lightweight base R bar chart. Supports the custom `sobol_result` class used in this package, compact `sobol_summary` data frames, and `sensitivity::sobol` objects.

Usage

```
autoplot(object, ...)

## S3 method for class 'sobol_result'
autoplot(
  object,
  show_uncertainty = FALSE,
  probs = c(0.1, 0.9),
  bootstrap = 200L,
  ...
)

## S3 method for class 'sobol'
autoplot(object, separate_panels = TRUE, ncol = 2, ...)

## S3 method for class 'sobol2007'
autoplot(object, ...)

## S3 method for class 'soboljansen'
autoplot(object, ...)

## S3 method for class 'sobolEff'
autoplot(object, ...)

## S3 method for class 'sobolmartinez'
autoplot(object, ...)

## S3 method for class 'sobol_summary'
autoplot(object, ...)
```

Arguments

object	A sobol_result, sobol_summary, or sensitivity::sobol instance.
...	Further arguments passed to the plotting backend.
show_uncertainty	Logical, when TRUE bootstrap quantiles are computed (if available) and displayed as error bars.
probs	Numeric vector of probabilities used for the uncertainty bars.
bootstrap	Integer indicating how many bootstrap resamples to draw when show_uncertainty = TRUE.
separate_panels	Should the indices be plotted on separate panels according to their order? If separate_panels = TRUE, the first order indices are separated from the higher orders ones.
ncol	If separate_panels = TRUE, the number of columns for the facet wrapping of the plot.

Value

A ggplot object when ggplot2 is installed, otherwise the bar centres invisibly.

---

bootstrap_indices *Bootstrap Sobol indices from stored samples*

Description

Recompute Sobol first- and total-order indices from stored sample matrices using bootstrap resampling. Falls back to deterministic values when no samples are available.

Usage

```
bootstrap_indices(result, bootstrap)
```

Arguments

result	A sobol_result object produced by sobol_indices().
bootstrap	Integer indicating how many bootstrap replicates to draw.

Value

A list with matrices first and total containing the bootstrap replications.

estimate_failure_probability *Estimate Failure Probability from Simulator Outputs*

Description

Convenient helper to compute the reliability-related probabilities described in Lebrun et al. (2021). The failure domain is controlled by a threshold and an inequality direction.

Usage

```
estimate_failure_probability(response, threshold, less = TRUE, weights = NULL)
```

Arguments

response	Numeric vector of simulator evaluations.
threshold	Numeric scalar defining the failure boundary.
less	Logical, failure is defined as response <= threshold when TRUE and response >= threshold otherwise.
weights	Optional numeric vector of non-negative weights. The vector is normalised internally when supplied.

Value

A list containing the estimated probability and its variance.

Examples

```
y <- rnorm(1000)
estimate_failure_probability(y, threshold = -1)
```

ishigami_model *Fast Ishigami Test Function*

Description

C++ implementation of the Ishigami function that is widely used as a benchmark for Sobol sensitivity indices. The implementation is vectorised and therefore convenient for Monte Carlo experiments.

Usage

```
ishigami_model(x, a = 7, b = 0.1)
```

Arguments

x	Numeric matrix with three columns representing the inputs.
a	Numeric scalar controlling the nonlinear term.
b	Numeric scalar controlling the interaction term.

Value

Numeric vector of simulator outputs.

Examples

```
x <- matrix(runif(30, -pi, pi), ncol = 3)
ishigami_model(x)
```

---

process_fun_indiv *Time to M successes for one individual*

Description

Stochastic model that simulates successive units until M successes occur, and returns the time when the M-th success happens.

Usage

```
process_fun_indiv(X_indiv, M = 50)
```

Arguments

X_indiv	Numeric vector `c(lambda1, lambda2, lambda3, p1, p2)`.
M	Target number of successes.

Value

Scalar time to M successes, with attribute `"success"`.

process_fun_mean_to_M *QoI wrapper for the process model*

Description

For each row of X, evaluates process_fun_row_wise several times and returns the mean time to M successes.

Usage

```
process_fun_mean_to_M(X, M = 50, nrep = 10)
```

Arguments

X	Matrix or data.frame of parameters.
M	Target number of successes.
nrep	Number of repetitions for the QoI.

Value

Numeric vector of QoI values.

process_fun_row_wise *Process model for a matrix of individuals*

Description

Applies process_fun_indiv row-wise to a matrix of parameters.

Usage

```
process_fun_row_wise(X, M = 50)
```

Arguments

X	Matrix or data.frame with columns lambda1, lambda2, lambda3, p1, p2.
M	Target number of successes.

