Package ‘Temporal’

March 16, 2020

Title Parametric Time to Event Analysis

Version 0.2.0

Description Performs maximum likelihood based estimation and inference on time to event data, possibly subject to non-informative right censoring. fitParaSurv() provides maximum likelihood estimates of model parameters and distributional characteristics, including the mean, median, variance, and restricted mean. compParaSurv() compares the mean, median, and restricted mean survival experiences of two treatment groups. Candidate distributions include the exponential, gamma, generalized gamma, log-normal, and Weibull.

Depends R (>= 3.5.0)

License GPL-3

Encoding UTF-8

LazyData true

LinkingTo Rcpp, RcppArmadillo

Imports expint, methods, numDeriv, plyr, Rcpp (>= 1.0.3), stats

RoxygenNote 7.0.2

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation yes

Author Zachary McCaw [aut, cre]

Maintainer Zachary McCaw <`zmccaw@alumni.harvard.edu`>

Repository CRAN

Date/Publication 2020-03-16 12:40:05 UTC

R topics documented:

`bootCI` *Bootstrap CIs*

Description

Generates bootstrap confidence intervals for location and RMST estimates.

Usage

```
bootCI(d1, d0, tau = NULL, dist1, dist0, B = 2000, sig)
```

Arguments

`d1`	Target data.frame containing time and status.
`d0`	Reference data.frame containing time and status.
`tau`	Optional truncation times for calculating restricted mean survival time.
`dist1`	String, target distribution.
`dist0`	String, reference distribution.
`B`	Number of resamples.
`sig`	Significance level, for CIs.

Value

List containing data.frames with the bootstrap CIs for the location and RMST estimates.

`checkArm` *Check Arm*

Description

Check whether treatment arm is properly formatted.

Usage

```
checkArm(arm, n)
```

Arguments

`arm`	0/1, treatment arm.
`n`	Integer, sample size.

Value

Logical indication of whether arm was properly formatted.

`checkDist` *Check Distribution*

Description

Check whether the distribution selected is available.

Usage

```
checkDist(dist)
```

Arguments

`dist`	String, distribution name.

Value

Logical indication of whether the distribution was available.

`checkInit` *Check Initialization*

Description

Check whether the initialization is valid.

Usage

```
checkInit(dist, init)
```

Arguments

`dist`	String, distribution name.
`init`	Numeric vector, initialization.

Value

Logical indication of whether the initialization was valid.

`checkStatus` *Status Check*

Description

Function to ensure the status indicator is properly formatted

Usage

```
checkStatus(n, status)
```

Arguments

`n`	Integer, sample size.
`status`	0/1 status indicator.

Value

Logical indicator of whether the status indicator was properly formatted.

`checkTheta` *Check Theta*

Description

Function to check the appropriate number of parameters are supplied for the selected distribution.

Usage

```
checkTheta(dist, theta)
```

Arguments

`dist`	String, distribution.
`theta`	Numeric, parameter vector.

Value

Logical indication of whether the appropriate number of parameters was provided.

`compParaSurv` *Compare Parametric Survival Distribution*

Description

Compares the means and medians of parametric survival distributions fit to each of two treatment arms. Available distributions include: exponential, gamma, generalized gamma, log-normal, and Weibull.

Usage

```
compParaSurv(
  time,
  status = NULL,
  arm,
  tau = NULL,
  dist1 = "weibull",
  dist0 = NULL,
  init1 = NULL,
  init0 = NULL,
  boot = NULL,
  perm = NULL,
  sig = 0.05,
  eps = 1e-06,
  maxit = 10,
  report = F
)
```

Arguments

`time`	Observation time.
`status`	Status indicator, coded as 1 if an event was observed, 0 if censored.
`arm`	Treatment indicator, coded as 1 for the target group, 0 for the reference group.
`tau`	Optional truncation times for calculating RMST.
`dist1`	Distribution to fit for the target group. Selected from among: exp, gamma, gengamma, log-normal, and weibull.
`dist0`	Distribution to fit for the reference group. Same choices as for the target group. If omitted, defaults to the distribution specified for the target group.
`init1`	Initial parameter values for the target group.
`init0`	Initial parameter values for the reference group.
`boot`	If provided, integer number of bootstraps for constructing CIs.
`perm`	If provided, integer number of permutations for calculating p-values.
`sig`	Significance level, for CIs.
`eps`	Tolerance for Newton-Raphson iterations.

