

Package ‘merlin’

October 13, 2022

Version 0.1.0

Title Mixed Effects Regression for Linear, Non-Linear and User-Defined Models

Description Fits linear, non-linear, and user-defined mixed effects regression models following the framework developed by Crowther (2017) <[arXiv:1710.02223](https://arxiv.org/abs/1710.02223)>. 'merlin' can fit multivariate outcome models of any type, each of which could be repeatedly measured (longitudinal), with any number of levels, and with any number of random effects at each level. Standard distributions/models available include the Bernoulli, Gaussian, Poisson, beta, negative-binomial, and time-to-event/survival models include the exponential, Gompertz, Royston-Parmar, Weibull and general hazard model. 'merlin' provides a flexible predictor syntax, allowing the user to define variables, random effects, spline and fractional polynomial functions, functions of other outcome models, and any interaction between each of them. Non-linear and time-dependent effects are seamlessly incorporated into the predictor. 'merlin' allows multivariate normal random effects, which are integrated out using Gaussian quadrature or Monte-Carlo integration. Note, 'merlin' is based on the 'Stata' package of the same name, described in Crowther (2018) <[arXiv:1806.01615](https://arxiv.org/abs/1806.01615)>.

License GPL (>= 3)

Depends R (>= 2.10)

Imports MASS, randtoolbox, statmod, stats, survival, tools

Suggests knitr, nlme, rmarkdown, testthat

URL <https://www.mjcrowther.co.uk/software/merlin>

VignetteBuilder knitr

RoxygenNote 7.1.1

LazyData true

Encoding UTF-8

ByteCompile true

Language en_GB

NeedsCompilation no

Author Emma Martin [aut],
Alessandro Gasparini [aut],
Michael Crowther [aut, cre]

Maintainer Michael Crowther <michael.crowther@le.ac.uk>

Repository CRAN

Date/Publication 2020-06-29 17:00:03 UTC

R topics documented:

coef.merlin	3
coef.mlrcs	3
coef.mlsurv	4
heart.valve	4
logLik.merlin	6
logLik.mlrcs	6
logLik.mlsurv	7
merlin	7
merlin_util_ap	12
merlin_util_ap_mod	13
merlin_util_depvar	14
merlin_util_ev	14
merlin_util_ev_deriv	15
merlin_util_ev_deriv2	16
merlin_util_ev_deriv2_mod	17
merlin_util_ev_deriv_mod	17
merlin_util_ev_integ	18
merlin_util_ev_integ_mod	19
merlin_util_ev_mod	20
merlin_util_timevar	20
merlin_util_xzb	21
merlin_util_xzb_deriv	22
merlin_util_xzb_deriv2	23
merlin_util_xzb_deriv2_mod	23
merlin_util_xzb_deriv_mod	24
merlin_util_xzb_integ	25
merlin_util_xzb_integ_mod	26
merlin_util_xzb_mod	26
mlrcs	27
mlsurv	28
pbc	30
predict.merlin	31
print.merlin	34
print.mlrcs	34
print.mlsurv	35
sim3	35
summary.merlin	36
summary.mlrcs	36
summary.mlsurv	37
vcov.merlin	37
vcov.mlrcs	38

<code>coef.merlin</code>	3
<code>vcov.mlsurv</code>	38
Index	39

<code>coef.merlin</code>	<i>Extract Model Coefficients</i>
--------------------------	-----------------------------------

Description

`coef` extracts model coefficients from a `merlin` model fit. `coefficients` is an alias for it.

Usage

```
## S3 method for class 'merlin'
coef(object, ...)

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

Arguments

<code>object</code>	An object of class <code>merlin</code> or <code>summary.merlin</code> .
<code>...</code>	Not used.

<code>coef.mlrcs</code>	<i>Extract Model Coefficients</i>
-------------------------	-----------------------------------

Description

`coef` extracts model coefficients from a `mlrcs` model fit. `coefficients` is an alias for it.

Usage

```
## S3 method for class 'mlrcs'
coef(object, ...)

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

Arguments

<code>object</code>	An object of class <code>mlrcs</code> or <code>summary.mlrcs</code> .
<code>...</code>	Not used.

coef.mlsurv	<i>Extract Model Coefficients</i>
-------------	-----------------------------------

Description

coef extracts model coefficients from a mlsurv model fit. coefficients is an alias for it.

Usage

```
## S3 method for class 'mlsurv'
coef(object, ...)

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

Arguments

object	An object of class mlsurv or summary.mlsurv.
...	Not used.

heart.valve	<i>Aortic valve replacement surgery data</i>
-------------	--

Description

This is longitudinal data on an observational study on detecting effects of different heart valves, differing on type of tissue, implanted in the aortic position. The data consists of longitudinal measurements on three different heart function outcomes, after surgery occurred. There are several baseline covariates available, and also survival data.

Usage

