Package 'OneShotEM'

August 21, 2026

Type Package

Title Efficient eM-Algorithm for One-Shot Device Data Analysis

Version 0.1.0

Description Implements the simple and efficient Expectation-Maximization (eM) algorithm proposed by Zhu, Li, Li, and Balakrishnan (2026) <doi:10.1080/03610918.2025.2515193> for parameter estimation in one-shot device accelerated life testing (ALT) data. Unlike traditional EM algorithms that impute exact failure times, this method treats failure counts between inspection intervals as missing data, resulting in faster convergence and enhanced numerical stability. Supports Exponential, Weibull, Lognormal, Gamma, and custom user-defined lifetime distributions under log-linear stress models. Standard errors, confidence intervals, model selection statistics (AIC, BIC, AICc, HQIC), residual diagnostics, and visualization tools are provided. References: Balakrishnan and Ling (2012) <doi:10.1016/j.csda.2011.09.010>, Fan, Balakrishnan, and Chang (2009) <doi:10.1080/00949650802142592>.

License GPL (>= 3)

LazyData true

Depends R (>= 4.0.0)

Imports stats, graphics, grDevices, utils, methods, numDeriv

Suggests testthat (>= 3.0.0), knitr, rmarkdown

Config/testthat/edition 3

Encoding UTF-8

Language en-US

RoxygenNote 7.3.3

VignetteBuilder knitr

NeedsCompilation no

Author Shikhar Tyagi [aut, cre] (ORCID:
<https://orcid.org/0000-0003-1606-0844>),
Arvind Pandey [aut],
Bhupendra Singh [aut],
Vrijesh Tripathi [aut]

Maintainer Shikhar Tyagi <shikhar1093tyagi@gmail.com>

Repository CRAN

Date/Publication 2026-08-21 12:41:11 UTC

Contents

AIC.oneshot_em 2
BIC.oneshot_em 3
coef.oneshot_em 3
electro_explosive 4
em_control 5
logLik.oneshot_em 6
oneshot_em 6
oneshot_fit 8
oneshot_residuals 9
oneshot_tm 9
plot.oneshot_em 11
predict.oneshot_em 11
print.oneshot_em 12
print.summary.oneshot_em 13
summary.oneshot_em 13
vcov.oneshot_em 14

Index **15**

`AIC.oneshot_em` *Extract Akaike Information Criterion (AIC)*

Description

Extract Akaike Information Criterion (AIC)

Usage

```
## S3 method for class 'oneshot_em'
AIC(object, ..., k = 2)
```

Arguments

`object`	an object of class `"oneshot_em"`.
`...`	optional additional objects.
`k`	penalty per parameter (default: 2).

Value

A numeric scalar of class `"numeric"` containing the calculated Akaike Information Criterion (AIC) value for the fitted model.

BIC.oneshot_em *Extract Bayesian Information Criterion (BIC)*

Description

Extract Bayesian Information Criterion (BIC)

Usage

```
## S3 method for class 'oneshot_em'
BIC(object, ...)
```

Arguments

object	an object of class "oneshot_em".
...	optional additional objects.

Value

A numeric scalar of class "numeric" containing the calculated Bayesian Information Criterion (BIC) value for the fitted model.

coef.oneshot_em *Extract Model Coefficients*

Description

Extract Model Coefficients

Usage

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

Arguments

object	an object of class "oneshot_em".
...	additional arguments (unused).

Value

A named numeric vector of class "numeric" containing the maximum likelihood estimates of the model parameters.

`electro_explosive` *Electro-Explosive Device Accelerated Life Test Data*

Description

One-shot device accelerated life test (ALT) dataset reported by Fan, Balakrishnan, and Chang (2009) and analyzed by Zhu et al. (2026).