`maxit`	Maximum number of NR iterations.
`report`	Report fitting progress?

Details

Status is encoded 0 for censored and 1 for observed. Arm is encoded 0 for reference, 1 for target. Tau is an optional numeric vector of truncation times for calculating restricted mean survival time, which is the area under the survival curve up to the specified truncation point.

Value

An object of class `contrast` containing the following:

Model1 The fitted model for the target group.

Model0 The fitted model for the reference group.

Contrast Contrasts of means and medians.

See Also

- Fitting function for parametric survival distributions `fitParaSurv`

Examples

```
## Not run:
set.seed(100);
# Weibull and Weibull, different means and medians
n = 1e3;
## Target group
d1 = genData(n=n,dist="weibull",theta=c(1,1),p=0.2);
d1$arm = 1;
## Reference group
d0 = genData(n=n,dist="weibull",theta=c(1,2),p=0.2);
d0$arm = 0;
## Overall data set
data = rbind(d1,d0);
## Comparison
comp = compParaSurv(time=data$time,status=data$status,arm=data$arm,tau=0.5,dist1="weibull");

# Gamma and Weibull, same mean, different medians, bootstrap CIs
## Target group
d1 = genData(n=n,dist="gamma",theta=c(2,2),p=0.2);
d1$arm = 1;
## Reference group
d0 = genData(n=n,dist="weibull",theta=c(2,sqrt(pi)/2),p=0.2);
d0$arm = 0;
## Overall data set
data = rbind(d1,d0);
## Comparison
comp = compParaSurv(time=data$time,status=data$status,arm=data$arm,
                    tau=0.5,dist1="gamma",dist0="weibull",boot=2e3);
```

```
# Weibull and Log-normal, different means, same median, permutation p.values
## Target group
d1 = genData(n=n,dist="log-normal",theta=c(0,2),p=0.2);
d1$arm = 1;
## Reference group
d0 = genData(n=n,dist="weibull",theta=c(2,sqrt(log(2))),p=0.2);
d0$arm = 0;
## Overall data set
data = rbind(d1,d0);
## Comparison
comp = compParaSurv(time=data$time,status=data$status,arm=data$arm,
                    tau=0.5,dist1="log-normal",dist0="weibull",perm=2e3);

## End(Not run)
```

`contrast-class` *Contrast of Survival Distributions.*

Description

Defines the object class returned by the comparison function.

Slots

`Dist1` Distribution fit to the target group, string.

`Dist0` Distribution fit to the reference group, string.

`Model1` Fitted model for the target group, fit.

`Model0` Fitted model for the reference group, fit.

`Location` Contrasts of means and medians, data.frame.

`RMST` Contrasts of RMSTs, data.frame.

`defaultParam` *Set Default Parameters*

Description

Function to select default parameter values for each distribution.

Usage

```
defaultParam(dist)
```

Arguments

`dist`	String, distribution name.

Value

Numeric parameter vector

`estDiff` *Difference of Estimates*

Description

Calculate CIs and p-value for the difference of estimated parameters

Usage

```
estDiff(t1, s1, t0, s0, sig = 0.05)
```

Arguments

`t1`	Treatment estimate
`s1`	Treatment standard error
`t0`	Reference estimate
`s0`	Reference standard error
`sig`	Significance level

Value

Data.frame containing estimated difference, its standard error, lower and upper confidence bounds, and a p-value assessing the null hypothesis of no difference.

`estRatio` *Ratio of Estimates*

Description

Calculate CIs and p-value for the ratio of estimated parameters

Usage

```
estRatio(t1, s1, t0, s0, sig = 0.05)
```

Arguments

`t1`	Treatment estimate
`s1`	Treatment standard error
`t0`	Reference estimate
`s0`	Reference standard error
`sig`	Significance level

Value

Data.frame containing estimated ratio, its standard error, lower and upper confidence bounds, and a p-value assessing the null hypothesis that the ratio is unity.

