

Package ‘prim’

April 19, 2009

Version 1.0.6

Date 2008/12/11

Title Patient Rule Induction Method (PRIM)

Author Tarn Duong <tarn.duong@gmail.com>

Maintainer Tarn Duong <tarn.duong@gmail.com>

Depends R (>= 1.4.0), ks

Suggests rgl (>= 0.66), misc3d (>= 0.4-0)

Description PRIM for bump hunting in high-dimensional data

License GPL-2

Repository CRAN

Date/Publication 2008-12-11 18:14:44

R topics documented:

plot.prim	2
prim	3
prim.box	4
quasiflow	6

Index	8
--------------	----------

plot.prim

PRIM plot for multivariate data

Description

PRIM plot for multivariate data.

Usage

```
## bivariate
## S3 method for class 'prim':
plot(x, col, xlim, ylim, xlab, ylab, add=FALSE,
     add.legend=FALSE, cex.legend=1, pos.legend, lwd=1, ...)

## trivariate
## S3 method for class 'prim':
plot(x, color, xlim, ylim, zlim, xlab, ylab, zlab,
     add.axis=TRUE, ...)

## d-variate
## S3 method for class 'prim':
plot(x, col, xmin, xmax, xlab, ylab, ...)
```

Arguments

x	an object of class <code>prim</code>
add.legend	flag for adding legend (2-d plot)
pos.legend	(x,y) co-ordinates for legend (2-d plot)
cex.legend	cex graphics parameter for legend (2-d plot)
col	vector of plotting colours, one for each box
xlab, ylab, zlab, xlim, ylim, zlim, add, lwd	usual graphics parameters
xmin, xmax	vector of minimum and maximum axis plotting values for scatter plot matrix
color	vector of colours, one for each box (3-d plot)
add.axis	flag for plotting axes (3-d plot)
...	other graphics parameters

Details

Default colours are `topo.colors()`, with one colour per box in the PRIM box sequence.

Value

Plot of 2-dim PRIM is a set of nested rectangles. Plot of 3-dim PRIM is a scatter point cloud. Plot of d-dim PRIM is a scatter plot matrix. The scatter plots indicate which points belong to which box.

References

Friedman, J.H. & Fisher, N.I. (1999) Bump-hunting for high dimensional data, *Statistics and Computing*, **9**, 123–143.

See Also

[prim.box](#)

Examples

```
## see ?prim.box
```

prim

Patient Rule Induction Method (PRIM)

Description

PRIM for bump-hunting for high-dimensional regression-type data.

Details

The data are $(\mathbf{X}_1, Y_1), \dots, (\mathbf{X}_n, Y_n)$ where \mathbf{X}_i is d-dimensional and Y_i is a scalar response. We wish to find the modal (and/or anti-modal) regions in the conditional expectation $m(\mathbf{x}) = E(Y|\mathbf{x})$.

PRIM is a bump-hunting technique introduced by Friedman & Fisher (1999), taken from data mining. PRIM estimates are a sequence of nested hyper-rectangles (boxes).

For an overview of this package, see `vignette("prim")` for PRIM estimation for 2- and 5-dimensional data.

Author(s)

Tarn Duong <tduong@maths.unsw.edu.au>

References

Friedman, J.H. & Fisher, N.I. (1999) Bump-hunting for high dimensional data, *Statistics and Computing*, **9**, 123–143.

Hyndman, R.J. Computing and graphing highest density regions. *The American Statistician*, **50**, 120–126.

 prim.box

PRIM for multivariate data

Description

PRIM for multivariate data.

Usage

```
prim.box(x, y, box.init=NULL, peel.alpha=0.05, paste.alpha=0.01,
        mass.min=0.05, threshold, pasting=TRUE, verbose=FALSE,
        threshold.type=0)
```

```
prim.hdr(prim, threshold, threshold.type)
prim.combine(prim1, prim2)
```

Arguments

x	matrix of data values
y	vector of response values
box.init	initial covering box
peel.alpha	peeling quantile tuning parameter
paste.alpha	pasting quantile tuning parameter
mass.min	minimum mass tuning parameter
threshold	threshold tuning parameter(s)
threshold.type	threshold direction indicator: 1 = ">= threshold", -1 = "<= threshold", 0 = ">= threshold[1] & <= threshold[2]"
pasting	flag for pasting
verbose	flag for printing output during execution
prim, prim1, prim2	objects of type prim

Details

The data are $(\mathbf{X}_1, Y_1), \dots, (\mathbf{X}_n, Y_n)$ where \mathbf{X}_i is d-dimensional and Y_i is a scalar response. PRIM finds modal (and/or anti-modal) regions in the conditional expectation $m(\mathbf{x}) = E(Y|\mathbf{x})$.

In general, Y_i can be real-valued. See `vignette("prim")`. Here, we focus on the special case for binary Y_i . Let $Y_i = 1$ when $\mathbf{X}_i \sim F^+$; and $Y_i = -1$ when $\mathbf{X}_i \sim F^-$ where F^+ and F^- are different distribution functions. In this set-up, PRIM finds the regions where F^+ and F^- are most different.

