

# Package ‘melt’

May 6, 2022

**Type** Package

**Title** Multiple Empirical Likelihood Tests

**Version** 1.5.1

**Description** Performs multiple empirical likelihood tests for regression analysis. Much of its functionality and syntax mimics the corresponding base R functions. The core computational routines are implemented using the 'Eigen' C++ library and 'RcppEigen' interface, with OpenMP for parallel computation. Additional functions are available for multiple testing for the analysis of experimental designs. Details of the testing procedures are given in Kim et al. (2021) <[arxiv:2112.09206](https://arxiv.org/abs/2112.09206)>.

**URL** <https://github.com/markean/melt>

**BugReports** <https://github.com/markean/melt/issues>

**License** GPL (>= 2)

**Depends** R (>= 4.0.0)

**Imports** methods, Rcpp (>= 1.0.8), stats

**LinkingTo** Rcpp, RcppEigen, RcppProgress

**Encoding** UTF-8

**RoxygenNote** 7.1.2

**NeedsCompilation** yes

**Suggests** spelling, covr, knitr, rmarkdown, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Language** en-US

**Author** Eunseop Kim [aut, cre],  
Steven MacEachern [aut, ths],  
Mario Peruggia [aut, ths]

**Maintainer** Eunseop Kim <[kim.7302@osu.edu](mailto:kim.7302@osu.edu)>

**Repository** CRAN

**Date/Publication** 2022-05-06 09:30:02 UTC

**R topics documented:**

CEL-class . . . . .	2
clothianidin . . . . .	4
coef-method . . . . .	4
confint . . . . .	5
confreg . . . . .	6
ConfregEL-class . . . . .	7
ControlEL-class . . . . .	8
EL-class . . . . .	9
eld . . . . .	10
ELD-class . . . . .	11
el_control . . . . .	12
el_eval . . . . .	13
el_glm . . . . .	14
el_lm . . . . .	16
el_mean . . . . .	18
el_pairwise . . . . .	19
el_test . . . . .	20
GLM-class . . . . .	21
lht . . . . .	21
LM-class . . . . .	23
logLik . . . . .	24
logLikEL-class . . . . .	24
plot . . . . .	25
print-method . . . . .	25
summary-method . . . . .	26
SummaryLM-class . . . . .	27
<b>Index</b>	<b>28</b>

---

CEL-class	<i>S4 class CEL</i>
-----------	---------------------

---

**Description**

S4 class for constrained empirical likelihood. It inherits from [EL](#) class. Note that `optim` slot has constrained optimization results with respect to parameters, not the Lagrange multiplier.

**Details**

Let  $l(\theta)$  denote the minus twice the empirical log-likelihood ratio function. We consider a linear hypothesis of the form

$$L\theta = r,$$

where the left-hand-side  $L$  is a  $q$  by  $p$  matrix and the right-hand-side  $r$  is a  $q$ -dimensional vector. Under some regularity conditions,  $l(\theta)$  converges in distribution to  $\chi_q^2$  under the constraint of hypothesis, i.e.,

$$\min_{\theta: L\theta=r} l(\theta) \rightarrow_d \chi_q^2.$$

Minimization of  $l(\theta)$  with respect to  $\theta$  is computationally expensive since it implicitly involves the evaluation step as described in [EL](#). Further, depending on the form of  $g(X_i, \theta)$  and the constraint, the optimization problem can be nonconvex and have multiple local minima. For this reason, **melt** only considers linear hypotheses and performs local minimization of  $l(\theta)$  using projected gradient descent method. With the orthogonal projection matrix  $P$  and a step size  $\gamma$ , the algorithm updates  $\theta$  as

$$\theta^{(k+1)} \leftarrow \theta^{(k)} - \gamma P \nabla l(\theta^{(k)}),$$

where  $\nabla l(\theta^{(k)})$  denotes the gradient of  $l$  at  $\theta^{(k)}$ . The first order optimality condition is  $P \nabla l(\theta) = 0$ , which is used as the stopping criterion.

## Slots

`optim` A list with the following optimization results:

- `method` Character for method dispatch in internal functions.
- `par` Parameter value that minimizes the constrained empirical likelihood.
- `lambda` Lagrange multiplier of the dual problem.
- `iterations` Number of iterations performed.
- `convergence` Convergence status.

`logp` Log probabilities obtained from constrained empirical likelihood.

`logl` Constrained empirical log-likelihood.

`loglr` Constrained empirical log-likelihood ratio.

`statistic` Minus twice the constrained empirical log-likelihood ratio statistic that has an asymptotic chi-square distribution.

`df` Degrees of freedom of the statistic.

`pval`  $p$ -value of the statistic.

`npar` Number of parameters.

`weights` Rescaled weights used for model fitting.

`data` Data matrix used for model fitting.

`coefficients` Maximum empirical likelihood estimates of the parameters.

## References

Adimari, Gianfranco, and Annamaria Guolo. 2010. "A Note on the Asymptotic Behaviour of Empirical Likelihood Statistics." *Statistical Methods & Applications* 19 (4): 463–76. doi:10.1007/s1026001001379.

