Package: WeibullR (via r-universe)

October 28, 2026

Version 1.2.4

Date 2025-10-8

Title Weibull Analysis for Reliability Engineering

Description Life data analysis in the graphical tradition of Waloddi Weibull. Methods derived from Robert B. Abernethy (2008, ISBN 0-965306-3-2), Wayne Nelson (1982, ISBN: 9780471094586), William Q. Meeker and Lois A. Escobar (1998, ISBN: 1-471-14328-6), John I. McCool, (2012, ISBN: 9781118217986).

License GPL (>= 3)

Copyright copyright (c) OpenReliability.org 2011-2019

URL http://www.openreliability.org/weibull-r-weibull-analysis-on-r/

LazyLoad yes

Imports methods, Rcpp (>= 0.11.1)

LinkingTo Rcpp, RcppArmadillo

Author David Silkworth [aut], Jurgen Symynck [aut], Jacob Ormerod [cre], OpenReliability.org [cph]

Maintainer Jacob Ormerod <jake@openreliability.org>

NeedsCompilation yes

Repository https://cran.r-universe.dev

Date/Publication 2025-10-09 05:10:26 UTC

RemoteUrl https://github.com/cran/WeibullR

RemoteRef HEAD

RemoteSha 87430c8b236a93b289b896e374959b2a96efb5aa

Contents

WeibullR-package 2
AbPval 3
BBB 4
contour.wblr 6

FMbounds . . . 7
getCCC2 . . . 8
getPercentilePlottingPositions . . . 9
getPPP . . . 10
hrbu . . . 11
LLln . . . 12
LLw . . . 13
LRbounds . . . 14
lslr . . . 15
MLEcontour . . . 16
mlefit . . . 18
mleframe . . . 19
MLEln2p . . . 21
MLEln3p . . . 22
MLEw2p . . . 23
MLEw3p . . . 24
MRRln2p . . . 25
MRRln3p . . . 26
MRRw2p . . . 27
MRRw3p . . . 28
options.wblr . . . 29
p2y . . . 32
pivotal.rr . . . 33
plot.wblr . . . 35
plot_contour . . . 37
rba . . . 38
wblr . . . 39
wblr.conf . . . 41
wblr.fit . . . 44
wblrLoglike . . . 45
weibayes . . . 46
weibayes.mle . . . 47
xbounds . . . 48
xfit . . . 49

Index **50**

`WeibullR-package` *Weibull-R : Weibull Analysis on R*

Description

Life data analysis in the graphical tradition of Waloddi Weibull

Details

The WeibullR package provides a flexible data entry capability with three levels of usage.

***Quick Fit* Functions** Functions with intuitive names `MLEw2p` through `MRRln3p` for preparing simple fits, bounds, and displays using default options. Only data sets with exact failure times and/or suspensions are processed. The quick fit functions return a simple named vector of the fitted parameters with appropriate goodness of fit measure(s). Optional preparation of appropriate interval bounds (at 90% confidence), or a display of fit and bounds are controlled by two final arguments taking logical entry, Such that a function call like `MLEw2p(input_data, T, T)` will generate a plot with the fitted data and confidence interval bounds. When the first logical for bounds is set to TRUE, the returned object will be a list with the fitted parameter vector first and dataframe of bounds values second.

wblr Object Model Construction of a wblr object is initiated by providing a data set through function `wblr`. Modification of the object with the progressive addition of fits and confidence interval bounds is made via functions `wblr.fit` and `wblr.conf`. Fine control over many aspects of fit, confidence, and display are made possible using a flexible options mechanism. Display for single object models is via S3 methods `plot` or `contour`, while multiple objects *(provided as a list)* can be displayed on a single plot using `plot.wblr`, `plot_contour`, or `contour.wblr`.

Backend Functions Access to backend functions providing all the functionality of the upper levels of usage are provided as exported functions. These functions may provide advanced users with resources to expand analysis further than has been implemented in the WeibullR package.

Data entry is made through the Quick Fit functions, `wblr`, or on the backend through `getPPP` for rank regression, `mleframe` for mle processing. In all cases the primary argument `x` can be a vector of exact time failures or a dataframe with `time`, and `event` columns as a minimum. An additional column `qty` may optionally be used to record duplicated data. If the dataframe entry is not used (in favor of an exact time failure vector), a second argument, `s`, can be used to enter a vector of last observed success times for right censored data (suspensions).
Beyond the entry of the first two data types, interval data (including discoveries with last known success time=0) are entered via argument `interval` as a dataframe with columns `left`, and `right` as a minimum. As with the primary argument dataframe entry, an additional column `qty` may optionally be used to record duplicated interval data. Such interval data entry is not supported with the Quick Fit functions.

Author(s)

David J. Silkworth <djsilk@openreliability.org> Jurgen Symynck <email withheld>

`AbPval` *Determination of the percentile of r and r-squared, by correlation. Here designated "Abernethy's P-value"*

Description

The percentile of r and r-squared (prr) generated by pivotal Monte Carlo analysis has been promoted as a goodness of fit measure by Robert B. Abernethy.

Usage

```
AbPval(F,R2,model="weibull")
```

Arguments

F	The quantity of complete failure data points under consideration.
R2	The square of the correlation coefficient derived from residuals of the linear model.
model	A string defining the distribution under consideration. Only entry of "lnorm", or "lognormal" will alter the default of "weibull".

Details

The value returned is derived from a correlation developed from previously run pivotal analysis with 10^8 random samples. Only the prr derived from 2 parameter models is judged to have usefullness in comparitive analysis. For validity of a 3rd parameter optimization on a given model over its 2 parameter fit, only the Likelihood Ratio Test should be considered.

Value

Returns a vector containing the P-value and the square of CCC (for comparison with R squared).

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

Wes Fulton, (2005) "Improved Goodness of Fit: P-value of the Correlation Coefficient"

Chi-Chao Lui, (1997) "A Comparison Between The Weibull And Lognormal Models Used To Analyse Reliability Data" (dissertation from University of Nottingham)

Examples

```
AbernethyPvalue<-AbPval(50, 0.996, "lnorm")
```

BBB *Beta Binomial Bounds*

Description

Generates confidence inference bounds by means of the beta binomial distribution applied to data ranks.

Usage

```
BBB(xdata, xfit,n=NULL, CI=.90, unrel=NULL, type="horizontal", nknots=NULL)
```

Arguments

xdata	A list containing either dpoints and/or dlines such as available from a wblr object.
xfit	A list containing dist and fit elements. A suitable object can be extracted from a wblr object containing the resuult of at least one modifying wblr.fit call. Function xfit has been provided to return such a list.
n	The total number of data entries fail, suspension, discovery and interval. This value is only required to be provided if not available from either xdata (as sum(xdatalrq_frameqty) or xfit (as attr(xfit$fit)$n), however, an entry for this argument will take precidence over the other argument sources.
CI	A scalar for the double-sided confidence interval of interest. Default = 0.9, for 90
unrel	An optional vector of unreliability values to be used as the descriptive percentiles at which the bound quantiles will be calculated.
type	A string indicating the adjustment direction(s) for application of the beta binomial. Since both "horizontal" and "vertical" directions on the rank positions of failure data are considered valid, a smoothed spline can be created by appling both directions using "valid". An extrapolation of the smoothed spline is provided by using "extrapolated".
nknots	An optional number of knots to be used to form the smooth spline in the event that disjoints are formed at the junction of horizontal and vertical bounds. Usually, applying 75 to 80 result in a pleasing smoothness.

Details

This non-parametric approach calculates confidence bounds for quantiles and/or failure probabilities applying the CFD of the beta distribution at ranks and reverse ranks of the failure data including intervals. Although a result is provided for 3-parameter models, it does not reflect uncertainty in the third, translation parameter.

Value

Returns a dataframe holding values of Prob, Lower, Fit, and Upper values for the bounds.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Tim-Gunnar Hensel, (2017) "weibulltools", A package on CRAN.

Examples

```
set.seed(4321)
obj<-wblr.fit(wblr(rweibull(10,1,1)))
beta_binomial_bounds<-BBB(obj$data,xfit(obj))
```

`contour.wblr` *S3 Likelihood Ratio Contour Maps From* `wblr` *Objects*

Description

This function adds the `.wblr` method to `contour` from the **graphics** package.

Usage

```
## S3 method for class 'wblr'
contour(x, ...)
```

Arguments

`x`	Object of class `"wblr"` or a list of `wblr` objects.
`...`	Entry for a limited set of graphic parameters (col, lty, lwd, xlim, ylim, main and sub)

Details

This function provides S3 object functionality for plotting a likelihood ratio contour map for any single wblr object with just the contour function.

Contour parameters are drawn from a contour existing in the object(s) passed in, or from the base object (sometimes just defaults) if no contour exists in the object.

In order to plot contour maps from multiple objects onto a single canvas it is necessary to call contour.wblr specifically with a list of wblr objects as primary argument.

Unlike the `contour` this map is not prepared from a matrix of z-values. Rather the specific x,y points for each CL level (Z-value) are provided for plotting lines connecting the points.

Value

The contour.wblr function itself returns no value, however, the full output of contour points and parameters from the internally called WeibullR::plot_contour function is displayed.

Examples

```
set.seed(1234)
da2 <- wblr.conf(wblr.fit(wblr(runif(5,100,1000),col="red")))
da3 <- wblr.conf(wblr.fit(wblr(rweibull(5,3,1000),col="green4")))

## Not run:
contour.wblr(list(da2,da3))

## End(Not run)
```

FMbounds *Fisher Matrix bounds*

Description

Generates the asymptotic bounds by means of the information matrix.

Usage

```
FMbounds(x, dist="weibull", CI=.90, unrel=NULL, debias="none", show=FALSE)
```

Arguments

x	A dataframe such as generated by mleframe with column names 'left', 'right' and optionally 'qty'. Exact failure data (occurrences) have same time entered in both 'left' and 'right' columns. Suspension data has last known life[time] entered in 'left' column and -1 entered in 'right' column. The left(early) interval bound for left-censored data must be entered as zero. (NA is not accepted).
dist	A string defining a distribution to be fit. Implemented distributions are "weibull" (default) and "lognormal". (Only 2-parameter models are accepted.)
CI	A scalar for the double-sided confidence interval of interest. Default = 0.9, for 90
unrel	An optional vector of unreliability values to be used as the descriptive quantiles at which the bounds will be calculated.
debias	A string argument indicating the adjustment to be applied to the shape or standard deviation parameter of the fitted data.
show	A logical determining whether a crude graphic of the bounds shall be displayed.