Value

Numeric vector of length nrow(X).

sobol4r_clinic_model *Two-step clinic model wrapper for Sobol designs*

Description

Simulate a simple clinic with separate registration and examination stages using **simmer**. The quantity of interest is the mean time in system over `nrep` replications for each parameter set.

Usage

```
sobol4r_clinic_model(
  X,
  cap_reg = 2,
  cap_exam = 3,
  horizon = 2000,
  warmup_prob = 0.2,
  nrep = 10L
)
```

Arguments

`X`	Design matrix or data.frame with columns `lambda` (arrival rate), `mu_reg` (registration service rate), and `mu_exam` (examination service rate).
`cap_reg`, `cap_exam`	Integer capacities for the registration and examination resources.
`horizon`	Simulation horizon.
`warmup_prob`	Fraction of the horizon treated as warmup and discarded before computing the mean time in system.
`nrep`	Number of replications used to average the mean time in system.

Value

Numeric vector of length `nrow(X)`.

sobol4r_design *Design generation for Sobol indices*

Description

Simple helper that wraps `sensitivity::sobol` with `model = NULL` to create the extended design matrix used to evaluate the model.

Usage

```
sobol4r_design(
  X1,
  X2,
  order = 2,
  nboot = 0,
  type = c("soboljansen", "sobol", "sobol2007", "sobolEff", "sobolmartinez"),
  ...
)
```

Arguments

X1	First sample (matrix or data.frame).
X2	Second sample (matrix or data.frame).
order	Maximum interaction order (1 or 2).
nboot	Number of bootstrap replicates for confidence intervals.
type	Type of Monte Carlo Estimation of Sobol' Indices to be used. Supported estimators mirror the **sensitivity** helpers: sobol, sobol2007, soboljansen, sobolEff, and sobolmartinez. Defaults to "soboljansen", which is the safest general-purpose choice for both deterministic and stochastic simulators.
...	Additional arguments passed to sensitivity::sobol.

Value

An object of class "sobol" whose $X field contains the design matrix. You should evaluate your model on $X and then call sensitivity::tell().

---

sobol4r_mm1_model *M/M/1 queue model wrapper for Sobol designs*

Description

Evaluate a simple M/M/1 queue built with **simmer** for each row of a Sobol design matrix. The quantity of interest is the mean time in system across nrep independent replications.

Usage

```
sobol4r_mm1_model(X, horizon = 1000, warmup = 200, nrep = 20L)
```

Arguments

X	Design matrix or data.frame with columns lambda (arrival rate) and mu (service rate).
horizon	Simulation horizon.
warmup	Warmup period; arrivals ending before this time are discarded from the summary statistic.
nrep	Number of replications used to average the mean time in system.

Value

Numeric vector of length `nrow(X)`.

`sobol4r_qoi_indices` *Generic QoI-based Sobol indices for a stochastic model*

Description

This function extends the classical Sobol indices to a stochastic simulator by first computing a quantity of interest (QoI) for each input point, such as the mean of repeated runs.

Usage

```
sobol4r_qoi_indices(
  model,
  X1,
  X2,
  qoi_fun = base::mean,
  nrep = 1000,
  order = 2,
  nboot = 0,
  type = c("soboljansen", "sobol", "sobol2007", "sobolEff", "sobolmartinez"),
  ...
)
```

Arguments

`model`	Stochastic model function that takes a matrix or data.frame `X` and returns a numeric vector of length `nrow(X)`.
`X1, X2`	Two base designs (matrices or data.frames).
`qoi_fun`	Function used to summarize the repetitions (default is `mean`).
`nrep`	Number of repetitions of the stochastic model for each design point.
`order`	Maximum interaction order (1 or 2).
`nboot`	Number of bootstrap replicates for Sobol indices.
`type`	Which estimator to use. Any **sensitivity** Sobol helper is supported: `"sobol"`, `"sobol2007"`, `"soboljansen"`, `"sobolEff"`, or `"sobolmartinez"`. Defaults to `"soboljansen"`, the most robust general-purpose choice.
`...`	Additional arguments passed to `model`.

Value

An object of class `"sobol"` with QoI-based Sobol indices.

sobol4r_run *Run Sobol analysis with optional QoI wrapper*

Description

Helper around `sensitivity::sobol` that mimics the structure of the original scripts. It never writes to disk.

