Package 'Yamm'

May 20, 2022

Title Multivariate Methods Based on Projections and Related Concepts

Version 1.3.2

Date 2022-04-19

Depends R (>= 3.0), depth, robustX, interp

Suggests animation

Maintainer Guy Nason <g.nason@imperial.ac.uk>

Description Functionality to compute the projection median via several algorithms. Also provides functions to plot different multivariate medians and multivariate quantiles in two-dimensional and three-dimensional data respectively. See Chen, F and Nason, G P (2020) ``A new method for computing the projection median, its influence curve and techniques for the production of projected quantile plots.'' PLOS One <doi:10.1371/journal.pone.0229845>.

License GPL-2

NeedsCompilation yes

Author Fan Chen [aut],
Guy Nason [aut, cre]

Repository CRAN

Date/Publication 2022-05-20 12:50:09 UTC

R topics documented:

`Yamm-package` *Multivariate Methods Based on Projections and Related Concepts*

Description

This package provides functions for computing the projection median. `PmedTrapz` approximates the projection median by the trapezoidal rule, which is only valid for the two- and three-dimensional cases, while `PmedMCInt` use Monte Carlo approximation, and it is valid for any multivariate median. `yamm` provides another method to compute the projection median based on an optimiser technique. This package also provides functions for plotting different multivariate medians, such as the Spatial, Component-wise, Tukey's, etc., for randomly generated data sets in both the two-dimensional and three-dimensional cases. In addition, this package also allows users to produce the two-dimensional and three-dimensional quantile plots with function `muqie` and `muqie3D` respectively.

Details

The DESCRIPTION file:

Package:	Yamm
Title:	Multivariate Methods Based on Projections and Related Concepts
Version:	1.3.2
Date:	2022-04-19
Authors@R:	c(person("Fan", "Chen", role=c("aut"),email="fan.chen1018@gmail.com"), person("Guy", "Nason",role=c("au
Depends:	R (>= 3.0), depth, robustX, interp
Suggests:	animation
Maintainer:	Guy Nason <g.nason@imperial.ac.uk>
Description:	Functionality to compute the projection median via several algorithms. Also provides functions to plot differen
License:	GPL-2
Author:	Fan Chen [aut], Guy Nason [aut, cre]

Index of help topics:

```
Plot2dMedian            Plot Two-dimensional Medians
Plot3dMedian            Plot Three-dimensional Medians
PmedMCInt               Projection Median Approximated by Monte Carlo
                        Integration
PmedTrapz               Projection Median Approximated by Trapezoidal
                        Rule
Yamm-package            Multivariate Methods Based on Projections and
                        Related Concepts
```

beetle	Six Measurements of Beetles
clusters2d	Three Clusters of 2-dimensional Data
clusters3d	Four Clusters of 3-dimensional Data
makeplot	Plot Two-dimensional Quantile
makeplot3D	Plot Three-dimensional Quantile
muqie	Two-dimensional Quantile
muqie3D	Three-dimensional Quantile
yamm	Yet Another Multivariate Median
yamm.obj	Objective Function for Yamm

Author(s)

NA

Maintainer: Guy Nason <g.nason@imperial.ac.uk>

References

Basu, R., Bhattacharya, B.B., and Talukdar, T. (2012) The projection median of a set of points in Rd *CCCG.*, **47**, 329-346. doi:10.1007/s0045401193806

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

Croux, C., Filzmoser, P., and Oliveira, M., (2007). Algorithms for Projection-Pursuit Robust Principal Component Analysis, *Chemometrics and Intelligent Laboratory Systems*, **87**, 218-225.

Durocher, S. and Kirkpatrick, D. (2009), The projection median of a set of points, *Computational Geometry*, **42**, 364-375.

Rousseeuw, P.J. and Ruts, I. (1996), Algorithm AS 307: Bivariate location depth, *Appl. Stat.-J. Roy. St. C*, **45**, 516-526.

Rousseeuw, P.J. and Ruts, I. (1998), Constructing the bivariate Tukey median, *Stat. Sinica*, **8**, 828-839.

Rousseeuw, P.J., Ruts, I., and Tukey, J.W. (1999), The Bagplot: A Bivariate Boxplot, *The Am. Stat.*, **53**, 382-387.