Details

An "observed" information matrix is derived numerically as the hessian by means of optimHess. From the variance-covariance matrix (inverse of the hessian) variation of life(-time) at given quantiles is determined.

Application of a bias adjustment alters the calculation of the hessian, thus the bounds are based on a "modified" Fisher Matrix.

Value

Returns a dataframe holding values of percentiles, lower, datum, and upper values of the bound.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Examples

```
set.seed(4321)
data<-rlnorm(30,2,1.2)
asymptotic_bounds<-FMbounds(mleframe(data[7:30],data[1:6]), dist="lognormal")
```

`getCCC2` *Determination of the square of the "Critical Correlation Coefficient" (CCC2).*

Description

Abernethy has promoted the 10th percentile of Correlation Coefficients generated by pivotal Monte Carlo analysis as a critical measure by which a fit should be designated suitable for further analysis. According to his practice, the difference between the square of the Correlation Coefficient and the CCC2 (R^2 - CCC^2) is used to make comparitive judgments between weibull and lognormal fitting on the same data.

Usage

```
getCCC2(F, model="weibull")
```

Arguments

`F`	The quantity of complete failure data points under consideration.
`model`	A string defining the distribution under consideration. Only a value of "lnorm" will be treated any differently from default of "weibull".

Details

The value returned is derived from a correlation developed from previously run pivotal analysis with 10^8 random samples. Project "Abernethy Reliability Methods" has judged that only the CCC^2 derived from 2 parameter models to have usefullness in such analysis. This is seen from the "Detect Power" presentations in Appendix D of "The New Weibull Handbook, Fifth Edition". For validity of a 3rd parameter optimization on a given model over its 2 parameter fit, only the Likelihood Ratio Test will be applied. This validity check requires an LRT-P greater than 50

Value

Returns a single valued vector for the square of CCC (for comparison with R squared).

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

Wes Fulton, (2005) "Improved Goodness of Fit: P-value of the Correlation Coefficient"

Chi-Chao Lui, (1997) "A Comparison Between The Weibull And Lognormal Models Used To Analyse Reliability Data" (dissertation from University of Nottingham)

Examples

```
thisCCC2<-getCCC2(50, "lnorm")
```

`getPercentilePlottingPositions` *Determination of percentile plotting positions for linear regression with many optional methods*

Description

Determination of plotting positions to be used for linear regression of life data analysis is a highly debated topic. This function implements a wide assortment of options for the Weibull-R project.

Usage

```
getPercentilePlottingPositions(x, s=NULL, interval=NULL, ppos="beta",
 aranks="Johnson", ties="none")
```

Arguments

`x`	Either a dataframe containing at least `$time` and `$event` columns and optionally a `$qty` column, or a vector of class `"numeric"` or `"integer"` with (life-)time observations.
`s`	An optional vector of suspension data.
`interval`	reserved argument for interval censored data - NOT YET IMPLEMENTED.
`ppos`	A string defining a plotting position method. Implemented options include "beta" the incomplete beta function (as default), "Benard", "mean" also known as Herd-Johnson, "Hazen" or modified Kaplan-Meier, "Kaplan-Meier" with modification for final complete failure, and "Blom.
`aranks`	A string defining the method for establishing adjusted ranks when suspension data (right censored) are present. Implemented options include "Johnson" (as default) and "KMestimator"
`ties`	A string defining a method of eliminating ties, or duplicate time valued data, from plotting. Implemented options include "highest" (used as 'Inspection Option #1' by Abernethy), "mean", "lowest", and "sequential".

Value

Returns a dataframe with the failure data (as potentially reduced by ties argument), the probability plotting positions, and adjusted ranks.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

Leonard C. Johnson (1964) "The Statistical Treatment of Fatigue Experiments"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
median_percentile_ranks<-getPPP(failures,suspensions)[,2]
```

`getPPP` *Alias for* `getPercentilePlottingPositions`*, sets data into the format required by lslr.*

Description

Determination of plotting positions to be used for linear regression of life data analysis is a highly debated topic. This function implements a wide assortment of options for the Weibull-R project.

Usage

```
getPPP(x, s=NULL, interval=NULL, ppos="beta", aranks="Johnson", ties="none")
```

Arguments

`x`	Either a dataframe containing at least `$time` and `$event` columns and optionally a `$qty` column, or a vector of class `"numeric"` or `"integer"` with (life-)time observations.
`s`	An optional vector of suspension data.
`interval`	reserved argument for interval censored data - NOT YET IMPLEMENTED.
`ppos`	A string defining a plotting position method. Implemented options include "beta" the incomplete beta function (as default), "Benard", "mean" also known as Herd-Johnson, "Hazen" or modified Kaplan-Meier, "Kaplan-Meier" with modification for final complete failure, and "Blom.
`aranks`	A string defining the method for establishing adjusted ranks when suspension data (right censored) are present. Implemented options include "Johnson" (as default) and "KMestimator"
`ties`	A string defining a method of eliminating ties, or duplicate time valued data, from plotting. Implemented options include "highest" (used as 'Inspection Option #1' by Abernethy), "mean", "lowest", and "sequential".

Value

Returns a dataframe with the failure data (as potentially reduced by ties argument), the probability plotting positions, and adjusted ranks.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition" Leonard C. Johnson (1964) "The Statistical Treatment of Fatigue Experiments"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
median_percentile_ranks<-getPPP(failures, suspensions)[,2]
```

---

hrbu *Hirose and Ross beta unbias factors for Weibull MLE*

Description

hrbu generates the reduction factor based on the mean bias of the weibull MLE beta parameter(roughly C4^6) for complete failure samples, modestly increased correction as number of suspensions increases.

Usage

```
hrbu(Qx, Qs=NULL)
```

Arguments

Qx	The quantity of actual failures
Qs	An optional quantity of suspensions

Details

This, as many references, discuss the bias reduction in terms of mean.

Value

A factor to be multiplied to the MLE Beta account for known bias.

References

Hirose, H. (1999) "Bias Correction for the Maximum Likelihood Estimation in Two-parameter Weibull Distribution" IEEE Transactions on Dielectrics and Electrical Insulation, Vol. 6, No.1 Ross, R. (1996) "Bias and Standard Deviation Due to Weibull Parameter Estimation for Small Data Sets" IEEE Transactions on Dielectrics and Electrical Insulation, Vol. 3, No.1

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
MLEfit<-mlefit(mleframe(failures,suspensions))
MLE_Unbiased<-c(MLEfit[1],MLEfit[2]*hrbu(length(failures),length(suspensions)))
```

LLln *Log Likelihood for log-normal fitted data, failures and suspensions only*

Description

Determination of log-likelihood values for the basic distributions covered by R should be ubiquitous, but an implementation is provided here to enable likelihood ratio testing particularly for comparison of 3-parameter optimized models with their 2-parameter counterpart.

Usage

```
LLln(x, s=NULL, Mulog, Sigmalog)
```

Arguments

x	A vector of failure data.
s	An optional vector of suspension data.
Mulog	The mean parameter from a log-normal fit.
Sigmalog	The standard deviation parameter from a log-normal fit.

Details

Function LLln can only handle datasets with failure and suspension data. For data including intervals wblrLikelihood can be used. This function is somewhat unique among likelihood functions in that it will scan the suspension data for removal of negative values. Such negative values are likely to be entered as data is provided for a 3-parameter fit by explicitly providing the original data with vector subtraction by the optimized t0. The primary intention for likelihood determination is to permit likelihood ratio testing for comparison of 3-parameter optimized models with their 2-parameter counterpart.

Value

Returns a log-likelihood value.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

Marie Laure Delignette-Muller, Christophe Dutang (2015). "fitdistrplus: An R Package for Fitting Distributions". Journal of Statistical Software, 64(4), 1-34. URL http://www.jstatsoft.org/v64/i04/.

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-lslr(getPPP(failures, suspensions),dist="lnorm")
LL<-LLln(failures, suspensions, fit[1], fit[2])
```

LLw *Log Likelihood for weibull fitted data, failures and suspensions only*

Description

Determination of log-likelihood values for the basic distributions covered by R should be ubiquitous, but an implementation is provided here to enable likelihood ratio testing particularly for comparison of 3-parameter optimized models with their 2-parameter counterpart.

Usage

```
LLw(x, s=NULL, Eta, Beta)
```

Arguments

x	A vector of failure data.
s	An optional vector of suspension data.
Eta	The scale parameter from a weibull fit.
Beta	The shape parameter from a weibull fit.

Details

Function LLw can only handle datasets with failure and suspension data. For data including intervals `wblrLikelihood` can be used. This function is somewhat unique among likelihood functions in that it will scan the suspension data for removal of negative values. Such negative values are likely to be entered as data is provided for a 3-parameter fit by explicitly providing the original data with vector subtraction by the optimized t0. The primary intention for likelihood determination is to permit likelihood ratio testing for comparison of 3-parameter optimized models with their 2-parameter counterpart.

Value

Returns a log-likelihood value.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

Marie Laure Delignette-Muller, Christophe Dutang (2015). "fitdistrplus: An R Package for Fitting Distributions". Journal of Statistical Software, 64(4), 1-34. URL http://www.jstatsoft.org/v64/i04/.