Usage

```
data(electro_explosive)
```

Format

A data frame with 9 rows and 4 variables:

temp temperature stress setting in degrees Celsius (35, 45, 55)

it inspection time in hours (10, 20, 30)

n number of devices tested at the specified inspection time (10 per row, total 90)

r number of observed device failures at inspection

Source

Fan, T., Balakrishnan, N., & Chang, C. (2009). The Bayesian approach for highly reliable electro-explosive devices using one-shot device testing. *Journal of Statistical Computation and Simulation*, 79(9), 1143–1154. doi:10.1080/00949650802142592

References

Zhu, X., Li, Y., Li, T., & Balakrishnan, N. (2026). A simple and efficient eM-algorithm for one-shot device data analysis. *Communications in Statistics - Simulation and Computation*, 55(3), 959–970. doi:10.1080/03610918.2025.2515193

Examples

```
data(electro_explosive)
head(electro_explosive)
```

em_control *Control Parameters for EM Algorithms*

Description

Sets control parameters for the Expectation-Maximization algorithms.

Usage

```
em_control(
  maxit = 1000,
  tol = 1e-06,
  trace = FALSE,
  optimizer = c("nlm", "optim"),
  max_m_iter = 100,
  reltol = 1e-10
)
```

Arguments

maxit	maximum number of EM iterations (default: 1000).
tol	convergence tolerance based on maximum relative difference between parameter iterations (default: 1e-6).
trace	logical; if TRUE, iteration progress is printed (default: FALSE).
optimizer	optimization method for the M-step, either "nlm" or "optim" (default: "nlm").
max_m_iter	maximum iterations for the inner M-step optimization (default: 100).
reltol	relative tolerance for M-step optimizer (default: 1e-10).

Value

A list containing the control options.

Examples

```
ctrl <- em_control(maxit = 500, tol = 1e-5, trace = FALSE)
```

logLik.oneshot_em *Extract Log-Likelihood*

Description

Extract Log-Likelihood

Usage

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

Arguments

object	an object of class "oneshot_em".
...	additional arguments (unused).

Value

An object of class "logLik" representing the maximized log-likelihood value, with additional attributes "df" (degrees of freedom / number of estimated parameters) and "nobs" (total number of tested devices).

oneshot_em *Maximum Likelihood Estimation via New eM-Algorithm for One-Shot Device Data*

Description

Fits lifetime models for one-shot device data using the new expectation-maximization (eM) algorithm proposed by Zhu et al. (2026).

Usage

```
oneshot_em(
  formula,
  data,
  it = "it",
  dist = c("exponential", "weibull", "lognormal", "gamma", "custom"),
  sf_fun = NULL,
  theta0 = NULL,
  control = em_control()
)
```

Arguments

formula	an object of class formula (e.g., cbind(r, n) ~ temp or r/n ~ temp).
data	a data frame containing the variables in the model.
it	character string or vector specifying inspection time variable. If data is supplied, can be the column name of inspection time.
dist	lifetime distribution name: "exponential", "weibull", "lognormal", "gamma", or custom. Default is "exponential".
sf_fun	custom survival function of signature function(t, theta, X). Required if dist = "custom".
theta0	numeric vector of initial parameter values. If omitted, calculated automatically.
control	list of control parameters from em_control.

Value

An object of class "oneshot_em" containing:

coefficients	estimated model parameters
loglik	maximized log-likelihood value
iterations	number of EM iterations to convergence
converged	logical indicating successful convergence
vcov	variance-covariance matrix
std_err	standard errors of parameter estimates
dist	distribution used
method	"npm" (new proposed method)
data	input data frame
formula	model formula
terms	model terms object
control	control list

References

Zhu, X., Li, Y., Li, T., & Balakrishnan, N. (2026). A simple and efficient eM-algorithm for one-shot device data analysis. *Communications in Statistics - Simulation and Computation*, 55(3), 959–970. doi:10.1080/03610918.2025.2515193

Examples

```
data(electro_explosive)
fit_exp <- oneshot_em(cbind(r, n) ~ temp, data = electro_explosive,
                      it = "it", dist = "exponential")
summary(fit_exp)
```

`oneshot_fit` *Unified Wrapper for Fitting One-Shot Device ALT Models*

Description

Unified high-level fitting interface for one-shot device accelerated life test data.