Qin, Jing, and Jerry Lawless. 1995. "Estimating Equations, Empirical Likelihood and Constraints on Parameters." *Canadian Journal of Statistics* 23 (2): 145–59. doi:10.2307/3315441.

## Examples

```
showClass("CEL")
```

---

clothianidin	<i>Clothianidin concentration in maize plants</i>
--------------	---

---

**Description**

A dataset summarizing field experiments result of seed treatments on clothianidin concentration.

**Usage**

```
data("clothianidin")
```

**Format**

A data frame with 102 rows and 3 variables:

**blk** New blocks constructed from original data. The format is 'days post planting\_original block\_year'.

**trt** Seed treatment.

**clo** Log transformed clothianidin concentration ( $\mu\text{g}$ ).

**Details**

The original data is provided by Alford and Krupke (2017). Only some of the shoot region observations are taken from the original data and processed for illustration.

**Source**

Alford, Adam, and Christian H. Krupke. 2017. "Translocation of the Neonicotinoid Seed Treatment Clothianidin in Maize." Edited by Michael J. Stout. PLOS ONE 12 (3): e0173836. [doi:10.1371/journal.pone.0173836](https://doi.org/10.1371/journal.pone.0173836).

---

coef-method	<i>Maximum empirical likelihood estimates</i>
-------------	---

---

**Description**

Extracts model coefficients from a model.

**Usage**

```
## S4 method for signature 'EL'
coef(object, ...)
```

**Arguments**

object	A fitted <a href="#">EL</a> object.
...	Not used.

**Examples**

```
fit <- el_lm(formula = mpg ~ wt, data = mtcars)
coef(fit)
```

---

 confint

*Confidence intervals for model parameters*


---

**Description**

Computes confidence intervals for one or more parameters in a fitted model. Package **melt** adds a method for objects inheriting from class [EL](#).

**Usage**

```
## S4 method for signature 'EL'
confint(
  object,
  parm,
  level = 0.95,
  ...,
  cv = qchisq(level, 1L),
  control = el_control()
)
```

**Arguments**

object	A fitted <a href="#">EL</a> object.
parm	A specification of which parameters are to be given confidence intervals, either a vector of numbers or a vector of names. If missing, all parameters are considered.
level	A confidence level required. Defaults to 0.95.
...	Not used.
cv	A critical value for calibration of empirical likelihood ratio statistic. Defaults to <code>qchisq(level, 1L)</code> .
control	A list of control parameters set by <a href="#">el_control</a> .

**Value**

A matrix with columns giving lower and upper confidence limits for each parameter. In contrast to other methods that rely on studentization, the lower and upper limits obtained from empirical likelihood do not correspond to the  $(1 - \text{level}) / 2$  and  $1 - (1 - \text{level}) / 2$  in %, respectively.

## References

Kim, E., MacEachern, S., and Peruggia, M., (2021), "Empirical Likelihood for the Analysis of Experimental Designs," [arxiv:2112.09206](https://arxiv.org/abs/2112.09206).

Owen, Art. 1990. "Empirical Likelihood Ratio Confidence Regions." *The Annals of Statistics* 18 (1): 90–120. [doi:10.1214/aos/11176347494](https://doi.org/10.1214/aos/11176347494).

## See Also

[confreg](#), [el\\_control](#), [lht](#)

## Examples

```
fit <- el_lm(formula = mpg ~ wt, data = mtcars)
confint(fit)
```

---

confreg

*Confidence region for model parameters*

---

## Description

Computes boundary points of a two-dimensional confidence region for model parameters.

## Usage

```
## S4 method for signature 'EL'
confreg(
  object,
  parm,
  level = 0.95,
  cv = qchisq(level, 2L),
  npoints = 50L,
  control = el_control()
)
```

## Arguments

object	A fitted <a href="#">EL</a> object.
parm	A specification of which parameters are to be given a confidence region, either a vector of numbers or a vector of names. It should be a vector of length two of the form $c(x, y)$ . If missing, the first two parameter in object are considered.
level	A confidence level required. Defaults to 0.95.
cv	A critical value for calibration of empirical likelihood ratio statistic. Defaults to <code>qchisq(level, 2L)</code> .
npoints	The number of boundary points to compute. Defaults to 50.
control	A list of control parameters set by <a href="#">el_control</a> .

**Value**

An object of class [ConfregEL](#).

**References**

Owen, Art. 1990. "Empirical Likelihood Ratio Confidence Regions." *The Annals of Statistics* 18 (1): 90–120. doi:10.1214/aos/1176347494.

**See Also**

[confint](#), [el\\_control](#), [lht](#), [plot](#)

**Examples**

```
par <- c(0, 0, 0)
x <- matrix(rnorm(90), ncol = 3)
fit <- el_mean(par, x)
confreg(fit, parm = c(1, 3))
```

---

ConfregEL-class

*S4 class* [ConfregEL](#)

---

**Description**

S4 class for confidence region.

