Package ‘sprm’

February 22, 2016

Type Package
Title Sparse and Non-Sparse Partial Robust M Regression and Classification
Version 1.2.2
Date 2016-02-22
Author Sven Sermeels (BASF Corp) and Irene Hoffmann
Maintainer Irene Hoffmann <irene.hoffmann@tuwien.ac.at>
Description Robust dimension reduction methods for regression and discriminant analysis are implemented that yield estimates with a partial least squares alike interpretability. Partial robust M regression (PRM) is robust to both vertical outliers and leverage points. Sparse partial robust M regression (SPRM) is a related robust method with sparse coefficient estimate, and therefore with intrinsic variable selection. For binary classification related discriminant methods are PRM-DA and SPRM-DA.
License GPL (>= 3)
Depends ggplot2 (>= 2.0.0)
Imports cvTools, graphics, grDevices, grid, pcaPP, reshape2, robustbase, stats
NeedsCompilation no
Repository CRAN
Date/Publication 2016-02-22 14:33:44

R topics documented:

  sprm-package ............................................. 2
  biplot.prm ............................................. 3
  biplot.prmda .......................................... 5
  biplot.sprm ........................................... 6
  biplot.sprmda .......................................... 7
  plot.prm ............................................... 8
  plot.sprm ............................................. 10
  predict.prm ........................................... 12
  predict.prmda ......................................... 13
Description

Robust dimension reduction methods for regression and discriminant analysis are implemented that yield estimates with a partial least squares alike interpretability. Partial robust M regression (PRM) is robust to both vertical outliers and leverage points. Sparse partial robust M regression (SPRM) is a related robust method with sparse coefficient estimate, and therefore with intrinsic variable selection. For binary classification related discriminant methods are PRM-DA and SPRM-DA.

Details

| Package: | sprm |
| Type:    | Package |
| Version: | 1.1 |
| Date:    | 2014-12-10 |
| License: | GPL(>=3) |

The main functions in this package are `prms` and `sprms` for non-sparses and sparse partial robust M regression, respectively, and `prmda` and `sprmda` for non-sparses and sparse partial robust M discriminant analysis. Further cross validation procedures for tuning parameter selection are implemented in `prmsCV`, `sprmsCV`, `prmdaCV` and `sprmdaCV`. Various plot options are available to visualize the results.

Author(s)

Sven Serneels, BASF Corp and Irene Hoffmann
biplot.prm

References


See Also

prms, sprms, prmda, sprmda

Examples

set.seed(50235)
U1 <- c(rep(3,20), rep(4,30))
U2 <- rep(3.5,50)
X1 <- replicate(5, U1+rnorm(50))
X2 <- replicate(20, U2+rnorm(50))
X <- cbind(X1,X2)
beta <- c(rep(1, 5), rep(0,20))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
ed$y <- y
mod <- prms(y~., data=d, a=2, fun="Hampel")
smod <- sprms(y~., data=d, a=2, eta=0.5, fun="Hampel")

biplot(mod)
biplot(smod)

biplot.prm

Biplot for prm objects

Description

This biplot for prm objects visualizes the original variables which contribute to the model and their impact on the latent components as well as the position of the observations in the transformed space. The data is projected onto two of the latent components.

Usage

## S3 method for class 'prm'
biplot(x, comps = c(1, 2),
       colors = list(scores = "#0000AA", loadings = "red", background = "#BBBBEE"),
       textsize = 6, arrowshapes = c(25, 0.03), labelpos=0.35, ...)
Arguments

x        object of class prm.
comps    vector with two integers, referring to the components to be plotted.
colors   list of three elements named scores, loadings and background with color
         codes or names of colors.
textsize the text size in which to print the scores and loading names.
arrowsizes vector of length two containing the angle of the arrowheads and their relative
         length in npc.
labpos   numeric value; determines distance of the arrow label to the arrowhead.
...      further arguments. Currently not used.

Author(s)

Sven Serneels, BASF Corp.

References


See Also

plot.prm, prms, biplot.sprm

Examples

set.seed(5023)
U <- c(rep(3,20), rep(4,30))
X <- replicate(6, U+rnorm(50))
beta <- c(rep(1, 3), rep(-1,3))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
mod <- prms(y~, data=d, a=2, fun="Hampel")
biplot(mod, comps = c(1, 2))
**Description**

This biplot for prmda objects visualizes the original variables which contribute to the model and their impact on the latent components as well as the position of the observations in the transformed space. The data is projected onto two of the latent components and colored according to class membership.

**Usage**

```r
## S3 method for class 'prmda'
biplot(x, comps = c(1, 2),
       colors = list(scores1 = "orange", scores2 = "darkgreen", loadings = "red", background = "gray84"),
       textsize = 6, arrowshapes = c(25, 0.03), labelpos = 0.35, ...)
```

**Arguments**

- **x**: object of class prmda.
- **comps**: vector with two integers, referring to the components to be plotted.
- **colors**: list of four elements named scores1 (for observations of group coded with 1), scores2 (for observations of group coded with -1), loadings and background with color codes or names of colors.
- **textsize**: the text size in which to print the scores and loading names.
- **arrowshapes**: vector of length two containing the angle of the arrowheads and their relative length innpc.
- **labelpos**: numeric value; determines distance of the arrow label to the arrowhead.
- **...**: further arguments. Currently not used.

**Author(s)**

Irene Hoffmann

**References**


**See Also**

*prmda*
## Examples

```r
data(iris)
data <- droplevels(subset(iris, iris$Species!="setosa"))
mod <- prmda(Species~., data, a=2, class="lda")
bipplot(mod)
```

## Description

This biplot for sprm objects visualizes the original variables which contribute to the model and their impact on the latent components as well as the position of the observations in the transformed space. The data is projected onto two of the latent components.

