

# Package ‘pvclust’

April 19, 2009

**Version** 1.2-1

**Date** 2009-02-25

**Title** Hierarchical Clustering with P-Values via Multiscale Bootstrap Resampling

**Author** Ryota Suzuki <suzuki@ef-prime.com>, Hidetoshi Shimodaira <shimo@is.titech.ac.jp>

**Maintainer** Ryota Suzuki <suzuki@ef-prime.com>

**Depends** R (>= 1.8.0)

**Suggests** MASS, snow, Rmpi

**Description** pvclust is a package for assessing the uncertainty in hierarchical cluster analysis. It provides AU (approximately unbiased) p-values as well as BP (bootstrap probability) values computed via multiscale bootstrap resampling.

**License** GPL (>= 2)

**URL** <http://www.is.titech.ac.jp/~shimo/prog/pvclust/>

**Repository** CRAN

**Date/Publication** 2009-02-25 11:43:35

## R topics documented:

lung . . . . .	2
msfit . . . . .	3
msplot . . . . .	5
plot.pvclust . . . . .	5
print.pvclust . . . . .	7
pvclust . . . . .	8
pvpick . . . . .	11
seplot . . . . .	12

<b>Index</b>	<b>13</b>
--------------	-----------

---

lung

*DNA Microarray Data of Lung Tumors*

---

### Description

DNA Microarray data of 73 lung tissues including 67 lung tumors. There are 916 observations of genes for each lung tissue.

### Usage

```
data(lung)
```

### Format

data frame of size  $916 \times 73$ .

### Details

This dataset has been modified from original data. Each one observation of duplicate genes has been removed. See `source` section in this help for original data source.

### Source

[http://genome-www.stanford.edu/lung\\_cancer/adeno/](http://genome-www.stanford.edu/lung_cancer/adeno/)

### References

Garber, M. E. et al. (2001) "Diversity of gene expression in adenocarcinoma of the lung", *Proceedings of the National Academy of Sciences*, 98, 13784-13789.

### Examples

```
## Reading the data
data(lung)

## Multiscale Bootstrap Resampling
lung.pv <- pvclust(lung, nboot=100)

## CAUTION: nboot=100 may be too small for actual use.
##           We suggest nboot=1000 or larger.
##           plot/print functions will be useful for diagnostics.

## Plot the result
plot(lung.pv, cex=0.8, cex.pv=0.7)

ask.bak <- par()$ask
par(ask=TRUE)

pvrect(lung.pv, alpha=0.9)
```

```

msplot(lung.pv, edges=c(51,62,68,71))

par(ask=ask.bak)

## Print a cluster with high p-value
lung.pp <- pvpick(lung.pv, alpha=0.9)
lung.pp$clusters[[2]]

## Print its edge number
lung.pp$edges[2]

## We recommend parallel computing for large dataset as this one
## Not run:
library(snow)
cl <- makeCluster(10, type="MPI")
lung.pv <- parPvclust(cl, lung, nboot=1000)
## End(Not run)

```

---

msfit

*Curve Fitting for Multiscale Bootstrap Resampling*


---

## Description

msfit performs curve fitting for multiscale bootstrap resampling. It generates an object of class msfit. Several generic methods are available.

## Usage

```

msfit(bp, r, nboot)

## S3 method for class 'msfit':
plot(x, curve=TRUE, main=NULL, sub=NULL, xlab=NULL, ylab=NULL, ...)

## S3 method for class 'msfit':
lines(x, col=2, lty=1, ...)

## S3 method for class 'msfit':
summary(object, digits=3, ...)

```

## Arguments

bp	numeric vector of bootstrap probability values.
r	numeric vector of relative sample size of bootstrap samples defined as $r = n'/n$ for original sample size $n$ and bootstrap sample size $n'$ .
nboot	numeric value (vector) of the number of bootstrap replications.
x	object of class msfit.
curve	logical. If TRUE, the fitted curve is drawn.

main, sub, xlab, ylab, col, lty  
                                   generic graphic parameters.

object                  object of class msfit.

digits                  integer indicating the precision to be used in rounding.

...                      other parameters to be used in the functions.

### Details

function `msfit` performs the curve fitting for multiscale bootstrap resampling. In package `pvc lust` this function is only called from the function `pvc lust` (or `parPvc lust`), and may never be called from users. However one can access a list of `msfit` objects by `x$msfit`, where `x` is an object of class `pvc lust`.