`fit-class` *Fitted Survival Distribution*

Description

Defines the object class returned by fitting functions.

Slots

`Distribution` Fitted distribution, string.

`Parameters` Parameters, data.frame.

`Information` Information components, matrix.

`Outcome` Properties of the fitted distribution, data.frame.

`RMST` Estimated restricted mean survival times, data.frame

`S` Fitted survival function, function.

`fit.Exp` *Exponential Distribution Parameter Estimation*

Description

Estimates parameters for exponential event times subject to non-informative right censoring. The exponential distribution is parameterized in terms of the rate λ:

$$f(t) = \lambda e^{-\lambda t}, t > 0$$

Usage

```
fit.Exp(time, status, sig = 0.05, tau = NULL)
```

Arguments

`time`	Numeric observation times.
`status`	Status indicator, coded as 1 if an event was observed, 0 if censored.
`sig`	Significance level, for CIs.
`tau`	Optional truncation times for calculating RMSTs.

Value

An object of class `fit` containing the following:

Parameters The estimated model parameters.

Information The observed information matrix.

Outcome The fitted mean, median, and variance of the time to event distribution.

RMST The estimated RMSTs, if tau was specified.

See Also

- Fitting function for parametric survival distributions `fitParaSurv`

Examples

```
# Generate exponential data with 20% censoring
data = genData(n=1e3,dist="exp",theta=c(2),p=0.2);
# Estimate
fit = fitParaSurv(time=data$time,status=data$status,dist="exp");
```

`fit.Gamma` *Gamma Distribution Parameter Estimation*

Description

Estimates parameters for gamma event times subject to non-informative right censoring. The gamma distribution is parameterized in terms of the shape α and rate λ:

$$f(t) = \frac{\lambda}{\Gamma(\alpha)}(\lambda t)^{\alpha-1}e^{-\lambda t}, t > 0$$

Usage

```
fit.Gamma(
  time,
  status,
  sig = 0.05,
  tau = NULL,
  init = NULL,
  eps = 1e-06,
  maxit = 10,
  report = FALSE
)
```

Arguments

`time`	Numeric observation times.
`status`	Status indicator, coded as 1 if an event was observed, 0 if censored.
`sig`	Significance level, for CIs.
`tau`	Optional truncation times for calculating RMSTs.
`init`	Vector of initial values for shape α and rate λ parameters, respectively.
`eps`	Tolerance for Newton-Raphson iterations.
`maxit`	Maximum number of NR iterations.
`report`	Report fitting progress?

Value

An object of class `fit` containing the following:

Parameters The estimated shape α and rate λ.

Information The observed information matrix.

Outcome The fitted mean, median, and variance.

RMST The estimated RMSTs, if tau was specified.

See Also

- Fitting function for parametric survival distributions `fitParaSurv`

Examples

```
# Generate Gamma data with 20% censoring.
data = genData(n=1e3,dist="gamma",theta=c(2,2),p=0.2);
# Estimate
fit = fitParaSurv(time=data$time,status=data$status,dist="gamma");
```

`fit.Gamma.Complete` *Gamma Distribution Parameter Estimation without Censoring*

Description

Estimates parameters for gamma event times not subject to censoring.

Usage

```
fit.Gamma.Complete(time, eps = 1e-06)
```

Arguments

`time`	Observation times.
`eps`	Tolerance for Newton-Raphson iterations.

`fit.GenGamma` *Generalized Gamma Distribution Parameter Estimation*

Description

Estimates parameters for generalized gamma event times subject to non-informative right censoring. The gamma distribution is parameterized in terms of the shape parameters (α, β), and the rate λ:

$$f(t) = \frac{\beta\lambda}{\Gamma(\alpha)}(\lambda t)^{\alpha\beta - 1}e^{-(\lambda t)^{\beta}}, t > 0$$

Usage

```
fit.GenGamma(
  time,
  status,
  sig = 0.05,
  tau = NULL,
  init,
  bL = 0.1,
  bU = 10,
  eps = 1e-06,
  maxit = 10,
  report = FALSE
)
```

Arguments

`time`	Numeric observation times.
`status`	Status indicator, coded as 1 if an event was observed, 0 if censored.
`sig`	Significance level, for CIs.
`tau`	Optional truncation times for calculating RMSTs.
`init`	Vector of initial values for the shape α, β, and rate λ parameters, respectively.
`bL`	Lower limit on possible values for beta.
`bU`	Upper limit on possible values for beta.
`eps`	Tolerance for Newton-Raphson iterations.
`maxit`	Maximum number of NR iterations.
`report`	Report fitting progress?

