Package 'robRatio'

September 17, 2025

Type Package

Title M-Estimators for Generalized Ratio and Linear Regression Models

Version 0.1.0

Author Kazumi Wada [aut, cre] (ORCID: <https://orcid.org/0000-0002-9578-1588>)

Maintainer Kazumi Wada <kazwd2008@gmail.com>

Description Robust estimators for generalized ratio model (Wada, Sakashita and Tsubaki, 2021)<doi:10.17713/ajs.v50i1.994> and linear regression model by the IRLS(iterative reweighted least squares) algorithm are contained.

License GPL (>= 3)

URL <https://github.com/kazwd2008/robRatio>

Encoding UTF-8

RoxygenNote 7.3.2

Depends stats

Suggests knitr, MASS, rmarkdown, testthat (>= 3.0.0), latex2exp, RColorBrewer

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2025-09-17 10:20:02 UTC

Contents

`Hirls.aad` *Robust estimator for the linear regression model with Huber's weight function and AAD scal by iteratively re-weighted least squares (IRLS) algorithm*

Description

Robust estimator for the linear regression model with Huber's weight function and AAD scal by iteratively re-weighted least squares (IRLS) algorithm

Usage

```
Hirls.aad(
  x1,
  y1,
  rt = rep(1, length(y1)),
  c1 = 1.15,
  rp.max = 150,
  cg.rt = 0.01
)
```

Arguments

`x1`	explanatory variable(s)
`y1`	objective variable
`rt`	sample weights
`c1`	tuning parameter from 1.15 to 2.30 for the scale parameter of AAD(Average Absolute Deviation)
`rp.max`	maximum number of iteration
`cg.rt`	convergence condition to stop iteration (default: cg1=0.001)

Value

a list with the following elements

`HB` results of robust regression

`wt` robust weights

`rp` total number of iteration

`s1` changes in scale through iterative calculation

`Hirls.mad` *Robust estimator for the linear regression model with Huber's weight function and MAD scal by iteratively re-weighted least squares (IRLS) algorithm*

Description

Robust estimator for the linear regression model with Huber's weight function and MAD scal by iteratively re-weighted least squares (IRLS) algorithm

Usage

```
Hirls.mad(
  x1,
  y1,
  rt = rep(1, length(y1)),
  c1 = 1.44,
  rp.max = 150,
  cg.rt = 0.01
)
```

Arguments

`x1`	explanatory variable(s)
`y1`	objective variable
`rt`	sample weights
`c1`	tuning parameter from 1.44 to 2.88 for the scale parameter of MAD(Median Absolute Deviation)
`rp.max`	maximum number of iteration
`cg.rt`	convergence condition to stop iteration (default: cg1=0.001)

Value

a list with the following elements

`HB` results of robust regression

`wt` robust weights

`rp` total number of iteration

`s1` changes in scale through iterative calculation

robGR *Robust estimator for a generalized ratio model*

Description

This function simultaneously estimates two parameters of the generalized ratio model doi:10.17713/ajs.v50i1.994. It uses Tukey's biweight function and AAD for scale of quasi residuals.

This robGR function simultaneously estimate two parameters of the generalized ratio model. It uses Tukey's biweight function and AAD for scale of quasi residuals.

Usage

```
robGR(x1, y1, g1 = 0, c1 = 8, rp.max = 100, cg.rt = 0.001)

robGR(x1, y1, g1 = 0, c1 = 8, rp.max = 100, cg.rt = 0.001)
```

Arguments

x1	single explanatory variable (a vector)
y1	objective variable to be imputed (a vector)
g1	initial gamma value (default g1=0.5)
c1	tuning constant for Tukey's biweight function. Supposed to choose 4 to 8. Smaller figure is more robust (default tp=8).
rp.max	maximum number of iteration (default: rp.max=50)
cg.rt	convergence condition to stop iteration (default: cg.rt=0.001)

