Package 'likelihood.contr'

April 1, 2026

Type Package

Title Likelihood Contribution Models for Heterogeneous Observation Types

Version 0.1.1

Depends R (>= 3.5.0)

Description Constructs likelihood models from heterogeneous observation types by composing named contributions. Each observation type (exact, left-censored, right-censored, interval-censored, or custom) contributes independently to the total log-likelihood, which is summed under an i.i.d. assumption. Provides contr_name() for standard R distributions and contr_fn() for user-defined contributions, composed via likelihood_contr() into objects compatible with the likelihood.model inference framework.

License MIT + file LICENSE

URL https://github.com/queelius/likelihood.contr

BugReports https://github.com/queelius/likelihood.contr/issues

Encoding UTF-8

Imports likelihood.model, numDeriv

Suggests testthat (>= 3.0.0), knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.3.3

Config/testthat/edition 3

NeedsCompilation no

Author Alexander Towell [aut, cre] (ORCID: <https://orcid.org/0000-0001-6443-9897>)

Maintainer Alexander Towell <lex@metafunctor.com>

Repository CRAN

Date/Publication 2026-04-01 20:00:07 UTC

Contents

`assumptions.likelihood_contr`
Assumptions for likelihood_contr

Description

Returns the character vector of model assumptions stored in the `likelihood_contr` object. The `"iid"` assumption is always included.

Usage

```
## S3 method for class 'likelihood_contr'
assumptions(model, ...)
```

Arguments

`model`	A `likelihood_contr` model.
`...`	Additional arguments (currently unused).

Value

Character vector of assumptions.

Examples

```
model <- likelihood_contr(
  obs_type = "status",
  exact = contr_name("exp", "exact", ob_col = "t"),
  assumptions = c("exponential distribution", "non-informative censoring")
)
assumptions(model)
```

contr_fn *Create a likelihood contribution from user-supplied functions*

Description

Wraps user-provided log-likelihood, score, and Hessian functions into a "contr" S3 object suitable for use in likelihood_contr().

Usage

```
contr_fn(loglik, score = NULL, hess = NULL)
```

Arguments

loglik	A function function(df, par, ...) returning the log-likelihood contribution (scalar).
score	Optional function function(df, par, ...) returning the score vector. If NULL, numerical differentiation via numDeriv::grad() is used.
hess	Optional function function(df, par, ...) returning the Hessian matrix. If NULL, numerical differentiation via numDeriv::hessian() is used.

Details

The loglik function must have signature function(df, par, ...) and return a scalar log-likelihood value. The optional score and hess functions must have the same signature and return a numeric vector (score) or matrix (Hessian), respectively.

Value

A "contr" S3 object (a list) with elements $loglik (function), $score (function or NULL), and $hess (function or NULL). Pass this object to likelihood_contr() to include it in a composed model.

Examples

```
# Exponential exact-observation contribution
my_contr <- contr_fn(
  loglik = function(df, par, ...) {
    sum(dexp(df$x, rate = par[1], log = TRUE))
  },
  score = function(df, par, ...) {
    n <- nrow(df)
    c(rate = n / par[1] - sum(df$x))
  }
)
```

contr_name *Create a likelihood contribution from a named R distribution*

Description

Generates a "contr" object for a standard R distribution using the d<name> (PDF) and p<name> (CDF) functions. Exact observations use the PDF, right-censored use the survival function, left-censored use the CDF, and interval-censored use the CDF difference.

Usage

```
contr_name(dist_name, type, ob_col = "x", ob_col_upper = NULL)
```

Arguments

dist_name	The distribution name (e.g., "norm", "weibull", "exp"). Must have both d<name> and p<name> functions available.
type	One of "exact", "right", "left", "interval".
ob_col	Name of the observation column in the data frame. For interval censoring, this is the lower bound column.
ob_col_upper	Name of the upper bound column for interval censoring. Required when type = "interval".

Details

The log-likelihood form for each contribution type:

- "exact": PDF via d<name>(x, ..., log = TRUE)
- "right": Survival function via p<name>(x, ..., lower.tail = FALSE, log.p = TRUE)
- "left": CDF via p<name>(x, ..., log.p = TRUE)
- "interval": log(F(upper) - F(lower)) via p<name>, computed in log-space for numerical stability

Value

A "contr" S3 object (a list) with a $loglik function derived from the named distribution. The $score and $hess elements are NULL; numerical differentiation is used automatically when the contribution is part of a likelihood_contr() model.

Examples

```
# Exact Weibull contribution
exact <- contr_name("weibull", "exact", ob_col = "t")

# Right-censored Weibull contribution
right <- contr_name("weibull", "right", ob_col = "t")
```

```
# Interval-censored normal contribution
interval <- contr_name("norm", "interval", ob_col = "lo", ob_col_upper = "hi")
```

`hess_loglik.likelihood_contr` *Hessian of the log-likelihood for likelihood_contr*

Description

Returns a closure that splits the data frame by observation type and sums the per-type Hessian matrices. If a contribution does not provide an analytical Hessian, numerical differentiation via `numDeriv::hessian()` is used.

Usage

```
## S3 method for class 'likelihood_contr'
hess_loglik(model, ...)
```

Arguments

`model`	A `likelihood_contr` model.
`...`	Additional arguments (currently unused).

Value

A function `function(df, par, ...)` returning the total Hessian matrix.

Examples

```
model <- likelihood_contr(
  obs_type = "status",
  exact = contr_name("exp", "exact", ob_col = "t")
)
df <- data.frame(t = c(0.5, 1.0, 1.5), status = "exact")
hess_fn <- hess_loglik(model)
hess_fn(df, par = c(rate = 2))
```

`likelihood_contr` *Compose a likelihood model from heterogeneous observation contributions*

Description

Constructs a `"likelihood_contr"` model by combining named `"contr"` objects. The model splits a data frame by observation type, evaluates each type's contribution, and sums the results (under the i.i.d. assumption).