### Value

`msfit` returns an object of class `msfit`. It contains the following objects:

`p`                      numeric vector of  $p$ -values. `au` is AU (Approximately Unbiased)  $p$ -value computed by multiscale bootstrap resampling, which is more accurate than BP value (explained below) as unbiased  $p$ -value. `bp` is BP (Bootstrap Probability) value, which is simple but tends to be unbiased when the absolute value of `c` (a value in `coef` vector, explained below) is large.

`se`                   numeric vector of estimated standard errors of  $p$ -values.

`coef`                  numeric vector related to geometric aspects of hypotheses. `v` is signed distance and `c` is curvature of the boundary.

`df`                   numeric value of the degree of freedom in curve fitting.

`rss`                  residual sum of squares.

`pchi`                   $p$ -value of chi-square test based on asymptotic theory.

### Author(s)

Ryota Suzuki (ryota.suzuki@is.titech.ac.jp)

### References

Shimodaira, H. (2004) "Approximately unbiased tests of regions using multistep-multiscale bootstrap resampling", *Annals of Statistics*, 32, 2616-2641.

Shimodaira, H. (2002) "An approximately unbiased test of phylogenetic tree selection", *Systematic Biology*, 51, 492-508.

---

`msplot`*Drawing the Results of Curve Fitting for Pvclust Object*

---

**Description**

draws the results of curve fitting for pvclust object.

**Usage**

```
msplot(x, edges=NULL, ...)
```

**Arguments**

<code>x</code>	object of class <code>pvclust</code> .
<code>edges</code>	numeric vector of edge numbers to be plotted.
<code>...</code>	other parameters to be used in the function.

**Author(s)**

Ryota Suzuki <ryota.suzuki@is.titech.ac.jp>

**See Also**

[plot.msfit](#)

---

`plot.pvclust`*Draws Dendrogram with P-values for Pvclust Object*

---

**Description**

plot dendrogram for a `pvclust` object and add  $p$ -values for clusters.

**Usage**

```
## S3 method for class 'pvclust':
plot(x, print.pv=TRUE, print.num=TRUE, float=0.01,
      col.pv=c(2,3,8), cex.pv=0.8, font.pv=NULL, col=NULL, cex=NULL,
      font=NULL, lty=NULL, lwd=NULL, main=NULL, sub=NULL, xlab=NULL, ...)

## S3 method for class 'pvclust':
text(x, col=c(2,3,8), print.num=TRUE, float=0.01, cex=NULL, font=NULL, ...)
```

**Arguments**

<code>x</code>	object of class <code>pvclust</code> , which is generated by function <code>pvclust</code> . See <a href="#">pvclust</a> for details.
<code>print.pv</code>	logical flag to specify whether print $p$ -values above the edges (clusters).
<code>print.num</code>	logical flag to specify whether print edge numbers below clusters.
<code>float</code>	numeric value to adjust the height of $p$ -values from edges.
<code>col.pv</code>	numeric vector of length three to specify the colors for $p$ -values and edge numbers. From the beginning each value corresponds to the color of AU values, BP values and edge numbers, respectively.
<code>cex.pv</code>	numeric value which specifies the size of characters for $p$ -values and edge numbers. See <code>cex</code> argument for <a href="#">par</a> .
<code>font.pv</code>	numeric value which specifies the font of characters for $p$ -values and edge numbers. See <code>font</code> argument for <a href="#">par</a> .
<code>col, cex, font</code>	in <code>text</code> function, they correspond to <code>col.pv</code> , <code>cex.pv</code> and <code>font.pv</code> in <code>plot</code> function, respectively. In <code>plot</code> function they are used as generic graphic parameters.
<code>lty, lwd, main, sub, xlab, ...</code>	generic graphic parameters. See <a href="#">par</a> for details.

**Details**

This function plots a dendrogram with  $p$ -values for given object of class `pvclust`. AU  $p$ -value (printed in red color in default) is the abbreviation of "approximately unbiased"  $p$ -value, which is calculated by multiscale bootstrap resampling. BP value (printed in green color in default) is "bootstrap probability" value, which is less accurate than AU value as  $p$ -value. One can consider that clusters (edges) with high AU values (e.g. 95%) are strongly supported by data.