**Slots**

`points` A numeric matrix with two columns for boundary points of a confidence region.

`estimates` A numeric vector of length two for parameter estimates.

`level` A confidence level required.

`cv` A critical value for calibration of empirical likelihood ratio statistic.

`pnames` A character vector of length two for the name of parameters.

**Examples**

```
showClass("ConfregEL")
```

---

ControlEL-class	<i>S4 class</i> <a href="#">ControlEL</a>
-----------------	---

---

### Description

S4 class for details of computation of empirical likelihood.

### Slots

`maxit` Maximum number of iterations for the optimization with respect to  $\theta$ .

`maxit_l` Maximum number of iterations for the optimization with respect to  $\lambda$ .

`tol` Convergence tolerance denoted by  $\epsilon$ . The iteration stops when

$$\|P\nabla l(\theta^{(k)})\| < \epsilon.$$

`tol_l` Relative convergence tolerance denoted by  $\delta$ . The iteration stops when

$$\|\lambda^{(k)} - \lambda^{(k-1)}\| < \delta \|\lambda^{(k-1)}\| + \delta^2.$$

`step` Step size  $\gamma$  for the projected gradient descent method.

`th` Threshold for the negative empirical log-likelihood ratio value. The iteration stops if the value exceeds the threshold. Defaults to NULL and sets the threshold to  $200 * d$ , where  $d$  corresponds to the degrees of freedom of the limiting chi-squared distribution of the statistic.

`nthreads` Number of threads for parallel computation via OpenMP (if available). Defaults to the half of the available threads. For better performance, it is recommended to limit the number of threads to the number of physical cores. Note that it only applies to the following functions that involve multiple evaluations or minimizations:

- [confreg](#)
- [el\\_lm](#)
- [el\\_glm](#)
- [eld](#)

### Examples

```
showClass("ControlEL")
```



---

 EL-class

 S4 class *EL*


---

**Description**

S4 class for empirical likelihood.

**Details**

Let  $X_i$  be independent and identically distributed  $p$ -dimensional random variable from an unknown distribution  $F$  for  $i = 1, \dots, n$ . For a parameter of interest  $\theta(F) \in \mathbb{R}^p$ , consider a  $p$ -dimensional smooth estimating function  $g(X_i, \theta)$  with a moment condition

$$E[g(X_i, \theta)] = 0.$$

We assume that there exists a unique  $\theta_0$  that solves the above equation. Given a value of  $\theta$ , the (profile) empirical likelihood ratio is defined by

$$\mathcal{R}(\theta) = \max_{p_i} \left\{ \prod_{i=1}^n np_i : \sum_{i=1}^n p_i g(X_i, \theta) = 0, p_i \geq 0, \sum_{i=1}^n p_i = 1 \right\}.$$

The Lagrange multiplier  $\lambda \equiv \lambda(\theta)$  of the dual problem leads to

$$p_i = \frac{1}{n} \frac{1}{1 + \lambda^\top g(X_i, \theta)},$$

where  $\lambda$  solves

$$\frac{1}{n} \sum_{i=1}^n \frac{g(X_i, \theta)}{1 + \lambda^\top g(X_i, \theta)} = 0.$$

Then the empirical log-likelihood ratio is given by

$$\log \mathcal{R}(\theta) = \max_{\lambda} \sum_{i=1}^n \log(1 + \lambda^\top g(X_i, \theta)).$$

This problem can be efficiently solved by the Newton-Raphson method when the zero vector is contained in the interior of the convex hull of  $\{g(X_i, \theta)\}_{i=1}^n$ .

Under some regularity conditions, it is known that  $-2 \log \mathcal{R}(\theta_0)$  converges in distribution to  $\chi_p^2$ , where  $\chi_p^2$  has a chi-square distribution with  $p$  degrees of freedom.

**Slots**

optim A list with the following optimization results:

- method Character for method dispatch in internal functions.
- par Parameter value specified.
- lambda Lagrange multiplier of the dual problem.
- iterations Number of iterations performed.

- convergence Convergence status.

logp Log probabilities obtained from empirical likelihood.

logl Empirical log-likelihood.

loglr Empirical log-likelihood ratio.

statistic Minus twice the empirical log-likelihood ratio statistic that has an asymptotic chi-square distribution.

df Degrees of freedom of the statistic.

pval  $p$ -value of the statistic.

npar Number of parameters.

weights Rescaled weights used for model fitting.

data Data matrix used for model fitting.

coefficients Maximum empirical likelihood estimates of the parameters.

## References

Glenn, N.L., and Yichuan Zhao. 2007. “Weighted Empirical Likelihood Estimates and Their Robustness Properties.” *Computational Statistics & Data Analysis* 51 (10): 5130–41. doi:10.1016/j.csda.2006.07.032.

Qin, Jin, and Jerry Lawless. 1994. “Empirical Likelihood and General Estimating Equations.” *The Annals of Statistics* 22 (1): 300–325. doi:10.1214/aos/1176325370.

