

The pcurve Package

February 16, 2008

Version 0.6-2

Date 2005/02/25

Title Principal Curve analysis

Author S original by Trevor Hastie <hastie@stat.stanford.edu> S+ library by Glenn De'ath <g.death@aims.gov.au> R port by Chris Walsh <Chris.Walsh@sci.monash.edu.au>

Depends R(>= 1.9.0), mgcv, vegan, MASS, stats

Description Fits a principal curve to a numeric multivariate dataset in arbitrary dimensions. Produces diagnostic plots.

Maintainer Chris Walsh <Chris.Walsh@sci.monash.edu.au>

License GPL

R topics documented:

pca	1
pcdiags.plt	2
pcurve	4
pcurve.data	7
pcurve.internal	8

Index **9**

pca *Principal Component Analysis*

Description

Calculates principal components from a matrix

Usage

```
pca(mat, cent = TRUE, scle = FALSE)
```

Arguments

mat	a numeric matrix.
cent	a logical value referring to center argument in <code>scale</code> .
scle	a logical value referring to scale argument in <code>scale</code> .

Value

a list containing	
pcs	a matrix of principal component loadings
d	a matrix containing the singular value (eigenvalue) of each principal component on its diagonal
v	a matrix of eigenvectors

Author(s)

R port by Chris Walsh <Chris.Walsh@sci.monash.edu.au> from S+ library by Glenn De'ath <g.death@aims.gov.au>.

Examples

```
data(soilspec)
species <- sqrt(soilspec[,2:9])
specpca <- pca(species)
eqsplot(specpca$pcs[,1], specpca$pcs[,2], type = "n",
        xlab = "Principal component 1",
        ylab = "Principal component 2")
text(specpca$pcs[,1], specpca$pcs[,2],
     soilspec$site)
mtext(paste("Grassland communities in 45 sites"))
```

pcdiags.plt

Diagnostic Plots for Principal Curve Analysis

Description

A menu of 12 plots for diagnosis of results from principal curve analysis, `pcurve`

Usage

```
pcdiags.plt(zz, xx, pch = 1, graphics = TRUE)
```

Arguments

zz	an object of class principal curve, being the value of the function <code>pcurve</code> .
xx	data.frame or matrix of explanatory (environmental) variables to be used in constrained pcs.
pch	symbol to be used in plots.
graphics	a logical argument of menu indicating whether a graphics menu should be used. Currently unused.

Details

Produces a menu of 12 (or thirteen if xx is not missing) options. Once a selection is made, return to the menu by left-mouse clicking on the plot.

0. Exit
1. Residuals plots for each variable on the PC (by the internal function `pcres1.plt`)
2. Absolute residuals plot for each variable on the PC (by the internal function `pcres2.plt`)
3. QQ normal residuals plot for each variable (by the internal function `pcqqnorm.plt`)
4. QQ chi-squared quantile residuals plot (by the internal function `pcchisq.plt`)
5. Response plot and residual plot for each variable (by the internal function `pcresid.plt`)
6. Differenced locations: Plot of distances between consecutive locations on the PC (by the internal function `pcfinder.plt`)
7. Response plots for each variable along the PC (by the internal function `pcresp.plt`)
8. Flip plots: Plot of the PC projected onto a bi-plot of the first two principal coordinates, showing fitted locations of the variables on the PC. Left-mouse click to scroll through biplots of other principal coordinate combinations. (Right-mouse-click to return to the menu) (Using the internal function `pcflip.plt`)
9. Fix curve: a utility to break the curve in up to two places (by left mouse-clicks), re-order the segments and rerun the PC analysis with a new start. (using the internal function `finder`)
10. Scatterplots of Eclidean and Bray-Curtis distances against the PC. (using the internal function `pcdists.plt`)
11. Histograms of Eclidean and Bray-Curtis distances against the PC. (using the internal function `pchist.plt`)
12. A toggle to use Case numbers or symbols in plots
13. Env. vars. vs Gradient: if xx is not missing, plots of distance along the PC and explanatory variables (using the internal function `pcenv.plt`)

Value

Produces plots

Author(s)

R port by Chris Walsh <Chris.Walsh@sci.monash.edu.au> from S+ library by Glenn De'ath <g.death@aims.gov.au>.

References

De'ath, G. 1999 Principal Curves: a new technique for indirect and direct gradient analysis. *Ecology* **80**, 2237–2253.

Description

Fits a principal curve to a numeric multivariate dataset in arbitrary dimensions. Produces diagnostic plots.