## Usage

```r
## S3 method for class 'sprm'
bipplot(x, comps = c(1, 2),
  colors = list(scores = "#0000AA", loadings = "red", background = "#BBBBEE"),
  textsize = 6, arrowshapes = c(25, 0.03), labelpos=0.35, ...)```

## Arguments

- `x`: object of class sprm.
- `comps`: vector with two integers, referring to the components to be plotted.
- `colors`: list of three elements named scores, loadings and background with color codes or names of colors.
- `textsize`: the text size in which to print the scores and loading names.
- `arrowshapes`: vector of length two containing the angle of the arrowheads and their relative length in npc.
- `labelpos`: numeric value; determines distance of the arrow label to the arrowhead.
- `...`: further arguments. Currently not used.

## Details

The sparsity of the biplot is inherited by the sparsity of the model. Only the contributing variables are included in the plot, which can lead to better visualization and easier interpretation.

## Author(s)

Sven Serneels, BASF Corp.
References


See Also

plot.sprm, sprm, biplot.prm

Examples

```r
set.seed(5023)
U1 <- c(rep(3,20), rep(4,30))
U2 <- rep(3.5,50)
X1 <- replicate(5, U1+rnorm(50))
X2 <- replicate(20, U2+rnorm(50))
X <- cbind(X1,X2)
beta <- c(rep(1, 5), rep(0,20))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
mod <- sprms(y~., data=d, a=2, eta=0.5, fun="Hampel")
biplopt(mod, comps = c(1, 2))
```

biplot.sprmda

**Biplot for sprmda objects of Sparse PRM discriminant analysis**

Description

This biplot for sprmda objects visualizes the original variables which contribute to the model and their impact on the latent components as well as the position of the observations in the transformed space. The data is projected onto two of the latent components and colored according to class membership.

Usage

```r
## S3 method for class 'sprmda'
biplot(x, comps = c(1, 2),
   colors = list(scores1="orange", scores2="darkgreen", loadings="red", background="gray84"),
   textsize = 6, arrowshapes = c(25, 0.03), labellpos = 0.35, ...
)```
Arguments

- **x**: object of class sprmda.
- **comps**: vector with two integers, referring to the components to be plotted.
- **colors**: list of four elements named `scores1` (for observations of group coded with 1), `scores2` (for observations of group coded with -1), `loadings` and `background` with color codes or names of colors.
- **textsize**: the text size in which to print the scores and loading names.
- **arrowshapes**: vector of length two containing the angle of the arrowheads and their relative length in npc.
- **labellpos**: numeric value; determines distance of the arrow label to the arrowhead.
- **...**: further arguments. Currently not used.

Details

The sparsity of the biplot is inherited by the sparsity of the model. Only the contributing variables are included in the plot, which can lead to better visualization and easier interpretation.

Author(s)

Irene Hoffmann

References


See Also

sprmda

Examples

data(iris)
data <- droplevels(subset(iris, iris$Species!="setosa"))
smod <- sprmda(Species~., data, a=2, eta=0.7, class="lda")
biplot(smod)
Usage

```r
## S3 method for class 'prm'
plot(x, type = "yp", alpha = 0.025, colors = list(bars = "#0000AA",
       errorbars = "red", background = "#BBBBEE", abline = "#21A0D2",
       scores = "#0000AA", cutoffs = "#00EEEE", badouts = "darkred",
       modouts = "black"), textsize = 6,
       errorbar_width = 1, data, yscale = NULL, ...)
```

Arguments

- `x` object of class `prm`.
- `type` choices are "yp", "weights", "coefficients", "dd" (see Details).
- `alpha` significance level. Default is 0.025. Will be ignored if `type="weights"`.
- `colors` list with six elements with color codes or names for bar, errorbars, background, abline, scores, cutoffs, badouts (outliers with weight zero) and modouts (moderate outliers).
- `textsize` the text size in which to print the scores and loading names. Will be ignored if `type` is "weights" or "coefficients".
- `errorbar_width` a numeric containing the width of the error bars for `type="yp"`.
- `data` optional data frame, containing new cases to predict and plot for `type="yp"` and `type="dd"`.
- `yscale` optional scale vector for the yscale in the y vs y predicted plot (e.g. if two different regression plots have to be on the same scale).
- `...` further arguments. Currently not used.

Details

The choices for `type` are:
- `type="yp"` - y vs y predicted plot with confidence intervals for each observation.
- `type="weights"` - plot of case weights used for robust weighted regression.
- `type="coefficients"` - plot of the value of each coefficient estimate with confidence interval.
- `type="dd"` - distance-distance plot for visualization of leverage points. Robust distances are plotted against Mahalanobis distances.

Author(s)

Sven Serneels, BASF Corp.

References


**Describe**

Four types of plot options are available:

- y vs y predicted plot,
- plot of case weights used for robust weighted regression,
- plot of estimated coefficients,
- distance-distance plot.

**Usage**

```r
## S3 method for class 'sprm'
plot(x, type = "yyp", alpha = 0.025, colors = list(bars = "#0000AA", errorbars = "red", background = "#BBBEEE", abline = "#21A0D2", scores = "#0000AA", cutoffs = "#00EEE", badouts="darkred", modouts="black"), textsize = 6, errorbar_width = 1, data, yscale = NULL, ...)
```

**Arguments**

- `x` object of class sprm.
- `type` choices are "yyp", "weights", "coefficients", "dd" (see Details).
- `alpha` significance level. Default is 0.025. Will be ignored if type="weights".
- `colors` list with six elements with color codes or names for bar, errorbars, background, abline, scores, cutoffs, badouts (outliers with weight zero) and modouts (moderate outliers).
- `textsize` the text size in which to print the scores and loading names. Will be ignored if type is "weights" or "coefficients".

**Examples**

```r
set.seed(5023)
U <- c(rep(2,20), rep(5,30))
X <- replicate(6, U+rnorm(50))
beta <- c(rep(1, 3), rep(-1,3))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
mod <- prms(y~., data=d, a=2, fun="Hampel")

plot.prms(mod, type="yyp", errorbar_width=0.001)
plot(mod, type="coefficients", colors = list(bars = "darkgreen", errorbars = "red", background = "lightgray"))
plot(mod, type="weights")
```
errorbar_width  a numeric containing the width of the error bars for type="yp".
data    optional data frame, containing new cases to predict and plot for type="yp" and type="dd".
yscale  optional scale vector for the yscale in the y vs y predicted plot (e.g. if two different regression plots have to be on the same scale)
...

