Package: linLIR (via r-universe)

November 15, 2026

Type Package

Title linear Likelihood-based Imprecise Regression

Version 1.1.9001

Date 2012-11-09

Author Andrea Wiencierz

Maintainer Andrea Wiencierz <Andrea.Wiencierz@stat.uni-muenchen.de>

Description This package implements the methodology of Likelihood-based Imprecise Regression (LIR) for the case of linear regression with interval data. Compatibility updates by Renova.

License GPL (>= 2)

Repository/R-Forge/Project lir

Repository/R-Forge/Revision 25

Repository/R-Forge/DateTimeStamp 2012-11-09 13:13:06

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

Date/Publication 2026-11-15 20:37:39 UTC

RemoteUrl https://github.com/renova-r/linLIR

RemoteRef renova-1.1.9001

RemoteSha 4fa7d126ffcd2efb27f8f53a5d75f784361fe80b

Contents

gen.lms 2
idf 3
kl.ku 5
pm10 6
s.linlir 7
toy.smps 10
undom 10

Index **13**

gen.lms *Finding LRM regression line(s)*

Description

Function within the s.linlir-function that determines the Likelihood-based Region Minimax (LRM) line(s).

Usage

```
gen.lms(dat, p = 0.5, bet, epsilon = 0, k.u = 0)
```

Arguments

dat	An nx4 data.frame containing the imprecise data of the analyzed variables. Columns 1 and 2 correspond to the interval-valued observations of the regressor variable, columns 3 and 4 to those of the dependent variable.
p	Quantile of the abolute residuals' distribution to be used as loss function in the LIR analysis. (0.5 corresponds to the median.)
bet	Cutoff-point for the normalized profile likelihood function.
epsilon	Fraction of coarsening errors considered.
k.u	As default k.u is calculated on the basis of p, bet and epsilon.

Details

The gen.lms-function implements the first part of the exact algorithm for the simple linear LIR analysis with interval data developed in M. Cattaneo, A. Wiencierz (2012c). This first part of the algorithm can be seen as a generalization of the basic algorithm for Least Median of Squares Regression (see, e.g., Steele / Steiger (1986) and Rousseeuw / Leroy (1987)).

Value

A list of two components.

lrm	A vector (or a matrix) of the intercept and slope parameter values of the LRM line(s) and the value of the p-quantile of the absolute residuals associated with the LRM line(s).
b.val	A vector of all slope values considered for the LRM line(s).
dat.unique	A data frame of all unique observations in the analyzed data set.
ind.dat	A matrix of the indices of all combinations of 2 unique data points.

References

M. Cattaneo, A. Wiencierz (2012c). On the implementation of LIR: the case of simple linear regression with interval data. Technical Report No. 127. Department of Statistics. LMU Munich.

A. Wiencierz, M. Cattaneo (2012b). An exact algorithm for Likelihood-based Imprecise Regression in the case of simple linear regression with interval data. In: R. Kruse et al. (Eds.). Advances in Intelligent Systems and Computing. Vol. 190. Springer. pp. 293-301.

M. Cattaneo, A. Wiencierz (2012a). Likelihood-based Imprecise Regression. International Journal of Approximate Reasoning. Vol. 53. pp. 1137-1154.

P. Rousseeuw, A. Leroy (1987). Robust Regression and Outlier Detection. Wiley.

J. Steele, W. Steiger (1986). Algorithms and complexity for least median of squares regression. Discret Appl Math 14. 93-100.

See Also

`s.linlir`, `kl.ku`, `undom.para`

---

idf *Interval data frame*

Description

Create an interval data frame (`idf`-object), summarize its content and visualize subsets of two variables.