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-lslr(getPPP(failures, suspensions))
LL<-LLw(failures, suspensions, fit[1], fit[2])
```

LRbounds *Likelihood Ratio bounds*

Description

Generates bounds across a double-sided confidence interval based on a likelihood ratio contour.

Usage

```
LRbounds(x,  dist="weibull", CL=0.9, unrel=NULL,  contour=NULL, dof=1,
               control=NULL, debias="none", show=c(FALSE,FALSE))
```

Arguments

x	A dataframe such as generated by mleframe with column names 'left', 'right' and optionally 'qty'. Exact failure data (occurrences) have same time entered in both 'left' and 'right' columns. Suspension data has last known life[time] entered in 'left' column and -1 entered in 'right' column. The left(early) interval bound for left-censored data must be entered as zero. (NA is not accepted).
dist	A string defining a distribution to be fit. Implemented distributions are "weibull" (default) and "lognormal". (Only 2-parameter models are accepted.)
CL	A scalar for the double-sided confidence interval of interest. Default = 0.9, for 90
unrel	An optional vector of unreliability values to be used as the descriptive quantiles at which the bounds will be calculated.
contour	An optional dataframe object previously generated by MLEcontour on x, consistent with CL and dist.
dof	An integer value indicating degrees of freedom to apply to the Chi square test, which defaults to dof=1 for confidence interval bound use. Should be set to 2 for comparison of two models each with 2 parameters.
control	A list having specifically named options for control of the bounds preparation. List item 'ptDensity' an integer value for the number of points to be plotted around the circumference of the (each) 2p contour. List item 'tzpoints' controls the number of t0 trials in a 3p iteration (see Details). List item RadLimit a convergence limit for the contour radials based on specific units such as Eta/Eta_hat and Beta/Beta_hat.
debias	A place holder for an optional string argument indicating the adjustment to be applied to the shape or standard deviation parameter of the fitted data.
show	A vector of logicals determining whether and how a graphic of the bounds shall be displayed. For 2-parameter bounds only the first element is considered and a crude graphic is provided for the bounds. For 3-parameter bounds the first element determines whether a wblr graphic is displayed for the bounds, while the second element determines whether the series of contours generated to develop the bounds is displayed.

Details

The approach used for 3p bounds has been to vary the threshold value through a selection of increments from the optimized MLE toward zero. Then 2-parameter contours based on a likelihood ratio determined from the optimized MLE are plotted as if looking down on a 3 dimensional plot with threshold values in the z-direction. The bounds are then derived from the farthest extent of all the contours. tzpoints is a control argument that can be used to alter the number of threshold values (tz's) to be sampled. The tzpoints control argument is a vector having 3 elements. The first element controls the number of positive tz values to be sampled. The second and third elements of the tzpoints vector control the sequence of negative tz values according to the relationship tzpoints_neg<-seq(tzpoints[3],tzpoints[2], by=tzpoints[3]).

A bias adjustment, if specified, is applied to the shape parameter of generated contours. In this way such adjustment is passed into the bounds preparation.Abernethy's "Justified Likelihood Function" has not been demonstrated to be academically accepted. For this reason no attempt has been made to implement similar modification related to bias adjustment. Contribution of evidence or a method contrary to this position will gladly be reviewed by the package maintainer.

Value

Returns a dataframe holding values of percentiles, lower, datum, and upper values of the bound.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Examples

```
set.seed(4321)
data<-rlnorm(30,2,1.2)
bounds<-LRbounds(mleframe(data[7:30],data[1:6]), dist="lognormal")
```

---

lslr *Least squares linear regression with many optional methods*

Description

This function implements a wide assortment of options for linear regression fitting of distributions specific to reliability analysis for the Abernethy Reliability Methods project.

Usage

```
lslr(x, dist="weibull", npar=2, reg_method="XonY", abpval=TRUE)
```

Arguments

x	A dataframe such as generated by getPPP with column names 'data' and 'ppp'.
dist	A string defining a distribution to be fit. Implemented distributions are "weibull" (default), "lnorm", and "gumbel" (Extreme Value Type 1)
npar	Number of parameters to evaluate. This really is only looking for potential value of 3 for 3rd parameter optimization. Any other value will resolve the same as default of 2.
reg_method	A string defining the order of axes presented for regression, whether "XonY" axis(default) or "YonX" axis for alternative study.
abpval	A logical indicating whether to include the Abernethy P value in the result. When set to False the output vector will further be returned unnamed.

Value

Returns a named vector with parameters of the distribution fit and goodness of fit in terms of R squared.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition" Leonard C. Johnson (1964) "The Statistical Treatment of Fatique Experiments"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
default_weibull_fit<-lslr(getPPP(failures,suspensions))
```

MLEcontour *Likelihood Ratio Contour for Weibull and Lognormal Fitted Data*

Description

MLEcontour This function generates points for a display of the likelihood contour at given confidence limit for the 2-parameter Weibull or lognormal distributions.

Usage

```
MLEcontour(x,  dist="weibull", CL=0.9,dof=1,MLLx=NULL, MLEfit=NULL,
              RadLimit=1e-5,ptDensity=120, debias="none", show=FALSE)
```

Arguments

x	A dataframe such as generated by mleframe with column names 'left', 'right' and optionally 'qty'. Exact failure data (occurrences) have same time entered in both 'left' and 'right' columns. Suspension data has last known life[time] entered in 'left' column and -1 entered in 'right' column. The left(early) interval bound for left-censored data must be entered as zero. (NA is not accepted).
dist	A string defining a distribution to be fit. Implemented distributions are "weibull" (default) and "lognormal".
CL	a confidence limit to be applied to the Chi square test.
dof	an integer value indicating degrees of freedom to apply to the Chi square test, which defaults to dof=1 for confidence interval bound use. Should be set to 2 for comparison of two models each with 2 parameters.
MLLx	an optional argument intended for 3p modelling so that the maximum log-likelihood from an optimized 3rd parameter fit can be used to generate subsequent contours with variation in the translation parameter.
MLEfit	an optional argument to use a fit made external to the function. Disaster awaits any mismatch between data entered and this fit. When provided, the MLEfit shall be a vector in the order of scale(or logmean), shape (or std dev) and Log-likelihood.
ptDensity	an integer value for the number of points to be plotted around the circumference of the contour.
RadLimit	a convergence limit for the contour radials based on specific units such as Eta/Eta_hat and Beta/Beta_hat.
debias	An optional string argument indicating the adjustment to be applied to the shape or standard deviation parameter of the fitted data. Recognized values are "rba", "mean", or "hrbu".
show	a logical value indicating whether a graphical output is desired (independent of wblr activity).

Details

The contour points (p1,p2) identified as satisfying the root of the equation, (log(ML(p1_hat,p2_hat))-log(RL(p1,p2)) - chisquare(CL,DF)/2=0,where ML is Maximum Likelihood for the data,and RL is Ratioed Likelihood for the data at selected points for the contour. The algorithm for this function (executed in compiled C++ code) uses a quinary search for the root optimization of each radial on a polar coordinate loop. It is believed to be unique from commercial implementations and appears to have improved stability for this notoriously challenging calculation.

Value

A dataframe of plotting points for the contour.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Examples

```
fig3cF<-c(1500,2250,4000,4300,7000)
fig3cS<-c(1750,5000)
Contour<-MLEcontour(mleframe(fig3cF,fig3cS))
```

`mlefit` *Maximum likelihood regression on Weibull and Lognormal distributions*

Description

Maximum likelihood regression on weibull and lognormal distributions

Usage

```
mlefit(x, dist="weibull", npar = 2, debias="none", optcontrol=NULL)
```

Arguments

`x`	A dataframe such as generated by mleframe with column names 'left', 'right' and optionally 'qty'. Exact failure data (occurrences) have same time entered in both 'left' and 'right' columns. Suspension data has last known life[time] entered in 'left' column and -1 entered in 'right' column. The left(early) interval bound for left-censored data must be entered as zero. (NA is not accepted).
`dist`	A string defining a distribution to be fit. Implemented distributions are "weibull" (default),"lognormal", "weibull3p" and "lognormal3p".
`npar`	An optional argument for specifying 3p optimization. Priority is given to any suffix to the distribution name.
`debias`	An optional string argument indicating the adjustment to be applied to the shape or standard deviation parameter of the fitted data. Adjustments recognized are: "rba" (default), "mean", and "hrbu" (for hirose-ross beta unbias). Any entry for debias with the lognormal distribution will be processed as "rba" with a warning if it was not clearly specified.
`optcontrol`	An optional list of arguments for control of the MLE optimization. The strict naming convention for the list items are: 'vstart', a vector holding starting estimate for parameters in the order used by the underlying R distribution [CAUTION: for weibull the order of parameters is (shape, scope)], 'limit', the convergence limit, 'maxit', the maximum number of interations permitted in the simplex optimization, and 'listout' a logical indicating whether output should list both the fit and the optimization progress as a dataframe. Additional list items 'num_points', 'err_t0_limit', and 'err_gor_limit' are handled for 3rd parameter processing -see details.

Details

The negative log-likelihood is minimized by the Nelder-Meade, simplex, algorithm. This algorithm requires a reasonable starting point for the estimate. Such a reasonable estimate is attempted by default, but in some cases this may be inadequate for proper function. the optcontrol argument has been provided to give debug developers a means of altering and examining the progress of the optimization. For 3rd parameter processing defaults are used for the simplex options, while 'num_points' can be used to alter the seek points, 'err_t0_limit', and 'err_gor_limit' control cut-offs for unstable optiimizations, while 'list_out' provides a different output on the optimization progress.

Value

Returns a named vector with parameters of the distribution fit and goodness of fit in terms of log likelihood.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Marie Laure Delignette-Muller, Christophe Dutang (2015). "fitdistrplus: An R Package for Fitting Distributions". Journal of Statistical Software, 64(4), 1-34. URL http://www.jstatsoft.org/v64/i04/.

