Package ‘ec50estimator’

May 25, 2026

Type Package

Title An Automated Way to Estimate EC50 for Stratified Datasets

Version 1.0.0

Date 2026-05-24

Maintainer Kaique dos S. Alves <`kaiquedsalves@gmail.com`>

Description Estimates effective concentrations that reduce growth by 50 percent (EC50) in multi-isolate and stratified dose-response experiments. The package wraps model fitting from drc, returns data-frame outputs, and provides helper functions for data checks, model selection, fitted-curve plotting, prediction, diagnostics, and reporting. Information about drc is available in Ritz C, Baty F, Streibig JC, Gerhard D (2015) <doi:10.1371/journal.pone.0146021>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Depends R (>= 3.5.0)

Imports drc, ggplot2

RoxygenNote 7.3.2

VignetteBuilder knitr

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

URL `https://alvesks.github.io/ec50estimator/`,
`https://github.com/AlvesKS/ec50estimator`

BugReports `https://github.com/AlvesKS/ec50estimator/issues`

NeedsCompilation no

Author Kaique dos S. Alves [aut, cre] (ORCID: <`https://orcid.org/0000-0001-9187-0252`>)

Repository CRAN

Date/Publication 2026-05-25 03:10:02 UTC

Contents

`check_ec50_data` *Check dose-response data before EC50 fitting*

Description

Run group-level checks for common dose-response data problems before fitting EC50 models.

Usage

```
check_ec50_data(data, response, dose, isolate, strata = NULL, log_x = TRUE)
```

Arguments

`data`	A data frame.
`response, dose, isolate`	Character scalars naming the response, dose, and isolate columns.
`strata`	Optional character vector of grouping columns.
`log_x`	Logical. If ‘TRUE‘, nonpositive dose values are flagged.

Value

A data frame with one row per isolate/strata group.

Examples

```
data(multi_isolate)
check_ec50_data(
  multi_isolate,
  response = "growth",
  dose = "dose",
  isolate = "isolate",
  strata = c("field", "fungicida")
)
```

`ec50_extractors` *Extract information from fitted EC50 objects*

Description

These helpers make fitted objects from [estimate_EC50()] and [ec50_multimodel()] easier to reuse. They return plain data frames, metadata, stored ‘drc‘ models, or fitted curve coordinates without requiring users to work with object attributes directly.

Usage

```
ec50_estimates(x)

ec50_metadata(x)

fitted_models(x)

curve_data(x, n_points = 200, log_x = TRUE, quiet = FALSE)
```

Arguments

`x`	An object returned by [estimate_EC50()] or [ec50_multimodel()].
`n_points`	Number of dose values used to draw each fitted curve.
`log_x`	Logical. If ‘TRUE‘, curve coordinates are generated only over positive dose values for log10 x-axis plotting.
`quiet`	Logical. If ‘FALSE‘, failed curve predictions are reported with a warning.

Value

‘ec50_estimates()‘ returns a plain data frame of EC estimates. ‘ec50_metadata()‘ returns a list with modeling metadata. ‘fitted_models()‘ returns a named list of stored ‘drc‘ model objects. ‘curve_data()‘ returns a data frame of fitted curve coordinates.

Examples

```
data(multi_isolate)
sample_data <- subset(
  multi_isolate,
  isolate %in% 1:3 & fungicida == "Fungicide A"
)

fit <- estimate_EC50(
  growth ~ dose,
  data = sample_data,
  isolate_col = "isolate",
  strata_col = "field",
  fct = drc::LL.3()
```

```
)

ec50_estimates(fit)
ec50_metadata(fit)
curve_data(fit)
```

---

estimate_EC50 *Estimate effective doses for grouped dose-response data*

Description

‘estimate_EC50()‘ fits one dose-response model per isolate, optionally within strata such as year, site, treatment, or fungicide. ‘ec50_multimodel()‘ repeats the same workflow for several ‘drc‘ model functions and returns model-selection statistics with the estimates.

Usage

```
estimate_EC50(
  formula,
  data,
  EC_lvl = 50,
  isolate_col,
  strata_col = NULL,
  fct,
  interval = c("none", "delta", "fls", "tfls"),
  type = c("relative", "absolute"),
  quiet = FALSE
)

ec50_multimodel(
  formula,
  data,
  EC_lvl = 50,
  isolate_col,
  strata_col = NULL,
  fct,
  interval = c("none", "delta", "fls", "tfls"),
  type = c("relative", "absolute"),
  quiet = FALSE
)
```

Arguments

`formula`	A two-sided formula identifying one numeric response and one numeric dose column, for example ‘growth ~ dose‘.
`data`	A data frame containing the numeric response, numeric dose, isolate, and optional stratification columns.