## Examples

```
showClass("EL")
```

---

eld	<i>Empirical likelihood displacement</i>
-----	--

---

## Description

Computes empirical likelihood displacement for model diagnostics and outlier detection.

## Usage

```
## S4 method for signature 'EL'
eld(object, control = el_control())
```

## Arguments

object	A fitted <a href="#">EL</a> object.
control	A list of control parameters set by <a href="#">el_control</a> .

**Details**

Let  $L(\theta)$  be the empirical log-likelihood function based on the full sample with  $n$  observations. The maximum empirical likelihood estimate is denoted by  $\hat{\theta}$ . Consider a reduced sample with the  $i$ th observation deleted and the corresponding estimate  $\hat{\theta}_{(i)}$ . The empirical likelihood displacement is defined by

$$\text{ELD}_i = 2\{L(\hat{\theta}) - L(\hat{\theta}_{(i)})\}.$$

If  $\text{ELD}_i$  is large, then the  $i$ th observation is an influential point and can be inspected as a possible outlier. `eld` computes  $\text{ELD}_i$  for  $i = 1, \dots, n$ .

**Value**

An object of class `ELD`.

**References**

Lazar, Nicole A. 2005. "Assessing the Effect of Individual Data Points on Inference From Empirical Likelihood." *Journal of Computational and Graphical Statistics* 14 (3): 626–42. doi:10.1198/106186005X59568.

Zhu, H., J. G. Ibrahim, N. Tang, and H. Zhang. 2008. "Diagnostic Measures for Empirical Likelihood of General Estimating Equations." *Biometrika* 95 (2): 489–507. doi:10.1093/biomet/asm094.

**See Also**

[el\\_control](#), [el\\_eval](#), [plot](#)

**Examples**

```
x <- rnorm(10L)
y <- 10
fit <- el_mean(0, c(x, y))
eld(fit)
```

---

ELD-class

*S4 class* `ELD`

---

**Description**

S4 class for empirical likelihood displacement.

**Slots**

`eld` A numeric vector of empirical likelihood displacement values.

**Examples**

```
showClass("ELD")
```

---

el\_control                      *Control parameters for computation*

---

### Description

Specifies details of computation of (constrained) empirical likelihood.

### Usage

```
el_control(
  maxit = 200L,
  maxit_l = 25L,
  tol = 1e-06,
  tol_l = 1e-06,
  step = NULL,
  th = NULL,
  nthreads
)
```

### Arguments

maxit	Maximum number of iterations for constrained minimization of empirical likelihood. Defaults to 200.
maxit_l	Maximum number of iterations of evaluation of empirical likelihood. Defaults to 25.
tol	Convergence tolerance for the constrained minimization. Defaults to 1e-06.
tol_l	Relative convergence tolerance for the evaluation. Defaults to 1e-06.
step	Step size for projected gradient method. Defaults to NULL and set to the reciprocal of sample size.
th	Threshold for negative empirical log-likelihood ratio value. The iteration stops if the value exceeds the threshold. Defaults to NULL.
nthreads	Number of threads for parallel computation via OpenMP (if available).

### Value

An object of class of [ControlEL](#).

### See Also

[el\\_eval](#), [lht](#)

### Examples

```
optcfg <- el_control(maxit = 300L, th = 200, nthreads = 1L)
```

---

el_eval	<i>Empirical likelihood for general estimating functions</i>
---------	--

---

**Description**

Computes empirical likelihood with general estimating functions.

**Usage**

```
el_eval(g, weights = NULL, control = el_control())
```

**Arguments**

g	A numeric matrix, or an object that can be coerced to a numeric matrix. Each row corresponds to an observation of an estimating function.
weights	An optional numeric vector of weights to be used in the fitting process. Defaults to NULL, corresponding to identical weights. If non-NULL, weighted empirical likelihood is computed.
control	A list of control parameters set by <a href="#">el_control</a> .

**Details**

`el_eval` evaluates empirical likelihood with a  $n$  by  $p$  numeric matrix argument `g`, whose  $i$ th row is  $g(X_i, \theta)$ . Since the estimating function can be arbitrary, `el_eval` does not return an object of class [EL](#), and the associated generics and methods are not applicable.

**Value**

A list with the following components:

optim	A list with the following optimization results: <ul style="list-style-type: none"> <li>• <code>lambda</code> Lagrange multiplier of the dual problem.</li> <li>• <code>iterations</code> Number of iterations performed.</li> <li>• <code>convergence</code> Convergence status.</li> </ul>
logp	Log probabilities obtained from empirical likelihood.
logl	Empirical log-likelihood.
loglr	Empirical log-likelihood ratio.
statistic	Minus twice the empirical log-likelihood ratio statistic that has an asymptotic chi-square distribution.
df	Degrees of freedom of the statistic.
pval	$p$ -value of the statistic.
npar	Number of parameters.
weights	Rescaled weights used for model fitting.

## References

Qin, Jin, and Jerry Lawless. 1994. "Empirical Likelihood and General Estimating Equations." *The Annals of Statistics* 22 (1): 300–325. doi:10.1214/aos/1176325370.