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
weibull_fit<-mlefit(mleframe(failures,suspensions))
```

`mleframe` *Set life(time) data into the format required by mlefit*

Description

Validate input data by types and build a dataframe to be used as the primary argument object to mlefit

Usage

```
mleframe(x, s=NULL, interval=NULL)
```

Arguments

x	An expected vector of failure data. Alternative support is provided for a dataframe holding time and event columns, where event markers for failure occurrences are 1 and suspensions (right censored data) are 0.
s	An optional vector of suspension data.
interval	A dataframe holding interval bounds for failure data in columns named 'left' and 'right'. The left(early) interval bound for left-censored data must be entered as zero (NA is not accepted). An optional column named 'qty' may contain the integer quantity of data entries having same interval values.

Value

Returns a dataframe of the life[time] data in columns named 'left', 'right', and 'qty' with an attribute of fsiq set to TRUE. Exact failure data (occurrences) have same time entered in both 'left' and 'right' columns. Suspension data has last known life[time] entered in 'left' column and -1 entered in 'right' column. The interval dataframe argument is appended. The 'qty' field is populated with a value of 1 for all failure and suspension entries and interval rows where qty was not provided.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Marie Laure Delignette-Muller, Christophe Dutang (2015). "fitdistrplus: An R Package for Fitting Distributions". Journal of Statistical Software, 64(4), 1-34. URL http://www.jstatsoft.org/v64/i04/.

Examples

```
fail<-c(10,40,40,50)
susp<-c(20,60)
left<-c(0,0,0,20,10)
right<-c(30,70,100,80,85)
qty<-c(2,1,1,2,1)
interval_ex<-data.frame(left,right,qty)
input_frame<-mleframe(fail,susp,interval_ex)
## time_event dataframe argument
failDF<-data.frame(time=fail, event=1)
suspDF<-data.frame(time=susp, event=0)
time_event_frame<-rbind(failDF, suspDF)
input_frame2<-mleframe(time_event_frame)
## now a time_event_qty dataframe argument
fa<-c(10,40,50)
fq<-c(1,2,1)
su<-susp
faDF<-data.frame(time=fa, event=rep(1,length(fa)), qty=fq)
## note: data.frame is forgiving about repeated single column entry
suDF<-data.frame(time=su, event=0, qty=1)
time_event_qty<-rbind(faDF, suDF)
```

```
input_frame3<-mleframe(time_event_qty)
```

MLEln2p *Quick Fit, Maximum Likelihood Estimate for 2-parameter lognormal distributions*

Description

Determination of lognormal fitting parameters, goodness of fit measures and confidence interval bounds with optional graphical display.

Usage

```
MLEln2p(x, s=NULL, bounds=FALSE, show=FALSE)
```

Arguments

x	A vector of failure data, or a dataframe wit time, event and optionally qty columns
s	An optional vector of suspension data.
bounds	A logical argument defining whether confidence interval bounds should be calculated by pivotal analysis.
show	A logical argument defining whether a simple graphical output is desired.

Details

This function is intended to provide a simple casual method of standard lognormal fitting based on default methods, without options.

Value

When the bounds argument is set to FALSE this function returns a vector with named elements for Mulog, Sigmalog, and LL (log-likelihood). When the bounds argument is set to TRUE a list is returned containing the vector as described and a dataframe of confidence interval bound values at a fixed set of descriptive quantiles, suitable for comparison with other software.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-MLEln2p(failures, suspensions)
```

MLEln3p *Quick Fit, Maximum Likelihood Estimate for 3-parameter lognormal distributions*

Description

Determination of lognormal fitting parameters with third, translation parameter optimization.

Usage

```
MLEln3p(x, s=NULL, bounds=FALSE, show=FALSE)
```

Arguments

x	A vector of failure data, or a dataframe wit time, event and optionally qty columns
s	An optional vector of suspension data.
bounds	A logical argument defining whether confidence interval bounds should be calculated by pivotal analysis.
show	A logical argument defining whether a simple graphical output is desired.

Details

This function is intended to provide a simple casual method of standard lognormal fitting based on default methods, without options.

Value

When the bounds argument is set to FALSE this function returns a vector with named elements for Mulog, Sigmalog, and LL (log-likelihood). Bounds are not determined for 3p models, hence any bounds argument is simply ignored.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-MLEln3p(failures, suspensions)
```

MLEw2p *Quick Fit, Maximum Likelihood Estimate for 2-parameter weibull distributions*

Description

Determination of Weibull fitting parameters, goodness of fit measures and confidence interval bounds with optional graphical display.

Usage

```
MLEw2p(x, s=NULL, bounds=FALSE, show=FALSE)
```

Arguments

x	A vector of failure data, or a dataframe wit time, event and optionally qty columns
s	An optional vector of suspension data.
bounds	A logical argument defining whether confidence interval bounds should be calculated by pivotal analysis.
show	A logical argument defining whether a simple graphical output is desired.

Details

This function is intended to provide a simple casual method of standard weibull analysis based on default methods, without options.

Value

When the bounds argument is set to FALSE this function returns a vector with named elements for Eta, Beta, and LL (log-likelihood). When the bounds argument is set to TRUE a list is returned containing the vector as described and a dataframe of confidence interval bound values at a fixed set of descriptive quantiles, suitable for comparison with other software.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-MLEw2p(failures, suspensions)
```

MLEw3p *Quick Fit, Maximum Likelihood Estimation for weibull distribution in 3-parameters*

Description

Determination of Weibull fitting parameters with third, translation parameter optimization. Result provided with goodness of fit measures with optional graphical display.

Usage

```
MLEw3p(x, s=NULL, bounds=FALSE, show=FALSE)
```

Arguments

x	A vector of failure data.
s	An optional vector of suspension data.
bounds	A logical argument defining whether confidence interval bounds should be calculated by pivotal analysis.
show	A logical argument defining whether a simple graphical output is desired.

Details

This function is intended to provide a simple casual method of standard weibull analysis based on default methods, without options.

Value

When the bounds argument is set to FALSE this function returns a vector with named elements for Eta, Beta, and LL (log-likelihood). Bounds are not determined for 3p models, hence any bounds argument is simply ignored.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-MLEw3p(failures, suspensions)
```

MRRln2p *Quick Fit, Median rank regression for 2-parameter log-normal distributions*

Description

Determination of fitting parameters, goodness of fit measures and confidence interval bounds with optional graphical display.

Usage

```
MRRln2p(x, s=NULL, bounds=FALSE, show=FALSE)
```

Arguments

x	A vector of failure data.
s	An optional vector of suspension data.
bounds	A logical argument defining whether confidence interval bounds should be calculated by pivotal analysis.
show	A logical argument defining whether a simple graphical output is desired.

Details

This function is intended to provide a simple casual method of standard lognormal fitting based on default methods, without options. It also provides an example for handling the pivotal values returned from pivotalMC.

Value

When bounds is set to FALSE this function returns a vector with named elements for Mulog, Sigmalog, Rsqr, AbPval (Abernethy's P-value), and LL (log-likelihood). When the bounds argument is set to TRUE a list is returned with the vector as described and a dataframe of confidence interval bound values at a fixed set of descriptive quantiles, dq<-c(.01, .02, .05, .10, .15, .20, .30, .40, .50, .60, .70, .80, .90, .95, .99), suitable for comparison with other software.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition" Leonard C. Johnson (1964) "The Statistical Treatment of Fatigue Experiments"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-MRRln2p(failures, suspensions)
```

MRRln3p *Quick Fit, Median rank regression for log-normal distribution with third parameter optimization*

Description

Determination of log-normal fitting parameterswith third parameter optimization. Goodness of fit measures and confidence interval bounds can be returned with optional graphical display.

Usage

```
MRRln3p(x, s=NULL, bounds=FALSE, show=FALSE)
```

Arguments

x	A vector of failure data.
s	An optional vector of suspension data.
bounds	A logical argument defining whether confidence interval bounds should be calculated by pivotal analysis.
show	A logical argument defining whether a simple graphical output is desired.

Details

This function is intended to provide a simple casual method of standard lognoirmal fitting based on default methods, without options. It also provides an example for handling the pivotal values returned from pivotalMC and the likelihood ratio test.

Value

When bounds is set to FALSE this function returns a vector with named elements for Eta, Beta, Rsqr, AbPval (Abernethy's P-value), and LL (log-likelihood). When the bounds argument is set to TRUE a list is returned with the vector as described and a dataframe of confidence interval bound values at a fixed set of descriptive quantiles, dq<-c(.01, .02, .05, .10, .15, .20, .30, .40, .50, .60, .70, .80, .90, .95, .99), suitable for comparison with other software.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition" Leonard C. Johnson (1964) "The Statistical Treatment of Fatigue Experiments"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-MRRln3p(failures, suspensions)
```

MRRw2p *Quick Fit, Median rank regression for 2-parameter weibull distributions*

Description

Determination of Weibull fitting parameters, goodness of fit measures and confidence interval bounds with optional graphical display.

Usage

```
MRRw2p(x, s=NULL, bounds=FALSE, show=FALSE)
```

Arguments

x	A vector of failure data.
s	An optional vector of suspension data.
bounds	A logical argument defining whether confidence interval bounds should be calculated by pivotal analysis.
show	A logical argument defining whether a simple graphical output is desired.

Details

This function is intended to provide a simple casual method of standard weibull analysis based on default methods, without options. It also provides an example for handling the pivotal values returned from pivotalMC.

Value

When the bounds argument is set to FALSE this function returns a vector with named elements for Eta, Beta, Rsqr, AbPval (Abernethy's P-value), and LL (log-likelihood). When the bounds argument is set to TRUE a list is returned containing the vector as described and a dataframe of confidence interval bound values at a fixed set of descriptive quantiles, dq<-c(.01, .02, .05, .10, .15, .20, .30, .40, .50, .60, .70, .80, .90, .95, .99), suitable for comparison with other software.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition" Leonard C. Johnson (1964) "The Statistical Treatment of Fatigue Experiments"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-MRRw2p(failures, suspensions)
```

MRRw3p *Quick Fit, Median rank regression for weibull distribution in 3-parameters*

Description

Determination of Weibull fitting parameters with third, translation parameter optimization. Result provided with goodness of fit measures and confidence interval bounds with optional graphical display.

Usage

```
MRRw3p(x, s=NULL, bounds=FALSE, show=FALSE)
```

Arguments

x	A vector of failure data.
s	An optional vector of suspension data.
bounds	A logical argument defining whether confidence interval bounds should be calculated by pivotal analysis.
show	A logical argument defining whether a simple graphical output is desired.