`EC_lvl`	Numeric effective-dose level(s) passed to [drc::ED()]. The default estimates EC50.
`isolate_col`	Character scalar naming the column that identifies each isolate.
`strata_col`	Optional character vector naming columns used to split the data before fitting models.
`fct`	A ‘drc‘ model function object such as ‘drc::LL.3()‘ for ‘estimate_EC50()‘. For ‘ec50_multimodel()‘, provide a list such as ‘list(drc::LL.3(), drc::LL.4())‘.
`interval`	Character scalar passed to [drc::ED()]. One of ‘"none"‘, ‘"delta"‘, ‘"fls"‘, or ‘"tfls"‘.
`type`	Character scalar passed to [drc::ED()]. One of ‘"relative"‘ or ‘"absolute"‘.
`quiet`	Logical. If ‘FALSE‘, failed isolate/model fits are reported with a warning.

Value

A data frame with one row per successful estimate. The first columns identify the isolate (‘ID‘) and strata, followed by columns returned by [drc::ED()]. ‘ec50_multimodel()‘ also appends model-selection statistics from [drc::mselect()] and a ‘model‘ column. The result keeps its data-frame behavior, but also stores the original data, formula, grouping columns, model functions, and fitted ‘drc‘ models so it can be passed directly to [plot_EC50_curves()].

Examples

```
data(multi_isolate)
sample_data <- subset(
  multi_isolate,
  isolate == 1 & fungicida == "Fungicide A"
)

estimate_EC50(
  growth ~ dose,
  data = sample_data,
  isolate_col = "isolate",
  strata_col = c("field", "fungicida"),
  fct = drc::LL.3()
)

ec50_multimodel(
  growth ~ dose,
  data = sample_data,
  isolate_col = "isolate",
  strata_col = c("field", "fungicida"),
  fct = list(drc::LL.3(), drc::LL.4())
)
```

`fit_quality` *Inspect EC50 fit quality and failures*

Description

‘fit_quality()‘ returns group-level quality information for successful fits. ‘fit_failures()‘ returns failed group/model combinations as data.

Usage

```
fit_quality(x)

fit_failures(x)
```

Arguments

`x`	An object returned by [estimate_EC50()] or [ec50_multimodel()].

Value

A data frame.

Examples

```
data(multi_isolate)
fit <- estimate_EC50(
  growth ~ dose,
  data = subset(multi_isolate, isolate %in% 1:3 & fungicida == "Fungicide A"),
  isolate_col = "isolate",
  strata_col = "field",
  fct = drc::LL.3()
)
fit_quality(fit)
fit_failures(fit)
```

`model_selection` *Select EC50 candidate models*

Description

Rank candidate models fitted with [ec50_multimodel()] within each isolate and stratum using an information criterion such as ‘IC‘.

Usage

```
model_selection(x, criterion = "IC")

best_model(x, criterion = "IC")
```

Arguments

`x`	An object returned by [ec50_multimodel()].
`criterion`	Character scalar naming the criterion column. Smaller values are considered better.

Value

A data frame with model rankings, delta criterion values, and weights.

Examples

```
data(multi_isolate)
sample_data <- subset(multi_isolate, isolate %in% 1:3 & fungicida == "Fungicide A")
fit <- ec50_multimodel(
  growth ~ dose,
  data = sample_data,
  isolate_col = "isolate",
  strata_col = "field",
  fct = list(drc::LL.3(), drc::LL.4())
)
model_selection(fit)
best_model(fit)
```

`multi_isolate` *Multi-isolate dose-response data*

Description

Simulated mycelial growth data for 50 fungal isolates exposed to increasing fungicide doses in conventional and organic fields.

Usage

```
multi_isolate
```

Format

A data frame with 3,500 rows and 5 columns:

isolate Isolate identifier.

field Field management system: conventional or organic.

fungicida Fungicide treatment.

dose Fungicide dose.

growth Observed mycelial growth.

Source

Simulated data included with the package.

`plot_EC50_curves` *Plot fitted dose-response curves for multiple isolates*

Description

‘plot_EC50_curves()‘ plots an object returned by [estimate_EC50()] or [ec50_multimodel()]. It uses the formula, original data, grouping columns, and fitted ‘drc‘ models stored in that result, so users do not need to repeat the modeling arguments. For compatibility, the function also accepts the original formula/data interface.