## See Also

[el\\_control](#)

## Examples

```
# test for variance with known mean
x <- rnorm(100L)
sigma <- 1
g <- x^2 - sigma^2
el_eval(g)
```

---

el\_glm

*Empirical likelihood for generalized linear models*

---

## Description

Fits a generalized linear model with empirical likelihood.

## Usage

```
el_glm(
  formula,
  family = gaussian,
  data,
  weights = NULL,
  na.action,
  control = el_control(),
  model = TRUE,
  start = NULL,
  etastart = NULL,
  mustart = NULL,
  ...
)
```

## Arguments

**formula** An object of class "[formula](#)" (or one that can be coerced to that class): a symbolic description of the model to be fitted.

**family** A description of the error distribution and link function to be used in the model. Only the result of a call to a family function is supported. See 'Details'.

data	An optional data frame, list or environment (or object coercible by <code>as.data.frame</code> to a data frame) containing the variables in the formula. If not found in data, the variables are taken from <code>environment(formula)</code> .
weights	An optional numeric vector of weights to be used in the fitting process. Defaults to NULL, corresponding to identical weights. If non-NULL, weighted empirical likelihood is computed.
na.action	A function which indicates what should happen when the data contain NAs. The default is set by the <code>na.action</code> setting of <code>options</code> , and is <code>na.fail</code> if that is unset.
control	A list of control parameters set by <code>el_control</code> .
model	A logical. If TRUE the data matrix used for fitting is returned.
start	Starting values for the parameters in the linear predictor. Defaults to NULL and is passed to <code>glm.fit</code> .
etastart	Starting values for the linear predictor. Defaults to NULL and is passed to <code>glm.fit</code> .
mustart	Starting values for the vector of means. Defaults to NULL and is passed to <code>glm.fit</code> .
...	Additional arguments to be passed to <code>glm.control</code> .

### Details

The available families and link functions are as follows:

- gaussian: identity, log, and inverse.
- binomial: logit, probit, and log.
- poisson: log, identity, and sqrt.

Included in the tests are the overall test with

$$H_0 : \beta_1 = \beta_2 = \cdots = \beta_{p-1} = 0,$$

and the tests for each parameter with

$$H_{0j} : \beta_j = 0, \quad j = 0, \dots, p-1.$$

The test results are returned as `optim` and `parTests`, respectively.

### Value

An object of class of `GLM`.

### References

Chen, Song Xi, and Hengjian Cui. 2003. "An Extended Empirical Likelihood for Generalized Linear Models." *Statistica Sinica* 13: 69–81.

### See Also

[el\\_control](#), [el\\_lm](#), [lht](#)

**Examples**

```
n <- 50
x <- rnorm(n)
x2 <- rnorm(n)
l <- -2 + 0.2 * x + 3 * x2
mu <- 1 / (1 + exp(-l))
y <- rbinom(n, 1, mu)
df <- data.frame(y, x, x2)
fit <- el_glm(y ~ x + x2, family = binomial, df)
summary(fit)
```

el\_lm

*Empirical likelihood for linear models***Description**

Fits a linear model with empirical likelihood.

**Usage**

```
el_lm(
  formula,
  data,
  weights = NULL,
  na.action,
  control = el_control(),
  model = TRUE,
  ...
)
```

**Arguments**

formula	An object of class " <b>formula</b> " (or one that can be coerced to that class): a symbolic description of the model to be fitted.
data	An optional data frame, list or environment (or object coercible by <a href="#">as.data.frame</a> to a data frame) containing the variables in the formula. If not found in data, the variables are taken from <code>environment(formula)</code> .
weights	An optional numeric vector of weights to be used in the fitting process. Defaults to NULL, corresponding to identical weights. If non-NULL, weighted empirical likelihood is computed.
na.action	A function which indicates what should happen when the data contain NAs. The default is set by the <code>na.action</code> setting of <a href="#">options</a> , and is <code>na.fail</code> if that is unset.
control	A list of control parameters set by <a href="#">el_control</a> .
model	A logical. If TRUE the data matrix used for fitting is returned.
...	Additional arguments to be passed to the low level regression fitting functions. See 'Details'.



## Details

Suppose that we observe  $n$  independent random variables  $(X_i, Y_i)$  from a common distribution, where  $X_i$  is the  $p$ -dimensional covariate (including the intercept if any) and  $Y_i$  is the response. We consider the following linear regression model:

$$Y_i = X_i^\top \beta + \epsilon_i,$$

where  $\beta = (\beta_0, \dots, \beta_{p-1})$  is an unknown  $p$ -dimensional parameter and the errors  $\epsilon_i$  are independent random variables that satisfy  $E(\epsilon_i | X_i) = 0$ . We assume that the errors have finite conditional variance. Then the least square estimator of  $\beta$  solves the following estimating equation:

$$\sum_{i=1}^n (Y_i - X_i^\top \beta) X_i = 0.$$

`el_lm` first computes the parameter estimates by calling `lm.fit` (with `...` if any) since the maximum empirical likelihood estimator is the same as the least square estimator in our model. Next, it performs hypothesis tests based on asymptotic chi-squared distribution of empirical likelihood ratio statistics. Included in the tests are the overall test with

$$H_0 : \beta_1 = \beta_2 = \dots = \beta_{p-1} = 0,$$

and the tests for each parameter with

$$H_{0j} : \beta_j = 0, \quad j = 0, \dots, p-1.$$

The test results are returned as `optim` and `parTests`, respectively.