Details

This function is intended to provide a simple casual method of standard weibull analysis based on default methods, without options. It also provides examples for handling the pivotal values returned from pivotalMC and performance of the likelihood ratio test.

Value

When the bounds argument is set to FALSE this function returns a vector with named elements for Eta, Beta, Rsqr, AbPval (Abernethy's P-value), and LL (log-likelihood). When the bounds argument is set to TRUE a list is returned containing the vector as described and a dataframe of confidence interval bound values at a fixed set of descriptive quantiles, dq<-c(.01, .02, .05, .10, .15, .20, .30, .40, .50, .60, .70, .80, .90, .95, .99), suitable for comparison with other software.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition" Leonard C. Johnson (1964) "The Statistical Treatment of Fatigue Experiments"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
fit<-MRRw3p(failures, suspensions)
```

`options.wblr` *Options list for* `wblr` *Objects*

Description

This function handles the various calculation, printing and plotting options for wblr objects.

Usage

```
options.wblr(...)
```

Arguments

`...`	Options for calculating, printing and plotting `wblr` objects using the **WeibullR** package.

Details

Typical usage of `wblr` objects involves calling a sequence of functions each forming the object, then modifying it in this typical manner:

```
da <- wblr(c(10,11,27), col="red")
da <- wblr.fit(da, col="darkgreen")
da <- wblr.conf(da), col="blue"
plot(da)
```

The correct time to specify an option is when it is needed for the first time. For example, when the color setting option `col = "red"` is passed as an argument of function `wblr`, it will be used for data points, fitted lines and confidence interval bounds. If supplied to `wblr.conf`, only the confidence bounds will have the re-specified color, hereby overriding any previously inherited color settings from `wblr.fit` or `wblr`.

Do not call `options.wblr` in between these functions because some options are locked and cannot be altered further in this chain. This is an implication of the way the `wblr` object is structured.

As a function, options.wblr, borrows its internal structure from the `par` function of package **graphics**. It can be used independently of the typical wblr, wblr.fit, wblr.conf sequence, but this is discouraged. However, when used independently the following guidance is provided:

```
options.wblr()
```

Returns the currently used options and their values.

```
options.wblr()$dist options.wblr("dist")
```

Returns the current value of an option.

```
options.wblr(ci=0.95) options.wblr(list(ci=0.95,S=5e4))
```

Sets the specified options.

Currently, there is no way to reset the options to the default values using this function. One might, before changing any options, store the option list in a temporary variable like

```
wblr.defaults <- options.wblr()
```

for restoring it later by running `options.wblr(wblr.defaults)`.

The function creates a globally accessible list named `options_wblr`, holding the options. One should always use the `options.wblr` function to access the option list, do not access this list directly.

Value

Executing `options.wblr` without arguments returns a named list containing the currently active global options of the `wblr` object.
When arguments are supplied, these are returned in a named list.

wblr options

Options specific to initial `wblr` object creation with data prepared for graphical display

`dist` A character string defining the distribution target. When used to establish the basis for contour mapping (without using wblr.conf with method.conf="lrb") only "weibull" (default) and "lognormal" are recognized.
Also used with `wblr.fit` for specific fitting control.

`pp` Plotting position method, it is a character string describing the method of determining vertical plot positions. Implemented methods are "median" (default), "benard","hazen","mean", "kaplan-meier", and "blom".

`rank.adj` The method employed for determining rank of failures when suspensions (right censored data) are present in the data set. Implemented methods are "johnson" (default) and "KMestimator".

`ties.handler` The method employed for handling duplicate values in the data set.
Implemented methods are "none" (default) "highest", "lowest", "mean", and "sequential".
It is expected that ties handling will be applied to large data sets that will be fitted using the maximum likelihood estimation method, where the effect is only on the graphical presentation. Employing a ties handler on a rank regression model will effectively remove data from the data set, which is likely not intended.
Use of simply `ties` as an argument to function `wblr` will silently be accepted as `ties.handler`.

Options for graphical control over data points see `par` `pch` Point choice defaults to `1`. For more info, see `points`.

`cex.points` Point size defaults to `1`.

`lwd.points` Line width defaults to `2`.

Independent graphical control over interval lines `interval.col` Color defaults to `"black"`.

`interval.lty` Line type, defaults to `"dashed"`.

`interval.lwd` Line width defaults to `1`.

Options specific to `wblr.fit`

`dist` A character string defining the distribution target. Recognized values are "weibull" (default), "lognormal","lnorm","lognormal2p", "weibull2p","lognormal3p", and "weibull3p".

`method.fit` A vector of class `"character"` with fitting options. Recognized values are "rr-xony" (default), "rr","rr-yonx", "mle","mle-rba", and "mle-unbias".

Options specific to `wblr.conf`

`method.conf` A character string describing the techniques used for calculating confidence interval bounds. Implemented methods are "pivotal-rr" (default), "bbb", "fm" "fmbounds", and "lrb". Methods must conform to the method.fit in the wblr.fit call immediately preceding the wblr.conf call. Method "pivotal-rr" requires a rank regression fit method. Methods "fm", "fmbounds", and "lrb" require a mle based fit.

`dq` A named series of quantiles at which confidence interval bounds will be calculated.

`"abrem"` Default. This is the original default by Jurgen Symynck for predecessor package abrem it produces evenly spaced points across the y limits of a weibull canvas attempting to hold a constant number of points (see `num_dq` below).

`"minitab"` Quantiles matching Minitab(TM) unchangeable defaults (27 values).

`"supersmith"` Quantiles for comparison with SuperSMITH(TM) (limit of 15 values)

`"user"` Provides for a user defined series of quantiles. (see `user_dq` below).

`num_dq` The number of points used for the `"abrem"` dq determination.

`user_dq` A vector of quantiles set by user. Default `c(seq(.01,.09,by=.01),seq(.10,.90,by=.10),seq(.91,.99,by=.01))`.

`ci` The double-sided confidence interval, also chi sq confidence level for likelihood ratio. Must be in a range <1 && >0, default is `0.9`.

`blife.pts` The probability points at which to report Blife on legend.

Specific controls for pivotal analysis only `seed` The RNG seed integer such that results are duplicated between runs, default is `1234`.

`S` The number of samples to be run during pivotal analysis, default is `1e4`.

Specific controls for likelihood ratio contour and bounds only `dof` The degrees of freedom, `dof=1` (default) for confidence interval, `dof=2` for comparison.

`ptDensity` The number of points calculated to form the contour outline, default is `120`.

General graphical options

Graphical control for fitted lines, confidence bounds, or contours can be set at wblr, wblr.fit, or wblr.conf functions fo
Color defaults to `"black"`.

`lty` Line type, defaults to `"solid"`.

`lwd` Line width defaults to `2`.

Graphical controls typically used with `plot` **or** `plot.wblr` `canvas` The plotting canvas to be used. This does not necessarily have to match the fit distribution. Only `"weibull"` (default) or `"lognormal"` are recognized.

`mar` Margins defaults to `c(5.1,4.1,5.1,2.1),`.

`main` Title, defaults to `"Probability Plot"`.

`main.contour` Contour plot title defaults to `"Contour Plot"`.

`sub` Subtitle defaults to `NULL`.

`sub.contour` Contour subtitle defaults to `NULL`.

`xlim` Plot x limits override to be presented as a vector c(lo,hi), default `NULL`.

`ylim` Plot y limits override to be presented as a vector c(lo,hi), default `NULL`.

`xlab` X axis label defaults to `"Time To Failure"`.

`ylab` Y axis label defaults to `"Unreliability [%]"`.

`coordinate.text.size` default `0.7`.

`signif` Used to control display of numbers in Legend, default `4`.

`col.grid` Color for chart gridlines defaults `"grey"`.

`is.plot.grid` default `TRUE`.

`is.plot.fit` default `TRUE`.

`is.plot.pp` default `TRUE`.

`is.plot.ppcoordinates` default `FALSE`.

`is.plot.legend` default `TRUE`.

`legend.position` default `"bottomright"`. See `legend` Details.

`legend.inset` default `c(0,0)`. legend inset values are fractions of graph width and height.

`legend.text.size` default `0.7`.

`label` defaults to `""`.

`in.legend` default `TRUE`.

`in.legend.blives` default `TRUE`.

`in.legend.gof` default `TRUE`.

`is.plot.cb` default `TRUE`.

`persistent` default `TRUE`.

Examples

```
## backup options ##
wblr.defaults <- options.wblr()

## setting new options ##
options.wblr(S=5e5,ci=0.99)

## listing options ##
options.wblr()
options.wblr()$main

## restore options ##
options.wblr(wblr.defaults)
```

---

p2y *Transform Probability Value to the Y-Axis of a "plot.wblr" Canvas*

Description

This function used to transform a probability value to the y-axis of a plot canvas.

Usage

```
p2y(p,canvas="weibull")
```

Arguments

p	The percentile or probability value
canvas	The name of the wblr canvas to plot to. (Does not have to match the fit distribution. Use of this transformation permits distributions to appear as curves on unrelated canvas.)

Details

This applies the inverse Cumulative Distribution function. When applying the default "weibull" canvas this function is equivalent to SPLEDA::qsev. It is particularly handy when adding points and lines to plots generated on wblr objects.

Value

The transformed y-axis value for a "plot.wblr" Canvas.

---

pivotal.rr *Pivotal 'Monte Carlo' re-sampling of least squares linear regression models*

Description

This function implements a fast pivotal engine enabling a wide assortment of options for linear regression models for the Abernethy Reliability Methods project. Pivotal analysis is used to determine a goodness of fit measure and confidence interval bounds.

Usage