Struyf, A. and Rousseeuw, P.J. (2000), High-dimensional computation of the deepest location, *Comput. Statist. Data Anal.*, **34**, 415-436.

See Also

yamm, PmedTrapz, PmedMCInt,

Examples

```
# Load a 2-dimensional data set.
data(clusters2d)
#
# Set seed for reproduction.
set.seed(5)
#
```

```
# Projection median approximated by Monte Carlo Integration.
PmedMCInt(clusters2d, nprojs = 30000)
# [1]  4.3369501 -0.1578591
#
#
# Projection median approximated by the trapezoidal rule.
PmedTrapz(clusters2d,no.subinterval=180)
# [1]  4.1556553 -0.3566614
#
#
# Yamm.
set.seed(5)
yamm(clusters2d,nprojs = 2500,reltol=1e-3,doabs=1,full.results=FALSE)
# [1]  4.3871582 -0.1070497
#
#
# Plot 2-D medians
# Remove the outliers of the dataset.
cluster_without_outlier <- clusters2d[c(1:101),]
myxvec <- c(min(cluster_without_outlier[,1]),
          max(cluster_without_outlier[,1]))
myyvec <- c(min(cluster_without_outlier[,2]),
          max(cluster_without_outlier[,2]))
#
# Plot the figure.
set.seed(5)
Plot2dMedian(clusters2d, myxvec, myyvec, yamm.nprojs = 2000,
             PmedMCInt.nprojs = 20000, no.subinterval = 36,
             opt.method = "BFGS", xlab = "Component1",
             ylab = "Component2")
```

`beetle` *Six Measurements of Beetles*

Description

Multivariate dataset containing six measurements on each of three species of flea-beetles: concinna, heptapotamica, and heikertingeri. The original data set contains one column identifying the species of the observations, which is irrelevant and has been deleted here.

Usage

```
data("beetle")
```

Format

A data frame with 74 observations on the following 6 variables.

`tars1` Width of the first joint of the first tarsus in microns (the sum of measurements for both tarsi).

`tars2` The same for the second joint.

`head` The maximal width of the head between the external edges of the eyes in 0.01 mm.

`aede1` The maximal width of the aedeagus in the fore-part in microns.

`aede2` The front angle of the aedeagus (1 unit = 7.5 degrees).

`aede3` The aedeagus width from the side in microns.

Source

Lubischew, A.A.(1962) On the Use of Discriminant Functions in Taxonomy, *Biometrics*,**18**, 455-477.

References

Cook, D.H. and Swayne, D.F. (2007). Interactive and Dynamic Graphics for Data Analysis: With Examples Using R and GGobi. http://ggobi.org/book/data/flea.xml

Examples

```
data(beetle)
```

---

clusters2d *Three Clusters of 2-dimensional Data*

Description

This dataset with 103 observations contains three clusters, which are generated from different independent normal distributions randomly, and two outliers (located in the last two rows).

Usage

```
data("clusters2d")
```

Format

The first cluster has 26 observations, and the two variables are generated from $N(3,1)$ and $N(4,1)$ respectively. The second cluster has 36 observations, and the two variables are generated from $N(10,1.5)$ and $N(-2,1.5)$ respectively. The third cluster has 39 observations, and the two variables are generated from $N(2,0.5)$ and $N(-2,0.5)$ respectively. The two outliers are $c(100.3,99.1)$ and $c(97.5,98.4)$.

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

Examples

```
data(clusters2d)
```

clusters3d *Four Clusters of 3-dimensional Data*

Description

This dataset with 105 observations contains four clusters, which are generated from different Laplace distributions randomly, and five outliers (located in the last five rows).

Usage

```
data("clusters3d")
```

Format

The four clusters are generated from different multivariate Laplace distributions. The first cluster has 20 observations, where the mean values μ of the Laplace distribution are equal to $(-8, -8, -8)$ and the covariance matrix Σ is the product of two times identity matrix. The second cluster has 35 observations, where $\mu = (-5, 5, 5)$ and Σ is the identity matrix. The third cluster has 30 observations, where $\mu = (12, -12, 12)$ and Σ is the identity matrix. The fourth cluster has 30 observations, where $\mu = (18, 18, -18)$ and Σ is the identity matrix. The five outliers are from the $\mu = (100, 100, -100)$ and Σ is the product of ten times identity matrix.