Usage

```
pcurve(x, xcan = NULL, start = "ca", rank = FALSE, cv.fit = FALSE,
       penalty= 1, cv.all = FALSE, df = "vary", fit.meth = "spline",
       canfit = "lm", candf = FALSE, vary.adj = FALSE, subset,
       robust = FALSE, lowf = 0.5, min.df, max.df, max.df.cv.fit,
       ext.dist = TRUE, ext.dc = 0.9, metric = "bray", latent = FALSE,
       plot.pca = TRUE, thresh = 0.001, plot.true = TRUE,
       plot.init = FALSE, plot.segs = TRUE, plot.resp = TRUE,
       plot.cov = TRUE, maxit = 10, stretch = 2, fits = FALSE,
       prnt.fits = TRUE, trace = TRUE, trace.all = FALSE, pch = 1,
       row.chk0 = FALSE, col.chk0 = TRUE, use.loc = FALSE)
```

Arguments

x	numeric data matrix or data.frame.
xcan	data.frame or matrix of explanatory variables to be used in constrained PCs.
start	specifies how to determine the starting configuration (location of points on initial curve): "ca" = correspondence analysis; "pca" = principal components analysis with Euclidan metric; "pca.bc" = principal components analysis with Bray-Curtis metric; "mds" = non-metric multidimensional scaling with Euclidean metric; "mds.bc" = non-metric multidimensional scaling with Bray-Curtis metric; "cs.bc" = classical scaling (metric multidimensional scaling) with Bray-Curtis metric; "ran" = random start. Or if start is numeric and of length dim(x)[1] a user supplied configuration will be used.
rank	if TRUE starting configuration is transformed to rank
cv.fit	if TRUE a final iteration using cross-validation is done.
penalty	penalty for smoothing spline. A value of 1 corresponds to no penalty with values > 1 giving a less-smoothed fit. Increasing the penalty for small data sets can reduce over-fitting. If penalty = "np", penalty = 1 for N > 1000, penalty = 2 for N <=100, and penalty = 4-log(N, 10) for N > 100 and N <= 1000.
cv.all	if TRUE a cross-validated smoothing spline fit at each iteration.
df	if numeric specifies the df for the smoothing spline.
fit.meth	specifies smoother. "spline" = smooth.spline, "poisson" = poisson general additive model, "binomial" = binomial general additive model, "lowess" = lowess smoother (this argument overridden by robust = TRUE).

canfit	"lm" or "gam", model used to relate pc to xcan.
candf	if canfit = "gam", df for model. May be a single value or a vector of FALSE or positive integers indicating dfs for each explanatory variable in xcan. If FALSE, this is equivalent to fx=FALSE in gam, and d.f. is selected by GCV.UBRE
vary.adj	if FALSE the same df are used for the smooth of each variable, otherwise each variable has its own df.
subset	used to take a subset of x and start (if numeric).
robust	if TRUE uses lowess smooths, if FALSE uses smoothing spline.
lowf	specifies the span of the lowess smooth.
min.df	specifies the min df for the smoothing.
max.df	specifies the max df for the smoothing.
max.df.cv.fit	
ext.dist	if TRUE extended dissimilarities in calculation of initial configuration using the flexible shortest path. If FALSE standard dissimilarities are used (see De'ath, 1999b and stepacross in package vegan).
ext.dc	critical distance, the tolong argument in stepacross.
metric	similarity metric, the method argument in vegdist in package vegan.
latent	if FALSE locations are rescaled after each iteration to give distance along the curve; if TRUE no rescaling is done.
plot.pca	if TRUE the fitting is plotted (assuming plot.true = TRUE) in the first 2 dimensions of PCA space.
thresh	threshold value of difference in cross-validation for ceasing iteration
plot.true	if TRUE the fitting process is plotted.
plot.init	if TRUE the initial fits to each variable are plotted.
plot.segs	if TRUE segments linking the fitted points on the curves to their corresponding data points are plotted.
plot.resp	if TRUE the final response curves are plotted.
plot.cov	if TRUE covariate partial effects are plotted (only if xcan is not null).
maxit	specifies the maximum number of iterations.
stretch	end segments of the curve are stretched by this factor at each iteration.
fits	if TRUE value of pcurve includes diagnostics for each variable.
prnt.fits	statistics on model fits printed.
trace	prints out useful fitting diagnostics at each iteration.
trace.all	if TRUE prints out all curve details at each iteration.
pch	symbol for plots
row.chk0	if TRUE checks for and removes rows of x identically 0.
col.chk0	if TRUE checks for and removes columns of x identically 0.
use.loc	if TRUE pauses during the fitting displays (left mouse-click to progress to next plot).

Details

See De'ath (1999a) for a full discussion of the functions and their application.

Value

An object of class principal curve containing a list comprising

s	fitted values
tag	order of points along the curve
lambda	locations along the curve
dist	sum of squared distances of points from the curve
c	call to pcurve
x	data to which the curve was fitted
df	degrees of freedom for the smoothers used in the fit
fit.list	diagnostics for each variable, only included if fits = TRUE.

Author(s)

R port by Chris Walsh (Chris.Walsh@sci.monash.edu.au) from S+ library by Glenn De'ath (g.death@aims.gov.au).
Original S code for principal curve analysis by Trevor Hastie (hastie@stat.stanford.edu).

References

- De'ath, G. 1999a Principal Curves: a new technique for indirect and direct gradient analysis. *Ecology* **80**, 2237–2253.
- De'ath, G. 1999b Extended dissimilarity: method of robust estimation of ecological distances with high beta diversity. *Plant Ecology* **144**, 191–199.
- Gittins, R. 1985 *Canonical Analysis. A review with applications in ecology*. Berlin: Springer-Verlag.
- Hastie, T.J and Tibshirani, R.J. 1990 *Generalized additive models*. London: Chapman and Hall.
- Hastie, T.J. and Stuetzle, W. 1989 Principal Curves. *Journal of the American Statistical Association* **84**, 502–516.