```
data(heart.valve)

data(heart.valve.merlin)
```

Format

This is a data frame in the unbalanced format, that is, with one row per observation. The data consists in columns for patient identification, time of measurements, longitudinal multiple longitudinal measurements, baseline covariates, and survival data. The column names are identified as follows:

num	number for patient identification.
sex	gender of patient (0=Male and 1=Female).
age	age of patient at day of surgery (years).

time observed time point, with surgery date as the time origin (years).
fuyrs maximum follow up time, with surgery date as the time origin (years).
status censoring indicator (1=died and 0=lost at follow up).
grad valve gradient at follow-up visit.
log.grad natural log transformation of grad.
lvmi left ventricular mass index (standardised) at follow-up visit.
log.lvmi natural log transformation of lvmi.
ef ejection fraction at follow-up visit.
bsa preoperative body surface area.
lvh preoperative left ventricular hypertrophy.
prenyha preoperative New York Heart Association (NYHA) classification (1=I/II and 3=III/IV).
redo previous cardiac surgery.
size size of the valve (millimetres).
con.cabg concomitant coronary artery bypass graft.
creat preoperative serum creatinine ($\mu\text{mol/mL}$).
dm preoperative diabetes.
acei preoperative use of ace inhibitor.
lv preoperative left ventricular ejection fraction (LVEF) (1=good, 2=moderate, and 3=poor).
emergenc operative urgency (0=elective, 1=urgent, and 3=emergency).
hc preoperative high cholesterol (0=absent, 1=present treated, and 2=present untreated).
sten.reg.mix aortic valve haemodynamics (1=stenosis, 2=regurgitation, 3=mixed).
hs implanted aortic prosthesis type (1=homograft and 0=stentless porcine tissue).

An object of class `data.frame` with 629 rows and 29 columns.

Note

`heart.valve.merlin` is a version of the `heart.valve` dataset in `merlin` format.

Source

Mr Eric Lim (<http://www.drericlim.com>)

References

Lim E, Ali A, Theodorou P, Sousa I, Ashrafian H, Chamageorgakis T, Duncan M, Diggle P, Pepper J. A longitudinal study of the profile and predictors of left ventricular mass regression after stentless aortic valve replacement. *Ann Thorac Surg.* 2008; **85(6)**: 2026-2029.

logLik.merlin	<i>Extract Log-Likelihood</i>
---------------	-------------------------------

Description

Extract log-likelihood of a merlin model.

Usage

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

Arguments

object	An object of class merlin or summary.merlin.
...	Not used.

logLik.mlrcs	<i>Extract Log-Likelihood</i>
--------------	-------------------------------

Description

Extract log-likelihood of a mlrcs model.

Usage

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

Arguments

object	An object of class mlrcs or summary.mlrcs.
...	Not used.

logLik.mlsurv	<i>Extract Log-Likelihood</i>
---------------	-------------------------------

Description

Extract log-likelihood of a mlsurv model.

Usage

```
## S3 method for class 'mlsurv'
logLik(object, ...)

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

Arguments

object	An object of class mlsurv or summary.mlsurv.
...	Not used.

merlin	<i>merlin - Mixed Effects Regression for Linear, Nonlinear and User-defined models</i>
--------	--

Description

merlin fits linear, non-linear and user-defined mixed effects regression models. merlin can fit multivariate outcome models of any type, each of which could be repeatedly measured (longitudinal), with any number of levels, and with any number of random effects at each level. Standard distributions/models available include the Bernoulli, Gaussian, Poisson, beta, negative-binomial, and time-to-event/survival models include the exponential, Gompertz, Weibull, Royston-Parmar, and general hazard model. merlin provides a flexible predictor syntax, allowing the user to define variables, random effects, spline and fractional polynomial functions, functions of other outcome models, and any interaction between each of them. Non-linear and time-dependent effects are seamlessly incorporated into the predictor. merlin allows multivariate normal random effects, which are integrated out using Gaussian quadrature or Monte-Carlo integration. Relative survival (excess hazard) models are supported. Utility functions are provided to allow user-defined models to be specified, in conjunction with the complex predictor.

Usage

```

merlin(
  model,
  from = NULL,
  family = "gaussian",
  link = NULL,
  timevar = NULL,
  covariance = "diagonal",
  data,
  userf = NULL,
  sweights = NULL,
  levels = NULL,
  predict = FALSE,
  predtype = NULL,
  predmodel = NULL,
  causes = NULL,
  at = NULL,
  contrast = NULL,
  modelfit = NULL,
  control = list()
)

```

Arguments

- `model` specify the fixed and random elements for each model outcome. Where there are multiple outcomes, the models should be specified in a list. Each model should be specified as a formula (e.g. $y \sim x$). A number of different element types can be specified, including
- `varname` - an independent variable from the data set
 - `random-effects` - a random-effect at the cluster level can be specified using `M#[cluster level]`, for example `M1[id]` would define a random intercept at the ID level. Each independent random-effect should be given a unique name, if two random-effects are given the same name they will be treated as shared random-effects.
 - `racs()` - restricted cubic spline terms, this option can be used to include a restricted cubic spline function, with the degrees of freedom (number of spline terms) specified using the `df` sub-option, with the boundary knots assumed to be at the minimum and maximum of the variable, with internal knots placed at equally spaced centiles. Other default options `orthog = TRUE`, which by default orthogonalises the splines, `log = FALSE`, which can be used to calculate splines of the log of the variable and `event = F`, which can be used to calculate the internal knots based only on observations that experienced the event of interest (for survival models).
 - `sracs()` is a shorthand element, equivalent to `racs(..., log = TRUE, event = TRUE)`, for use with survival models.
 - `fp()` - fractional polynomials of order 1 or 2 can be specified, the sub-option `powers` is used to specify the powers of the fp model.