Usage

```
idf.create(dat, var.labels = NULL)

## S3 method for class 'idf'
summary(object, ...)
## S3 method for class 'idf'
plot(x, y=NULL, ..., var = NULL, typ="hist", k.x = 1, k.y = 1, inf.margin=10, p.cex=1, col.lev=15, plot.g
```

Arguments

`dat`	A `data.frame` containing 2 neighboring columns for each variable, the first column for the left endpoints of the interval observations, the second for the right endpoints.
`var.labels`	Names of the variables corresponding to the interval-valued observations in the `data.frame`.
`object`	The `idf`-object to be summarized.
`...`	Argument of the generic functions `plot` and `summary`: Other parameters.
`x`	Argument of the generic function `plot`. Here `x` is the `idf`-object to be plotted.
`y`	Argument of the generic function `plot`. Here `y=NULL`.

var	Names of the two variables out of the idf-object to be plotted. (Optional)
typ	Type of the plot. Possible values are "hist": plot 2-dim. histogram (default) and "draft".
k.x	Particular plot function parameter. 1/k.x is the step width along the abscissa.
k.y	Particular plot function parameter. 1/k.y is the step width along the ordinate.
inf.margin	Particular parameter for plot type "draft". inf.margin is the number of steps that the infinite observations are drawn beyond the limits of the plot.
p.cex	Particular parameter for plot type "draft". p.cex is the point size to fill the rectangles with grey color.
col.lev	Particular parameter for plot type "hist" indicating the number of different grey levels in the 2-dim. histogram.
plot.grid	Logical for plot type "hist". If plot.grid=TRUE dashed lines are added to the plot to indicate the location of the interval endpoints. This is particularly useful for categorized data.
x.adj	Horizontal position of the text for the abscissa.
x.padj	Vertical position of the text for the abscissa.
y.las	Orientation of the text for the ordinate. y.las=1 will turn the axis labels and the text in reading direction.
y.adj	y.adj regulates the position of the text for the ordinate in reading direction, i.e. if y.las=0 it sets the vertical position and if y.las=1 the horizontal position.
y.padj	y.padj regulates the position of the text for the ordinate orthogonal to the reading direction, i.e. if y.las=0 it sets the horizontal position and if y.las=1 the vertical position.
x.lim	The limits for the abscissa of the plot.
y.lim	The limits for the ordinate of the plot.
x.lab	Title of the abscissa.
y.lab	Title of the ordinate.

Details

Within the LIR framework all types of interval data are possible, including the particular cases of actually precise data (i.e., lower endpoint = upper endpoint) or missing data (i.e., in case of a real valued variable, lower endpoint = -Inf and upper endpoint = Inf). For the LIR analysis it makes practically no difference if the intervals are closed or not, therefore, the created idf-object does not contain this information.

Value

An idf-object of m variables, which is a list of m+1 entries.

Var1 ... varm	m different data.frames with 2 columns each, one for each of the 1st to mth variables.
n	Number of observations.

References

M. Cattaneo, A. Wiencierz (2012c). On the implementation of LIR: the case of simple linear regression with interval data. Technical Report No. 127. Department of Statistics. LMU Munich.

A. Wiencierz, M. Cattaneo (2012b). An exact algorithm for Likelihood-based Imprecise Regression in the case of simple linear regression with interval data. In: R. Kruse et al. (Eds.). Advances in Intelligent Systems and Computing. Vol. 190. Springer. pp. 293-301.

M. Cattaneo, A. Wiencierz (2012a). Likelihood-based Imprecise Regression. International Journal of Approximate Reasoning. Vol. 53. pp. 1137-1154.

Examples

```
data('toy.smps')
toy.idf <- idf.create(toy.smps, var.labels=c("x","y"))

summary(toy.idf)

plot(toy.idf, typ="draft", k.x=10, k.y=10, p.cex=1.5, y.las=1, y.adj=6)
plot(toy.idf, typ="draft", k.x=10, k.y=10, x.adj=0.7, y.las=1, y.adj=6, y.padj=-3)
plot(toy.idf, k.x=10, k.y=10, x.adj=0.7, x.padj=4, y.adj=0.7, y.padj=-4)

data('pm10')
pm.idf <- idf.create(pm10)

summary(pm.idf)

plot(pm.idf, typ="draft", k.x=10, k.y=20, p.cex=0.35, x.adj=0.5, x.padj=4, y.las=0, y.adj=0.5, y.padj=-4, x.lab="t
```

`kl.ku` *Finding* `k.l` *and* `k.u`

Description

Internal function to determine the quantities `k.l` and `k.u` on the basis of `n`, `p`, `bet`. This function is used within the (internal) functions `gen.lms`, `undom.a` and `undom.para`.