Usage

```
sobol4r_run(
  model,
  X1,
  X2,
  order = 2,
  nboot = 100L,
  qoi_fun = NULL,
  nrep = 1L,
  type = c("soboljansen", "sobol", "sobol2007", "sobolEff", "sobolmartinez"),
  ...
)
```

Arguments

`model`	Deterministic or stochastic model that takes a design X and returns a numeric vector of length nrow(X).
`X1`, `X2`	Matrices or data.frames used to build the Sobol design.
`order`	Order of the Sobol indices (1 or 2).
`nboot`	Number of bootstrap replicates for confidence intervals.
`qoi_fun`	Optional quantity of interest function. If not NULL, the model is evaluated repeatedly and QoI is computed row wise.
`nrep`	Number of replications per design row when `qoi` is not NULL.
`type`	Type of Monte Carlo Estimation of Sobol' Indices to be used. Supported estimators mirror the **sensitivity** helpers: `sobol`, `sobol2007`, `soboljansen`, `sobolEff`, and `sobolmartinez`. Defaults to `"soboljansen"` because it offers robust first and total order indices on both centred and non-centred outputs.
`...`	Extra arguments passed to `model`.

Value

A `sobol` object (output of `sensitivity::tell`).

sobol_design *Create Sobol Sampling Designs*

Description

Generate the two-sample matrices (A and B) that are required to apply Monte Carlo Sobol estimators. The helper can rely on pseudo random numbers or on a light-weight Halton low-discrepancy sequence to increase coverage.

Usage

```
sobol_design(
  n,
  d,
  lower = rep(0, d),
  upper = rep(1, d),
  quasi = FALSE,
  seed = NULL
)
```

Arguments

n	Integer, number of rows per design matrix.
d	Integer, number of model parameters.
lower	Numeric vector of length d containing lower bounds.
upper	Numeric vector of length d containing upper bounds.
quasi	Logical, when TRUE a Halton sequence is used.
seed	Optional integer used to initialise the RNG state.

Value

A list with matrices A and B plus the column names.

Examples

```
design <- sobol_design(n = 64, d = 3, quasi = TRUE)
str(design)
```

`sobol_example_covariate_large`
Example 3: Large covariate dependent random effect

Description

Third input C3 is uniform on [1, 100], used as the mean of a Gaussian noise term added to the G-function. Quantity of interest is the mean of repeated evaluations.

Usage

```
sobol_example_covariate_large(
  n = 50000,
  nrep_qoi = 1000,
  order = 2,
  nboot = 100
)
```

Arguments

`n`	Monte Carlo sample size for each base design.
`nrep_qoi`	Number of repetitions for the QoI.
`order`	Maximum interaction order.
`nboot`	Number of bootstrap replicates.

Value

A list with two `"sobol"` objects: `x_single` (single noisy run), `x_qoi` (QoI-based indices).

`sobol_example_covariate_small`
Example 4: Slight covariate dependent random effect

Description

Same as `sobol_example_covariate_large` but with C3 uniform on [1, 1.5], that is with a much smaller range for the mean of the Gaussian noise.

Usage

```
sobol_example_covariate_small(
  n = 50000,
  nrep_qoi = 1000,
  order = 2,
  nboot = 100
)
```

Arguments

n	Monte Carlo sample size for each base design.
nrep_qoi	Number of repetitions for the QoI.
order	Maximum interaction order.
nboot	Number of bootstrap replicates.

Value

A list with two "sobol" objects: x_single (single noisy run), x_qoi (QoI-based indices).

---

sobol_example_g_deterministic

Example 1: Deterministic G-function (reference case)

Description

Reproduces the classical non-random Sobol analysis on the G-function with k = 8 inputs on [0, 1].

Usage

```
sobol_example_g_deterministic(
  n = 50000,
  a = c(0, 1, 4.5, 9, 99, 99, 99, 99),
  order = 2,
  nboot = 100
)
```

Arguments

n	Monte Carlo sample size for each base design.
a	Parameter vector for the G-function.
order	Maximum interaction order for Sobol indices.
nboot	Number of bootstrap replicates.

Value

An object of class "sobol".

sobol_example_process *Example 5: Sobol indices for the process model*

Description

Computes Sobol indices for the simple process example with random distributional parameters. Uses both a single trajectory and a QoI based on repeated runs.

Usage

```
sobol_example_process(n = 100, M = 50, nrep_qoi = 10, order = 1, nboot = 10)
```

Arguments

n	Monte Carlo sample size for each base design.
M	Target number of successes.
nrep_qoi	Number of repetitions for the QoI.
order	Maximum interaction order.
nboot	Number of bootstrap replicates.

Value

A list with two "sobol" objects: xp_single and xp_qoi.

sobol_example_random_output *Example 2: Random effect on the output (constant Gaussian noise)*

Description

Two inputs in [0, 1], Sobol G-function with k = 2, plus additive Gaussian noise, and a QoI based on the mean of repeated evaluations.