- EV[depvar] - the expected value of the response of a submodel.
- dEV[depvar] - the first derivative with respect to time of the expected value of the response of a submodel.
- d2EV[depvar] - the second derivative with respect to time of the expected value of the response of a submodel.
- iEV[depvar] - the integral with respect to time of the expected value of the response of a submodel.
- XB[depvar] - the expected value of the complex predictor of a submodel.
- dXB[depvar] - the first derivative with respect to time of the expected value of the complex predictor of a submodel.
- d2XB[depvar] - the second derivative with respect to time of the expected value of the complex predictor of a submodel.
- iXB[depvar] - the integral with respect to time of the expected value of the complex predictor of a submodel.
- bhazard(varname) - invokes a relative survival (excess hazard) model. varname specifies the expected hazard rate at the event time.
- exposure(varname) - include $\log(\text{varname})$ in the linear predictor, with a coefficient of 1. For use with family = "poisson".
- ap(#) - defines the number of ancillary parameters. Used with family="user".

from

this is an optional argument giving the initial values for the full parameter vector, for more details on how to specify the initial estimates see the vignette.

family

a vector of strings specifying the family for each outcome specified in model. The currently available models include,

- gaussian - Gaussian distribution
- bernoulli - Bernoulli distribution
- poisson - Poisson distribution
- beta - Beta distribution
- negbinomial - Negative binomial distribution

with survival models including,

- exponential - exponential distribution
- weibull - Weibull distribution
- gompertz - Gompertz distribution
- rp - Royston-Parmar model (complex predictor on the log cumulative hazard scale)
- loghazard - general log hazard model (complex predictor on the log hazard scale)

and user-defined,

- user - fit a user-defined model, which should be written using merlin's utility functions. The name of your user-defined function should be passed through the userf option.
- null - is a convenience tool to define additional complex predictors, that do not contribute to the log likelihood. For use with family = "user".

link	string vector defining the link functions for each model. Default is "identity" for all models. If specified, you must define a link function for all submodels. Options include "identity", "log" and "logit".
timevar	specifies the variable which represents time, this is necessary when a function of time is used in the linear predictor of a survival model as it may interact with other elements of the model.
covariance	the structure of the variance-covariance matrix can be varied, the default is diagonal where all diagonal elements of the variance-covariance matrix are estimated uniquely, identity assumes all diagonal elements are equal and unstructured estimates all elements of the variance-covariance matrix.
data	a data frame containing all variables required for fitting the model. Can be a tibble object.
userf	string vector defining the name of the user-written functions for each family="user". Each function must be in memory and should return the observation level contribution to the log-likelihood.
sweights	Not documented.
levels	if the model contains random-effects then a vector giving the order of levels must be specified, from the highest level to the lowest, e.g. levels=c("practice", "id").
predict	Not documented.
predtype	Not documented.
predmodel	Not documented.
causes	Not documented.
at	Not documented.
contrast	Not documented.
modelfit	Not documented.
control	A list of parameters that control the estimation algorithm. Generally it should not be modified, unless there are convergence issues. Possible values are: <ul style="list-style-type: none"> • ip An optional vector of integers specifying the number of integration points to be used when integrating out the random effects. A different number of ip can be specified for each level (from highest to lowest level). If only a single number is given then merlin will assume that number of integration points at all levels. Default is ip = 7 for each level using Gauss-Hermite quadrature, or ip = 100 for each level using Monte-Carlo integration; • intmethod The method used for numerically integrating out the random-effects in order to calculate the likelihood for a mixed effects model which includes random effects. Options include ghermite for non-adaptive Gauss-Hermite quadrature, halton for Monte-Carlo integration using Halton sequences, sobol for Monte-Carlo integration using Sobol sequences, or mc for standard Monte-Carlo integration using normal draws. The default is ghermite. Level-specific integration techniques can be specified, for example, with a three level model, we may use Gauss-Hermite quadrature at the highest level, and Monte-Carlo integration with Halton sequences at level 2, using intmethod = c("ghermite", "halton").

- debug Not documented.
- verbose Not documented.
- optim.method The optim method to be used. Defaults to Nelder-Mead, see [optim](#) for available methods.
- maxit The maximum number of iterations for the optimisation routine. Defaults to 5000.

Author(s)

Emma C. Martin, Alessandro Gasparini and Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear, non-linear and user-defined models.

See Also

[predict.merlin](#)

[merlin_util_depvar](#), [merlin_util_timevar](#), [merlin_util_xzb](#), [merlin_util_xzb_mod](#) [merlin_util_xzb_deriv](#), [merlin_util_xzb_deriv_mod](#) [merlin_util_xzb_deriv2](#), [merlin_util_xzb_deriv2_mod](#) [merlin_util_xzb_integ](#), [merlin_util_xzb_integ_mod](#) [merlin_util_ev](#), [merlin_util_ev_mod](#) [merlin_util_ev_deriv](#), [merlin_util_ev_deriv_mod](#) [merlin_util_ev_deriv2](#), [merlin_util_ev_deriv2_mod](#) [merlin_util_ev_integ](#), [merlin_util_ev_integ_mod](#) [merlin_util_ap](#), [merlin_util_ap_mod](#)