```
pivotal.rr(x, event=NULL, dist="weibull", reg_method="XonY", R2, CI, unrel=NULL,
  P1=1.0, P2=1.0, npar=2, S=10^4, seed=1234, ProgRpt=FALSE)
```

Arguments

x	A dataframe such as generated by getPPP with column names 'data' and 'ppp'.
event	An optional integer vector with 0's identifying the rank position of suspension data for sample masking. Event masking will modestly narrow the confidence interval consistent with the addition of information. The user has full control over determination of significance of the suspension data for this process.
dist	A string defining a distribution to be fit. Implemented distributions are "weibull" (default), "lnorm", "weibull3p", and lognormal3p" are implemented. ["gumbel" (Extreme Value Type 1) experimental support is currently suspended]
reg_method	A string defining the order of regression whether "XonY" axis(default and "best practice for standard small sample fitting) or "YonX" axis for alternative study ("YonX" recommended for an inspection option by Abernethy).

R2	Output control: for prr and ccc2 the explained variance (R squared) from a linear regression of a data sample of interest. An entry of 0 (zero) suppresses output, 1.0 requests the entire vector of r^2 values.
CI	Output control: for the bounds provide the double sided confidence interval of interest. An entry of 0 (zero) suppresses output, 1.0 requests an entire matrix of generated values.
unrel	An optional vector of unreliability values to be used as the descriptive quantiles at which the bounds will be calculated.
P1	The scale parameter to be used in random sampling. Default = 1.0 (For lnorm or gumbel distributions this most likely will be preferred to be set to zero.)
P2	The shape parameter to be used in random sampling. Default = 1.0
npar	An alternate way to identify the number of distribution parameters (2 or 3 are only possibilities). A dist of "weibull3p" or "lognormal3p" will override any value entered as npar.
S	The number of random samples to be drawn for Monte Carlo simulation. S must be a multiple of 10. If S is less than 1,000 R2 and/or CI will be altered from non-zero quantities to 1.0 to return the full vector of R2 CDF or the full matrix of pivotal values. The default of 10^4 is adequate for most instances. S is implemented as an unsigned int in C++ code. The maximum limit is 4x10^9 if system memory permits.
seed	an integer used to set the RNG seed. Default = 1234
ProgRpt	A boolean value to control the generation of percent completion feedback in the R terminal.

Details

Pivotal quantities are determined by establishing the x-axis value at each descriptive quantile position for each sampled regression. The output pivotals determined at the double sided confidence interval must be interpreted for application to any given data fit.

Value

Returns an appropriate object for the input R2 and CI values. There are 8 output configurations that can be generated depending on the argument values for R2 (for r^2, coefficient of determination) and CI (for confidence interval). When either of these values is entered as zero, no output for the corresponding prr value or confidence interval are generated. Specifically defined input values for either argument are in the range 0.0<value<1.0 . When a specifically defined value has been provided the output for R2 will be a vector containing the prr value and the CCC^2. The output for a specifically defined CI will be a dataframe containing the pivotal quantities. Should an absolute value of 1.0 be given for either R2 or CI then a full vector of the r^2 CDF, or the full matrix of pivotal quantities generated for all samples will be returned. When non-zero entries are provided for both R2 and CI the appropriate output objects are returned in a list.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

Jerald F. Lawless, (2003) "Statistical Models and Methods for Lifelime Data, Second Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Chi-Chao Lui, "A Comparison Between The Weibull And Lognormal Models Used To Analyse Reliability Data" (dissertation from University of Nottingham, 1997)

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
data_positions<-getPPP(failures,suspensions)
fit<-lslr(data_positions)
prr_value<-pivotal.rr(data_positions,R2=fit[3],CI=0, unrel=.5)
## note: unrel value has no meaning when CI=0
```

`plot.wblr` *S3* `wblr` *Object Plotting on pretty canvax*

Description

This function adds the `.wblr` method to `plot.default` from the **graphics** package.

Currently, the function plots the (life-)time observations, fits (if any) and confidence bounds for B-lives (if any) of an `wblr` object or a list of `wblr` objects on Weibull or Lognormal probability paper.

For each fit in the (list of) `wblr` object(s), legends are added to the plot, displaying the fit parameters and (if available) goodness-of-fit indicators and confidence information.

Usage

```
## S3 method for class 'wblr'
plot(x, ...)
```

Arguments

`x`	Object of class `"wblr"` or a list of `wblr` objects.
`...`	Options for plotting the `wblr` object; see section "Details".

Details

The `...` argument can be any graphical parameter that can be supplied to `plot.default`, and any option that can be set by the function `options.wblr`. The options set in this way are applied to all graphical elements of the plot, overriding any previously supplied options.

One can pass a list of `wblr` objects to `plot.wblr`; in that case it is mandatory to use the full method name: `plot.wblr(...)` and not `plot(...)`.

The calculated Weibull or Lognormal distribution fits are plotted on Weibull probability paper by default, but by passing the argument `canvas = "lognormal"` to the function, Lognormal paper is used.

When a *list* of wblr objects is passed, the plot window is generated with the options of the first `wblr` object in the list.

Graphical controls typically used with `plot` or `plot.wblr`:

`canvas` The plotting canvas to be used. This does not necessarily have to match the fit distribution. Only `"weibull"` (default) or `"lognormal"` are recognized.

`mar` Margins defaults to `c(5.1,4.1,5.1,2.1),`.

`main` Title, defaults to `"Probability Plot"`.

`main.contour` Contour plot title defaults to `"Contour Plot"`.

`sub` Subtitle defaults to `NULL`.

`sub.contour` Contour subtitle defaults to `NULL`.

`xlim` Plot x limits override to be presented as a vector c(lo,hi), default `NULL`.

`ylim` Plot y limits override to be presented as a vector c(lo,hi), default `NULL`.

`xlab` X axis label defaults to `"Time To Failure"`.

`ylab` Y axis label defaults to `"Unreliability [%]"`.

`coordinate.text.size` default `0.7`.

`signif` Used to control display of numbers in Legend, default `4`.

`col.grid` Color for chart gridlines defaults `"grey"`.

`is.plot.grid` default `TRUE`.

`is.plot.fit` default `TRUE`.

`is.plot.pp` default `TRUE`.

`is.plot.ppcoordinates` default `FALSE`.

`is.plot.legend` default `TRUE`.

`legend.position` default `"bottomright"`. See `legend` Details.

`legend.inset` default `c(0,0)`. legend inset values are fractions of graph width and height.

`legend.text.size` default `0.7`.

`label` defaults to `""`.

`in.legend` default `TRUE`.

`in.legend.blives` default `TRUE`.

`in.legend.gof` default `TRUE`.

`is.plot.cb` default `TRUE`.

`persistent` default `TRUE`.

Value

Currently, the function returns no value.

References

Jurgen Symynck, Filip De Bal, Weibull analysis using R, in a nutshell (New Technologies and Products in Machine Manufacturing Technology, Stefan cel Mare University of Suceava, 2010).

Jurgen Symynck, Filip De Bal, Monte Carlo pivotal confidence bounds for Weibull analysis, with implementations in R (New Technologies and Products in Machine Manufacturing Technology, Stefan cel Mare University of Suceava, 2011).

Examples

```
options.wblr(blives=0.1) # make the legend boxes a bit shorter...
da2 <- wblr.conf(wblr.fit(wblr(runif(5,10,100),col="red")))
da3 <- wblr.conf(wblr.fit(wblr(rweibull(5,2,1000),col="green4",pch=3)))
da4 <- wblr.conf(wblr.fit(wblr(rlnorm(5,log(500),log(2)),col="blue3",pch=8),
    dist="lognormal"))
## Not run:
plot.wblr(list(da2,da3,da4),xlim=c(1,1e6),
    main="Uniformly distributed observations")

## End(Not run)
```

---

plot_contour *Plotting of Likelihood Ratio Contours from* wblr *Objects*

Description

Plots likelihood ratio contours from a wblr object or a list of wblr objects.

Usage

```
plot_contour(x, CL=NULL, AL=TRUE, ...)
```

Arguments

x	Object of class "wblr" or a list of "wblr" objects.
CL	An optional argument for establishing the confidence limit(s) for calculated contours. A vector of values may be provided to generate concentric contours on a single object.
AL	A Logical defining whether parameters of lognormal contours should be presented as AntiLog values.
...	Entry for a limited set of graphic parameters (col, lty, lwd, xlim, ylim, main and sub)

Details

If no CL argument is provided, the contours generated in the provided wblr objects for likelihood confidence bounds will be plotted. If a vector is provided for CL, multiple contours for each provided object will be calculated to display concentric contours according to values in the CL vector. In all cases options for distribution, degrees of freedom and graphic options will be extracted from the underlying objects.

Value

This function returns the list of contours that were prepared for plotting.

Examples

```
set.seed(1234)
fail<-rweibull(5,2,10)
ctest<-wblr(fail)
ctest<-wblr.fit(ctest, method.fit="mle")
ctest<-wblr.conf(ctest, method.conf="lrb",col="red")
fail2<-c(10,40,40,50)
susp<-c(20,60)
ctest2<-wblr(fail2, susp)
ctest2<-wblr.fit(ctest2, method.fit="mle")
ctest2<-wblr.conf(ctest2, method.conf="lrb",col="blue")
## Not run:
 plot_contour(list(ctest, ctest2))

## End(Not run)
```

---

rba *Reduced Bias Adjustment for Weibull and Lognormal MLE*

Description

rba This function generates reduction factors based on the median bias of the Weibull Beta parameter MLE (C4^3.5), alternatively based on the mean bias(C4^6), and a single factor for both median and mean of the symmetrical lognormal distribution.

Usage

```
rba(Qx, dist="weibull", basis="median")
```

Arguments

Qx	The quantity of failures in data (right-censored data, or suspensions excluded)
dist	A string defining a distribution to be fit. Implemented distributions are "weibull" (default) and "lognormal".
basis	a string value indicating the basis for bias reduction, "median" is the default, but "mean" may be chosen.

Details

Many references discuss the bias reduction in terms of mean reduction. Abernethy has promoted consideration of the RBA in terms of median bias reduction.

Value

A factor to be multiplied to the MLE Beta or log-Sigma for bias reduction.