## Value

An object of class of [LM](#).

## References

Owen, Art. 1991. "Empirical Likelihood for Linear Models." *The Annals of Statistics* 19 (4): 1725–47. [doi:10.1214/aos/1176348368](https://doi.org/10.1214/aos/1176348368).

## See Also

[el\\_control](#), [el\\_glm](#), [lht](#)

## Examples

```
fit <- el_lm(mpg ~ wt, mtcars)
summary(fit)
```

---

el_mean	<i>Empirical likelihood for the mean</i>
---------	--

---

### Description

Computes empirical likelihood for the mean.

### Usage

```
el_mean(par, x, weights = NULL, control = el_control(), model = TRUE)
```

### Arguments

par	A numeric vector of parameter values to be tested.
x	A numeric matrix, or an object that can be coerced to a numeric matrix. Each row corresponds to an observation.
weights	An optional numeric vector of weights to be used in the fitting process. Defaults to NULL, corresponding to identical weights. If non-NULL, weighted empirical likelihood is computed.
control	A list of control parameters set by <a href="#">el_control</a> .
model	A logical. If TRUE the data matrix used for model fitting is returned.

### Value

An object of class [EL](#).

### References

Owen, Art. 1990. "Empirical Likelihood Ratio Confidence Regions." *The Annals of Statistics* 18 (1): 90–120. doi:10.1214/aos/1176347494.

### See Also

[el\\_control](#), [el\\_eval](#), [lht](#)

### Examples

```
# scalar mean
par <- 0
x <- rnorm(100)
el_mean(par, x)

# vector mean
par <- c(0, 0)
x <- matrix(rnorm(100), ncol = 2)
el_mean(par, x)
```

```
# weighted data
par <- c(0, 0)
x <- matrix(rnorm(100), ncol = 2)
w <- rep(c(1, 2), each = 25)
el_mean(par, x, w)
```

---

el\_pairwise

*Empirical likelihood pairwise comparisons*


---

## Description

Tests all pairwise comparisons or comparisons with control for general block designs. Two single step asymptotic  $k$ -FWER (generalized family-wise error rate) controlling procedures are available: asymptotic Monte Carlo (AMC) and nonparametric bootstrap (NB).

## Usage

```
el_pairwise(
  formula,
  data,
  control = NULL,
  k = 1,
  alpha = 0.05,
  method = c("AMC", "NB"),
  B,
  nthread = 1,
  progress = TRUE,
  maxit = 10000,
  abstol = 1e-08
)
```

## Arguments

formula	A formula object. It must specify variables for response, treatment, and block as 'response ~ treatment   block'. Note that the use of vertical bar ( ) separating treatment and block.
data	A data frame containing the variables in the formula.
control	Optional character specifying the treatment for comparisons with control.
k	Single integer for $k$ in $k$ -FWER. Defaults to 1.
alpha	Level of the test. Defaults to 0.05.
method	Single character for the procedure to be used; either 'AMC' or 'NB' is supported. Defaults to 'AMC'.
B	Number of Monte Carlo samples for the AMC (number of bootstrap replicates for the NB).
nthread	Number of threads to be used with OpenMP. Only applied when the NB is chosen as the method. Defaults to 1.

progress	If TRUE, will show progress status.
maxit	Maximum number of iterations for optimization. Defaults to 10000.
abstol	Absolute convergence tolerance for optimization. Defaults to 1e-08.

**Value**

A list with class `c("pairwise", "melt")`.

**References**

Kim, E., MacEachern, S., and Peruggia, M., (2021), "Empirical Likelihood for the Analysis of Experimental Designs," [arxiv:2112.09206](https://arxiv.org/abs/2112.09206).

**Examples**

```
# all pairwise comparisons
data("clothianidin")
el_pairwise(clo ~ trt | blk, clothianidin, B = 1000)

# comparisons with control
el_pairwise(clo ~ trt | blk, clothianidin,
  control = "Naked", method = "NB",
  B = 500
)
```

---

el\_test

*Tests single hypothesis for general block designs*

---

**Description**

*This function is deprecated in favor of [lht](#) and will be removed in a future release.*

**Usage**

```
el_test(formula, data, lhs, rhs = NULL, maxit = 10000, abstol = 1e-08)
```

**Arguments**

formula	A formula object. It must specify variables for response, treatment, and block as 'response ~ treatment   block'. Note that the use of vertical bar ( ) separating treatment and block.
data	A data frame containing the variables in the formula.
lhs	Numeric matrix specifying linear hypothesis in terms of parameters.
rhs	Optional numeric vector for the right hand side of lhs. If not specified, it is set to 0 vector.
maxit	Maximum number of iterations for optimization. Defaults to 10000.
abstol	Absolute convergence tolerance for optimization. Defaults to 1e-08.