Value

An object of class `fit` containing the following:

Parameters The estimated shape (α, β) and rate λ parameters.

Information The observed information matrix.

Outcome The fitted mean, median, and variance.

RMST The estimated RMSTs, if tau was specified.

See Also

- Fitting function for parametric survival distributions `fitParaSurv`

Examples

```
set.seed(103);
# Generalized Gamma data with 20% censoring.
data = genData(n=1e4,dist="gen-gamma",theta=c(2,2,2),p=0.2);
# Estimate
fit = fitParaSurv(time=data$time,status=data$status,dist="gen-gamma",report=TRUE);
```

`fit.GenGamma.Complete` *Initialization for Generalized Gamma*

Description

Initializes the parameters for the generalized gamma distribution via maximum likelihood.

Usage

```
fit.GenGamma.Complete(time, bL, bU)
```

Arguments

`time`	Observed event times.
`bL`	Lower limit on possible values for beta.
`bU`	Upper limit on possible values for beta.

Value

3x1 numeric vector of estimated parameters, α, β, λ.

`fit.LogNormal` *Log-Normal Distribution Parameter Estimation*

Description

Estimates parameters for log-normal event times subject to non-informative right censoring. The log-normal distribution is parameterized in terms of the location μ and scale σ:

$$f(t) = \phi\left(\frac{\ln t - \mu}{\sigma}\right)\frac{1}{t\sigma}, t > 0$$

Usage

```
fit.LogNormal(
  time,
  status,
  sig = 0.05,
  tau = NULL,
  init = NULL,
  eps = 1e-06,
  maxit = 10,
  report = F
)
```

Arguments

`time`	Numeric observation times.
`status`	Status indicator, coded as 1 if an event was observed, 0 if censored.
`sig`	Significance level, for CIs.
`tau`	Optional truncation times for calculating RMSTs.
`init`	Vecotor of initial values for the location μ and scale σ.
`eps`	Tolerance for Newton-Raphson iterations.
`maxit`	Maximum number of NR iterations.
`report`	Report fitting progress?

Value

An object of class `fit` containing the following:

Parameters The estimated location μ and scale σ.

Information The observed information matrix.

Outcome The fitted mean, median, and variance.

RMST The estimated RMSTs, if tau was specified.

See Also

- Fitting function for parametric survival distributions `fitParaSurv`

Examples

```
# Generate Log-Normal data with 20% censoring.
data = genData(n=1e3,dist="log-normal",theta=c(0,2),p=0.2);
# Estimate
fit = fitParaSurv(time=data$time,status=data$status,dist="log-normal");
```

`fit.Weibull` *Weibull Distribution Parameter Estimation*

Description

Estimates parameters for Weibull event times subject to non-informative right censoring. The Weibull distribution is parameterized in terms of the shape α and rate λ:

$$f(t) = \alpha\lambda^{\alpha}t^{\alpha-1}e^{-(\lambda t)^{\alpha}}, t > 0$$

Usage

```
fit.Weibull(time, status, sig = 0.05, tau = NULL, init = NULL)
```

Arguments

`time`	Numierc observation times.
`status`	Status indicator, coded as 1 if an event was observed, 0 if censored.
`sig`	Significance level, for CIs.
`tau`	Optional truncation times for calculating RMSTs
`init`	Numeric vector containing the initial value for α.

Value

An object of class `fit` containing the following:

Parameters The estimated shape α and rate λ.

Information The observed information matrix.

Outcome The fitted mean, median, and variance.

RMST The estimated RMSTs, if tau was specified.