Details

The choices for type are:
type="yp" - y vs y predicted plot with confidence intervals for each observation.
type="weights" - plot of case weights used for robust weighted regression.
type="coefficients" - plot of the value of each coefficient estimate with confidence interval.
type="dd" - distance-distance plot for for visualization of leverage points. Robust distances are plotted against Mahalanobis distances.

Author(s)

Sven Serneels, BASF Corp.

References


See Also

sprm, biplot.sprm

Examples

set.seed(5023)
U1 <- c(rep(2,20), rep(5,30))
U2 <- rep(3.5,50)
X1 <- replicate(5, U1+rnorm(50))
X2 <- replicate(20, U2+rnorm(50))
X <- cbind(X1,X2)
alpha <- c(rep(1, 5), rep(0,20))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%alpha + e
d <- as.data.frame(X)
dy <- y
smod <- sprm(y~., data=d, a=1, eta=0.5, fun="Hampel")
mod <- prms(y~., data=d, a=1, fun="Hampel")
plot(smod, type="yp", errorbar_width=0.001)
plot(smod, type="coefficients")
plot(mod, type="coefficients")
plot(smod, type="weights")
plot(smod, type="dd", colors=list(background="lightgray", scores="darkblue", cutoffs="red"))

---

predictprm

Predict method for models of class prm

Description

Predictions from a partial robust M regression model.

Usage

## S3 method for class 'prm'
predict(object, newdata, ...)

Arguments

- **object**: object of class prm.
- **newdata**: optional data frame with new observations.
- **...**: further arguments. Currently not used.

Details

If newdata is specified the prm model is used to predict the fitted values for this data set, otherwise the fitted values of the model are returned.

Value

predictprm returns a vector of the predicted response.

Author(s)

Sven Serneels, BASF Corp and Irene Hoffmann

References


predict.prmda

See Also

prms, prmsCV

Examples

```r
set.seed(5023)
U <- c(rep(2,20), rep(5,30))
X <- replicate(6, U+rnorm(50))
beta <- c(rep(1, 3), rep(-1,3))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
mod <- prms(y~., data=d, a=2, fun="Hampel")

dnew <- as.data.frame(replicate(6, U+rnorm(10)))
ynewp <- predict(mod, newdata=dnew)
```

predict.prmda

**Predict method for models of class prmda**

**Description**

Predictions for a PRM-DA classification model.

**Usage**

```r
## S3 method for class 'prmda'
predict(object, newdata, ...)
```

**Arguments**

- `object`: object of class prmda.
- `newdata`: optional data frame with new observations.
- `...`: further arguments. Currently not used.

**Details**

If `newdata` is specified the PRM-DA classification model is used to predict the fitted values for this data set, otherwise the fitted values of the model are returned.

**Value**

`predict.prmda` returns a vector of the predicted class membership.

**Author(s)**

Irene Hoffmann
References


See Also

prmda, prmdacv

Examples

data(iris)
data <- droplevels(subset(iris, iris$Species!="setosa"))
mod <- prmda(Species~., data, a=2, class="lda")
table(data$Species, predict(mod))

predict.sprm  Predict method for models of class sprm

Description

Predictions from a sparse partial robust M regression model.

Usage

## S3 method for class 'sprm'
predict(object, newdata, ...)

Arguments

object  object of class sprm.
newdata  optional data frame with new observations.
...  further arguments. Currently not used.

Details

If newdata is specified the sprm model is used to predict the fitted values for this data set, otherwise the fitted values of the model are returned.

Value

predict.sprm returns a vector of the predicted response.

Author(s)

Sven Serneels, BASF Corp and Irene Hoffmann
predict.sprmda

References


See Also

sprms, sprmsCV

Examples

```r
set.seed(5023)
U1 <- c(rep(2,20), rep(5,30))
U2 <- rep(3.5,50)
X1 <- replicate(5, U1+rnorm(50))
X2 <- replicate(20, U2+rnorm(50))
X <- cbind(X1,X2)
beta <- c(rep(1, 5), rep(0,20))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
smod <- sprms(y~., data=d, a=1, eta=0.5, fun="Hampel")

dnew <- as.data.frame(cbind(replicate(5, U1+rnorm(10)), replicate(20, U2+rnorm(10))))
ynewp <- predict(smod, newdata=dnew)
```

predict.sprmda

**Predict method for models of class sprmda**

Description

Predictions from SPRM-DA classification model.

Usage

```r
## S3 method for class 'sprmda'
predict(object, newdata, ...)  
```

Arguments

- `object` : object of class sprmda.
- `newdata` : optional data frame with new observations.
- `...` : further arguments. Currently not used.
prmda

Details

If `newdata` is specified the SPRM-DA model is used to predict the fitted values for this data set, otherwise the fitted values of the model are returned.

Value

`predict.sprmda` returns a vector of the predicted class membership.

Author(s)

Irene Hoffmann

References


See Also

`sprmda`, `sprmdacv`

Examples

data(iris)
data <- droplevels(subset(iris, iris$Species != "setosa"))smod <- sprmda(Species ~ ., data, a = 2, eta = 0.5, class = "lda")table(data$Species, predict(smod))

---

prmda

Robust PLS for binary classification

Description

Robust PLS and discriminant analysis for binary classification problems. This method for dimension reduction and discriminant analysis yields a classification model with a partial least squares alike interpretability that is robust to both vertical outliers and leverage points.

Usage

```r
prmda(formula, data, a, fun = "Hampel", probp1 = 0.95, hampelp2 = 0.975,
      hampelp3 = 0.999, probp4 = 0.01, yweights = TRUE,
      class = c("regfit", "lda"), prior = c(0.5, 0.5),
      center = "median", scale = "qn",
      numit = 100, prec = 0.01)
```
Arguments

formula a formula, e.g. group ~ X1 + X2 with group a factor with two levels or a numeric vector coding class membership with 1 and -1 and X1,X2 numeric variables.
data a data frame or list which contains the variables given in formula. The response specified in the formula needs to be a numeric vector coding the class membership with 1 and -1 or a vector of factors with two levels.
a the number of PRM components to be estimated in the model.
fun an internal weighting function for case weights. Choices are "Hampel" (preferred), "Huber" or "Fair".
probp1 a quantile close to 1 at which to set the first outlier cutoff for the weighting function.
hampelp2 a quantile close to 1 with probp1<hampelp2 for second cutoff. Only applies to fun="Hampel".
hampelp3 a quantile close to 1 with probp1<hampelp2<hampelp3 for third cutoff. Only applies to fun="Hampel".
probp4 a quantile close to zero for the cutoff for potentially wrong class labels (see Reference). Ignored if yweights=FALSE.
yweights logical; if TRUE weights are calculated for observations with potentially wrong class labels.
class type of classification; choices are "regfit" or "lda" (see Detail). If "regfit" an object of class prm is returned.
prior vector of length 2 with prior probabilities of the groups; only used if class="lda".
center type of centering of the data in form of a string that matches an R function, e.g. "mean" or "median".
scale type of scaling for the data in form of a string that matches an R function, e.g. "sd" or "qn" or alternatively "no" for no scaling.
numit the number of maximal iterations for the convergence of the case weights.
prec a value for the precision of the convergence of the case weights.