Examples

```
## Not run:
library(merlin)
data(pbc.merlin, package = "merlin")

# Linear fixed-effects model
merlin(logb ~ year,
       family = "gaussian",
       data = pbc.merlin)

# Linear mixed-effects model with random intercept and slope at ID level
merlin(logb ~ year + M1[id] * 1 + year:M2[id] * 1,
       family = "gaussian",
       levels = "id",
       data = pbc.merlin)

# Joint longitudinal and survival model with shared random effects
merlin(model = list(logb ~ year + M1[id] * 1,
                   Surv(stime, died) ~ trt + M1[id]),
       family = c("gaussian", "weibull"),
```

```

      levels = "id",
      data = pbc.merlin)

# Joint longitudinal and survival model with expected value
merlin(model = list(logb ~ year + M1[id] * 1,
                  Surv(stime, died) ~ trt + EV[logb]),
      family = c("gaussian", "weibull"),
      levels = "id",
      timevar = c("year", "stime"),
      data = pbc.merlin)

# Gaussian distribution - implemented as a user-written family
logl_gaussian <- function(gml)
{
  y <- merlin_util_depvar(gml)
  xzb <- merlin_util_xzb(gml)
  se <- exp(merlin_util_ap(gml,1))

  mu <- (sweep(xzb,1,y,"-"))^2
  logl <- ((-0.5 * log(2*pi) - log(se)) - (mu/(2 * se^2)))
  return(logl)
}

merlin(logb ~ year + ap(1), family = "user", data = pbc.merlin,
      userf = "logl_gaussian")

# 3-level Weibull model
merlin(Surv(stime1,dead1) ~ age + M1[id1]*1 + M2[id2]*1,
      levels=c("id1","id2"), family="weibull", data=sim3)

## End(Not run)

```

merlin_util_ap	<i>merlin_util_ap</i> - returns the current estimate of the <i>i</i> th ancillary parameter for the associated model
----------------	--

Description

merlin_util_ap - returns the current estimate of the *i*th ancillary parameter for the associated model

Usage

```
merlin_util_ap(gml, i)
```

Arguments

gml	merlin object - should not be edited
i	index for which ancillary parameter to extract

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_ap_mod	<i>merlin_util_ap_mod - returns the current estimate of the ith ancillary parameter for the specified model</i>
--------------------	---

Description

merlin_util_ap_mod - returns the current estimate of the ith ancillary parameter for the specified model

Usage

```
merlin_util_ap_mod(gml, m, i)
```

Arguments

gml	merlin object - should not be edited
m	specifies the merlin submodel
i	index for which ancillary parameter to extract

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_depvar	<i>merlin_util_depvar - returns the response variable(s)</i>
--------------------	--

Description

Utility function to extract the dependent variable(s) for the current model. If the model is a survival/time-to-event model, then it will contain the event times in the first column and the event indicator in the second.

Usage

```
merlin_util_depvar(gml)
```

Arguments

gml	merlin object - should not be edited
-----	--------------------------------------

Author(s)

Emma C. Martin, Alessandro Gasparini and Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_ev	<i>merlin_util_ev - returns the observation-level expected value of the outcome</i>
----------------	---

Description

Utility function to extract the expected value of the outcome at the current parameter estimates for a particular model.

Usage

```
merlin_util_ev(gml, t = NULL)
```

Arguments

gml	merlin object - should not be edited
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_ev_deriv *merlin_util_ev_deriv - returns the first derivative with respect to time of the observation-level expected value of the outcome*

Description

Utility function to extract d/dt of the expected value of the outcome at the current parameter estimates for a particular model.

Usage

```
merlin_util_ev_deriv(gml, t = NULL)
```

Arguments

gml	merlin object - should not be edited
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

Examples

```
library(merlin)
data(pbc.merlin, package = "merlin")

# Linear fixed-effects model
merlin(model = list(logb ~ year),
       family = "gaussian",
       data = pbc.merlin)
```

merlin_util_ev_deriv2 *merlin_util_ev_deriv2 - returns the second derivative with respect to time of the observation-level expected value of the outcome*

Description

merlin_util_ev_deriv2 - returns the second derivative with respect to time of the observation-level expected value of the outcome

Usage

```
merlin_util_ev_deriv2(gml, t = NULL)
```

Arguments

gml	merlin object - should not be edited
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

- Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>
- Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>
- Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_ev_deriv2_mod

merlin_util_ev_deriv2_mod - returns the second derivative with respect to time of the observation-level expected value of the specified outcome

Description

merlin_util_ev_deriv2_mod - returns the second derivative with respect to time of the observation-level expected value of the specified outcome

Usage

```
merlin_util_ev_deriv2_mod(gml, m, t = NULL)
```

Arguments

gml	merlin object - should not be edited
m	specifies the merlin submodel
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_ev_deriv_mod

merlin_util_ev_deriv - returns the first derivative with respect to time of the observation-level expected value of the specified outcome

Description

Utility function to extract d/dt of the expected value of the outcome at the current parameter estimates for a specified model.

Usage

```
merlin_util_ev_deriv_mod(gml, m, t = NULL)
```

Arguments

gml	merlin object - should not be edited
m	specifies the merlin submodel
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_ev_integ *merlin_util_ev_integ - returns the integral with respect to time of the observation-level expected value of the outcome*

Description

merlin_util_ev_integ - returns the integral with respect to time of the observation-level expected value of the outcome

Usage

```
merlin_util_ev_integ(gml, t = NULL)
```

Arguments

gml	merlin object - should not be edited
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_ev_integ_mod

merlin_util_ev_integ_mod - returns the integral with respect to time of the observation-level expected value of the specified outcome

Description

merlin_util_ev_integ_mod - returns the integral with respect to time of the observation-level expected value of the specified outcome

Usage

```
merlin_util_ev_integ_mod(gml, m, t = NULL)
```

Arguments

gml	merlin object - should not be edited
m	specifies the merlin submodel
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_ev_mod	<i>merlin_util_ev_mod - returns the observation-level expected value of the outcome for a specified model</i>
--------------------	---

Description

Utility function to extract the expected value of the outcome at the current parameter estimates for a specified model.