Usage

```
kl.ku(n, p = 0.5, bet, epsilon = 0)
```

Arguments

`n`	Number of observations.
`p`	Quantile of the abolute residuals' distribution to be used as loss function in the LIR analysis. (0.5 corresponds to the median.)
`bet`	Cutoff-point for the normalized profile likelihood function.
`epsilon`	Fraction of coarsening errors considered.

Value

A vector with 2 elements, `k.l` and `k.u`.

References

M. Cattaneo, A. Wiencierz (2012c). On the implementation of LIR: the case of simple linear regression with interval data. Technical Report No. 127. Department of Statistics. LMU Munich.

A. Wiencierz, M. Cattaneo (2012b). An exact algorithm for Likelihood-based Imprecise Regression in the case of simple linear regression with interval data. In: R. Kruse et al. (Eds.). Advances in Intelligent Systems and Computing. Vol. 190. Springer. pp. 293-301.

M. Cattaneo, A. Wiencierz (2012a). Likelihood-based Imprecise Regression. International Journal of Approximate Reasoning. Vol. 53. pp. 1137-1154.

See Also

`s.linlir`, `gen.lms`, `undom.para`

---

pm10 *Simulated data set with 514 interval-valued observations of two variables.*

Description

This data set was analyzed M. Cattaneo, A. Wiencierz (2012c).

Usage

```
data('pm10')
```

Format

A `data.frame` containing 514 rows and 4 columns, i.e. 2 neighboring columns for each variable, the first column for the left endpoints of the interval observations, the second for the right endpoints.

References

M. Cattaneo, A. Wiencierz (2012c). On the implementation of LIR: the case of simple linear regression with interval data. Technical Report No. 127. Department of Statistics. LMU Munich.

A. Wiencierz, M. Cattaneo (2012b). An exact algorithm for Likelihood-based Imprecise Regression in the case of simple linear regression with interval data. In: R. Kruse et al. (Eds.). Advances in Intelligent Systems and Computing. Vol. 190. Springer. pp. 293-301.

M. Cattaneo, A. Wiencierz (2012a). Likelihood-based Imprecise Regression. International Journal of Approximate Reasoning. Vol. 53. pp. 1137-1154.

s.linlir *Simple linear Likelihood-based Imprecise Regression*

Description

Conducts a LIR analysis for 2 variables with interval-valued observations whose relation is assumed to be linear.

Usage

```
s.linlir(dat.idf, var=NULL, p=0.5, bet, epsilon=0, a.grid=100)

## S3 method for class 's.linlir'
print(x, ...)
## S3 method for class 's.linlir'
summary(object, ...)
## S3 method for class 's.linlir'
plot(x, y=NULL, ..., typ, para.typ="polygon", b.grid=500, nb.func=1000, seed.func=NULL, pl.lrm=TRUE, p
```

Arguments

dat.idf	The idf-object to be analyzed.
var	Names of the two variables out of the idf-object to be analyzed.
p	Quantile of the abolute residuals' distribution to be used as loss function in the LIR analysis. (0.5 corresponds to the median.)
bet	Cutoff-point for the normalized profile likelihood function.
epsilon	Fraction of coarsening errors considered.
a.grid	Particular parameter of the internal function undom.para determining the undominated parameter combinations.
x	Argument of the generic functions plot and print. Here x is the s.linlir-object to be plotted or printed.
...	Argument of the generic functions plot, print and summary: Other parameters.
object	The s.linlir-object to be summarized.
y	Argument of the generic function plot. Here y=NULL.
typ	Type of the plot. Possible values are "para": plot undominated parameter set, "lrm": plot LRM regression line(s), "func": plot undominated regression functions.
para.typ	Options for plot of typ="para" are "polygon" (default) or "points" (approximation).
b.grid	Parameter for plot of typ="para" with default option "polygon". b.grid is the number of points over the range of slope values at which the corresponding undominated intercept values are displayed.
nb.func	Number of (randomly chosen) plotted undominated lines for plots of typ="func".