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

Examples

```
data(clusters3d)
```

makeplot *Plot Two-dimensional Quantile*

Description

This function calls `muqie` for multiple values of quantiles from 0.5 to 0.95 and then produces a set of plots with these quantiles for producing an animated GIF using package **animation**.

Usage

```
makeplot(xdata, dm=c(1,2), nsegs=20,
         quantile.increment= 0.001,
         nprojs=2000, reltol=0.001)
```

Arguments

xdata	The data as a matrix or dataframe with the number of columns greater than or equal to two, with each row being viewed as one multivariate observation.
dm	A numeric vector with two entries representing the selected columns of the data considered. The default value is $c(1, 2)$, which means the first two columns of data are chosen if the dimension of data is greater than two.
nsegs	The number of the unit-length direction vectors u, which is computed by dividing a unit circle into nsegs equal sectors.
quantile.increment	A numeric value specifies the increment of the set of different quantiles.
nprojs	The number of projections for the dataset when computing yamm. The default value is 2000.
reltol	The tolerance of the optimisation process in the function yamm. The default value is 0.001.

Value

This function returns a set of plots with various specified quantiles.

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

See Also

yamm muqie

Examples

```
# Load a data frame with 103 rows and 2 columns.
# The last two rows of the data are the outliers.
data(clusters2d)
#
# Remove the outliers of the dataset.
cluster_without_outlier <- clusters2d[c(1:101),]
#
# Produce an animation of a set of multivariate quantile plots.

if (requireNamespace("animation")) {

library("animation")
# Generate temporary file
f <- tempfile(fileext=".gif")
#
# Now generate movie into the temporary file.
# Here nprojs=40: for a real example, for production quality you should increase
```

```
# it to 1000, 2000 or even higher
#
# Here quantile.increment=0.1, for production quality this should be reduced to
# e.g. 0.01, of even smaller
#
saveGIF(makeplot(cluster_without_outlier, nprojs=40, quantile.increment=0.1),
diff.col=3, interval=0.1,width=500, height=500, movie.name=f)
cat("Movie saved to: ", f, "\n")
}
```

`makeplot3D` *Plot Three-dimensional Quantile*

Description

This function calls `muqie3D` for multiple values of quantiles from 0.5 to 0.95 and then produces a set of perspective plots of a surface over the x-y plane with these quantiles, which are used to produce an animated GIF using package **animation**.

Usage

```
makeplot3D(xdata, dm=c(1,2,3), nsegs=30,
         quantile.increment= 0.005,
         nprojs=2000, reltol=0.001)
```

Arguments

`xdata`	The data as a matrix or dataframe with the number of columns greater than or equal to three, with each row being viewed as one multivariate observation.
`dm`	A numeric vector with three entries representing the selected columns of the data considered. The default value is $c(1, 2, 3)$, which means the first three columns of data are chosen if the dimension of data is more than three.
`nsegs`	The number of the three-dimensional unit-length direction vectors `u`, which is computed by dividing a unit sphere into `nsegs` equal sectors.
`quantile.increment`	A numeric value specifies the increment of the set of different quantiles.
`nprojs`	The number of projections for the dataset when computing `yamm`. The default value is 2000.
`reltol`	The tolerance of the optimisation process in the function `yamm`. The default value is 0.001.

Value

This function returns a set of perspective plots of a surface over the x-y plane with various specified quantiles.

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

See Also

yamm muqie3D

Examples

```
#
data(beetle)
#
# Produce an animation of a set of multivariate quantile plots.
if (requireNamespace("animation")) {

library("animation")
# Generate temporary file
f <- tempfile(fileext=".gif")
#
# Now generate movie into the temporary file.
# Here nprojs=40: for a real example, for production quality you should increase
# it to 1000, 2000 or even higher
#
# Here quantile.increment=0.1, for production quality this should be reduced to
# e.g. 0.01, of even smaller
#
saveGIF(makeplot3D(beetle, dm=c(1,3,6), nprojs=40, quantile.increment=0.1),
diff.col=3, interval=0.1,width=500, height=500, movie.name=f)
cat("Movie saved to: ", f, "\n")
}
```

---

muqie *Two-dimensional Quantile*

Description

This function plots the collection of all MUltivariate QuantIlE points in two dimensions (muqie) over all unit-length direction vectors u, which projects the yamm-centred multivariate data onto the chosen vector u to obtain a univariate set. The muqie point is merely the vector u rescaled to have length equal to the quantile of the univariate set.