**Author(s)**

Ryota Suzuki (ryota.suzuki@is.titech.ac.jp)

**References**

- Shimodaira, H. (2004) "Approximately unbiased tests of regions using multistep-multiscale bootstrap resampling", *Annals of Statistics*, 32, 2616-2641.
- Shimodaira, H. (2002) "An approximately unbiased test of phylogenetic tree selection", *Systematic Biology*, 51, 492-508.

**See Also**

[text.pvclust](#)

---

print.pvclust      *Print Function for Pvclust Object*

---

### Description

print clustering method and distance measure used in hierarchical clustering,  $p$ -values and related statistics for a `pvclust` object.

### Usage

```
## S3 method for class 'pvclust':  
print(x, which=NULL, digits=3, ...)
```

### Arguments

<code>x</code>	object of class <code>pvclust</code> .
<code>which</code>	numeric vector which specifies the numbers of edges (clusters) of which the values are printed. If <code>NULL</code> is given, it prints the values of all edges. The default is <code>NULL</code> .
<code>digits</code>	integer indicating the precision to be used in rounding.
<code>...</code>	other parameters used in the function.

### Value

this function prints  $p$ -values and some related statistics.

<code>au</code>	AU (Approximately Unbiased) $p$ -value, which is more accurate than BP value as unbiased $p$ -value. It is computed by multiscale bootstrap resampling.
<code>bp</code>	BP (Bootstrap Probability) value, which is a simple statistic computed by bootstrap resampling. This value tends to be biased as $p$ -value when the absolute value of <code>c</code> (explained below) is large.
<code>se.au</code> , <code>se.bp</code>	estimated standard errors for <code>au</code> and <code>bp</code> , respectively.
<code>v</code> , <code>c</code>	values related to geometric aspects of hypotheses. <code>v</code> is signed distance and <code>c</code> is curvature of the boundary.
<code>pchi</code>	$p$ -values of chi-square test based on asymptotic theory.

### Author(s)

Ryota Suzuki (ryota.suzuki@is.titech.ac.jp)

**Description**

calculates  $p$ -values for hierarchical clustering via multiscale bootstrap resampling. Hierarchical clustering is done for given data and  $p$ -values are computed for each of the clusters.

**Usage**

```
pvclust(data, method.hclust="average",
        method.dist="correlation", use.cor="pairwise.complete.obs",
        nboot=1000, r=seq(.5,1.4,by=.1), store=FALSE, weight=FALSE)

parPvclust(cl, data, method.hclust="average",
           method.dist="correlation", use.cor="pairwise.complete.obs",
           nboot=1000, r=seq(.5,1.4,by=.1), store=FALSE, weight=FALSE,
           init.rand=TRUE, seed=NULL)
```

**Arguments**

<code>data</code>	numeric data matrix or data frame.
<code>method.hclust</code>	the agglomerative method used in hierarchical clustering. This should be (an abbreviation of) one of "average", "ward", "single", "complete", "mcquitty", "median" or "centroid". The default is "average". See method argument in <a href="#">hclust</a> .
<code>method.dist</code>	the distance measure to be used. This should be (an abbreviation of) one of "correlation", "uncentered", "abscor" or those which are allowed for method argument in <code>dist</code> function. The default is "correlation". See <i>details</i> section in this help and method argument in <a href="#">dist</a> .
<code>use.cor</code>	character string which specifies the method for computing correlation with data including missing values. This should be (an abbreviation of) one of "all.obs", "complete.obs" or "pairwise.complete.obs". See the use argument in <a href="#">cor</a> function.
<code>nboot</code>	the number of bootstrap replications. The default is 1000.
<code>r</code>	numeric vector which specifies the relative sample sizes of bootstrap replications. For original sample size $n$ and bootstrap sample size $n'$ , this is defined as $r = n'/n$ .
<code>store</code>	logical. If <code>store=TRUE</code> , all bootstrap replications are stored in the output object. The default is <code>FALSE</code> .
<code>cl</code>	<b>snow</b> cluster object which may be generated by function <code>makeCluster</code> . See <a href="#">snow-startstop</a> in <b>snow</b> package.

weight	logical. If <code>weight=TRUE</code> , resampling is made by weight vector instead of index vector. Useful for large <code>r</code> value ( $r > 10$ ). Currently, available only for distance "correlation" and "abscor".
init.rand	logical. If <code>init.rand=TRUE</code> , random number generators are initialized at child processes. Random seeds can be set by <code>seed</code> argument.
seed	integer vector of random seeds. It should have the same length as <code>cl</code> . If <code>NULL</code> is specified, <code>1:length(cl)</code> is used as seed vector. The default is <code>NULL</code> .