**Value**

A list of class `c("el_test", "melt")`.

**References**

Kim, E., MacEachern, S., and Peruggia, M., (2021), "Empirical Likelihood for the Analysis of Experimental Designs," [arxiv:2112.09206](https://arxiv.org/abs/2112.09206).

**Examples**

```
## Not run:
# test of no treatment effect
data("clothianidin")
el_test(clo ~ trt | blk, clothianidin,
  lhs = matrix(c(
    1, -1, 0, 0,
    0, 1, -1, 0,
    0, 0, 1, -1
  ), byrow = TRUE, nrow = 3)
)

## End(Not run)
```

---

GLM-class

*S4 class GLM*


---

**Description**

S4 class for generalized linear models with empirical likelihood. It inherits from [LM](#) class.

**Examples**

```
showClass("GLM")
```

---

lht

*Linear hypothesis test*


---

**Description**

Tests a linear hypothesis for objects that inherit from class [EL](#).

**Usage**

```
lht(object, rhs = NULL, lhs = NULL, control = el_control())
```

**Arguments**

object	A fitted <a href="#">EL</a> object.
rhs	A numeric vector for the right-hand-side of hypothesis, with as many entries as the rows in lhs. Defaults to NULL. See ‘Details’.
lhs	A numeric matrix, or an object that can be coerced to a numeric matrix. It specifies the left-hand-side of hypothesis. Each row gives a linear combination of parameters. The number of columns should be equal to the number of parameters in object. Defaults to NULL. See ‘Details’.
control	A list of control parameters set by <a href="#">el_control</a> .

**Details**

[lht](#) performs the constrained minimization of  $l(\theta)$  described in [CEL](#). `rhs` and `lhs` cannot be both NULL. For non-NULL `lhs`, it is required that `lhs` have full row rank  $q \leq p$  and  $p$  be equal to `object$npars`, the number of parameters in the fitted model.

Depending on the specification of `rhs` and `lhs`, we have the following three cases:

1. If both `rhs` and `lhs` are non-NULL, the constrained minimization is performed with the right-hand-side  $r$  and the left-hand-side  $L$  as

$$\min_{\theta: L\theta=r} l(\theta).$$

2. If `rhs` is NULL,  $r$  is set to the zero vector as

$$\min_{\theta: L\theta=0} l(\theta).$$

3. If `lhs` is NULL,  $L$  is set to the identity matrix and the problem reduces to evaluating at  $r$  as

$$l(r).$$

**Value**

If `lhs` is NULL, an object of class [EL](#) is returned. Otherwise, an object of class [CEL](#) is returned.

**References**

Adimari, Gianfranco, and Annamaria Guolo. 2010. “A Note on the Asymptotic Behaviour of Empirical Likelihood Statistics.” *Statistical Methods & Applications* 19 (4): 463–76. doi:10.1007/s1026001001379.

Qin, Jing, and Jerry Lawless. 1995. “Estimating Equations, Empirical Likelihood and Constraints on Parameters.” *Canadian Journal of Statistics* 23 (2): 145–59. doi:10.2307/3315441.

**See Also**

[el\\_control](#)

**Examples**

```

n <- 100L
x1 <- rnorm(n)
x2 <- rnorm(n)
y <- 1 + x1 + x2 + rnorm(n)
df <- data.frame(y, x1, x2)
fit <- el_lm(y ~ x1 + x2, df)
lhs <- matrix(c(0, 1, -1), nrow = 1L)
lht(fit, lhs = lhs)

# test of no treatment effect
data("clothianidin")
lhs2 <- matrix(c(
  1, -1, 0, 0,
  0, 1, -1, 0,
  0, 0, 1, -1
), byrow = TRUE, nrow = 3L)
fit2 <- el_lm(clo ~ -1 + trt, clothianidin)
lht(fit2, lhs = lhs2)

```

LM-class

*S4 class LM***Description**

S4 class for linear models with empirical likelihood. It inherits from [CEL](#) class.

**Details**

If there is no intercept in a model, `optim` slot need to be understood in terms of [EL](#) class since constrained optimization is not involved in the overall test.

**Slots**

`parTests` A list with the test results for each parameter:

- `statistic` A numeric vector of chi-squared statistics.
- `convergence` Convergence status of tests for each parameter.

`misc` A list with miscellaneous outputs from a model fitting function. They are used in other generics and methods.

**Examples**

```
showClass("LM")
```

---

logLik	<i>Empirical log-likelihood</i>
--------	---------------------------------

---

**Description**

Extracts empirical log-likelihood from a model evaluated at the estimated coefficients.