See Also

- Fitting function for parametric survival distributions `fitParaSurv`

Examples

```
# Generate Weibull data with 20% censoring
data = genData(n=1e3,dist="weibull",theta=c(2,2),p=0.2);
# Estimate
fit = fitParaSurv(time=data$time,status=data$status,dist="weibull");
```

---

fitParaSurv *Fit Parametric Survival Distribution*

Description

Estimates parametric survival distributions using event times subject to non-informative right censoring. Available distributions include: exponential, gamma, generalized gamma, log-normal, and Weibull.

Usage

```
fitParaSurv(
  time,
  status = NULL,
  dist = "weibull",
  tau = NULL,
  sig = 0.05,
  init = NULL,
  bL = 0.1,
  bU = 10,
  eps = 1e-06,
  maxit = 10,
  report = F
)
```

Arguments

`time`	Numeric observation times.
`status`	Status indicator, coded as 1 if observed, 0 if censored.
`dist`	String, distribution to fit, selected from among: exp, gamma, gen-gamma log-normal, and weibull.
`tau`	Optional truncation time for calculating RMSTs.
`sig`	Significance level, for CIs.
`init`	Numeric vector of initial parameters. See individual distributions for parameter order.
`bL`	If dist="gen-gamma", lower limit on possible values for beta.
`bU`	If dist="gen-gamma", upper limit on possible values for beta.
`eps`	Tolerance for Newton-Raphson iterations.
`maxit`	Maximum number of NR iterations.
`report`	Report fitting progress?

Value

An object of class `fit` containing the following:

Parameters The estimated shape and rate parameters.

Information The observed information matrix.

Outcome The fitted mean, median, and variance.

RMST The estimated RMSTs, if tau was specified.

See Also

- Between group comparison of survival experience `compParaSurv`
- Exponential distribution `fit.Exp`
- Gamma distribution `fit.Gamma`
- Generalized gamma distribution `fit.GenGamma`
- Log-normal distribution `fit.LogNormal`
- Weibull distribution `fit.Weibull`

Examples

```
# Generate Gamma data with 20% censoring
D = genData(n=1e3,dist="gamma",theta=c(2,2),p=0.2);
# Fit gamma distribution
M = fitParaSurv(time=D$time,status=D$status,dist="gamma");

# Generate Weibull data with 10% censoring
D = genData(n=1e3,dist="weibull",theta=c(2,2),p=0.1);
# Fit weibull distribution, calculate RMST at tau=0.5
M = fitParaSurv(time=D$time,status=D$status,dist="weibull",tau=0.5);
```

`genData` *Data Generation with Censoring*

Description

Generates data from survival distributions as parameterized in this package, with optional non-informative random right censoring.

Usage

```
genData(n, dist = "exp", theta = NULL, p = 0)
```

Arguments

`n`	Integer sample size.
`dist`	String, distribution name selected from among: "exp","gamma","gen-gamma","log-normal","weibull".
`theta`	Numeric parameter vector. Elements will vary according to the distribution.
`p`	Expected censoring proportion.

Details

The parameter vector theta should contain the following elements, in order, depending on the distribution:

Exponential Rate λ.

Gamma Shape α, rate λ.

Generalized Gamma Shape 1 α, shape 2 β, rate λ.

Log-Normal Locaion μ, scale σ.

Weibull Shape α, rate λ.

Value

data.frame including the observation times and status indicators.

Examples

```
# Gamma event times with shape 2 and rate 2
# Expected censoring proportion of 20%
data = genData(n=1e3,dist="gamma",theta=c(2,2),p=0.20);

# Generalized gamma event times with shapes (2,3) and rate 1
# Expected censoring proportion of 15%
data = genData(n=1e3,dist="gen-gamma",theta=c(2,3,1),p=0.15);

# Log-normal event times with location 0 and rate 1
# Expected censoring proportion of 10%
data = genData(n=1e3,dist="log-normal",theta=c(0,1),p=0.10);

# Weibull event times with shape 2 and rate 2
# Expected censoring proportion of 5%
data = genData(n=1e3,dist="weibull",theta=c(2,2),p=0.05);
```

`matInv` *Matrix Inverse*

Description

Calcualtes A^{-1}.

Usage

```
matInv(A)
```

Arguments

`A`	Numeric matrix.

Value

Numeric matrix.

`matIP` *Matrix Inner Product*

Description

Calculates the product $A'B$.