## Details

Function `pvclust` conducts multiscale bootstrap resampling to calculate  $p$ -values for each cluster in the result of hierarchical clustering. `parPvclust` is the parallel version of this procedure which depends on `snow` package for parallel computation.

For data expressed as  $(n \times p)$  matrix or data frame, we assume that the data is  $n$  observations of  $p$  objects, which are to be clustered. The  $i$ 'th row vector corresponds to the  $i$ 'th observation of these objects and the  $j$ 'th column vector corresponds to a sample of  $j$ 'th object with size  $n$ .

There are several methods to measure the dissimilarities between objects. For data matrix  $X = \{x_{ij}\}$ , "correlation" method takes

$$1 - \frac{\sum_{i=1}^n (x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k)}{\sqrt{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2} \sqrt{\sum_{i=1}^n (x_{ik} - \bar{x}_k)^2}}$$

for dissimilarity between  $j$ 'th and  $k$ 'th object, where  $\bar{x}_j = \frac{1}{n} \sum_{i=1}^n x_{ij}$  and  $\bar{x}_k = \frac{1}{n} \sum_{i=1}^n x_{ik}$ .

"uncentered" takes uncentered sample correlation

$$1 - \frac{\sum_{i=1}^n x_{ij} x_{ik}}{\sqrt{\sum_{i=1}^n x_{ij}^2} \sqrt{\sum_{i=1}^n x_{ik}^2}}$$

and "abscor" takes the absolute value of sample correlation

$$1 - \left| \frac{\sum_{i=1}^n (x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k)}{\sqrt{\sum_{i=1}^n (x_{ij} - \bar{x}_j)^2} \sqrt{\sum_{i=1}^n (x_{ik} - \bar{x}_k)^2}} \right|.$$

## Value

<code>hclust</code>	hierarchical clustering for original data generated by function <code>hclust</code> . See <a href="#">hclust</a> for details.
<code>edges</code>	data frame object which contains $p$ -values and supporting informations such as standard errors.
<code>count</code>	data frame object which contains primitive information about the result of multiscale bootstrap resampling.
<code>msfit</code>	list whose elements are results of curve fitting for multiscale bootstrap resampling, of class <code>msfit</code> . See <a href="#">msfit</a> for details.
<code>nboot</code>	numeric vector of number of bootstrap replications.
<code>r</code>	numeric vector of the relative sample size for bootstrap replications.
<code>store</code>	list contains bootstrap replications if <code>store=TRUE</code> was given for function <code>pvclust</code> or <code>parPvclust</code> .

**Author(s)**

Ryota Suzuki <ryota.suzuki@is.titech.ac.jp>

**References**

Shimodaira, H. (2004) "Approximately unbiased tests of regions using multistep-multiscale bootstrap resampling", *Annals of Statistics*, 32, 2616-2641.

Shimodaira, H. (2002) "An approximately unbiased test of phylogenetic tree selection", *Systematic Biology*, 51, 492-508.

Suzuki, R. and Shimodaira, H. (2004) "An application of multiscale bootstrap resampling to hierarchical clustering of microarray data: How accurate are these clusters?", *The Fifteenth International Conference on Genome Informatics 2004*, P034.

<http://www.is.titech.ac.jp/~shimo/prog/pvclust/>

**See Also**

[lines.pvclust](#), [print.pvclust](#), [msfit](#), [plot.pvclust](#), [text.pvclust](#), [pvrect](#) and [pvpick](#).

**Examples**

```
## using Boston data in package MASS
library(MASS)
data(Boston)

## multiscale bootstrap resampling
boston.pv <- pvclust(Boston, nboot=100)

## CAUTION: nboot=100 may be too small for actual use.
##           We suggest nboot=1000 or larger.
##           plot/print functions will be useful for diagnostics.

## plot dendrogram with p-values
plot(boston.pv)

ask.bak <- par()$ask
par(ask=TRUE)

## highlight clusters with high au p-values
pvrect(boston.pv)

## print the result of multiscale bootstrap resampling
print(boston.pv, digits=3)

## plot diagnostic for curve fitting
msplot(boston.pv, edges=c(2,4,6,7))

par(ask=ask.bak)

## Print clusters with high p-values
```

```

boston.pp <- pvpick(boston.pv)
boston.pp

## Not run:
## parallel computation via snow package
library(snow)
cl <- makeCluster(10, type="MPI")

## parallel version of pvclust
boston.pv <- parPvclust(cl, Boston, nboot=1000)
## End(Not run)

```

pvpick

*Find Clusters with High/Low P-values***Description**

find clusters with relatively high/low  $p$ -values. `pvrect` and `lines` (S3 method for class `pvclust`) highlight such clusters in existing plot, and `pvpick` returns a list of such clusters.