Usage

```
merlin_util_ev_mod(gml, m, t = NULL)
```

Arguments

gml	merlin object - should not be edited
m	specifies the model
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_timevar	<i>merlin_util_timevar - returns the time variable</i>
---------------------	--

Description

Utility function to extract the time variable(s) for the current model.

Usage

```
merlin_util_timevar(gml)
```

Arguments

gml merlin object - should not be edited

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_xzb *merlin_util_xzb - returns the observation-level complex predictor*

Description

Utility function to extract the complex predictor evaluated at the current parameter estimates for a particular model.

Usage

```
merlin_util_xzb(gml, t = NULL)
```

Arguments

gml merlin object - should not be edited
t specifies the variable which represents time

Author(s)

Emma C. Martin, Alessandro Gasparini and Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

Examples

```
library(merlin)
data(pbc.merlin, package = "merlin")

# Linear fixed-effects model
merlin(model = list(logb ~ year),
       family = "gaussian",
       data = pbc.merlin)
```

merlin_util_xzb_deriv *merlin_util_xzb_deriv* - returns the first derivative with respect to time of the observation-level complex predictor

Description

Utility function to extract d/dt of the complex predictor evaluated at the current parameter estimates for a particular model.

Usage

```
merlin_util_xzb_deriv(gml, t = NULL)
```

Arguments

gml	merlin object - should not be edited
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

Examples

```
library(merlin)
data(pbc.merlin, package = "merlin")

# Linear fixed-effects model
merlin(model = list(logb ~ year),
       family = "gaussian",
       data = pbc.merlin)
```

merlin_util_xzb_deriv2

merlin_util_xzb_deriv2 - returns the second derivative with respect to time of the observation-level complex predictor

Description

merlin_util_xzb_deriv2 - returns the second derivative with respect to time of the observation-level complex predictor

Usage

```
merlin_util_xzb_deriv2(gml, t = NULL)
```

Arguments

gml	merlin object - should not be edited
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_xzb_deriv2_mod

merlin_util_xzb_deriv2_mod - returns the second derivative with respect to time of the observation-level complex predictor of the specified model

Description

merlin_util_xzb_deriv2_mod - returns the second derivative with respect to time of the observation-level complex predictor of the specified model

Usage

```
merlin_util_xzb_deriv2_mod(gml, m, t = NULL)
```

Arguments

gml	merlin object - should not be edited
m	specifies the merlin submodel
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_xzb_deriv_mod

merlin_util_xzb_deriv_mod - returns the first derivative with respect to time of the observation-level complex predictor of a specified model

Description

merlin_util_xzb_deriv_mod - returns the first derivative with respect to time of the observation-level complex predictor of a specified model

Usage

```
merlin_util_xzb_deriv_mod(gml, m, t = NULL)
```

Arguments

gml	merlin object - should not be edited
m	specifies the merlin submodel
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_xzb_integ *merlin_util_xzb_integ* - returns the integral with respect to time of the observation-level complex predictor

Description

merlin_util_xzb_integ - returns the integral with respect to time of the observation-level complex predictor

Usage

```
merlin_util_xzb_integ(gml, t = NULL)
```

Arguments

gml	merlin object - should not be edited
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_xzb_integ_mod

merlin_util_xzb_integ_mod - returns the integral with respect to time of the observation-level complex predictor of the specified model

Description

merlin_util_xzb_integ_mod - returns the integral with respect to time of the observation-level complex predictor of the specified model

Usage

```
merlin_util_xzb_integ_mod(gml, m, t = NULL)
```

Arguments

gml	merlin object - should not be edited
m	specifies the merlin submodel
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

merlin_util_xzb_mod

merlin_util_xzb_mod - returns the observation-level complex predictor of the specified model

Description

Utility function to extract the complex predictor evaluated at the current parameter estimates for a particular model.

Usage

```
merlin_util_xzb_mod(gml, m, t = NULL)
```

Arguments

gml	merlin object - should not be edited
m	specifies the merlin submodel
t	specifies the variable which represents time

Author(s)

Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear and non-linear models.

mlrcs

Fit multi-level restricted cubic spline model

Description

Fit multi-level restricted cubic spline model

Usage

```
mlrcs(formula, random = NULL, data, ...)
```

Arguments

formula	A model formula for the fixed-effects in the model. Must include one restricted cubic spline term, specified as <code>rcs(variable, df = #)</code> .
random	A formula for the random-effects in the model. Random-effects should be specified as a one-sided formula, e.g. <code>~ 1 + trt id</code> for random effect on the intercept and treatment at the <code>id</code> level. Random-effects can be estimated at any number of nested random-effect levels by providing a list of one-sided formulas. When specifying random-effect at multiple levels. The one-sided formula should be given in order, starting with the highest level. Only required when <code>rcs = TRUE</code> .
data	A data frame containing all variables required for fitting the model.
...	Further arguments passed to merlin .

Value

An object of class `mlrcs`.