Usage

```
plot_EC50_curves(
  x,
  data = NULL,
  isolate_col = NULL,
  strata_col = NULL,
  fct = NULL,
  color_col = NULL,
  facet_col = NULL,
  facet_row = NULL,
  models = "all",
  n_points = 200,
  log_x = TRUE,
  point_size = 2,
  point_alpha = 0.8,
  line_width = 1,
  quiet = FALSE
)
```

Arguments

`x`	An object returned by [estimate_EC50()] or [ec50_multimodel()]. A two-sided formula such as ‘growth ~ dose‘ is also accepted for compatibility.
`data`	A data frame containing the response, dose, isolate, and optional stratification columns. Required only when ‘x‘ is a formula.
`isolate_col`	Character scalar naming the column that identifies each isolate. Required only when ‘x‘ is a formula.
`strata_col`	Optional character vector naming columns used to split the data before fitting models. Used only when ‘x‘ is a formula.
`fct`	A ‘drc‘ model function object such as ‘drc::LL.3()‘, or a list of model function objects such as ‘list(drc::LL.3(), drc::LL.4())‘. Required only when ‘x‘ is a formula.
`color_col`	Character scalar naming the column mapped to curve and point color. Defaults to the isolate column and is always converted to a factor before plotting.
`facet_col`, `facet_row`	Optional character scalars naming columns used for faceting. When omitted, the first two ‘strata_col‘ values are used.
`models`	One of ‘"all"‘, ‘"best"‘, or a character vector of model names to draw. ‘"best"‘ uses [best_model()] for multimodel fits.
`n_points`	Number of dose values used to draw each fitted curve.
`log_x`	Logical. If ‘TRUE‘, use a log10 x-axis and omit non-positive dose values from the plotted data and prediction grid.
`point_size`, `point_alpha`	Size and alpha for raw observation points.
`line_width`	Width for fitted curves.
`quiet`	Logical. If ‘FALSE‘, failed group/model fits are reported with a warning.

Value

A ‘ggplot2‘ object. The plotted curve data, observed data, and fitted models are attached to the returned object as ‘curve_data‘, ‘observed_data‘, and ‘fitted_models‘ attributes and list elements.

Examples

```
data(multi_isolate)
sample_data <- subset(
  multi_isolate,
  isolate %in% 1:4 & fungicida == "Fungicide A"
)

fit <- estimate_EC50(
  growth ~ dose,
  data = sample_data,
  isolate_col = "isolate",
  strata_col = "field",
  fct = drc::LL.3()
```

```
)
plot_EC50_curves(fit)

multi_fit <- ec50_multimodel(
  growth ~ dose,
  data = sample_data,
  isolate_col = "isolate",
  strata_col = "field",
  fct = list(drc::LL.3(), drc::LL.4())
)
plot_EC50_curves(multi_fit)
```

`predict_ec50` *Predict responses from fitted EC50 models*

Description

Predict response values at user-supplied doses from stored ‘drc‘ model fits.

Usage

```
predict_ec50(x, dose, models = "all")
```

Arguments

`x`	An object returned by [estimate_EC50()] or [ec50_multimodel()].
`dose`	Numeric vector of dose values for prediction.
`models`	One of ‘"all"‘, ‘"best"‘, or a character vector of model names.

Value

A data frame with group identifiers, model, dose, and predicted value.

Examples

```
data(multi_isolate)
fit <- estimate_EC50(
  growth ~ dose,
  data = subset(multi_isolate, isolate == 1 & fungicida == "Fungicide A"),
  isolate_col = "isolate",
  fct = drc::LL.3()
)
predict_ec50(fit, dose = c(0.001, 0.01, 0.1))
```

`report_ec50` *Build EC50 report tables*

Description

Return a plain data frame suitable for reporting estimates from all, best, or selected models.

Usage

```
report_ec50(x, models = "all")
```

Arguments

`x`	An object returned by [estimate_EC50()] or [ec50_multimodel()].
`models`	One of ‘"all"‘, ‘"best"‘, or a character vector of model names.

Value

A plain data frame.

Examples

```
data(multi_isolate)
fit <- estimate_EC50(
  growth ~ dose,
  data = subset(multi_isolate, isolate %in% 1:3 & fungicida == "Fungicide A"),
  isolate_col = "isolate",
  strata_col = "field",
  fct = drc::LL.3()
)
report_ec50(fit)
```

`residual_data` *Extract and plot EC50 residual diagnostics*

Description

‘residual_data()‘ returns observed, fitted, and residual values from stored models. ‘plot_residuals()‘ returns a ‘ggplot2‘ diagnostic plot.

Usage

```
residual_data(x, models = "all")

plot_residuals(x, models = "all", type = c("fitted", "dose"))
```

Arguments

`x`	An object returned by [estimate_EC50()] or [ec50_multimodel()].
`models`	One of ‘"all"‘, ‘"best"‘, or a character vector of model names.
`type`	For ‘plot_residuals()‘, plot residuals against fitted values or dose.

Value

‘residual_data()‘ returns a data frame; ‘plot_residuals()‘ returns a ‘ggplot2‘ object.

Examples

```
data(multi_isolate)
fit <- estimate_EC50(
  growth ~ dose,
  data = subset(multi_isolate, isolate == 1 & fungicida == "Fungicide A"),
  isolate_col = "isolate",
  fct = drc::LL.3()
)
residual_data(fit)
plot_residuals(fit)
```

Index