Usage

```
muqie(xdata, dm=c(1,2), probs=0.5, nsegs=20,
      nprojs=2000, reltol=0.001, plot.it=FALSE,
      full.return=FALSE, xlab=NULL, ylab=NULL)
```

Arguments

xdata	The data as a matrix or dataframe with the number of columns greater than or equal to two, with each row being viewed as one multivariate observation.
dm	A numeric vector with two entries representing the selected columns of the data considered. The default value is $c(1, 2)$, which means the first two columns of data are chosen if the dimension of data is more than two.
probs	The quantile of the data after projected to obtain a univariate set.
nsegs	The number of the two-dimensional unit-length direction vectors u, which is computed by dividing a unit circle into nsegs equal sectors.
nprojs	The number of projections for the dataset when computing yamm. The default value is 2000.
reltol	The tolerance of the optimisation process in the function yamm. The default value is 0.001.
plot.it	Logical. If TRUE, the function muqie will produce a two-dimensional quantile plot.
full.return	Logical. If TRUE, the function muqie will return a list of full results. See "Value".
xlab	x-axis label for the quantile plot.
ylab	y-axis label for the quantile plot.

Value

If full.results = TRUE, it returns a list comprising of

ans	A data matrix with four rows. The first row represents the angle of the unit-length projection vector u to the positive x-axis, while the second and third row are the x- and y-coordinates of the projection vector respectively. The last row is univariate quantile of the projected data matrix.
uvd	A data matrix after projecting the yamm-centred multivariate data onto a set of projection vectors u.
cdata	The yamm-centred multivariate data matrix.
yamm	The yamm value of the multivariate data. See yamm for more details.

If full.results = FALSE (default), it will only return ans.

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

See Also

yamm

Examples

```
data(beetle)
#
# Compute the 0.7-quantile for the first two columns of the beetle data.
muqie(beetle,dm=c(1,4), probs=0.7)
```

---

muqie3D *Three-dimensional Quantile*

Description

This function plots the collection of all MUltivariate QuantIlE points in three dimensions (muqie3D) over all unit-length direction vectors u, which projects the yamm-centred multivariate data onto the chosen vector u to obtain a univariate set. The muqie3D point is merely the vector u rescaled to have length equal to the quantile of the univariate set.

Usage

```
muqie3D (xdata, dm=c(1,2,3), probs=0.5,
         nsegs=30, nprojs=2000, reltol=0.001,
         plot.it=FALSE, full.return=FALSE)
```

Arguments

xdata	The data as a matrix or dataframe with the number of columns greater than or equal to three, with each row being viewed as one multivariate observation.
dm	A numeric vector with three entries representing the selected columns of the data considered. The default value is $c(1, 2, 3)$, which means the first three columns of data are chosen if the dimension of data is more than three.
probs	The quantile of the data after projected to obtain a univariate set.
nsegs	The number of the three-dimensional unit-length direction vectors u, which is computed by dividing a unit sphere into nsegs equal sectors.
nprojs	The number of projections for the dataset when computing yamm. The default value is 2000.
reltol	The tolerance of the optimisation process in the function yamm. The default value is 0.001.
plot.it	Logical. If TRUE, the function muqie will produce a three-dimensional quantile plot.
full.return	Logical. If TRUE, the function muqie will return a list of full results. See "Value".

Value

If `full.results` = `TRUE`, it returns a list comprising of

`ans`	A data matrix with four rows. The first three rows represent the x-, y- and z-coordinates of the projection vector `u` respectively. The last row is univariate quantile of the projected data matrix.
`uvd`	A data matrix after projecting the `yamm`-centred multivariate data onto a set of projection vectors `u`.
`cdata`	The `yamm`-centred multivariate data matrix.
`yamm`	The yamm value of the multivariate data. See `yamm` for more details.