seed.func	Set seed for the random selection of plotted regression lines for plots of typ="func". (Optional)
pl.lrm	Logical for plots of typ=c("para","func"). If pl.lrm=TRUE (default), the LRM regression line(s) is highlighted in the plot.
lrm.col	Color used to highlight the LRM regression lines in the plots of typ=c("para","func") with option pl.lrm=TRUE.
pl.band	Logical for plots of typ="func". If pl.band=TRUE, the band(s) around the LRM regression line(s) is added to the plot.
pl.dat	Logical for plots of typ=c("lrm","func"). If pl.dat=TRUE, the data are plotted in the background of the plot.
pl.dat.typ	Type of the data plot. Possible values are "hist": plot 2-dim. histogram (default) and "draft".
k.x	Particular data plot function parameter. 1/k.x is the step width along the abscissa.
k.y	Particular data plot function parameter. 1/k.y is the step width along the ordinate.
inf.margin	Particular parameter for data plot with pl.dat.typ="draft". inf.margin is the number of steps that the infinite observations are drawn beyond the limits of the plot.
p.cex	Particular parameter for data plot with pl.dat.typ="draft". p.cex is the point size to fill the rectangles with grey color.
col.lev	Particular parameter for data plot with pl.dat.typ="hist" indicating the number of different grey levels in the 2-dim. histogram.
plot.grid	Logical for data plot with pl.dat.typ="hist". If plot.grid=TRUE dashed lines are added to the plot to indicate the location of the interval endpoints. This is particularly useful for categorized data.
x.adj	Horizontal position of the text for the abscissa.
x.padj	Vertical position of the text for the abscissa.
y.las	Orientation of the text for the ordinate. y.las=1 will turn the axis labels and the text in reading direction.
y.adj	y.adj regulates the position of the text for the ordinate in reading direction, i.e. if y.las=0 it sets the vertical position and if y.las=1 the horizontal position.
y.padj	y.padj regulates the position of the text for the ordinate orthogonal to the reading direction, i.e. if y.las=0 it sets the horizontal position and if y.las=1 the vertical position.
x.lim	The limits for the abscissa of the plot.
y.lim	The limits for the ordinate of the plot.
x.lab	Title of the abscissa.
y.lab	Title of the ordinate.

Value

f.lrm	Intercept and slope value(s) of the Likelihood-based Region Minimax (LRM) regression line(s).
q.lrm	Value of the p-quantile of the absolute residuals associated with the LRM regression line(s).
a.undom	Range of intercept values of the undominated regression lines.
b.undom	Range of slope values of the undominated regression lines.
undom.para	A matrix of undominated parameter combinations approximating the entire set of parameters corresponding to the set of undominated regression lines.
config	A list containing information about the settings of the LIR analysis.
dat	An nx4 data.frame containing the imprecise data of the analyzed variables. Columns 1 and 2 correspond to the interval-valued observations of the regressor variable, columns 3 and 4 to those of the dependent variable.
n	Number of observations.
call	Call of the function s.linlir.

References

M. Cattaneo, A. Wiencierz (2012c). On the implementation of LIR: the case of simple linear regression with interval data. Technical Report No. 127. Department of Statistics. LMU Munich.

A. Wiencierz, M. Cattaneo (2012b). An exact algorithm for Likelihood-based Imprecise Regression in the case of simple linear regression with interval data. In: R. Kruse et al. (Eds.). Advances in Intelligent Systems and Computing. Vol. 190. Springer. pp. 293-301.

M. Cattaneo, A. Wiencierz (2012a). Likelihood-based Imprecise Regression. International Journal of Approximate Reasoning. Vol. 53. pp. 1137-1154.

See Also

idf.create, gen.lms, kl.ku, undom.para

Examples

```
data('toy.smps')
toy.idf <- idf.create(toy.smps, var.labels=c("x","y"))

test <- s.linlir(toy.idf, bet=0.5)
test

summary(test)

plot(test, typ="para", x.adj=0.7, y.las=1, y.adj=6, y.padj=-3)
plot(test, typ="func", pl.lrm=FALSE, x.adj=0.7, y.adj=0.7, y.padj=-3)
plot(test, typ="lrm", lrm.col="red", pl.band=TRUE, pl.dat=TRUE, pl.dat.typ="draft",k.x=10, k.y=10, y.las=1, y.adj
```

toy.smps — *Simulated data set containing 17 interval-valued observations of 2 variables.*

Description

This data set was used for the illustrations in A. Wiencierz, M. Cattaneo (2012b).