Usage

```
matIP(A, B)
```

Arguments

`A`	Numeric matrix.
`B`	Numeric matrix.

Value

Numeric matrix.

`matOP` *Matrix Outer Product*

Description

Calculates the outer product AB'.

Usage

```
matOP(A, B)
```

Arguments

`A`	Numeric matrix.
`B`	Numeric matrix.

Value

Numeric matrix.

`matQF` *Quadratic Form*

Description

Calculates the quadratic form $X'AX$.

Usage

```
matQF(X, A)
```

Arguments

`X`	Numeric matrix.
`A`	Numeric matrix.

Value

Numeric matrix.

`MMP` *Matrix Matrix Product*

Description

Calculates the product AB.

Usage

```
MMP(A, B)
```

Arguments

`A`	Numeric matrix.
`B`	Numeric matrix.

Value

Numeric matrix.

`paraRMST` *Restricted Mean Survival Time*

Description

Calculates the tau-year RMST for a fitted parametric model.

Usage

```
paraRMST(fit, sig = 0.05, tau)
```

Arguments

`fit`	Fitted parametric survival distribution.
`sig`	Significance level, for CIs.
`tau`	Numeric vector of truncation times.

Value

Data.table containing the estimated RMST at each truncation time.

Examples

```
# Generate Weibull data with 20% censoring.
data = genData(n=1e3,dist="weibull",theta=c(2,0.5),p=0.2);
fit = fitParaSurv(time=data$time,status=data$status,dist="weibull");
rmst = paraRMST(fit=fit,tau=c(0.5,1.0,1.5,2.0));

# Generate Gamma data with 10% censoring.
data = genData(n=1e3,dist="gamma",theta=c(2,2),p=0.10);
fit = fitParaSurv(time=data$time,status=data$status,dist="gamma");
rmst = paraRMST(fit=fit,tau=c(0.5,1.0,1.5,2.0));
```

`permP` *Permutation P Value*

Description

Calculates permutation p-values for location and RMST estimates.

Usage

```
permP(d1, d0, tau, dist1, dist0, B = 1999)
```

Arguments

`d1`	Target data.frame containing time and status.
`d0`	Reference data.frame containing time and status.
`tau`	Optional truncation times for calculating restricted mean survival time.
`dist1`	String, target distribution.
`dist0`	String, reference distribution.
`B`	Number of resamples.

Value

List containing data.frames with the bootstrap CIs for the location and RMST estimates.

`print.contrast` *Print Method for a Contrast of Survival Distributions.*

Description

Print method for an object of class `contrast`.

Usage

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

Arguments

`x`	A `contrast` object.
`...`	Unused.

`print.fit` *Print Method for Fitted Survival Distributions*

Description

Print method for objects of class `fit`.

Usage

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

Arguments

`x`	An object of class `fit`.
`...`	Unused.

`qWeibull` *Quantile Function for the Weibull Distribution*

Description

Quantile function for the Weibull distribution. See `fit.Weibull` for the parameterization.

Usage

```
qWeibull(p, a = 1, l = 1)
```

Arguments

`p`	Probability.
`a`	Shape.
`l`	Rate.

Value

Scalar quantile.

`resample` *Resample Data*

Description

Generates a resampled data.frame with replacement.

Usage

```
resample(data)
```

Arguments

`data`	Input data.farme

Value

Data.frame resampled from the input.

`rGamma` *Simulation from the Gamma Distribution*

Description

Generates gamma event times with shape parameter α and rate parameter λ. See `fit.Gamma` for the parameterization. If a censoring proportion p is provided, the event times are subject to non-informative random right censoring.

Usage

```
rGamma(n, a = 1, l = 1, p = 0)
```

Arguments

`n`	Sample size.
`a`	Shape.
`l`	Rate.
`p`	Expected censoring proportion.

Value

Data.frame including the observation times and status indicators.

`rGenGamma` *Simulation from the Generalized Gamma Distribution*

Description

Generates generalized gamma event times with shape parameters (α, β), and rate parameter λ. See `fit.GenGamma` for the parameterization. If a censoring proportion p is provided, the event times are subject to non-informative random right censoring.