References

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
MLEfit<-mlefit(mleframe(failures,suspensions))
MLE_RBA<-c(MLEfit[1],MLEfit[2]*rba(length(failures)))
```

---

wblr *Create a* wblr *Object for Life Data Analysis*

Description

This function creates an object of class `"wblr"` for further processing by the other functions of **wblr**.

Usage

```
wblr(x, s=NULL, interval=NULL,...)
```

Arguments

`x`	Either a dataframe containing at least `$time` and `$event` columns and optionally a `$qty` column, or a vector of class `"numeric"` or `"integer"` with (life-)time observations.
`s`	An optional vector of right-censored data, or suspensions.
`interval`	An optional dataframe of interval data having columns specifically named "left" and "right". Left values are the last time at which no failure was evident and may be zero for discovery. Right values are the earliest time at which failure was observed.
`...`	Graphical options for plotting the `wblr` object. See section "Details" for other data passing arguments.

Details

There are several methods to passing arguments for building an `wblr` object.

1. If argument `x` is of class `"data.frame"` then it must contain `$time` and `$event` columns. Additional columns in the dataframe will be ignored.
2. When a single unnamed vector of class `"numeric"` or `"integer"` is supplied it is treated as a vector of (life-)time observations.
3. If argument `time` or `fail` is provided it is treated as a vector of (life-)time observations. Take care NOT to supply both `time` and `fail` in the same function call.

4. If argument `event` is provided it is treated as a vector of even time indicators with possible values of `0` and `1`. `0` and `1`. See section "Value" for more details on event vectors.
5. If the `x` argument is not provided as a dataframe and `susp` is provided, it is treated as a vector of right-censored (life-)time observations (also called suspended observations or suspensions).

`wblr` always generates (probability) plot positions for graphically displaying the (life-)time observations and for (possible) later usage by `wblr.fit`. The following optiona arguments are most appropriate for passing in with `wblr`:

`dist` A character string defining the distribution target. When used to establish the basis for contour mapping (without using `wblr.conf` with method.conf="lrb") only "weibull" (default) and "lognormal" are recognized.
Also used with `wblr.fit` for specific fitting control.

`pp` Plotting position method, it is a character string describing the method of determining vertical plot positions. Implemented methods are "median" (default), "benard","hazen","mean", "kaplan-meier", and "blom".

`rank.adj` The method employed for determining rank of failures when suspensions (right censored data) are present in the data set. Implemented methods are "johnson" (default) and "KMestimator".

`ties.handler` The method employed for handling duplicate values in the data set.
Implemented methods are "none" (default) "highest", "lowest", "mean", and "sequential".
It is expected that ties handling will be applied to large data sets that will be fitted using the maximum likelihood estimation method, where the effect is only on the graphical presentation. Employing a ties handler on a rank regression model will effectively remove data from the data set, which is likely not intended.
Use of simply `ties` as an argument to function `wblr` will silently be accepted as `ties.handler`.

Options for graphical control over data points see `par` `pch` Point choice defaults to `1`. For more info, see `points`.

`cex.points` Point size defaults to `1`.

`lwd.points` Line width defaults to `2`.

Independent graphical control over interval lines `interval.col` Color defaults to `"black"`.

`interval.lty` Line type, defaults to `"dashed"`.

`interval.lwd` Line width defaults to `1`.

Subsequent calls to `wblr.fit` and `wblr.conf` will inherit these options.

Value

A named list of class `"wblr"`. The first list item (`$data`) is a list with up to least three items:

`$lrq_frame` A dataframe containing the provided data formatted with `"left"`, `"right"`, and `"qty"` columns. This is the output of WeibullR function `"mleframe"`.

`$data$dpoints` A dataframe contianing graphical data for exact failure point with their probability plotting positions and adjusted ranks.

`$data$dlines` If interval data has been provided this dataframe will contain the graphical data for display similar to `$data$dpoints`, but with endpoints t1 and t2 for the interval.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Jurgen Symynck, Filip De Bal, Weibull analysis using R, in a nutshell (New Technologies and Products in Machine Manufacturing Technology, Stefan cel Mare University of Suceava, 2010).

Examples

```
## These code lines all generate the same object ##
wblr(c(500,1200,900,1300,510))
wblr(time=c(500,1200,900,1300,510))
## this input format works, but not recommended.
wblr(time=c(500,1200,900,1300,510),event=c(1,1,1,1,1))
wblr(fail=c(500,1200,900,1300,510))
wblr(fail=c(500,1200,900,1300,510),susp=c())
da1 <- data.frame(
    serial=c("S12","S16","S17","S3","S5"),
    time=c(500,1200,900,1300,510),
    event=c(1,1,1,1,1))
## it is best practice set named objects
obj1 <- wblr(da1,label="complete dataset",pch=3,col="orange3")
obj2 <- wblr(da1,label="complete dataset",pch=4,pp="benard",col="red")


## Generate a similar dataset, but with suspensions ##
wblr(time=c(500,1200,900,1300,510),event=c(1,1,1,0,0))
wblr(data.frame(time=c(500,1200,900,1300,510),event=c(1,1,1,0,0)))
wblr(fail=c(500,1200,900),susp=c(1300,510))
wblr(time=c(500,1200,900),susp=c(1300,510))
da3 <- wblr(fail=c(500,1200,900,1300,510),
    event=c(1,1,1,0,0),label="censored dataset",pch=1,col="blue")

## plot datasets ##
## Not run:
plot.wblr(list(da1,da3))

## End(Not run)
```

`wblr.conf` *Add Confidence Interval Bounds to* `wblr` *Objects*

Description

This function adds confidence calculations to the last fit entity in a `wblr` object.

Usage

```
wblr.conf(x,...)
```

Arguments

x	Object of class "wblr".
...	Options for calculating confidence, and for plotting the results.

Details

This function adds confidence calculations to the last fit entity in a wblr object and adds the result to the object alongside any pre-existing confidence calculations.

Certain confidence calculations pertain to specific fit methods. The pivotal.rr confidence method can only apply to a rank regression fit, while a likelihood ratio calculation can only be performed on an mle fit.

Option arguments suitable for passing into wblr.conf:

`method.conf` A character string describing the techniques used for calculating confidence interval bounds. Implemented methods are "pivotal-rr" (default), "bbb", "fm" "fmbounds", and "lrb". Methods must conform to the method.fit in the wblr.fit call immediately preceding the wblr.conf call. Method "pivotal-rr" requires a rank regression fit method. Methods "fm", "fmbounds, and "lrb" require a mle based fit.

`dq` A named series of quantiles at which confidence interval bounds will be calculated.

- `"abrem"` Default. This is the original default by Jurgen Symynck for predecessor package abrem it produces evenly spaced points across the y limits of a weibull canvas attempting to hold a constant number of points (see num_dq below).
- `"minitab"` Quantiles matching Minitab(TM) unchangeable defaults (27 values).
- `"supersmith"` Quantiles for comparison with SuperSMITH(TM) (limit of 15 values)
- `"user"` Provides for a user defined series of quantiles. (see user_dq below).

`num_dq` The number of points used for the `"abrem"` dq determination.

`user_dq` A vector of quantiles set by user. Default `c(seq(.01,.09,by=.01),seq(.10,.90,by=.10),seq(.91,.99,by=.01))`.

`ci` The double-sided confidence interval, also chi sq confidence level for likelihood ratio. Must be in a range <1 && >0, default is `0.9`.

`blife.pts` The probability points at which to report Blife on legend.

Specific controls for pivotal analysis only `seed` The RNG seed integer such that results are duplicated between runs, default is `1234`.

`S` The number of samples to be run during pivotal analysis, default is `1e4`.

Specific controls for likelihood ratio contour and bounds only `dof` The degrees of freedom, `dof=1` (default) for confidence interval, `dof=2` for comparison.

`ptDensity` The number of points calculated to form the contour outline, default is `120`.

Additionally, graphical control options for the confidence interval bounds can be passed in with the following options:

`col` Color defaults to `"black"`.

`lty` Line type, defaults to `"solid"`.

`lwd` Line width defaults to 2.

Subsequent calls to `wblr.fit` and `wblr.conf` will inherit these options.

Value

The function returns its argument `x`, extended with the confidence calculations and any optional graphical and calculation arguments as passed to the function.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Examples

```
## full dataset ##
da1 <- wblr(runif(10,100,1e4),label="Complete data")
da1 <- wblr.fit(da1)
da1 <- wblr.conf(da1,method.conf="pivotal-rr",col="red")
da1 <- wblr.conf(da1,method.conf="bbb",col="orange")
da1 <- wblr.fit(da1, method.fit="mle")
da1 <- wblr.conf(da1,method.conf="lrb",col="yellow3")

## plot datasets ##
## Not run:
plot(da1,main="Comparison between different bound types.")

## End(Not run)

## with right-censored data
da2 <- runif(8,100,1e4)
da2 <- wblr(fail=da2,susp=rep(max(da2),2),label="Type II right-censored data")
da2 <- wblr.fit(da2)
da2 <- wblr.conf(da2,method.conf="pivotal-rr",col="blue1")
da2 <- wblr.conf(da2,method.conf="bbb",col="steelblue")
da2 <- wblr.fit(da2, method.fit="mle")
da2 <- wblr.conf(da2,method.conf="lrb",col="cyan3")
## Not run:
plot(da2,main="Comparison between different bound types with right-censored data.")