Examples

```
## Not run:
# Two-level model
data("pbc.merlin", package = "merlin")
fit <- mlrcs(formula = logp ~ 1 + rcs(year, df = 4) + age + trt,
             random = ~ 1 + trt | id,
             data = pbc.merlin
)
summary(fit)

# Three-level model
fit <- mlrcs(formula = logp ~ 1 + rcs(year, df = 4) + age + trt,
             random = list(~ 1 | region,
                          ~ 1 + trt | id),
             data = pbc.merlin
)
summary(fit)

## End(Not run)
```

mlsurv

Fit proportional hazards survival models

Description

Fit proportional hazards survival models

Usage

```
mlsurv(
  formula,
  distribution,
  df = NULL,
  powers = NULL,
  rcs = TRUE,
  data,
  from.null = NULL,
  ...
)
```

Arguments

formula A model formula, where the left-hand side is a [Surv](#) object.

distribution A parametric distribution for the baseline hazard. Possible values are `exponential`, `weibull`, `gompertz`, `rp`, `logchazard`, and `loghazard`. `rp` is equivalent to a logchazard model with restricted cubic splines (argument `rcs = TRUE`).

df	Represents the number of degrees of freedom used for the restricted cubic splines when flexibly modelling the baseline hazard. Only required when <code>rcs = TRUE</code> .
powers	A vector representing the degree of the fractional polynomials used to model the baseline hazard (with a maximum degree of 2). Only required when <code>rcs = FALSE</code> .
rcs	Use restricted cubic splines when flexibly modelling the baseline hazard? Defaults to <code>TRUE</code> , and the alternative is using fractional polynomials.
data	A data frame containing all variables required for fitting the model. Can be a tibble object.
from.null	A vector of starting values for the null model (used to get improved starting values). This is mostly useful when experiencing issues with default starting values or convergence issues.
...	Further arguments passed to merlin .

Value

An object of class `mlsurv`.

Examples

```
# Weibull model
library(survival)
data("pbc.merlin", package = "merlin")
fit <- mlsurv(
  formula = Surv(stime, died) ~ trt,
  distribution = "weibull",
  data = pbc.merlin
)
summary(fit)

# Royston-Parmar model with 3 degrees of freedom
fit <- mlsurv(
  formula = Surv(stime, died) ~ trt,
  distribution = "rp",
  df = 3,
  data = pbc.merlin
)
summary(fit)

## Not run:
# Flexible parametric model on the log-hazard scale with fractional polynomials
fit <- mlsurv(
  formula = Surv(stime, died) ~ trt,
  distribution = "loghazard",
  powers = c(0, 1),
  rcs = FALSE,
  data = pbc.merlin
)
summary(fit)
```

```
## End(Not run)
```

pbc

Mayo Clinic primary biliary cirrhosis data

Description

This data is from the Mayo Clinic trial in primary biliary cirrhosis (PBC) of the liver conducted between 1974 and 1984. A total of 424 PBC patients, referred to Mayo Clinic during that ten-year interval met eligibility criteria for the randomized placebo controlled trial of the drug D-penicillamine, but only the first 312 cases in the data set participated in the randomized trial. Therefore, the data here are for the 312 patients with largely complete data.

Usage

```
data(pbc)
```

```
data(pbc.merlin)
```

Format

A data frame with 1945 observations on the following 20 variables:

`id` patients identifier; in total there are 312 patients.

`years` number of years between registration and the earlier of death, transplantation, or study analysis time.

`status` a factor with levels `alive`, `transplanted` and `dead`.

`drug` a factor with levels `placebo` and `D-penicil`.

`age` at registration in years.

`sex` a factor with levels `male` and `female`.

`year` number of years between enrollment and this visit date, remaining values on the line of data refer to this visit.

`ascites` a factor with levels `No` and `Yes`.

`hepatomegaly` a factor with levels `No` and `Yes`.

`spiders` a factor with levels `No` and `Yes`.

`edema` a factor with levels `No edema` (i.e. no edema and no diuretic therapy for edema), `edema no diuretics` (i.e. edema present without diuretics, or edema resolved by diuretics), and `edema despite diuretics` (i.e. edema despite diuretic therapy).

`serBilir` serum bilirubin in mg/dl.

`serChol` serum cholesterol in mg/dl.

`albumin` albumin in mg/dl.

`alkaline` alkaline phosphatase in U/liter.

`SGOT` SGOT in U/ml.

platelets platelets per cubic ml/1000.

prothrombin prothrombin time in seconds.

histologic histologic stage of disease.

status2 a numeric vector with the value 1 denoting if the patient was dead, and 0 if the patient was alive or transplanted.

An object of class `data.frame` with 1945 rows and 25 columns.

Note

`pbcr.merlin` is a version of the `pbcr` dataset in `merlin` format.

Source

[pbcr](#).

References

Fleming T, Harrington D. *Counting Processes and Survival Analysis*. 1991; New York: Wiley.