Details

For class="lda" a robust LDA model is estimated in the PRM score space for class="regfit" the model ist a robust PLS regression model on the binary response.

Value

prmda returns an object of class prmda.
Functions summary, predict and biplot are available. Also the generic functions coefficients, fitted.values and residuals can be used to extract the corresponding elements from the sprmda object.
scores the matrix of scores.
R Direction vectors (or weighting vectors or rotation matrix) to obtain the scores.

scores=xs%*%R.
loadings the matrix of loadings.

\( w \) the overall case weights used for robust dimension reduction and classification (depending on the weight function). \( w = \sqrt{w_y \times wt} \).

\( wt \) the group wise obtained case weights in the score space.

\( w_y \) the case weights for potentially mislabeled observations.

Results from LDA model:

\( \text{ldamod} \) list with robust pooled within-group covariance (cov) and the two robust group centers (m1, m2) in the score space.

\( \text{ldafit} \) posterior probabilities from robust LDA in the score space.

\( \text{ldaclass} \) predicted class labels from robust LDA in the score space.

Results from the regression model with binary response:

\( \text{coefficients} \) vector of coefficients of the weighted regression model.

\( \text{intercept} \) intercept of weighted regression model.

\( \text{residuals} \) vector of residuals, true response minus estimated response.

\( \text{fitted.values} \) the vector of estimated response values.

\( \text{coefficients.scaled} \) vector of coefficients of the weighted regression model with scaled data.

\( \text{intercept.scaled} \) intercept of weighted regression model with scaled data.

Data preprocessing:

\( \text{yMeans} \) value used internally to center response.

\( \text{xMean} \) vector used internally to center data.

\( \text{xScales} \) vector used internally to scale data.

\( \text{yScales} \) value used internally to scale response.

\( \text{inputs} \) list of inputs: parameters, data and scaled data.

Author(s)

Irene Hoffmann and Sven Serneels

References


See Also

\texttt{prmdaCV}

Examples

\begin{verbatim}
data(iris)
data <- droplevels(subset(iris,iris$Species!="setosa"))
mod <- prmda(Species~,data, a=2, class="lda")
\end{verbatim}
prmdaCV

Cross validation method for PRM classification models.

Description

k-fold cross validation for the selection of the number of components for PRM classification.

Usage

prmdaCV(formula, data, as, nfold = 10, fun = "Hampel", probp1 = 0.95, hampelp2 = 0.975, hampelp3 = 0.999, probp4 = 0.01, yweights = TRUE, class = c("regfit", "lda"), prior = c(0.5, 0.5), center = "median", scale = "qn", plot = TRUE, numit = 100, prec = 0.01)

Arguments

formula a formula, e.g. group ~ X1 + X2 with group a factor with two levels and X1,X2 numeric variables.

data a data frame or list which contains the variables given in formula. The response specified in the formula needs to be a numeric vector coding the class membership with 1 and-1 or a vector of factors with two levels.

as a vector with positive integers, which are the number of PRM components to be estimated in the models.

nfold the number of folds used for cross validation, default is nfold=10 for 10-fold CV.

fun an internal weighting function for case weights. Choices are "Hampel" (preferred), "Huber" or "Fair".

probp1 the 1-alpha value at which to set the first outlier cutoff for the weighting function.

hampelp2 the 1-alpha values for second cutoff. Only applies to fun="Hampel".

hampelp3 the 1-alpha values for third cutoff. Only applies to fun="Hampel".

probp4 a quantile close to zero for the cutoff for potentially wrong class labels (see Reference). Ignored if yweights=FALSE.

yweights logical; if TRUE weights are calculated for observations with potentially wrong class labels.

class type of classification; choices are "regfit" or "lda". If "regfit" an object of class prm is returned.

prior vector of length 2 with prior probabilities of the groups; only used if class="lda".

center type of centering of the data in form of a string that matches an R function, e.g. "mean" or "median".

scale type of scaling for the data in form of a string that matches an R function, e.g. "sd" or "qn" or alternatively "no" for no scaling.
prms

plot logical, default is TRUE. If TRUE a plot is generated with a mean weighted misclassification rate for each model (see Details).

numit the number of maximal iterations for the convergence of the coefficient estimates.

prec a value for the precision of estimation of the coefficients.

Details

The robust cross validation criterion is a weighted misclassification rate. Class assignment of outliers is unreliable. Therefore, the case weights from the model are used to downweight the influence observations which were detected as outliers on the misclassification rate.

Value

opt.mod object of class prmda. (see prmda)

pcm matrix with predicted class membership for each observation and each number of components.