Usage

```
sobol_example_random_output(
  n = 50000,
  sd = 1,
  nrep_qoi = 1000,
  order = 2,
  nboot = 100
)
```

Arguments

n	Monte Carlo sample size for each base design.
sd	Standard deviation of the Gaussian noise.
nrep_qoi	Number of repetitions for the QoI.
order	Maximum interaction order.
nboot	Number of bootstrap replicates.

Value

A list with three "sobol" objects: x_det (deterministic G-function), x_noise (single noisy output), x_qoi (QoI-based indices).

---

sobol_g2_additive_noise

Additive Gaussian noise on the Sobol G-function (k = 2) - C++ backend

---

Description

Additive Gaussian noise on the Sobol G-function (k = 2) - C++ backend

Usage

```
sobol_g2_additive_noise(X, sd = 1, a = c(0, 1, 4.5, 9, 99, 99, 99, 99))
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
sd	Standard deviation of the Gaussian noise.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of model outputs with noise.

sobol_g2_additive_noise_R

Additive Gaussian noise on the Sobol G-function (k = 2)

Description

Additive Gaussian noise on the Sobol G-function (k = 2)

Usage

```
sobol_g2_additive_noise_R(X, sd = 1, a = c(0, 1, 4.5, 9, 99, 99, 99, 99))
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
sd	Standard deviation of the Gaussian noise.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of model outputs with noise.

sobol_g2_function

Sobol G-function restricted to the first two inputs - C++ backend

Description

Convenience wrapper around `sobol_g_function` that uses only the first two columns of `X`.

Usage

```
sobol_g2_function(X, a = c(0, 1, 4.5, 9, 99, 99, 99, 99))
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of length `nrow(X)` with model outputs.

sobol_g2_qoi_covariate_mean

QoI wrapper for covariate noisy G-function (k = 2) - C++ backend

Description

Computes a mean over repeated evaluations of the noisy model.

Usage

```
sobol_g2_qoi_covariate_mean(
  X,
  nrep = 1000,
  a = c(0, 1, 4.5, 9, 99, 99, 99, 99)
)
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
nrep	Number of replicates used for the QoI.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of QoI values (means over nrep runs).

sobol_g2_qoi_covariate_mean_R

Quantity-of-interest wrapper for the covariate noisy G-function (k = 2)

Description

Computes a mean over repeated evaluations of the noisy model.

Usage

```
sobol_g2_qoi_covariate_mean_R(
  X,
  nrep = 1000,
  a = c(0, 1, 4.5, 9, 99, 99, 99, 99)
)
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
nrep	Number of replicates used for the QoI.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of QoI values (means over nrep runs).

---

sobol_g2_qoi_mean *QoI wrapper for the noisy G-function (k = 2) - C++ backend*

Description

Computes a mean over repeated evaluations of the noisy model.

Usage

```
sobol_g2_qoi_mean(X, nrep = 1000, sd = 1, a = c(0, 1, 4.5, 9, 99, 99, 99, 99))
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
nrep	Number of replicates used for the QoI.
sd	Standard deviation of the Gaussian noise.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of QoI values (means over nrep runs).

---

sobol_g2_qoi_mean_R *Quantity-of-interest wrapper for the noisy G-function (k = 2)*

Description

Computes a mean over repeated evaluations of the noisy model.

Usage

```
sobol_g2_qoi_mean_R(
  X,
  nrep = 1000,
  sd = 1,
  a = c(0, 1, 4.5, 9, 99, 99, 99, 99)
)
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
nrep	Number of replicates used for the QoI.
sd	Standard deviation of the Gaussian noise.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of QoI values (means over nrep runs).

---

sobol_g2_R *Sobol G-function restricted to the first two inputs*

Description

Convenience wrapper around sobol_g_function that uses only the first two columns of X.

Usage

```
sobol_g2_R(X, a = c(0, 1, 4.5, 9, 99, 99, 99, 99))
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of model outputs.

---

sobol_g2_with_covariate_noise *Covariate dependent Gaussian noise on the Sobol G-function (k = 2) - C++ backend*

Description

Covariate dependent Gaussian noise on the Sobol G-function (k = 2) - C++ backend

Usage

```
sobol_g2_with_covariate_noise(X, a = c(0, 1, 4.5, 9, 99, 99, 99, 99))
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of model outputs with noise.

---

sobol_g2_with_covariate_noise_R

Additive Gaussian noise on the Sobol G-function (k = 2)

Description

Additive Gaussian noise on the Sobol G-function (k = 2)

Usage