Usage

```
oneshot_fit(
  formula,
  data,
  it = "it",
  dist = c("exponential", "weibull", "lognormal", "gamma", "custom"),
  method = c("npm", "tm"),
  sf_fun = NULL,
  theta0 = NULL,
  control = em_control()
)
```

Arguments

`formula`	an object of class `formula` (e.g., `cbind(r, n) ~ temp` or `r/n ~ temp`).
`data`	a data frame containing the variables in the model.
`it`	character string or vector specifying inspection time variable. If `data` is supplied, can be the column name of inspection time.
`dist`	lifetime distribution name: `"exponential"`, `"weibull"`, `"lognormal"`, `"gamma"`, or custom. Default is `"exponential"`.
`method`	estimation method: `"npm"` (new proposed eM-algorithm) or `"tm"` (traditional EM algorithm).
`sf_fun`	custom survival function of signature `function(t, theta, X)`. Required if `dist = "custom"`.
`theta0`	numeric vector of initial parameter values. If omitted, calculated automatically.
`control`	list of control parameters from `em_control`.

Value

An object of class `"oneshot_em"` containing model fitting results (see `oneshot_em` for full list of output elements).

Examples

```
data(electro_explosive)
fit <- oneshot_fit(cbind(r, n) ~ temp, data = electro_explosive,
                   it = "it", dist = "weibull", method = "npm")
print(fit)
```

oneshot_residuals *Standardized and Generalized Residuals for One-Shot Device EM Fit*

Description

Computes standardized Pearson residuals and randomized quantile residuals for one-shot device data models.

Usage

```
oneshot_residuals(fit)
```

Arguments

fit	an object of class "oneshot_em".

Value

A data frame containing:

observed	observed failure count
expected	expected failure count
pearson	standardized Pearson residual
quantile	randomized quantile residual

Examples

```
data(electro_explosive)
fit <- oneshot_em(cbind(r, n) ~ temp, data = electro_explosive, it = "it")
res <- oneshot_residuals(fit)
head(res)
```

oneshot_tm *Maximum Likelihood Estimation via Traditional EM-Algorithm for One-Shot Device Data*

Description

Fits lifetime models for one-shot device data using the traditional failure-time expectation-maximization (EM) algorithm for comparison with the new eM-algorithm (oneshot_em).

Usage

```
oneshot_tm(
  formula,
  data,
  it = "it",
  dist = c("exponential", "weibull", "lognormal", "gamma", "custom"),
  sf_fun = NULL,
  theta0 = NULL,
  control = em_control()
)
```

Arguments

formula	an object of class formula (e.g., cbind(r, n) ~ temp or r/n ~ temp).
data	a data frame containing the variables in the model.
it	character string or vector specifying inspection time variable. If data is supplied, can be the column name of inspection time.
dist	lifetime distribution name: "exponential", "weibull", "lognormal", "gamma", or custom. Default is "exponential".
sf_fun	custom survival function of signature function(t, theta, X). Required if dist = "custom".
theta0	numeric vector of initial parameter values. If omitted, calculated automatically.
control	list of control parameters from em_control.

Value

An object of class "oneshot_em" fitted via the traditional EM approach, containing:

coefficients	estimated model parameters
loglik	maximized log-likelihood value
iterations	number of EM iterations to convergence
converged	logical indicating successful convergence
vcov	variance-covariance matrix
std_err	standard errors of parameter estimates
dist	distribution used
method	"tm" (traditional method)
data	input data frame
formula	model formula
terms	model terms object
control	control list

References

Balakrishnan, N., & Ling, M. (2012). EM algorithm for one-shot device testing under the exponential distribution. *Computational Statistics & Data Analysis*, 56(3), 502–509.

Examples

```
data(electro_explosive)
fit_tm <- oneshot_tm(cbind(r, n) ~ temp, data = electro_explosive,
                     it = "it", dist = "exponential")
summary(fit_tm)
```

---

plot.oneshot_em *Diagnostic Plots for One-Shot Device EM Fit*

Description

Generates fitted survival curves, empirical vs fitted failure proportions, or residual plots.