Author(s)

Irene Hoffmann

References


See Also

prmda, biplot.prmda, predict.prmda, sprmdacv

Examples

data(iris)
data <- droplevels(subset(iris, iris$Species!="setosa"))mod <- prmdacv(Species~., data, as=1:2, class="lda", numit=10, prec=0.1)biplot(mod$opt.mod)

---

Partial robust M regression

Description

Partial robust M regression for models with univariate response. This method for dimension reduction and regression analysis yields estimates with a partial least squares alike interpretability that are robust to both vertical outliers and leverage points.
Usage

prms(formula, data, a, fun = "Hampel", probp1 = 0.95, hampelp2 = 0.975, hampelp3 = 0.999, center = "median", scale = "qn", usesvd = FALSE, numit = 100, prec = 0.01)

Arguments

formula an object of class formula.
data a data frame or list which contains the variables given in formula.a the number of PRMS components to be estimated in the model.fun an internal weighting function for case weights. Choices are "Hampel" (preferred), "Huber" or "Fair".probp1 the 1-alpha value at which to set the first outlier cutoff for the weighting function.hampelp2 the 1-alpha values for second cutoff. Only applies to fun="Hampel".hampelp3 the 1-alpha values for third cutoff. Only applies to fun="Hampel".center type of centering of the data in form of a string that matches an R function, e.g. "mean" or "median".scale type of scaling for the data in form of a string that matches an R function, e.g. "sd" or "qn" or alternatively "no" for no scaling.usesvd logical, default is FALSE. If TRUE singular value decomposition is performed.numit the number of maximal iterations for the convergence of the coefficient estimates.prec a value for the precision of estimation of the coefficients.

Details

The NIPLS algorithm combined with weighted regression is used for the model estimation.
a is the number of components in the model. Note that it is not possible to simply reduce the number of weighting vectors to obtain a model with a smaller number of components. Each model has to be estimated separately due to its dependence on robust case weights.

Value

prms returns an object of class prm.

Functions summary, predict and plot are available. Also the generic functions coefficients, fitted.values and residuals can be used to extract the corresponding elements from the prm object.

coefficients vector of coefficients of the weighted regression model.
intercept intercept of weighted regression model.
wy the case weights in the y space.
wt the case weights in the score space.
the overall case weights used for weighted regression (depending on the weight function). \( w = w_y \times w_t \).

scores

the matrix of scores.

\( R \)

Direction vectors (or weighting vectors or rotation matrix) to obtain the scores.

\( \text{scores}=Xs\times R \).

loadings

the matrix of loadings.

fitted.values

the vector of estimated response values.

residuals

vector of residuals, true response minus estimated response.

coefficients.scaled

vector of coefficients of the weighted regression model with scaled data.

intercept.scaled

intercept of weighted regression model with scaled data.

\( \text{ymeans} \)

value used internally to center response.

\( \text{xmeans} \)

vector used internally to center data.

Xscales

vector used internally to scale data.

Yscales

value used internally to scale response.

Yvar

percentage of contribution for each component to the explanation of the variance of the response.

Xvar

percentage of contribution for each component to the explanation of the variance of the variables.

inputs

list of inputs: parameters, data and scaled data.

Author(s)

Sven Serneels, BASF Corp and Irene Hoffmann

References


See Also

prmsCV, plot.prm, biplot.prm, predict.prm, sprms

Examples

\[
\text{set.seed(5023)} \\
U <- c(rep(2,20), rep(5,30)) \\
X <- replicate(6, U+rnorm(50)) \\
\beta <- c(rep(1, 3), rep(-1,3)) \\
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1)) \\
y <- X%*%\beta + e
\]
```r
prmsCV <- as.data.frame(x)
d$y <- y
mod <- prms(y~, data=d, a=2, fun="Hampel")
summary(mod)
```

**prmsCV**  
Cross validation method for PRM regression models.

**Description**  
k-fold cross validation for the selection of the number of components for partial robust M regression.

**Usage**  
```r
prmsCV(formula, data, as, nfold = 10, fun = "Hampel", probp1 = 0.95, hampelp2 = 0.975, hampelp3 = 0.999, center = "median", scale = "qn", usesvd = FALSE, plot = TRUE, numit = 100, prec = 0.01, alpha = 0.15)
```

**Arguments**  

- `formula`: an object of class formula.
- `data`: a data frame or list which contains the variables given in formula.
- `as`: a vector with positive integers, which are the number of PRM components to be estimated in the models.
- `nfold`: the number of folds used for cross validation, default is `nfold=10` for 10-fold CV.
- `fun`: an internal weighting function for case weights. Choices are "Hampel" (preferred), "Huber" or "Fair".
- `probp1`: the 1-alpha value at which to set the first outlier cutoff for the weighting function.
- `hampelp2`: the 1-alpha values for second cutoff. Only applies to `fun="Hampel"`.
- `hampelp3`: the 1-alpha values for third cutoff. Only applies to `fun="Hampel"`.
- `center`: type of centering of the data in form of a string that matches an R function, e.g. "mean" or "median".
- `scale`: type of scaling for the data in form of a string that matches an R function, e.g. "sd" or "qn" or alternatively "no" for no scaling.
- `usesvd`: logical, default is FALSE. If TRUE singular value decomposition is performed.
- `plot`: logical, default is TRUE. If TRUE a plot is generated with a measure of the prediction accuracy for each model (see Details).
- `numit`: the number of maximal iterations for the convergence of the coefficient estimates.
- `prec`: a value for the precision of estimation of the coefficients.
- `alpha`: value used for alpha trimmed mean squared error, which is the cross validation criterion (see Details).
Details

The alpha - trimmed mean squared error of the predicted response over all observations is used as robust decision criterion to choose the optimal model. For plot=TRUE a graphic visualizes the alpha - trimmed mean squared error for each model.

Value

- opt.mod: object of class prm. (see prms)
- spe: matrix with squared prediction error for each observation and each number of components.

Author(s)

Irene Hoffmann

References


See Also

prms, plot.prm, predict.prm, sprmsCV

Examples

```r
set.seed(5023)
U <- c(rep(2,20), rep(5,30))
X <- replicate(6, U+rnorm(50))
beta <- c(rep(1, 3), rep(-1,3))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
res <- prmsCV(y~., data=d, as=2:4, plot=TRUE, prec=0.05)
summary(res$opt.mod)
```

sprmda  

**Sparse and robust PLS for binary classification**

Description

This method for dimension reduction and discriminant analysis yields a sparse classification model with a partial least squares alike interpretability that is robust to both vertical outliers and leverage points.
Usage

sprmda(formula, data, a, eta, fun = "Hampel", probp1 = 0.95, hampelp2 = 0.975, hampelp3 = 0.999, probp4=0.01, yweights = TRUE, class = c("regfit", "lda"), prior = c(0.5, 0.5), center = "median", scale = "qn", print = FALSE, numit = 100, prec = 0.01)