**Usage**

```
## S4 method for signature 'EL'
logLik(object, ...)
```

**Arguments**

object	A fitted <a href="#">EL</a> object.
...	Not used.

**Value**

An object of class [logLikEL](#).

**Examples**

```
fit <- el_lm(formula = mpg ~ wt, data = mtcars)
logLik(fit)
```

---

logLikEL-class	<i>S4 class <a href="#">logLikEL</a></i>
----------------	--

---

**Description**

S4 class for empirical log-likelihood.

**Slots**

logLik	Empirical log-likelihood.
df	Degrees of freedom or the number of (estimated) parameters in the model.

**Examples**

```
showClass("logLikEL")
```



---

plot	<i>Plot methods</i>
------	---------------------

---

### Description

Provides plot methods for objects that inherit from class [EL](#).

### Usage

```
## S4 method for signature 'ConfregEL'  
plot(x, y, ...)  
  
## S4 method for signature 'ELD'  
plot(x, y, ...)
```

### Arguments

x	An object to be plotted.
y	Not used.
...	Further graphical parameters (see <a href="#">par</a> ).

### Methods (by class)

- `ConfregEL`: Plots a two-dimensional confidence region for model parameters.
- `ELD`: Plots empirical likelihood displacement values versus observation index.

### See Also

[confreg.eld](#)

---

print-method	<i>Print methods</i>
--------------	----------------------

---

### Description

Provides print methods for objects that inherit from class [EL](#).

**Usage**

```
## S4 method for signature 'EL'
print(x, digits = max(3L, getOption("digits") - 3L), ...)

## S4 method for signature 'SummaryLM'
print(
  x,
  digits = max(3L, getOption("digits") - 3L),
  signif.stars = getOption("show.signif.stars"),
  ...
)

## S4 method for signature 'logLikEL'
print(x, digits = getOption("digits"), ...)
```

**Arguments**

x	An object to be printed.
...	Further arguments passed to other methods.
digits	The number of significant digits to be passed to <a href="#">format</a> .
signif.stars	Logical. If TRUE, ‘significance stars’ are printed for each coefficient.

---

summary-method

*Summary methods*


---

**Description**

Provides summary methods for objects that inherit from class [EL](#).

**Usage**

```
## S4 method for signature 'LM'
summary(object, ...)
```

**Arguments**

object	An object for which a summary is desired.
...	Additional arguments affecting the summary produced.

**Functions**

- `summary, LM-method`: Summarizes the results of the overall test and the tests for each parameter.

---

SummaryLM-class      *S4 class* [SummaryLM](#)

---

**Description**

S4 class for a summary of [LM](#) objects.

**Slots**

`statistic` Minus twice the constrained empirical log-likelihood ratio for the overall test of the model.

`df` Degrees of freedom of the statistic.

`convergence` Convergence status of the minimization.

`parMatrix` A numeric matrix of the test results of the parameters.

`weighted` A logical for whether the given model is weighted or not.

`na.action` Information returned by [model.frame](#) on the special handling of NAs.

`call` Matched call.

`terms` [terms](#) object used.

`aliased` Named logical vector showing if the original coefficients are aliased.

**Examples**

```
showClass("SummaryLM")
```

# Index

## \* datasets

- clothianidin, 4
- as.data.frame, 15, 16
- CEL, 2, 22, 23
- CEL-class, 2
- clothianidin, 4
- coef, EL-method (coef-method), 4
- coef-method, 4
- confint, 5, 7
- confint, EL-method (confint), 5
- confreg, 6, 6, 8, 25
- confreg, EL-method (confreg), 6
- ConfregEL, 7
- ConfregEL-class, 7
- ControlEL, 8, 12
- ControlEL-class, 8
- EL, 2–6, 9, 10, 13, 18, 21–26
- EL-class, 9
- el\_control, 5–7, 10, 11, 12, 13–18, 22
- el\_eval, 11, 12, 13, 18
- el\_glm, 8, 14, 17
- el\_lm, 8, 15, 16, 17
- el\_mean, 18
- el\_pairwise, 19
- el\_test, 20
- ELD, 11
- eld, 8, 10, 25
- eld, EL-method (eld), 10
- ELD-class, 11
- format, 26
- formula, 14, 16
- GLM, 15, 21
- GLM-class, 21
- glm.control, 15
- glm.fit, 15
- lht, 6, 7, 12, 15, 17, 18, 20, 21, 22
- LM, 17, 21, 23, 27
- LM-class, 23
- lm.fit, 17
- logLik, 24
- logLik, EL-method (logLik), 24
- logLikEL, 24
- logLikEL-class, 24
- model.frame, 27
- options, 15, 16
- par, 25
- plot, 7, 11, 25
- plot, ConfregEL-method (plot), 25
- plot, ELD-method (plot), 25
- print, EL-method (print-method), 25
- print, logLikEL-method (print-method), 25
- print, SummaryLM-method (print-method), 25
- print-method, 25
- summary, LM-method (summary-method), 26
- summary-method, 26
- SummaryLM, 27
- SummaryLM-class, 27
- terms, 27