Therneau T, Grambsch P. *Modeling Survival Data: Extending the Cox Model*. 2000; New York: Springer-Verlag.

predict.merlin

predict.merlin - post-estimation tools for merlin

Description

predictions following the fit of a `merlin` model

Usage

```
## S3 method for class 'merlin'
predict(
  object,
  stat = "eta",
  type = "fixedonly",
  predmodel = 1,
  causes = NULL,
  at = NULL,
  contrast = NULL,
  ...
)
```

Arguments

object	merlin model object
stat	<p>specifies which prediction, which can be one of:</p> <ul style="list-style-type: none"> • <code>eta</code> the expected value of the complex predictor • <code>mu</code> the expected value of the response variable • <code>hazard</code> the hazard function • <code>chazard</code> the cumulative hazard function • <code>logchazard</code> the log cumulative hazard function • <code>survival</code> the survival function • <code>cif</code> the cumulative incidence function • <code>rmst</code> calculates the restricted mean survival time, which is the integral of the survival function within the interval $(0,t]$, where t is the time at which predictions are made. If multiple survival models have been specified in your merlin model, then it will assume all of them are cause-specific competing risks models, and include them in the calculation. If this is not the case, you can override which models are included by using the <code>causes</code> option. $rmst = t - totaltimelost$. • <code>timelost</code> calculates the time lost due to a particular event occurring, within the interval $(0,t]$. In a single event survival model, this is the integral of the cif between $(0,t]$. If multiple survival models are specified in the merlin model then by default all are assumed to be cause-specific event time models contributing to the calculation. This can be overridden using the <code>causes</code> option. • <code>totaltimelost</code> total time lost due to all competing events, within $(0,t]$. If multiple survival models are specified in the merlin model then by default all are assumed to be cause-specific event time models contributing to the calculation. This can be overridden using the <code>causes</code> option. <code>totaltimelost</code> is the sum of the <code>timelost</code> due to all causes. • <code>cifdifference</code> calculates the difference in cif predictions between values of a covariate specified using the <code>contrast</code> option. • <code>hdifference</code> calculates the difference in hazard predictions between values of a covariate specified using the <code>contrast</code> option. • <code>rmstdifference</code> calculates the difference in rmst predictions between values of a covariate specified using the <code>contrast</code> option. • <code>mudifference</code> calculates the difference in mu predictions between values of a covariate specified using the <code>contrast</code> option. • <code>etadifference</code> calculates the difference in eta predictions between values of a covariate specified using the <code>contrast</code> option.
type	<p>the type of prediction, either:</p> <ul style="list-style-type: none"> • <code>fixedonly</code> prediction calculated based only on the fixed effects; the default. • <code>marginal</code> prediction calculated marginally with respect to the latent variables. the <code>stat</code> is calculated by integrating the prediction function with respect to all the latent variables over their entire support.

predmodel	specifies which model to obtain predictions from; default is predmodel=1
causes	is for use when calculating predictions from a competing risks merlin model. By default, cif, rmst, timelost and totaltimelost assume that all survival models included in the merlin model are cause-specific hazard models contributing to the calculation. If this is not the case, then you can specify which models (indexed using the order they appear in your merlin model by using the causes option, e.g. causes=c(1,2).
at	specify covariate values for prediction. Fixed values of covariates should be specified in a list e.g. at = c("trt" = 1, "age" = 50).
contrast	specifies the values of a covariate to be used when comparing statistics, such as when using the cifdifference option to compare cumulative incidence functions, e.g. contrast = c("trt" = 0, "trt" = 1).
...	other options

Author(s)

Emma C. Martin, Alessandro Gasparini and Michael J. Crowther

References

Crowther MJ. Extended multivariate generalised linear and non-linear mixed effects models. <https://arxiv.org/abs/1710.02223>

Crowther MJ. merlin - a unified framework for data analysis and methods development in Stata. <https://arxiv.org/abs/1806.01615>

Martin EC, Gasparini A, Crowther MJ. merlin - an R package for mixed effects regression of linear, non-linear and user-defined models.

See Also

[merlin](#)

Examples

```
library(merlin)
data(pbc.merlin, package = "merlin")

# Linear fixed-effects model
mod <- merlin(model = list(logb ~ year),
             family = "gaussian",
             data = pbc.merlin)
predict(mod, stat="eta", type="fixedonly")
```

print.merlin	<i>Print merlin Fits</i>
--------------	--------------------------

Description

Print the coefficients from a merlin fit.

Usage

```
## S3 method for class 'merlin'
print(x, digits = max(3L, getOption("digits") - 3L), ...)
```

Arguments

x	An object of class merlin.
digits	The number of significant digits to use when printing.
...	Not used.

print.mlrcs	<i>Print mlrcs Fits</i>
-------------	-------------------------

Description

Print the coefficients from a mlrcs fit.

Usage

```
## S3 method for class 'mlrcs'
print(x, digits = max(3L, getOption("digits") - 3L), simplify = TRUE, ...)
```

Arguments

x	An object of class mlrcs.
digits	The number of significant digits to use when printing.
simplify	Should non-interpretable coefficients be hidden (e.g. splines and flexible polynomials terms)? Defaults to TRUE.
...	Not used.

print.mlsurv	<i>Print mlsurv Fits</i>
--------------	--------------------------

Description

Print the coefficients from a mlsurv fit.

Usage

```
## S3 method for class 'mlsurv'
print(x, digits = max(3L, getOption("digits") - 3L), simplify = TRUE, ...)
```

Arguments

x	An object of class mlsurv.
digits	The number of significant digits to use when printing.
simplify	Should non-interpretable coefficients be hidden (e.g. splines and flexible polynomials terms)? Defaults to TRUE.
...	Not used.

sim3	<i>Simulated 3-level survival data</i>
------	--

Description

Simulated 3-level survival data

Usage

```
data(sim3)
```

Format

A data frame...

summary.merlin	<i>Summarizing merlin Fits</i>
----------------	--------------------------------

Description

These functions are all methods for class merlin or summary.merlin objects.