Usage

```
likelihood_contr(obs_type, ..., rdata_fn = NULL, assumptions = character(0))
```

Arguments

`obs_type`	Either a string (column name) or function for determining observation types. See Details.
`...`	Named `"contr"` objects, one per observation type.
`rdata_fn`	Optional function `function(theta, n, ...)` that generates random data from the model's data-generating process. Required by the default `likelihood.model::fim.likelihood_mod` method.
`assumptions`	Character vector of model assumptions. `"iid"` is always included.

Details

Observation type dispatch is controlled by `obs_type`:

- If a **string**, it names a column in the data frame whose values are matched against the contribution names.
- If a **function**, it is called as `obs_type(df)` and must return a character vector of the same length as `nrow(df)`.

Value

A `"likelihood_contr"` S3 object (inheriting from `"likelihood_model"`) containing the contributions, dispatch method, and assumptions. Use `loglik()`, `score()`, `hess_loglik()` to obtain inference closures, or `generics::fit()` from the likelihood.model package to perform maximum likelihood estimation.

Examples

```
# Weibull model with exact and right-censored observations
model <- likelihood_contr(
  obs_type = "status",
  exact = contr_name("weibull", "exact", ob_col = "t"),
  right = contr_name("weibull", "right", ob_col = "t"),
```

```
  assumptions = c("Weibull distribution", "non-informative censoring")
)
```

`loglik` *Re-exported generics from likelihood.model*

Description

These generics are re-exported so that users can call them directly after loading `likelihood.contr`, without loading `likelihood.model` separately.

Arguments

`model`	A likelihood model object (e.g., a `"likelihood_contr"` object).
`...`	Additional arguments passed to methods.

Details

`loglik(model, ...)` Returns a closure `function(df, par, ...)` that evaluates the total log-likelihood (a scalar).

`score(model, ...)` Returns a closure `function(df, par, ...)` that evaluates the score vector (gradient of the log-likelihood).

`hess_loglik(model, ...)` Returns a closure `function(df, par, ...)` that evaluates the Hessian matrix of the log-likelihood.

`assumptions(model, ...)` Returns a character vector of model assumptions.

`rdata(model, ...)` Returns a closure `function(theta, n, ...)` that generates a random data frame from the model.

See `loglik` for full documentation.

Value

The return type depends on the generic; see Details. `loglik`, `score`, `hess_loglik`, and `rdata` each return a **closure** (function). `assumptions` returns a character vector.

loglik.likelihood_contr

Log-likelihood for likelihood_contr

Description

Returns a closure that splits the data frame by observation type, evaluates each contribution's log-likelihood, and sums them. The data-frame split is cached so repeated calls with the same `df` (e.g., during optimization) skip the split step.

Usage

```
## S3 method for class 'likelihood_contr'
loglik(model, ...)
```

Arguments

`model`	A `likelihood_contr` model.
`...`	Additional arguments (currently unused).

Value

A function `function(df, par, ...)` returning the total log-likelihood (scalar).

Examples

```
model <- likelihood_contr(
  obs_type = "status",
  exact = contr_name("exp", "exact", ob_col = "t"),
  right = contr_name("exp", "right", ob_col = "t")
)
df <- data.frame(t = c(1, 2, 3, 4), status = c("exact", "exact", "right", "right"))
ll_fn <- loglik(model)
ll_fn(df, par = c(rate = 0.5))
```

print.likelihood_contr

Print method for likelihood_contr

Description

Prints a summary of the model's observation types, dispatch method (column name or function), and assumptions.

Usage

```
## S3 method for class 'likelihood_contr'
print(x, ...)
```

Arguments

x	A likelihood_contr model.
...	Additional arguments (ignored).

Value

The model object, invisibly.

Examples

```
model <- likelihood_contr(
  obs_type = "status",
  exact = contr_name("weibull", "exact", ob_col = "t"),
  right = contr_name("weibull", "right", ob_col = "t"),
  assumptions = c("Weibull distribution")
)
print(model)
```

---

rdata.likelihood_contr

Random data generation for likelihood_contr

Description

Returns the user-supplied rdata_fn or errors if none was provided.

Usage

```
## S3 method for class 'likelihood_contr'
rdata(model, ...)
```

Arguments

model	A likelihood_contr model.
...	Additional arguments (currently unused).

Value

A function function(theta, n, ...) returning a data frame.

Examples

```
rdata_fn <- function(theta, n, ...) {
  data.frame(t = rexp(n, rate = theta[1]), status = "exact")
}
model <- likelihood_contr(
  obs_type = "status",
  exact = contr_name("exp", "exact", ob_col = "t"),
  rdata_fn = rdata_fn
)
gen <- rdata(model)
gen(theta = c(rate = 2), n = 5)
```

---

score.likelihood_contr

Score for likelihood_contr

Description

Returns a closure that splits the data frame by observation type and sums the per-type score vectors. If a contribution does not provide an analytical score, numerical differentiation via `numDeriv::grad()` is used.

Usage

```
## S3 method for class 'likelihood_contr'
score(model, ...)
```

Arguments

`model`	A `likelihood_contr` model.
`...`	Additional arguments (currently unused).

Value

A function `function(df, par, ...)` returning the total score vector.

Examples

```
model <- likelihood_contr(
  obs_type = "status",
  exact = contr_name("exp", "exact", ob_col = "t")
)
df <- data.frame(t = c(0.5, 1.0, 1.5), status = "exact")
score_fn <- score(model)
score_fn(df, par = c(rate = 2))
```

Index