Package ‘hlr’

February 20, 2015

Type Package
Title Hidden Logistic Regression
Version 0.0-4
Date 2008-09-02
Author Original S-PLUS code by Peter J. Rousseeuw and Andreas Christmann, 
R port by Tobias Verbeke
Maintainer Tobias Verbeke, <tobias.verbeke@gmail.com>
Description Package implementing the methods described in 
Rousseeuw and Christman (2003) to cope with separation issues 
and outliers in logistic regression
Depends MASS
License GPL-3
LazyData yes
NeedsCompilation no
Repository CRAN
Date/Publication 2008-09-03 09:52:06

R topics documented:

MEL ................................................................. 1
WEMEL ............................................................ 3

Index 6

---

MEL

Logistic Regression Using Maximum Estimated Likelihood

Description

Logistic regression using maximum estimated likelihood (MEL) in order to cope with separation issues.
Usage

MEL(x, y, delta = 0.01, epsilon = 1e-06, maxit = 100)
## S3 method for class 'MEL'
print(x, ...)
## S3 method for class 'MEL'
summary(object, ...)
## S3 method for class 'MEL'
plot(x, ...)

Arguments

x          design matrix (n, p) for function MEL; object of class 'MEL' for the print and plot methods
y          response vector
delta      constant
epsilon    precision constant
maxit      maximum number of iterations
object     object of class 'MEL'
...        further arguments to be passed to the methods

Details

The MEL function fits the MEL-model to the data. The print method displays the model coefficients. The summary method displays the model coefficients and displays the names of the components of the MEL output object. The plot function plots the observed response values against the predicted values (using the MEL linear predictor) on link scale. The actual logistic cdf is added as well.

Value

Object of class 'MEL' with following components

MEL        MEL estimates of the coefficients
outMEL     object of class 'glm' corresponding to the final fit

Author(s)

Tobias Verbeke, largely based on original S-PLUS code by Peter J. Rousseeuw and Andreas Christmann

References


Original S-PLUS code available at http://www.stoch.uni-bayreuth.de/de/CHRISTMANN
Examples

```r
par(mfrow = c(2,1))
### Example 1 for function MEL: data set has overlap
set.seed(314)
N <- 500
beta <- matrix(c(2), ncol=1)
x <- matrix(rnorm(N), ncol=1)
eta <- -3 + x
y <- rbinom(nrow(x), 1, plogis(eta))
out <- MEL(x, y)
print(out)
summary(out)
plot(out)
title("overlap")

### Example 2 for function MEL: data set has no overlap
y[eta <= -1] <- 0
y[eta > -1] <- 1
out <- MEL(x, y, delta=0.01)
out
plot(out)
title("no overlap")

### Example 3 for function MEL: all responses are equal to 0
y <- rep(0, nrow(x))
MEL(x, y)

### Example 4 for function MEL: all responses are equal to 1
y <- rep(1, nrow(x))
out <- MEL(x, y)
out
```

Description

Logistic regression using weighted maximum estimated likelihood (WEMEL) in order to cope with separation issues and outliers.

Usage

```r
WEMEL(x, x1, y, delta=0.01, q=0.75, method="MCD", w=rep(1,length(y)), epsilon = 1e-06, maxit = 100)
## S3 method for class 'WEMEL'
print(x, ...)
## S3 method for class 'WEMEL'
summary(object, ...)
## S3 method for class 'WEMEL'
plot(x, which = c(1, 2), ...)
```
WEMEL

Arguments

x design matrix (n, p) for function WEMEL; object of class 'WEMEL' for the print and plot methods
x1 (sub-)matrix of the design matrix. The robust weights are computed w.r.t. to x1. E.g. x1 contains only continuous variables.
y response vector
delta constant
delta quantile used for MCD and for the robust weights; defaults to 0.75
method method to define weights; one of "MCD" (default) or "PCA"
w input vector of weights
epsilon precision constant for the algorithm, default: 1.E-6
maxit maximum number of iterations for the algorithm; defaults to 100
object object of class 'WEMEL'
which which plot should be plotted? An index plot of the robust weights (which=1), the observed response values against the predicted values (which=2) or both (which=c(1, 2), the default).
... further arguments to be passed to the methods

Details

The WEMEL function fits the WEMEL-model to the data. The print method displays the model coefficients. The summary method displays the model coefficients and displays the names of the components of the WEMEL output object. The plot function plots either the index plot of the robust weights (which=1) or the observed response values against the predicted values (using the WEMEL linear predictor) on link scale with a logistic cdf overplotted (which=2). The default value (which=c(1, 2)) plots both.

Value

Object of class 'WEMEL' with following components

WEMEL WEMEL estimates of the coefficients
outWEMEL object of class 'glm' corresponding to the final fit

Author(s)

Tobias Verbeke, largely based on original S-PLUS code by Peter J. Rousseeuw and Andreas Christmann

References


Original S-PLUS code available at http://www.stoch.uni-bayreuth.de/de/CHRISTMANN
Examples

### Example 1 for function WEMEL: data set has overlap

```r
set.seed(314)
n <- 500
beta <- matrix(c(3), ncol=1)
x <- matrix(rnorm(n), ncol=1)
eta <- -2 + x
y <- rbinom(nrow(x), 1, plogis(eta))
out <- WEMEL(x, x, y)
print(out)
summary(out)
plot(out)
```

### Example 2 for function WEMEL: 2 bad leverage points

```r
x[499:500] <- c(-10, 10)
y[499:500] <- c(1, 0)
out <- WEMEL(x, x, y, delta=0.01)
out
plot(out)
```

### Example 3 for function WEMEL: data set has no overlap

```r
eta <- -2 + x
y[eta <= -1] <- 0
y[eta > -1] <- 1
out <- WEMEL(x, x, y, delta=0.01)
out
plot(out)
```
Index

*Topic models
  MEL, 1
  WEMEL, 3

MEL, 1
plot.MEL (MEL), 1
plot.WEMEL (WEMEL), 3
print.MEL (MEL), 1
print.WEMEL (WEMEL), 3

summary.MEL (MEL), 1
summary.WEMEL (WEMEL), 3

WEMEL, 3