Arguments

formula a formula, e.g. group ~ X1 + X2 with group a factor with two levels or a numeric vector coding class membership with 1 and -1 and X1,X2 numeric variables.
data a data frame or list which contains the variables given in formula. The response specified in the formula needs to be a numeric vector coding the class membership with 1 and-1 or a vector of factors with two levels.
a the number of SPRM components to be estimated in the model.
etta a tuning parameter for the sparsity with 0\le eta<1.
fun an internal weighting function for case weights. Choices are "Hampel" (preferred), "Huber" or "Fair".
probp1 the 1-alpha value at which to set the first outlier cutoff for the weighting function.
hampelp2 the 1-alpha values for second cutoff. Only applies to fun="Hampel".
hampelp3 the 1-alpha values for third cutoff. Only applies to fun="Hampel".
probp4 a quantile close to zero for the cutoff for potentially wrong class labels (see Reference). Ignorred if yweights=FALSE.
yweights logical; if TRUE weights are calculated for observations with potentially wrong class labels.
class type of classification; choices are "regfit" or "lda". If "regfit" an object of class prm is returned.
prior vector of length 2 with prior probabilities of the groups; only used if class="lda".
center type of centering of the data in form of a string that matches an R function, e.g. "mean" or "median".
scale type of scaling for the data in form of a string that matches an R function, e.g. "sd" or "qn" or alternatively "no" for no scaling.
print logical, default is FALSE. If TRUE the variables included in each component are reported.
numit the number of maximal iterations for the convergence of the coefficient estimates.
prec a value for the precision of estimation of the coefficients.

details

For class="lda" a robust LDA model is estimated in the SPRM score space for class="regfit" the model ist a robust sparse PLS regression model on the binary response.
Value

sprmda returns an object of class sprmda.

Functions summary, predict and biplot are available. Also the generic functions coefficients, fitted.values and residuals can be used to extract the corresponding elements from the sprmda object.

scores the matrix of scores.
R Direction vectors (or weighting vectors or rotation matrix) to obtain the scores. scores=XS%*%R.
loadings the matrix of loadings.
w the overall case weights used for robust dimension reduction and classification (depending on the weight function). w=sqrt(wy*wt).
wt the group wise obtained case weights in the score space.
wy the case weights for potentially mislabeled observations.
used.vars Indices of variables included in the model.
Yvar percentage of contribution for each component to the explanation of the variance of the response.
Xvar percentage of contribution for each component to the explanation of the variance of the variables.

Results from LDA model:

ldamod list with robust pooled within-group covariance (cov) and the two robust group centers (m1, m2) in the score space
ldafit posterior probabilities from robust LDA in the score space.
ldaclass predicted class labels from robust LDA in the score space.

Results from the regression model with binary response:

coefficients vector of coefficients of the weighted regression model.
intercept intercept of weighted regression model.
residuals vector of residuals, true response minus estimated response.
fitted.values the vector of estimated response values.
coefficients.scaled vector of coefficients of the weighted regression model with scaled data.
intercept.scaled intercept of weighted regression model with scaled data.

Data preprocessing:

YMeans value used internally to center response.
XMean vector used internally to center data.
Xscales vector used internally to scale data.
Yscales value used internally to scale response.
inputs list of inputs: parameters, data and scaled data.
sprmdaCV

**Author(s)**
Irene Hoffmann and Sven Serneels

**References**

**See Also**
sprmdaCV

**Examples**
```r
data(iris)
data <- droplevels(subset(iris, iris$Species != "setosa"))
smod <- sprmda(Species ~ ., data, a=2, eta=0.7, class="lda")
```

---

**sprmdaCV**

Cross validation method for sparse PRM classification models.

**Description**
k-fold cross validation for the selection of the number of components for sparse PRM classification.

**Usage**
```
sprmdaCV(formula, data, as, etas, nfold = 10, fun = "Hampel", probp1 = 0.95, hampelp2 = 0.975, hampelp3 = 0.999, probp4=0.01, yweights = TRUE, class = c("regfit", "lda"), prior = c(0.5, 0.5), center = "median", scale = "qn", print = FALSE, plot = TRUE, numit = 100, prec = 0.01)
```

**Arguments**

- **formula**: a formula, e.g. group ~ X1 + X2 with group a factor with two levels or a numeric vector coding class membership with 1 and -1 or a numeric vector of factors with two levels.
- **data**: a data frame or list which contains the variables given in formula. The response specified in the formula needs to be a numeric vector coding the class membership with 1 and -1 or a vector of factors with two levels.
- **as**: a vector with positive integers, which are the number of SPRM components to be estimated in the models.
- **etas**: vector of values for the tuning parameter for the sparsity. Values have to between 0 and 1.
- **nfold**: the number of folds used for cross validation, default is `nfold=10` for 10-fold CV.
fun an internal weighting function for case weights. Choices are "Hampel" (preferred), "Huber" or "Fair".

probp1 the 1-alpha value at which to set the first outlier cutoff for the weighting function.

hampelp2 the 1-alpha values for second cutoff. Only applies to fun="Hampel".

hampelp3 the 1-alpha values for third cutoff. Only applies to fun="Hampel".

probp4 a quantile close to zero for the cutoff for potentially wrong class labels (see Reference). Ignored if yweights=FALSE.

yweights logical; if TRUE weights are calculated for observations with potentially wrong class labels.

class type of classification; choices are "regfit" or "lda". If "regfit" an object of class prm is returned.

prior vector of length 2 with prior probabilities of the groups; only used if class="lda".

center type of centering of the data in form of a string that matches an R function, e.g. "mean" or "median".

scale type of scaling for the data in form of a string that matches an R function, e.g. "sd" or "qn" or alternatively "no" for no scaling.

print logical, default is FALSE. If TRUE the variables included in each component are reported.

plot logical, default is TRUE. If TRUE two contour plots are generated for number of components and sparsity parameter. The first contour plot shows the mean weighted misclassification rate (see Details) the second the number of variables in the model.

numit the number of maximal iterations for the convergence of the coefficient estimates.

prec a value for the precision of estimation of the coefficients.

Details

The robust cross validation criterion is a weighted misclassification rate. Class assignment of outliers is unreliable. Therefore, the case weights from the model are used to downweight the influence observations which were detected as outliers on the misclassification rate.