See Also

[pcdiags.plt](#), [vegdist](#), [stepacross](#)

Examples

```
#a simulated dataset with 4 response variables (taxa 1-4),
#n=100. The response curve is Gaussian and noise is Poisson.
data(sim4var)
sim4fit <- pcurve(sim4var, plot.init = FALSE, use.loc = TRUE)

#Limestone grassland community example worked by De'ath (1999a),
#from data in Gittins (1985)
```

```

data(soilspec)
species <- sqrt(soilspec[,2:9])
envvar <- soilspec[,10:12]
#indirect gradient analysis
spec.fit <- pcurve(species, start = "mds.bc", plot.init = FALSE,
                  use.loc = TRUE)
#direct gradient analysis
soilspec.fit <- pcurve(species, xcan = envvar,
                      start = "mds.bc", plot.init = FALSE,
                      fits = TRUE, prnt.fits = TRUE,
                      use.loc = TRUE)

```

pcurve.data

Example data for pcurve

Description

Example data sets for pcurve package.

Usage

```

data(sim4var)
data(sim10var)
data(fish)
data(soilspec)

```

Details

sim4var.txt

Simulated data. Comprises 4 response variables (Taxa1 - Taxa4) and the generating locations (Location). Number of cases = 100. This example was worked by De'ath (1999). The response curves are Gaussian and noise is Poisson. Most starting configurations are adequate, square root transformation helps.

sim10var.txt

Simulated data. Comprises 10 response variables (Taxa1 - Taxa10) and the generating locations (Location). Number of cases = 100. The response curves are Gaussian and noise is Poisson. The beta-diversity is high and recovering the generating locations is difficult. A more difficult exercise. Transformation is a must (square-root is ok). Many starting configurations fail. CA or MDS-BC succeed with appropriate smoothness.

fish.txt

Comprises counts on 10 families of reef fish (n = 33) and a factor variable IMO denoting the position of the sites across the reef. Log-transformation is effective and a final cross-validation helps improve the fit. The locations vary systematically with cross shelf position (IMO).

soilspec.txt

Comprises data on 8 species of plants and 3 soil characteristics and their interactions. Source Gittins (1985), where a relatively complex canonical analysis was used to model the data. This example was worked by De'ath (1999).

Author(s)

R port by Chris Walsh <Chris.Walsh@sci.monash.edu.au> from S+ library by Glenn De'ath <g.death@aims.gov.au>.

References

De'ath, G. 1999 Principal Curves: a new technique for indirect and direct gradient analysis. *Ecology* **80**, 2237–2253.

Gittins, R. 1985 *Canonical Analysis. A review with applications in ecology*. Berlin: Springer-Verlag.

pcurve.internal *Internal functions for pcurve*

Description

Internal functions for pcurve

Index

- *Topic **datasets**
 - `pcurve.data`, 7
- *Topic **hplot**
 - `pcdiags.plt`, 2
 - `pcurve`, 4
- *Topic **internal**
 - `pcurve.internal`, 8
- *Topic **iplot**
 - `pcdiags.plt`, 2
- *Topic **loess**
 - `pcurve`, 4
- *Topic **multivariate**
 - `pca`, 1
 - `pcdiags.plt`, 2
 - `pcurve`, 4
- *Topic **smooth**
 - `pcdiags.plt`, 2
 - `pcurve`, 4

`cavecs(pcurve.internal)`, 8

`finder(pcdiags.plt)`, 2

`fish(pcurve.data)`, 7

`gdsqform(pcurve.internal)`, 8

`mdsform(pcurve.internal)`, 8

`n.plt(pcurve.internal)`, 8

`pca`, 1

`pcchisq.plt(pcdiags.plt)`, 2

`pcdiags.plt`, 2, 6

`pcdists.plt(pcdiags.plt)`, 2

`pcenv.plt(pcdiags.plt)`, 2

`pcfinder.plt(pcdiags.plt)`, 2

`pcflip.plt(pcdiags.plt)`, 2

`pcget.lam(pcurve.internal)`, 8

`pchist.plt(pcdiags.plt)`, 2

`pcqqnorm.plt(pcdiags.plt)`, 2

`pcres1.plt(pcdiags.plt)`, 2

`pcres2.plt(pcdiags.plt)`, 2

`pcresid.plt(pcdiags.plt)`, 2

`pcresp.plt(pcdiags.plt)`, 2

`pcurve`, 4

`pcurve.data`, 7

`pcurve.internal`, 8

`print.gd(pcurve.internal)`, 8

`scaletol(pcurve.internal)`, 8

`sim10var(pcurve.data)`, 7

`sim4var(pcurve.data)`, 7

`soilspec(pcurve.data)`, 7

`startPC(pcurve.internal)`, 8

`stepacross`, 6

`vegdist`, 6