Value

a list with the following elements

par robustly estimated ratio of y1 to x1 (beta)

g1 robustly estimated power (gamma)

res homoscedastic quasi-residuals

wt robust weights

rp total number of iteration

efg error flag. 1: calculation not coverged, 0: successful termination

rt.cg change of par(beta)

g1.cg changes of g1(gamma)

s1.cg changes of the scale(AAD)

a list with the following elements

par robustly estimated ratio of y1 to x1 (beta)

g1 robustly estimated power (gamma)

`res` homoscedastic quasi-residuals

`wt` robust weights

`rp` total number of iteration

`efg` error flag. 1: calculation not coverged, 0: successful termination

`rt.cg` change of par(beta)

`g1.cg` changes of g1(gamma)

`s1.cg` changes of the scale(AAD)

`robRatio` *Robust estimator for ratio models*

Description

This function integrates 4 functions (RrT.aad, RrT.mad, RrH.aad and RrH.mad) for estimating generalized ratio model. Please note that the values for the tuning parameter `tp` allowed in this function is standardized. See the vignette for the detail.

Usage

```
robRatio(
  x1,
  y1,
  gm = "b",
  wf = "T",
  scale = "AAD",
  rt = 1,
  tp = 8,
  rp.max = 100,
  cg.rt = 0.01
)
```

Arguments

`x1`	single explanatory variable (a vector)
`y1`	objective variable to be imputed (a vector)
`gm`	indication of gamma value as follows: gm="a": gamma=1 gm="b": gamma=1/2 (conventional ratio model) gm="c"; gamma=0 (regression model without intercept)
`wf`	weight function (wf=T : Tukey, wf=H : Huber)
`scale`	scale for residuals. "AAD"(default) or "MAD".
`rt`	sample weight (default 1)

tp	standardized tuning parameter. choose 4, 6 or 8. Smaller figure is more robust (default tp=8). See details.
rp.max	maximum number of iteration (default: rp.max=50)
cg.rt	convergence condition to stop iteration (default: cg1=0.001)

Value

a list with the following elements

cond Weight function, scale, and other arguments choosed

par robustly estimated ratio of y1 to x1 (beta)

res homoscedastic quasi-residuals

wt robust weights

rp total number of iteration

s1 changes of the scale (AAD or MAD)

efg error flag. 1: acalculia (all weights become zero) 0: successful termination

Examples

```
require(robRatio)

x1 <- seq(1, 10, by=0.1)
#e <- rnorm(length(x1))
e <- rt(length(x1), df=3)   # error term following t distribution

b <- 2 # true value of slope

y1 <- b*x1 + x1*e # example 1: gamma=1
y2 <- b*x1 + sqrt(x1)*e   # example 2: gamma=1/2

o1 <- robRatio(x1, y1, gm="a")
o2 <- robRatio(x1, y2, gm="b")

o1$par;  o2$par     # estimated slope

cols = RColorBrewer::brewer.pal(11, "PiYG")
cl1 <- round((o1$wt)*10+1)
cl2 <- round((o2$wt)*10+1)

oldpar <- par(mfrow=c(1,2))
plot(x1, y1, col=cols[cl1], pch=20)
plot(x1, y2, col=cols[cl2], pch=20)
par(oldpar)
```

robReg *[Robust estimator for regression models*

Description

This function is for Robust regression by the IRLS algorithm. It integrates child functions contained in Tirls.r and Hirls.r.

Usage

```
robReg(
  x1,
  y1,
  wf = "T",
  scale = "AAD",
  rt = rep(1, length(y1)),
  tp = 8,
  rp.max = 150,
  cg.rt = 0.01
)
```

Arguments

x1	explanatory variable in regression (a vector or a matrix)
y1	objective variable in regression (a vector)
wf	weight function ("T" for Tukey's biweight, and "H" for Huber weight)
scale	scale for residuals. "AAD"(default) or "MAD".
rt	sample weight (default 1)
tp	tuning parameter (tp=4, 6 or 8) for weight function. Smaller figure is more robust.
rp.max	The maximum number of iteration (default 150)
cg.rt	convergence condition to stop iteration (default: cg.rt=0.001)