Usage

```
## S3 method for class 'merlin'
summary(object, sig = 0.95, ...)

## S3 method for class 'summary.merlin'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

object	An object of class merlin
sig	Significance level for confidence intervals. Defaults to 0.95.
...	Not used.
x	An object of class summary.merlin
digits	The number of significant digits to use when printing.

summary.mlrcs	<i>Summarizing mlrcs Fits</i>
---------------	-------------------------------

Description

These functions are all methods for class mlrcs or summary.mlrcs objects.

Usage

```
## S3 method for class 'mlrcs'
summary(object, sig = 0.95, ...)

## S3 method for class 'summary.mlrcs'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

object	An object of class mlrcs
sig	Significance level for confidence intervals. Defaults to 0.95.
...	Not used.
x	An object of class summary.mlrcs
digits	The number of significant digits to use when printing.

summary.mlsurv	<i>Summarizing mlsurv Fits</i>
----------------	--------------------------------

Description

These functions are all methods for class `mlsurv` or `summary.mlsurv` objects.

Usage

```
## S3 method for class 'mlsurv'
summary(object, sig = 0.95, ...)

## S3 method for class 'summary.mlsurv'
print(x, digits = max(3, getOption("digits") - 3), simplify = TRUE, ...)
```

Arguments

<code>object</code>	An object of class <code>mlsurv</code>
<code>sig</code>	Significancy level for confidence intervals. Defaults to 0.95.
<code>...</code>	Not used.
<code>x</code>	An object of class <code>summary.mlsurv</code>
<code>digits</code>	The number of significant digits to use when printing.
<code>simplify</code>	Should non-interpretable coefficients be hidden (e.g. splines and flexible polynomials terms)? Defaults to TRUE.

<code>vcov.merlin</code>	<i>Calculate Variance-Covariance Matrix for a merlin Model Object</i>
--------------------------	---

Description

Returns the variance-covariance matrix of all estimated parameters of a fitted `merlin` model.

Usage

```
## S3 method for class 'merlin'
vcov(object, ...)

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

Arguments

<code>object</code>	An object of class <code>merlin</code> or <code>summary.merlin</code> .
<code>...</code>	Not used.

vcov.mlrcs

Calculate Variance-Covariance Matrix for a mlrcs Model Object

Description

Returns the variance-covariance matrix of all estimated parameters of a fitted mlrcs model.

Usage

```
## S3 method for class 'mlrcs'
vcov(object, ...)

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

Arguments

object	An object of class mlrcs or summary.mlrcs.
...	Not used.

vcov.mlsurv

Calculate Variance-Covariance Matrix for a mlsurv Model Object

Description

Returns the variance-covariance matrix of all estimated parameters of a fitted mlsurv model.

Usage

```
## S3 method for class 'mlsurv'
vcov(object, ...)

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

Arguments

object	An object of class mlsurv or summary.mlsurv.
...	Not used.

Index

* datasets

- heart.valve, 4
 - psc, 30
 - sim3, 35
- coef.merlin, 3
- coef.mlrcs, 3
- coef.mlsurv, 4
- coef.summary.merlin (coef.merlin), 3
- coef.summary.mlrcs (coef.mlrcs), 3
- coef.summary.mlsurv (coef.mlsurv), 4
- heart.valve, 4
- logLik.merlin, 6
- logLik.mlrcs, 6
- logLik.mlsurv, 7
- logLik.summary.merlin (logLik.merlin), 6
- logLik.summary.mlrcs (logLik.mlrcs), 6
- logLik.summary.mlsurv (logLik.mlsurv), 7
- merlin, 7, 27, 29, 33
- merlin_util_ap, 11, 12
- merlin_util_ap_mod, 11, 13
- merlin_util_depvar, 11, 14
- merlin_util_ev, 11, 14
- merlin_util_ev_deriv, 11, 15
- merlin_util_ev_deriv2, 11, 16
- merlin_util_ev_deriv2_mod, 11, 17
- merlin_util_ev_deriv_mod, 11, 17
- merlin_util_ev_integ, 11, 18
- merlin_util_ev_integ_mod, 11, 19
- merlin_util_ev_mod, 11, 20
- merlin_util_timevar, 11, 20
- merlin_util_xzb, 11, 21
- merlin_util_xzb_deriv, 11, 22
- merlin_util_xzb_deriv2, 11, 23
- merlin_util_xzb_deriv2_mod, 11, 23
- merlin_util_xzb_deriv_mod, 11, 24
- merlin_util_xzb_integ, 11, 25
- merlin_util_xzb_integ_mod, 11, 26
- merlin_util_xzb_mod, 11, 26
- mlrcs, 27
- mlsurv, 28
- optim, 11
- psc, 30, 31
- predict.merlin, 11, 31
- print.merlin, 34
- print.mlrcs, 34
- print.mlsurv, 35
- print.summary.merlin (summary.merlin), 36
- print.summary.mlrcs (summary.mlrcs), 36
- print.summary.mlsurv (summary.mlsurv), 37
- sim3, 35
- summary.merlin, 36
- summary.mlrcs, 36
- summary.mlsurv, 37
- Surv, 28
- vcov.merlin, 37
- vcov.mlrcs, 38
- vcov.mlsurv, 38
- vcov.summary.merlin (vcov.merlin), 37
- vcov.summary.mlrcs (vcov.mlrcs), 38
- vcov.summary.mlsurv (vcov.mlsurv), 38