Usage

```
rGenGamma(n, a = 1, b = 1, l = 1, p = 0)
```

Arguments

`n`	Sample size.
`a`	First shape parameter, α.
`b`	Second shape parameter, β. For the standard gamma distribution, set $\beta = 1$.
`l`	Rate.
`p`	Expected censoring proportion.

Value

DSata.frame including the observation times and status indicators.

`rLogNormal` *Simulation from the Log-Normal Distribution*

Description

Generates log-normal event times with location parameter μ and scale parameter σ. See `fit.LogNormal` for the parameterization. If a censoring proportion p is provided, the event times are subject to non-informative random right censoring.

Usage

```
rLogNormal(n, m = 0, s = 1, p = 0)
```

Arguments

`n`	Sample size.
`m`	Location.
`s`	Scale.
`p`	Expected censoring proportion.

Value

Data.frame including the observation times and status indicators.

`rWeibull` *Simulation from the Weibull Distribution*

Description

Generates Weibull event times with shape parameter α and rate parameter λ. See `fit.Weibull` for the parameterization. If a censoring proportion p is provided, the deviates are subject to non-informative random right censoring.

Usage

```
rWeibull(n, a = 1, l = 1, p = 0)
```

Arguments

`n`	Sample size.
`a`	Shape.
`l`	Rate.
`p`	Expected censoring proportion.

Value

Data.frame including the observation times and status indicators.

`show,contrast-method` *Show Method for a Contrast of Survival Distributions.*

Description

Show Method for a Contrast of Survival Distributions.

Usage

```
## S4 method for signature 'contrast'
show(object)
```

Arguments

`object`	An object of class `contrast`.

`show,fit-method` *Show Method for Fitted Survival Distributions*

Description

Show Method for Fitted Survival Distributions

Usage

```
## S4 method for signature 'fit'
show(object)
```

Arguments

`object`	An object of class `fit`.

`survFunc` *Survival Functions*

Description

Constructs the survival function for a parameter distribution.

Usage

```
survFunc(dist, theta)
```

Arguments

`dist`	String, distribution name.
`theta`	Numeric parameter vector.

Details

The parameter vector theta should contain the following elements, in order, according to the distribution:

Exponential Rate λ.

Gamma Shape α, rate λ.

Generalized Gamma Shape 1 α, shape 2 β, rate λ.

Log-Normal Locaion μ, scale σ.

Weibull Shape α, rate λ.

Value

Survival function.

`survLogLik` *Log Likelihood*

Description

Evaluates the log-likelihood for a parametric survival distribution.

Usage

```
survLogLik(time, status = NULL, dist, theta, log.scale = FALSE)
```

Arguments

`time`	Numeric observation times.
`status`	Status indicator, coded as 1 if an event was observed, 0 if censored.
`dist`	Distribution, from among: "exp","gamma","gen-gamma","log-normal","weibull".
`theta`	Parameters, which will vary according to the distribution.
`log.scale`	Are strictly positive parameters on log-scale?

Details

The parameter vector theta should contain the following elements, in order, depending on the distribution:

Exponential Rate λ.

Gamma Shape α, rate λ.

Generalized Gamma Shape 1 α, shape 2 β, rate λ.

Log-Normal Locaion μ, scale σ.

Weibull Shape α, rate λ.

Value

Scalar value of the log likelihood.

Examples

```
# Generate Weibull data with 20% censoring.
data = genData(n=1e3,dist="weibull",theta=c(2,2),p=0.2);
# Log likelihood
ll = survLogLik(time=data$time,status=data$status,dist="weibull",theta=c(2,2));

# Generate Gamam data with 10% censoring.
data = genData(n=1e3,dist="gamma",theta=c(2,2),p=0.1);
# Log likelihood
ll = survLogLik(time=data$time,status=data$status,dist="gamma",theta=c(2,2));
```

`Temporal` *Temporal: Utilities for Parametric Survival Analysis*

Description

This package performs estimation and inference on parametric survival curves. See `fitParaSurv` for maximum likelihood estimation of model parameters. See `compParaSurv` for contrasting the survival experience of two treatment arms.

Author(s)

Zachary R. McCaw

Index