Value

a list with the following elements

cond Weight function, scale, and other arguments choosed

TK robustly estimated regression coefficients using Tukey's biweight

HB robustly estimated regression coefficients using Huber weight

wt final robust weights

rp total number of iteration

s1 iterative changes in the sclae of residuals (AAD or MAD)

Examples

```
require(robRatio)

set.seed(4)
cov1 <- matrix(c(3, 2.8, 2.8, 3), 2, 2)
cov2 <- matrix(c(2.5, 0, 0, 3), 2, 2)
dat1 <-  MASS::mvrnorm(n=400, mu=c(100, 100), Sigma=cov1, empirical=TRUE)
dat2 <-  cbind(runif(100,  min=96, max=104),  runif(50,  min=95, max=105))
dat3 <- matrix(c(103, 103.5, 104.5, 104.8, 96, 98, 94, 95), 4, 2)
dat <- rbind(dat1, dat2, dat3)
plot(dat)
y1 <- dat[,2]
x1 <- dat[,1]

R0  <- lm(y1~x1)        # regression by OLS

o1 <- robReg(x1, y1, tp=4)  # robust regression by IRLS (more robust)
o2 <- robReg(x1, y1, tp=8)  # robust regression by IRLS (less robust)

oldpar <- par(mfrow=c(2,2))

# non-robust regression
  plot(dat, pch=20, main="non-robust regression")
  abline(R0, col="red", lwd=2)

# robust regression with coloring robust weight
  f.o1 <- rep(1, length(x1))
  f.o1[which(o1$wt < 0.8)] <- 3
  f.o1[which(o1$wt < 0.5)] <- 7
  f.o1[which(o1$wt < 0.2)] <- 2
  f.o1[which(o1$wt == 0)] <- 8

  plot(x1, y1, pch=20, col=f.o1)
  abline(R0, col="red", lty=3)
  abline(o1$TK, col="blue", lwd=2)
  abline(o2$TK, col="cyan", lwd=2)

# robust weights (more robust)
  hist(o1$wt, main="tp=4(more robust)")

# robust weights (less robust)
  hist(o2$wt, main="tp=4(less robust)")

par(oldpar)
```

RrH.aad *Robust estimator for a generalized ratio model with Huber's weight function and AAD scal by iteratively re-weighted least squares (IRLS) algorithm for M-estimation*

Description

Robust estimator for a generalized ratio model with Huber's weight function and AAD scal by iteratively re-weighted least squares (IRLS) algorithm for M-estimation

Usage

```
RrH.aad(x1, y1, g1 = 0.5, c1 = 2.3, rp.max = 100, cg.rt = 0.01)
```

Arguments

x1	single explanatory variable
y1	objective variable
g1	power (default: g1=0.5(conventional ratio model))
c1	tuning parameter usually from 1.15 to 2.30 (smaller figure is more robust)
rp.max	maximum number of iteration
cg.rt	convergence condition to stop iteration (default: cg1=0.001)

Value

a list with the following elements

par	robustly estimated ratio of y1 to x1
res	homoscedastic quasi-residuals
wt	robust weights
rp	total number of iteration
s1	changes in scale through iterative calculation
efg	error flag. 1: acalculia (all weights become zero) 0: successful termination

---

RrH.mad — *Robust estimator for a generalized ratio model with Huber's weight function and MAD scal by iteratively re-weighted least squares (IRLS) algorithm for M-estimation*

Description

Robust estimator for a generalized ratio model with Huber's weight function and MAD scal by iteratively re-weighted least squares (IRLS) algorithm for M-estimation

Usage

```
RrH.mad(x1, y1, g1 = 0.5, c1 = 2.88, rp.max = 100, cg.rt = 0.01)
```

Arguments

x1	single explanatory variable
y1	objective variable
g1	power (default: g1=0.5(conventional ratio model))
c1	tuning parameter usually from 1.44 to 2.88 (equivalent to those for AAD scale)
rp.max	maximum number of iteration
cg.rt	convergence condition to stop iteration (default: cg1=0.001)

Value

a list with the following elements

par	robustly estimated ratio of y1 to x1
res	homoscedastic quasi-residuals
wt	robust weights
rp	total number of iteration
s1	changes in scale through iterative calculation
efg	error flag. 1: acalculia (all weights become zero) 0: successful termination