The tuning parameters `peel.alpha` and `paste.alpha` control the ‘patience’ of PRIM. Smaller values involve more patience. Larger values less patience. The peeling steps remove data from a box

till either the box mean is smaller than `threshold` or the box mass is less than `mass.min`. Peeling is optional, and is used to correct any possible over-peeling. The default values for `peel.alpha`, `paste.alpha` and `mass.min` are taken from Friedman & Fisher (1999).

The type of PRIM estimate is controlled `threshold` and `threshold.type`:

For `threshold.type=1`, we search for $\{m(\mathbf{x}) \geq \text{threshold}\}$.

For `threshold.type=-1`, we search for $\{m(\mathbf{x}) \leq \text{threshold}\}$.

For `threshold.type=0`, we search for both $\{m(\mathbf{x}) \geq \text{threshold}[1]\}$ and $\{m(\mathbf{x}) \leq \text{threshold}[2]\}$.

There are two ways of using PRIM. One is `prim.box` with pre-specified threshold(s). This is appropriate when the threshold(s) are known to produce good estimates.

On the other hand, if the user doesn't provide threshold values then `prim.box` computes box sequences which cover the data range. These can then be pruned at a later stage. `prim.hdr` allows the user to specify many different threshold values in an efficient manner, without having to recomputing the entire PRIM box sequence. `prim.combine` can be used to join the regions computed from `prim.hdr`. See the examples below.

Value

– `prim.box` produces a PRIM estimate, an object of type `prim`, which is a list with 8 fields:

<code>x</code>	list of data matrices
<code>y</code>	list of response variable vectors
<code>y.mean</code>	list of vectors of box mean for y
<code>box</code>	list of matrices of box limits (first row = minima, second row = maxima)
<code>mass</code>	vector of box masses (proportion of points inside a box)
<code>num.class</code>	total number of PRIM boxes
<code>num.hdr.class</code>	total number of PRIM boxes which form the HDR
<code>ind</code>	threshold direction indicator: 1 = " \geq threshold", -1 = " \leq threshold"

The above lists have `num.class` fields, one for each box.

– `prim.hdr` takes a `prim` object and prunes it using different threshold values. Returns another `prim` object. This is much faster for experimenting with different threshold values than calling `prim.box` each time.

– `prim.combine` combines two `prim` objects into a single `prim` object. Usually used in conjunction with `prim.hdr`. See examples below.

References

Friedman, J.H. & Fisher, N.I. (1999) Bump-hunting for high dimensional data, *Statistics and Computing*, **9**, 123–143.

Examples

```

n <- 1000
set.seed(88192)

mus.p <- rbind(c(0,0), c(2,0), c(1, 2), c(2.5, 2))
Sigmas.p <- 0.125*rbind(diag(2), diag(c(0.5, 0.5)),
  diag(c(0.125, 0.25)), diag(c(0.125, 0.25)))
props.p <- c(0.5, 0.25, 0.125, 0.125)

mus.n <- rbind(c(0,0), c(2,0), c(2.5, 2))
Sigmas.n <- 0.125*rbind(matrix(c(1,-0.6,-0.6,1), nrow=2),
  diag(c(0.5, 0.5)),diag(c(0.125, 0.25)))
props.n <- c(0.625, 0.25, 0.125)

x.p <- rmvnorm.mixt(n, mus.p, Sigmas.p, props.p)
x.n <- rmvnorm.mixt(n, mus.n, Sigmas.n, props.n)
x <- rbind(x.p, x.n)
y <- c(rep(1, nrow(x.p)), rep(-1, nrow(x.n)))
  ## 1 = positive sample, -1 = negative sample

y.thr <- c(0.8, -0.35)

## using only one command

x.prim1 <- prim.box(x=x, y=y, threshold=y.thr, threshold.type=0)

## alternative - requires more commands but allows more control
## in intermediate stages

x.prim.hdr.p <- prim.box(x=x, y=y, threshold.type=1,
  threshold=0.8)

x.prim.n <- prim.box(x=x, y=y, threshold.type=-1)
summary(x.prim.n)
  ## threshold too high, try lower one

x.prim.hdr.n <- prim.hdr(x.prim.n, threshold=-0.35,
  threshold.type=-1)
x.prim2 <- prim.combine(x.prim.hdr.p, x.prim.hdr.n)

plot(x.prim2)

summary(x.prim1)
summary(x.prim2) ## should be exactly the same as command above

```

Description

This data set is simulated data from two normal mixture distributions, mimicking a flow cytometry data set. It contains 10000 observations from an HIV+ patient and 10000 observations an HIV-patient.

Usage

```
data(quasiflow)
```

Format

quasiflow is a matrix with 6 columns and 20000 rows. Each row corresponds to measurements for one cell. The first 5 columns are flow cytometric measurements and the sixth column is a binary indicator, with 1 = HIV+ and -1 = HIV-.

Source

Generated by package author.

Index

*Topic **datasets**

quasiflow, 6

*Topic **hplot**

plot.prim, 1

*Topic **multivariate**

prim.box, 3

*Topic **package**

prim, 3

plot.prim, 1

prim, 3

prim.box, 2, 3

prim.combine (*prim.box*), 3

prim.hdr (*prim.box*), 3

quasiflow, 6