Usage

```
## S3 method for class 'oneshot_em'
plot(x, type = c("fitted", "survival", "residuals"), ...)
```

Arguments

x	an object of class "oneshot_em".
type	plot type: "fitted" (fitted vs observed failure rates), "survival" (fitted survival curves over time), or "residuals".
...	additional graphical parameters.

Value

Invisibly returns the original input object x of class "oneshot_em". This function is called for its side effect of producing graphical diagnostic plots (fitted vs observed failure rates, survival curves, or standardized residuals).

---

predict.oneshot_em *Predict Survival Probability or Mean Time To Failure*

Description

Predicts survival probability $S(t)$ or Mean Time To Failure (MTTF) under specified stress levels.

Usage

```
## S3 method for class 'oneshot_em'
predict(object, newdata = NULL, time = NULL, type = c("survival", "mttf"), ...)
```

Arguments

object	an object of class "oneshot_em".
newdata	data frame containing new stress levels.
time	numeric vector of inspection times for survival prediction.
type	prediction type: "survival" or "mttf".
...	additional arguments (unused).

Value

A numeric vector or matrix of class "numeric" containing predicted survival probabilities (when type = "survival") or Mean Time To Failure (MTTF) values (when type = "mttf") calculated under the specified stress levels and times.

---

print.oneshot_em *Print Method for One-Shot Device EM Fit*

Description

Print Method for One-Shot Device EM Fit

Usage

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

Arguments

x	an object of class "oneshot_em".
digits	number of significant digits to print.
...	additional arguments (unused).

Value

Invisibly returns the original input object x of class "oneshot_em". This function is called primarily for its side effect of printing model fit summary information (distribution, convergence status, estimated coefficients, log-likelihood) to the console.

print.summary.oneshot_em *Print Method for Summary of One-Shot Device EM Fit*

Description

Print Method for Summary of One-Shot Device EM Fit

Usage

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

Arguments

x	an object of class "summary.oneshot_em".
digits	number of significant digits to print.
...	additional arguments (unused).

Value

Invisibly returns the summary object x of class "summary.oneshot_em". This function is called primarily for its side effect of formatting and printing model summary results, coefficient matrices, and information criteria to the console.

summary.oneshot_em *Summary Method for One-Shot Device EM Fit*

Description

Computes summary statistics including standard errors, z-values, p-values, 95% Wald confidence intervals, and model selection criteria.

Usage

```
## S3 method for class 'oneshot_em'
summary(object, level = 0.95, ...)
```

Arguments

object	an object of class "oneshot_em".
level	confidence level for confidence intervals (default: 0.95).
...	additional arguments (unused).

Value

An object of class "summary.oneshot_em", which is a list containing the following components:

coefficients	a matrix of parameter estimates, standard errors, z-values, p-values, and confidence intervals
loglik	maximized log-likelihood value
npar	number of estimated parameters
nobs	total number of tested devices
aic	Akaike Information Criterion
bic	Bayesian Information Criterion
aicc	corrected Akaike Information Criterion
hqic	Hannan-Quinn Information Criterion
iterations	number of EM iterations executed
converged	logical flag indicating convergence status
dist	fitted lifetime distribution name
method	estimation algorithm name ("npm" or "tm")
call_object	original fitted "oneshot_em" object

---

vcov.oneshot_em *Extract Variance-Covariance Matrix*

Description

Extract Variance-Covariance Matrix

Usage

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

Arguments

object	an object of class "oneshot_em".
...	additional arguments (unused).

Value

A symmetric numeric matrix of class "matrix" representing the estimated variance-covariance matrix of the parameter estimates.

Index

∗ **datasets**
electro_explosive, 4

AIC.oneshot_em, 2

BIC.oneshot_em, 3

coef.oneshot_em, 3

electro_explosive, 4
em_control, 5, *7, 8*, *10*

formula, *7, 8*, *10*

logLik.oneshot_em, 6

oneshot_em, 6, *8, 9*
oneshot_fit, 8
oneshot_residuals, 9
oneshot_tm, 9

plot.oneshot_em, 11
predict.oneshot_em, 11
print.oneshot_em, 12
print.summary.oneshot_em, 13

summary.oneshot_em, 13

vcov.oneshot_em, 14