There may occur combinations of "a" and "eta" where the model cannot be estimated. Then the function issues a warning "CV broke off at "a" and "eta"".

Value

opt.mod object of class sprmda with the selected parameters. (see sprms)

pcm array with predicted class membership of each observation and for each combination of tuning parameters

nzcoef array with the number of variables in the model for each cross validation subset and each combination of tuning parameters
**sprms**

**Author(s)**
Irene Hoffmann

**References**

**See Also**
`sprmda`, `biplot.sprmda`, `predict.sprmda`, `prmdaCV`

**Examples**
```r
data(iris)
data <- droplevels(subset(iris, iris$Species != "setosa"))
## for demonstration with only two values in etas
smod <- sprmdaCV(Species ~., data, as=2:3, etas=c(0.1, 0.9), nfold=5,
    class="lda", numit=10, prec=0.1)
biplot(smod$opt.mod)
## Not run:
## in practis a finer grid of as and etas should be searched
## at the expense of computation time
smod <- sprmdaCV(Species ~., data, as=1:4, etas=seq(0.1, 0.9, 0.1), nfold=5,
    class="lda", numit=10, prec=0.1)
## End(Not run)
```

---

**sprms**  
*Sparse partial robust M regression*

**Description**
Sparse partial robust M regression for models with univariate response. This method for dimension reduction and regression analysis yields estimates with a partial least squares alike interpretability that are both sparse and robust to both vertical outliers and leverage points. The sparsity is tuned with an L1 penalty.

**Usage**
```r
sprms(formula, data, a, eta, fun = "Hampel", probp1 = 0.95, hampelp2 = 0.975,
    hampelp3 = 0.999, center = "median", scale = "qn", print = FALSE,
    numit = 100, prec = 0.01)
```
Arguments

- **formula**: an object of class formula.
- **data**: a data frame which contains the variables given in formula or a list of two elements, where the first element is the response vector and the second element is a matrix of the explanatory variables.
- **a**: the number of SPRMS components to be estimated in the model.
- **eta**: a tuning parameter for the sparsity with 0\le eta<1.
- **fun**: an internal weighting function for case weights. Choices are "Hampel" (preferred), "Huber" or "Fair".
- **probp1**: the 1-alpha value at which to set the first outlier cutoff for the weighting function.
- **hampelp2**: the 1-alpha values for second cutoff. Only applies to fun="Hampel".
- **hampelp3**: the 1-alpha values for third cutoff. Only applies to fun="Hampel".
- **center**: type of centering of the data in form of a string that matches an R function, e.g. "mean" or "median".
- **scale**: type of scaling for the data in form of a string that matches an R function, e.g. "sd" or "qn" or alternatively "no" for no scaling.
- **print**: logical, default is FALSE. If TRUE the variables included in each component are reported.
- **numit**: the maximum number of iterations for the convergence of the coefficient estimates.
- **prec**: a value for the precision of estimation of the coefficients.

Details

The NIPLS algorithm with a L1 sparsity constrained combined with weighted regression is used for the model estimation.

- **a**: is the number of components in the model. Note that it is not possible to simply reduce the number of weighting vectors to obtain a model with a smaller number of components. Each model has to be estimated separately due to its dependence on robust case weights.

Value

- **sprms**: returns an object of class sprm.

Functions summary, predict and plot are available. Also the generic functions coefficients, fitted.values and residuals can be used to extract the corresponding elements from the sprm object.

- **coefficients**: vector of coefficients of the weighted regression model.
- **intercept**: intercept of weighted regression model.
- **wy**: the case weights in the y space.
- **wt**: the case weights in the score space.
the overall case weights used for weighted regression (depending on the weight function). \( w = w_y * w_t \).

scores the matrix of scores.

\( R \) Direction vectors (or weighting vectors or rotation matrix) to obtain the scores.

scores = \( X_s * \%R \).

loadings the matrix of loadings.

fitted.values the vector of estimated response values.

residuals vector of residuals, true response minus estimated response.

coefficients.scaled vector of coefficients of the weighted regression model with scaled data.

intercept.scaled intercept of weighted regression model with scaled data.

\( yMean \) value used internally to center response.

\( xMean \) vector used internally to center data.

\( Xscales \) vector used internally to scale data.

\( Yscales \) value used internally to scale response.

\( yVar \) percentage of contribution for each component to the explanation of the variance of the response.

\( xVar \) percentage of contribution for each component to the explanation of the variance of the variables.

inputs list of inputs: parameters, data and scaled data.

used.vars Indices of variables included in the model.

Author(s)

Sven Serneels, BASF Corp and Irene Hoffmann

References


See Also

sprmsCV, plot.sprm, biplot.sprm, predict.sprm, prms
Examples

```r
set.seed(50235)
U1 <- c(rep(3,20), rep(4,30))
U2 <- rep(3.5,50)
X1 <- replicate(5, U1+rnorm(50))
X2 <- replicate(20, U2+rnorm(50))
X <- cbind(X1,X2)
beta <- c(rep(1, 5), rep(0,20))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
mod <- sprms(y~., data=d, a=1, eta=0.5, fun="Hampel")
sprmfit <- predict(mod)

plot(y,sprmfit, main="SPRM")
abline(0,1)
```

**sprmsCV**

Cross validation method for SPRM regression models.

Description

k-fold cross validation for the selection of the number of components and the sparsity parameter for sparse partial robust M regression.