---

RrT.aad — *Robust estimator for a generalized ratio model with Tukey biweight function and AAD scale by iteratively re-weighted least squares (IRLS) algorithm for M-estimation*

Description

Robust estimator for a generalized ratio model with Tukey biweight function and AAD scale by iteratively re-weighted least squares (IRLS) algorithm for M-estimation

Usage

```
RrT.aad(x1, y1, g1 = 0.5, c1 = 8, rp.max = 100, cg.rt = 0.01)
```

Arguments

x1	single explanatory variable
y1	objective variable
g1	power (default: g1=0.5(conventional ratio model))
c1	tuning parameter usually from 4 to 8 (smaller figure is more robust)
rp.max	maximum number of iteration
cg.rt	convergence condition to stop iteration (default: cg1=0.001)

Value

a list with the following elements

par	robustly estimated ratio of y1 to x1
res	homoscedastic quasi-residuals
wt	robust weights
rp	total number of iteration
s1	changes in scale through iterative calculation
efg	error flag. 1: acalculia (all weights become zero) 0: successful termination

---

RrT.mad — *Robust estimator for a generalized ratio model with Tukey biweight function and MAD scale by iteratively re-weighted least squares (IRLS) algorithm for M-estimation*

Description

Robust estimator for a generalized ratio model with Tukey biweight function and MAD scale by iteratively re-weighted least squares (IRLS) algorithm for M-estimation

Usage

```
RrT.mad(x1, y1, g1 = 0.5, c1 = 10.03, rp.max = 100, cg.rt = 0.01)
```

Arguments

x1	single explanatory variable
y1	objective variable
g1	power (default: g1=0.5(conventional ratio model))
c1	tuning parameter usually from 5.01 to 10.03 (equivalent to those for AAD scale)
rp.max	maximum number of iteration
cg.rt	convergence condition to stop iteration (default: cg1=0.001)

Value

a list with the following elements

par	robustly estimated ratio of y1 to x1
res	homoscedastic quasi-residuals
wt	robust weights
rp	total number of iteration
s1	changes of the scale (AAD or MAD)
efg	error flag. 1: acalculia (all weights become zero) 0: successful termination

Tirls.aad — *Robust estimator for the linear regression model with Tukey's biweight function and AAD scal by iteratively re-weighted least squares (IRLS) algorithm*

Description

Robust estimator for the linear regression model with Tukey's biweight function and AAD scal by iteratively re-weighted least squares (IRLS) algorithm

Usage

```
Tirls.aad(x1, y1, rt = rep(1, length(y1)), c1 = 8, rp.max = 150, cg.rt = 0.01)
```

Arguments

x1	explanatory variable(s)
y1	objective variable
rt	sample weights
c1	tuning parameter from 4 to 8 for the scale parameter of AAD(Average Absolute Deviation)
rp.max	maximum number of iteration
cg.rt	convergence condition to stop iteration (default: cg1=0.001)

Value

a list with the following elements

- TK results of robust regression
- wt robust weights
- rp total number of iteration
- s1 changes in scale through iterative calculation

Tirls.mad — *Robust estimator for the linear regression model with Tukey's biweight function and MAD scale by iteratively re-weighted least squares (IRLS) algorithm*

Description

Robust estimator for the linear regression model with Tukey's biweight function and MAD scale by iteratively re-weighted least squares (IRLS) algorithm

Usage

```
Tirls.mad(
  x1,
  y1,
  rt = rep(1, length(y1)),
  c1 = 10.03,
  rp.max = 150,
  cg.rt = 0.01
)
```

Arguments

x1	explanatory variable(s)
y1	objective variable
rt	sample weights
c1	tuning parameter from 5.01 to 10.03 for the scale parameter of MAD(Median Absolute Deviation)
rp.max	maximum number of iteration
cg.rt	convergence condition to stop iteration (default: cg1=0.001)

Value

a list with the following elements

- TK results of robust regression
- wt robust weights
- rp total number of iteration
- s1 changes in scale through iterative calculation

Index