**Usage**

```

pvpick(x, alpha=0.95, pv="au", type="geq", max.only=TRUE)

pvrect(x, alpha=0.95, pv="au", type="geq", max.only=TRUE, border=2, ...)

## S3 method for class 'pvclust':
lines(x, alpha=0.95, pv="au", type="geq", col=2, lwd=2, ...)

```

**Arguments**

<code>x</code>	object of class <code>pvclust</code> .
<code>alpha</code>	threshold value for $p$ -values.
<code>pv</code>	character string which specifies the $p$ -value to be used. It should be either of "au" or "bp", corresponding to AU $p$ -value or BP value, respectively. See <code>plot.pvclust</code> for details.
<code>type</code>	one of "geq", "leq", "gt" or "lt". If "geq" is specified, clusters with $p$ -value <i>greater than or equals</i> the threshold given by "alpha" are returned or displayed. Likewise "leq" stands for <i>lower than or equals</i> , "gt" for <i>greater than</i> and "lt" for <i>lower than</i> the threshold value. The default is "geq".
<code>max.only</code>	logical. If some of clusters with high/low $p$ -values have inclusion relation, only the largest cluster is returned (or displayed) when <code>max.only=TRUE</code> .
<code>border</code>	numeric value which specifies the color of borders of rectangles.
<code>col</code>	numeric value which specifies the color of lines.
<code>lwd</code>	numeric value which specifies the width of lines.
<code>...</code>	other graphic parameters to be used.

**Value**

`pvpick` returns a list which contains the following values.

<code>clusters</code>	a list of character string vectors. Each vector corresponds to the names of objects in each cluster.
<code>edges</code>	numeric vector of edge numbers. The $i$ 'th element (number) corresponds to the $i$ 'th name vector in <code>clusters</code> .

**Author(s)**

Ryota Suzuki (ryota.suzuki@is.titech.ac.jp)

---

seplot

*Diagnostic Plot for Standard Error of  $p$ -value*

---

**Description**

draws diagnostic plot for standard error of  $p$ -value for `pvclust` object.

**Usage**

```
seplot(object, type=c("au", "bp"), identify=FALSE, main=NULL,
        xlab=NULL, ylab=NULL, ...)
```

**Arguments**

<code>object</code>	object of class <code>pvclust</code> .
<code>type</code>	the type of $p$ -value to be plotted, one of "au" or "bp".
<code>identify</code>	logical. If TRUE, edge numbers can be identified interactively. See <code>identify</code> for basic usage.
<code>main, xlab, ylab</code>	generic graphic parameters. See <code>par</code> for details.
<code>...</code>	other graphical parameters to be passed to generic <code>plot</code> or <code>identify</code> function.

**Author(s)**

Ryota Suzuki (ryota.suzuki@is.titech.ac.jp)

# Index

- \*Topic **aplot**
  - pvpick, 11
- \*Topic **cluster**
  - pvclust, 8
- \*Topic **datasets**
  - lung, 2
- \*Topic **hplot**
  - msplot, 5
  - plot.pvclust, 5
  - seplot, 12
- \*Topic **htest**
  - msfit, 3
- \*Topic **print**
  - print.pvclust, 7

cor, 8

dist, 8

hclust, 8, 9

identify, 12

lines.msfit (*msfit*), 3

lines.pvclust, 10

lines.pvclust (*pvpick*), 11

lung, 2

msfit, 3, 9, 10

msplot, 5

par, 6, 12

parPvclust (*pvclust*), 8

plot.msfit, 5

plot.msfit (*msfit*), 3

plot.pvclust, 5, 10

print.pvclust, 7, 10

pvclust, 6, 8

pvpick, 10, 11

pvrect, 10

pvrect (*pvpick*), 11

seplot, 12

snow-startstop, 8

summary.msfit (*msfit*), 3

text.pvclust, 6, 10

text.pvclust (*plot.pvclust*), 5