```
sobol_g2_with_covariate_noise_R(X, a = c(0, 1, 4.5, 9, 99, 99, 99, 99))
```

Arguments

X	Numeric matrix or data.frame with at least two columns.
a	Numeric vector of parameters (at least length 2).

Value

Numeric vector of model outputs with noise.

---

sobol_g_function

Sobol G-function (Saltelli reference function) - C++ backend

Description

Generic implementation of the Sobol G-function for k inputs. Columns of `X` are interpreted as inputs X1, X2, ..., Xk.

Usage

```
sobol_g_function(X, a = c(0, 1, 4.5, 9, 99, 99, 99, 99))
```

Arguments

X	Numeric matrix or data.frame of inputs in [0, 1].
a	Numeric vector of parameters a_j controlling importance. Its length must be at least `ncol(X)`.

Value

Numeric vector of length `nrow(X)` with model outputs.

`sobol_g_R` *Sobol G-function (Saltelli reference function)*

Description

Generic implementation of the Sobol G-function for k inputs. Columns of `X` are interpreted as inputs X1, X2, ..., Xk.

Usage

```
sobol_g_R(X, a = c(0, 1, 4.5, 9, 99, 99, 99, 99))
```

Arguments

`X`	Numeric matrix or data.frame of inputs in [0, 1].
`a`	Numeric vector of parameters a_j controlling importance. Its length must be at least `ncol(X)`.

Value

Numeric vector of length `nrow(X)` with model outputs.

`sobol_indices` *Sobol Indices for Stochastic Simulators*

Description

Estimate first-order and total-order Sobol indices using Monte Carlo estimators that support noisy outputs via independent replicates.

Usage

```
sobol_indices(
  model,
  design,
  replicates = 1L,
  estimator = c("jansen", "saltelli"),
  keep_samples = FALSE,
  ...
)
```

Arguments

model	Function receiving a numeric matrix and returning a numeric vector of responses. The function may include internal randomness.
design	Output of sobol_design().
replicates	Integer, number of repeated evaluations to average out the model noise. Defaults to one replicate (deterministic behaviour).
estimator	Character string, either "saltelli" or "jansen". Defaults to "jansen".
keep_samples	When TRUE, store all simulated values.
...	Further arguments passed to model.

Details

Two families of estimators are available:

- "saltelli": Saltelli-type estimator with internal centering of the model outputs before variance and index computation.
- "jansen": Jansen-type estimator based on variances of output differences, which is numerically stable in many settings.

Value

An object of class sobol_result containing the indices, intermediate estimates, and the Monte Carlo variance.

Examples

```
design <- sobol_design(n = 128, d = 3, quasi = TRUE)
model <- function(x) ishigami_model(x)
result <- sobol_indices(model, design, replicates = 4)
result$data
```

---

sobol_reliability *Reliability-Oriented Sobol Indices*

Description

Transform stored simulator samples into Sobol indices for the binary failure indicator described by Lebrun et al. (2021). The function reuses the Saltelli-type estimator from sobol_indices() and therefore requires a previous call with keep_samples = TRUE.

Usage

```
sobol_reliability(result, threshold, less = TRUE)
```

Arguments

result	Output of sobol_indices() computed with keep_samples = TRUE.
threshold	Numeric scalar defining the failure boundary.
less	Logical, when TRUE failures correspond to response <= threshold; otherwise, failures correspond to response >= threshold.

Value

A sobol_result instance storing the Sobol indices of the failure indicator along with the estimated failure probability and its variance.

Examples

```
design <- sobol_design(n = 128, d = 3, lower = rep(-pi, 3), upper = rep(pi, 3))
stochastic <- sobol_indices(ishigami_model, design, replicates = 3,
                            keep_samples = TRUE)
failure <- sobol_reliability(stochastic, threshold = -1)
Sobol4R::autoplot(failure, show_uncertainty = TRUE)
```

---

summarise_sobol *Summarise Sobol Indices*

Description

Compute compact summaries of the Sobol indices and their Monte Carlo variability. The function is intended to feed diagnostic plots.

Usage

```
summarise_sobol(result, probs = c(0.1, 0.5, 0.9), bootstrap = 200L)
```

Arguments

result	A sobol_result object.
probs	Numeric vector of probabilities used to report quantiles of the empirical bootstrap distribution.
bootstrap	Integer, number of bootstrap resamples used to quantify the estimator uncertainty.

Value

A data frame (class sobol_summary) with the requested statistics. Quantile columns are added when probs is not empty.

Examples

```
design <- sobol_design(n = 64, d = 3)
model <- function(x) ishigami_model(x)
sob <- sobol_indices(model, design, keep_samples = TRUE)
summarise_sobol(sob, probs = c(0.1, 0.9))
```

Index