Usage

```r
sprmsCV(formula, data, as, etas, nfold = 10, fun = "Hampel", probp1 = 0.95, hampelp2 = 0.975, hampelp3 = 0.999, center = "median", scale = "qn", plot = TRUE, numit = 100, prec = 0.01, alpha = 0.15)
```

Arguments

- **formula** an object of class formula.
- **data** a data frame or list which contains the variables given in formula.
- **as** a vector with positive integers, which are the number of SPRM components to be estimated in the models.
- **etas** vector of values for the tuning parameter for the sparsity. Values have to be between 0 and 1.
- **nfold** the number of folds used for cross validation, default is nfold=10 for 10-fold CV.
- **fun** an internal weighting function for case weights. Choices are "Hampel" (preferred), "Huber" or "Fair".
the 1-alpha value at which to set the first outlier cutoff for the weighting function.

hampelp2
the 1-alpha values for second cutoff. Only applies to fun="Hampel".

hampelp3
the 1-alpha values for third cutoff. Only applies to fun="Hampel".

center
type of centering of the data in form of a string that matches an R function, e.g. "mean" or "median".

scale
type of scaling for the data in form of a string that matches an R function, e.g. "sd" or "qn" or alternatively "no" for no scaling.

plot
logical, default is TRUE. If TRUE two contour plots are generated for number of components and sparsity parameter. The first contour plot shows the trimmed mean squared error of the prediction of the response (see Details) the second the number of variables in the model.

numit
the number of maximal iterations for the convergence of the coefficient estimates.

prec
a value for the precision of estimation of the coefficients.

alpha
value used for alpha trimmed mean squared error, which is the cross validation criterion (see Details).

Details

The alpha - trimmed mean squared error of the predicted response over all observations is used as robust decision criterion to choose the optimal model.

There may occur combinations of "a" and "eta" where the model cannot be estimated. Then the function issues a warning "CV broke off at "a" and "eta"".

Value

opt.mod
object of class sprm with the selected parameters. (see sprms)

spe
array with squared prediction error for each observation and each combination of tuning parameters

nzcoef
array with the number of variables in the model for each cross validation subset and each combination of tuning parameters

Author(s)
Irene Hoffmann

References


See Also

sprms, plot.sprm, predict.sprm, prmsCV
Examples

```r
set.seed(50235)
U1 <- c(rep(3,20), rep(4,30))
U2 <- rep(3.5,50)
X1 <- replicate(5, U1+rnorm(50))
X2 <- replicate(20, U2+rnorm(50))
X <- cbind(X1,X2)
beta <- c(rep(1, 5), rep(0,20))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
res <- sprmsCV(y~, data=d, as=1:2, etas=seq(0,0.9,0.2), nfold=5, fun="Hampel", prec=0.1)
summary(res$opt.mod)
```

---

**summary.prm**  

**Summary of a prm model**

**Description**

Summarizing models of class prm.

**Usage**

```r
## S3 method for class 'prm'
summary(object, ...)  
## S3 method for class 'prm'
print(x, ...)  
```

**Arguments**

- `object, x` object of class prm
- `...` optional arguments for internal print function.

**Value**

- `summary` prints model parameters and explained variances.
- `print` prints model parameters.

**Author(s)**

Irene Hoffmann

**References**


summary.prmda

See Also

prms

Examples

set.seed(5023)
U <- c(rep(2,20), rep(5,30))
X <- replicate(6, U+rnorm(50))
beta <- c(rep(1, 3), rep(-1,3))
e <- c(rnorm(45,0,1.5), rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
mod <- prms(y~, data=d, a=2, fun="Hampel")
summary(mod)

summary.prmda  Summary of a prmda model

Description

Summarizing models of class prmda.

Usage

## S3 method for class 'prmda'
summary(object, ...)
## S3 method for class 'prmda'
print(x, ...)

Arguments

object, x  object of class prmda
...

optional arguments for internal print function.

Value

summary prints model parameters and explained variances.
print prints model parameters.

Author(s)

Irene Hoffmann

References

See Also

prmda

Examples

data(iris)
data <- droplevels(subset(iris, iris$Species!="setosa"))
mod <- prmda(Species~., data, a=2, class="lda")
mod
summary(mod)

summary.sprm  Summary of a sprm model

Description

Summarizing models of class sprm.

Usage

## S3 method for class 'sprm'
summary(object, ...)
## S3 method for class 'sprm'
print(x, ...)

Arguments

object, x  object of class sprm.
...	optional arguments for internal print function.

Value

summary prints model parameters and explained variances.
print prints model parameters.

Author(s)

Irene Hoffmann

References

See Also

sprms

Examples

```r
set.seed(50235)
U1 <- c(rep(3,20), rep(4,30))
U2 <- rep(3.5,50)
X1 <- replicate(5, U1+rnorm(50))
X2 <- replicate(20, U2+rnorm(50))
X <- cbind(X1,X2)
beta <- c(rep(1, 5), rep(0,20))
e <- c(rnorm(45,0,1.5),rnorm(5,-20,1))
y <- X%*%beta + e
d <- as.data.frame(X)
d$y <- y
mod <- sprms(y~., data=d, a=1, eta=0.5, fun="Hampel")
summary(mod)
```
References


See Also

sprmda

Examples

```r
data(iris)
data <- droplevels(subset(iris, iris$Species!="setosa"))
smod <- sprmda(Species~., data, a=2, eta=0.7, class="lda")
smod
summary(smod)
```
Index

*Topic \textasciitilde kwd1
  biplot.sprm, 6
*Topic \textasciitilde kwd2
  biplot.sprm, 6
*Topic biplot
  biplot.prmda, 5
  biplot.sprm, 7
*Topic classification
  biplot.prmda, 5
  biplot.sprm, 7
*Topic package
  sprm-package, 2

biplot.prm, 3, 7, 10, 22
biplot.prmda, 5, 20
biplot.sprm, 4, 6, 11, 31
biplot.sprmda, 7, 29

plot.prm, 4, 8, 22, 24
plot.sprm, 7, 10, 31, 33
predict.prm, 12, 22, 24
predict.prmda, 13, 20
predict.sprm, 14, 31, 33
predict.sprmda, 15, 29
print.prm (summary.prm), 34
print.prmda (summary.prmda), 35
print.sprm (summary.sprm), 36
print.sprmda (summary.sprmda), 37
prmda, 3, 5, 14, 16, 20, 36
prmdaCV, 14, 18, 19, 29
prms, 3, 4, 10, 13, 20, 24, 31, 35
prmsCV, 13, 22, 23, 33

sprm, 7, 11
sprm (sprm-package), 2
sprm-package, 2
sprmda, 3, 8, 16, 24, 29, 38
sprmdaCV, 16, 20, 27, 27
sprms, 3, 15, 22, 28, 29, 33, 37
sprmsCV, 15, 24, 31, 32

summary.prm, 34
summary.prmda, 35
summary.sprm, 36
summary.sprmda, 37