Usage

```
data('toy.smps')
```

Format

A `data.frame` containing 17 rows and 4 columns, i.e. 2 neighboring columns for each variable, the first column for the left endpoints of the interval observations, the second for the right endpoints.

References

M. Cattaneo, A. Wiencierz (2012c). On the implementation of LIR: the case of simple linear regression with interval data. Technical Report No. 127. Department of Statistics. LMU Munich.

A. Wiencierz, M. Cattaneo (2012b). An exact algorithm for Likelihood-based Imprecise Regression in the case of simple linear regression with interval data. In: R. Kruse et al. (Eds.). Advances in Intelligent Systems and Computing. Vol. 190. Springer. pp. 293-301.

M. Cattaneo, A. Wiencierz (2012a). Likelihood-based Imprecise Regression. International Journal of Approximate Reasoning. Vol. 53. pp. 1137-1154.

undom — *Finding undominated parameters*

Description

Functions within the `s.linlir`-function that determine the parameter combinations corresponding to undominated regression lines. The `undom.a`-function finds the set of undominated intercept values associated with a given slope and the `undom.para`-function finds the set of undominated intercept values associated with a given vector of slope values.

Usage

```
undom.a(dat, b, q.lrm, p = 0.5, bet, epsilon = 0)

undom.para(dat, b.range, a.grid = 100, q.lrm, p = 0.5, bet, epsilon = 0)
```

Arguments

dat	An nx4 data.frame containing the imprecise data of the analyzed variables. Columns 1 and 2 correspond to the interval-valued observations of the regressor variable, columns 3 and 4 to those of the dependent variable.
b	A given value for the slope of a regression line.
q.lrm	Value of the p-quantile of the absolute residuals associated with the LRM line(s).
p	Quantile of the abolute residuals' distribution to be used as loss function in the LIR analysis. (0.5 corresponds to the median.)
bet	Cutoff-point for the normalized profile likelihood function.
epsilon	Fraction of coarsening errors considered.
b.range	Vector of slope values handed over to the function undom.para.
a.grid	Particular parameter of the function undom.para indicating how fine the set of undominated parameter combinations is approximated with respect to the intercept values.

Details

The undom.para-function together with some preparational steps in the s.linlir-function implement the second part of the exact algorithm for the simple linear LIR analysis with interval data developed in M. Cattaneo, A. Wiencierz (2012c).

Value

The undom.a-function returns a list of 2 components:

result1	A 2-column matrix of possibly degenerate intervals for the undominated intercept values associated with the given slope b.
result2	The information of result1 reduced to the fewest intervals possible.

The undom.para-function returns a list of 3 components:

a.undom	Range of intercept values of the undominated regression lines.
b.undom	Range of slope values of the undominated regression lines.
undom.para	A matrix of undominated parameter combinations approximating the entire set of parameters corresponding to the set of undominated regression lines.

References

M. Cattaneo, A. Wiencierz (2012c). On the implementation of LIR: the case of simple linear regression with interval data. Technical Report No. 127. Department of Statistics. LMU Munich.

A. Wiencierz, M. Cattaneo (2012b). An exact algorithm for Likelihood-based Imprecise Regression in the case of simple linear regression with interval data. In: R. Kruse et al. (Eds.). Advances in Intelligent Systems and Computing. Vol. 190. Springer. pp. 293-301.

M. Cattaneo, A. Wiencierz (2012a). Likelihood-based Imprecise Regression. International Journal of Approximate Reasoning. Vol. 53. pp. 1137-1154.

See Also

s.linlir, gen.lms, kl.ku

Index

gen.lms, 2, *6*, *9*, *12*

idf, 3
idf.create, *9*

kl.ku, *3*, 5, *9*, *12*

plot.idf *(*idf*)*, 3
plot.s.linlir *(*s.linlir*)*, 7
pm10, 6
print.s.linlir *(*s.linlir*)*, 7

s.linlir, *3*, *6*, 7, *12*
summary.idf *(*idf*)*, 3
summary.s.linlir *(*s.linlir*)*, 7

toy.smps, 10

undom, 10
undom.para, *3*, *6*, *9*