## End(Not run)
```

wblr.fit *Add Fit Distributions to* wblr *Objects*

Description

This function fits probability distributions to wblr objects.

Usage

```
wblr.fit(x, modify.by.t0=FALSE, ...)
```

Arguments

x	Object of class "wblr".
modify.by.t0	A logical value to signifying whether to revise object data by subtraction of the "t0" (translation) parameter of a 3-parameter fit. A value of TRUE generates a linearized view of the fit on its base distribution canvas. It is recommended that the resulting object have an altered name perhaps adding a ".3p" suffix to the original wblr object to preserve original data.
...	Options for fitting the (life-)time observations, and for plotting the results.

Details

This function calculates fits for the (life-)time observations in the wblr object and adds them to the object alongside any pre-existing fits.

Fitting options are passed with the dist and method.fit arguments:

dist A character string with the target distribution for fitting. Possible values are "weibull", "weibull2p", "weibull3p" (three parameter Weibull), "lognormal" , "lognormal2p"or "lognormal3p".

Defaults to "weibull".

in.legend Logical value controlling the inclusion of various elements in the legend.

If in.legend=FALSE is passed, the resulting fit calculations will be omitted from the legend, leaving only observation summary data.

Defaults to TRUE.

method.fit A vector of class "character" with fitting options.

Defaults to "rr-xony".

- "rr": Rank Regression (RR). Depending on the method for calculating probability plot positions chosen during the creation of the wblr object (see option pp and function wblr), this can either be "exact median rank regression" or "Benard's approximate median rank regression". If this method is used then it is mandatory to additionally specify either X-on-Y or Y-on-X regression.
- "xony","yonx": Differentiate between X-on-Y and Y-on-X regression, respectively. For rank regression in lifetime analysis, it is best practice to use the X values ((life-)time observations) as the response variables whose horizontal distance to the fit line must be minimized, and the Y values (unreliabilities) as the explanatory variable.

- `"mle"`: Maximum Likelihood Estimation (MLE), using many functions of the **debias** package.
- `"mle-rba"`: Maximum Likelihood Estimation with Reduced Bias Adjustment as popularized by Abernethy based on the median bias of MLE fitted distributions.
- `"mle-unbias"`: Maximum Likelihood Estimation with bias adjustment as popularized by Reliasoft software based on the mean bias of MLE fitted distributions.

Additionally, one can pass any options available from `options.wblr`, such as `col` or `is.plot.legend`. The graphical options will be used when plotting the (life-)time observations using `plot.wblr`. Subsequent calls to `wblr.conf` will inherit these options.

Currently, there is no graceful error recovery after attempting to fit lifetime data including negative time observations, for example `wblr.fit(wblr(-5:10))`.

Value

The function returns its argument object `x`, extended with the calculated fit and the optional graphical and calculation arguments as provided to the function.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Marie Laure Delignette-Muller, Christophe Dutang (2015). "fitdistrplus: An R Package for Fitting Distributions". Journal of Statistical Software, 64(4), 1-34. URL http://www.jstatsoft.org/v64/i04/.

`wblrLoglike` *Log likelihood for Weibull and Lognormal fitted data including intervals*

Description

This function generates the Log likelihood for Wiebull and lognormal distributions from life(-time) data containing fixed time occurrences, and suspensions (progressive right-censored data), discoveries (left-censored data), and interval censored data.

Usage

```
wblrLoglike(par, x, dist="weibull", sign=1, tz=0 )
```

Arguments

par	A vector of parameters taken in the same order as the underlying R distribution. That is, for weibull the order of parameters is (shape, scale). For lognormal (meanlog, sdlog). (Unsuitable parameters cannot be pre-tested, but may cause the function to fail with NaN, or inf values.)
x	A dataframe such as generated by mleframe with column names 'left', 'right' and optionally 'qty'. Exact failure data (occurrences) have same time entered in both 'left' and 'right' columns. Suspension data has last known life[time] entered in 'left' column and -1 entered in 'right' column. The left(early) interval bound for left-censored data must be entered as zero. (NA is not accepted).
dist	A string defining a distribution to be fit. Implemented distributions are "weibull" (default) and "lognormal". (Only 2-parameter models are accepted.)
sign	An integer of 1 or -1 assigning a multiplier for the returned value. A value of -1 results in a minimization function suitable for use as an argument to optim.
tz	A scalar vector with the translation parameter, t0, to be applied to data.

Value

Returns a scalar vector holding the value of log-likelihood. Unsuitable parameters may cause generation of NaN.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

Marie Laure Delignette-Muller, Christophe Dutang (2015). "fitdistrplus: An R Package for Fitting Distributions". Journal of Statistical Software, 64(4), 1-34. URL http://www.jstatsoft.org/v64/i04/.

Examples

```
failures<-c(90,96,30,49,82)
suspensions<-c(100,45,10)
weibull_loglike<-wblrLoglike(c( 2.26, 4900), mleframe(failures,suspensions))
```

---

weibayes *Fitting for Minimal Failure Datasets*

Description

A simplistic fitting method also called "1-parameter Weibull" based on a "prior" known beta value.

Usage

```
weibayes(x, s=NULL, beta)
```

Arguments

x	Either a vector of class "numeric" or "integer" with (life-)time observations, or a dataframe containing at least $time and $event columns and optionally a $qty column.
s	An optional vector of suspension data.
beta	A weibull shape parameter based on prior like kind study.

Details

This function implements a very simplistic relationship defined by two lines of R code:
`t_eta<-(times^beta)/nfail`
`out_val<-sum(t_eta)^(1/beta)`
Note: for a single failure dataset, with beta = 1, the return is simply the sum of the failure time value plus sum of all suspension times. Should no failures exist, then the nfail will be set to 1 for a pessimistic solution.

Value

Returns the Weibull scale, Eta, paramater value.

Examples

```
fail<-5
susp<-rweibull(10, 1, 10)
eta<-weibayes(fail, susp, beta=1)
```

---

weibayes.mle *Fitting for Minimal Failure Datasets using likelihood optimization*

Description

An optimized search method for "1-parameter Weibull" modeling based on a "prior" known beta, or eta value.

Usage

```
weibayes.mle(x, beta=NULL, eta=NULL, incr=1e-7, listout=FALSE)
```

Arguments

x	A dataframe such as generated by mleframe with column names 'left', 'right' and optionally 'qty'. Exact failure data (occurrences) have same time entered in both 'left' and 'right' columns. Suspension data has last known life[time] entered in 'left' column and -1 entered in 'right' column. The left(early) interval bound for left-censored data must be entered as zero. (NA is not accepted).
beta	A weibull shape parameter based on prior like kind study.

eta	A weibull scope parameter based on prior like kind study.
incr	An increment used to establish the slope of the contour at given points.
listout	A logical determining whether details of the optimization loop should be presented in a list output. (For debugging purposes.)

Details

This function searches the contour map to identify the local maximum liklihood on a given axis through the contour. Since this function takes the lrq_frame for data input, it will handle interval data as well as exact time failures and suspensions. This function has been written in pure R, hence, is expected to be too slow for production work. It is intended to be usefull as a study tool.

Value

Returns either the Weibull scale, Eta, paramater value if known beta is provided, or the Weibull shape, Beta, parameter if knonw eta is provided.

Examples

```
fail<-5
susp<-rweibull(10, 1, 10)
eta<-weibayes.mle(mleframe(fail, susp), beta=1)
```

---

xbounds *Extract a bounds dataframe from a* wblr *Object*

Description

Creates a list of bounds and ,if appropriate, t0 modifier from a wblr object.

Usage

```
xbounds(obj, fit_num=1, conf_num)
```

Arguments

obj	An object created by wblr and the result of at least one modifying wblr.fit call.
fit_num	The numbered fit in the wblr object.
conf_num	The numbered confidence calculation in the specified fit of the wblr object.

Value

A list holding the bounds dataframe and, if modified.by.t0, the t0 value as extracted from the wblr object.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Jurgen Symynck, Filip De Bal, Weibull analysis using R, in a nutshell (New Technologies and Products in Machine Manufacturing Technology, Stefan cel Mare University of Suceava, 2010).

Examples

```
obj<-wblr.fit(wblr(rweibull(10,1,1)))
obj<-wblr.conf(obj)
xbounds(obj)
```

---

xfit *Extract a Fit Summary from a* wblr *Object*

Description

Creates a list of distribution and fit elements from a wblr object.

Usage

```
xfit(obj, fit_num=1)
```

Arguments

obj	An object created by wblr and the result of at least one modifying wblr.fit call.
fit_num	The numbered fit in the wblr object.

Value

A list holding dist the distribution name and fit a vector as extracted from the wblr object.

References

William Q. Meeker and Luis A. Escobar, (1998) "Statistical Methods for Reliability Data", Wiley-Interscience, New York

Robert B. Abernethy, (2008) "The New Weibull Handbook, Fifth Edition"

John I. McCool, (2012) "Using the Weibull Distribution: Reliability, Modeling and Inference"

Jurgen Symynck, Filip De Bal, Weibull analysis using R, in a nutshell (New Technologies and Products in Machine Manufacturing Technology, Stefan cel Mare University of Suceava, 2010).

Examples

```
obj<-wblr.fit(wblr(rweibull(10,1,1)))
xfit(obj)
```

Index

∗ **interval**
  mleframe, 19
∗ **likelihood bias**
  hrbu, 11
∗ **likelihood reliability**
  LLln, 12
  LLw, 13
∗ **likelihood**
  MLEcontour, 16
  rba, 38
∗ **median ranks**
  getPercentilePlottingPositions, 9
∗ **regression reliability goodness-of-fit**
  AbPval, 3
  getCCC2, 8
∗ **regression reliability**
  BBB, 4
  getPPP, 10
  lslr, 15
  mlefit, 18
  MLEln2p, 21
  MLEw2p, 23
  MLEw3p, 24
  MRRln2p, 25
  MRRln3p, 26
  MRRw2p, 27
  MRRw3p, 28
  pivotal.rr, 33
  wblrLoglike, 45

AbPval, 3

BBB, 4

contour, *6*
contour.wblr, *3*, 6

FMbounds, 7

getCCC2, 8
getPercentilePlottingPositions, 9
getPPP, *3*, 10

hrbu, 11

legend, *32*, *36*
LLln, 12
LLw, 13
LRbounds, 14
lslr, 15

MLEcontour, 16
mlefit, 18
mleframe, *3*, 19
MLEln2p, 21
MLEln3p, 22
MLEw2p, *3*, 23
MLEw3p, 24
MRRln2p, 25
MRRln3p, *3*, 26
MRRw2p, 27
MRRw3p, 28

options.wblr, 29, *35*

p2y, 32
par, *29–31*, *40*
pivotal.rr, 33
plot.default, *35*
plot.wblr, *3*, *31*, 35
plot_contour, *3*, 37
points, *30*, *40*

rba, 38

wblr, *3*, *29*, *30*, 39, *44*
wblr.conf, *3*, *29*, *30*, *40*, 41
wblr.fit, *3*, *29*, *30*, *40*, 44
wblrLoglike, 45
weibayes, 46
weibayes.mle, 47
WeibullR-package, 2

xbounds, 48
xfit, 49