If `full.results` = `FALSE` (default), it will only return `ans`.

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

See Also

`yamm`

Examples

```
data(beetle)
#
# Compute the 0.7-quantile for the first three columns of the beetle data.
muqie3D(beetle, dm=c(1,3,6), probs=0.7)
```

Plot2dMedian *Plot Two-dimensional Medians*

Description

This function plots various multivariate medians in the two-dimensional case. The grey dots presented in the figure are the data points and the Spatial, Component-wise (CWmed), Tukey's, Liu's, Projection median as well as the mean value of the data set are plotted in the figure. Oja's median is no longer used, as the package used to produce this function has been archived

Usage

```
Plot2dMedian(data, xvec, yvec, yamm.nprojs = 2000,
             PmedMCInt.nprojs = 20000,
             no.subinterval = 36, opt.method = "BFGS",
             xlab = "Component1", ylab = "Component2")
```

Arguments

data	The data as a matrix or data frame, with each row being viewed as one multivariate observation.
xvec	A numeric vector containing the maximum and minimum values you desire for the x-axis.
yvec	A numeric vector containing the maximum and minimum values you desire for the y-axis.
yamm.nprojs	The number of projections for the dataset when computing yamm. The default value is 2000.
PmedMCInt.nprojs	The number of projections for the dataset when computing PmedMCInt. The default value is 20000, since PmedMCInt requires large number of projections while doing the Monte Carlo integration to ensure accuracy.
no.subinterval	The number of subintervals while using the trapezoidal rule to approximate the projection median with PmedTrapz function. The default value is 36, and small values (e.g. less than 10) of no.subinterval should not be used, to safeguard accuracy.
opt.method	The method chosen for the optimiser when computing the yamm, with default function "BFGS". optim is used to minimise the objective function yamm.obj. Apart from "BFGS", other functions in optim like "Nelder-Mead", "CG", "L-BFGS-B", and "SANN" can also be used.
xlab	Title for x-axis. Must be a character string.
ylab	Title for y-axis. Must be a character string.

Details

The Spatial median is obtained using L1median in the Rpackage **robustX**. The Component-wise (CWmed), Liu's and Tukey's median are produced using function med in the Rpackage **depth**. Oja's median is NO LONGER produced using function ojaMedian in the Rpackage OjaNP, as this package has been archived. When computing the projection median, three approximations are implemented and diplayed in the plot, where PmedMCInt uses Monte Carlo method, PmedTrapz is computed by the trapezoidal rule, and yamm uses an optimiser.

The argument xvec and yvec are useful when there are outliers in the data set, which are not expected to be shown in the figure in some cases. Determining the x-axis and y-axis allows you to zoom in the plot and see the difference between multivariate medians and mean value.

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

See Also

PmedTrapz, PmedMCInt, yamm, yamm.obj, optim.

Examples

```
# Load a data frame with 103 rows and 2 columns.
# The last two rows of the data are the outliers.
data(clusters2d)
#
# Remove the outliers of the dataset.
cluster_without_outlier <- clusters2d[c(1:101),]
myxvec <- c(min(cluster_without_outlier[,1]),
          max(cluster_without_outlier[,1]))
myyvec <- c(min(cluster_without_outlier[,2]),
          max(cluster_without_outlier[,2]))
#
# Plot the figure.
set.seed(5)
Plot2dMedian(clusters2d, myxvec, myyvec, yamm.nprojs = 2000,
             PmedMCInt.nprojs = 20000, no.subinterval = 36,
             opt.method = "BFGS", xlab = "Component1",
             ylab = "Component2")
```

Plot3dMedian *Plot Three-dimensional Medians*

Description

This function plots multivariate medians in the three-dimensional case. The grey dots presented in the figure are the data points and the Spatial, Component-wise (CWmed), Tukey's, Liu's, Projection medians as well as the mean value of the data set are plotted in the figure. Oja's median is no longer used, as the package used to compute it has been archived from CRAN.

Usage

```
Plot3dMedian(data, xvec, yvec, zvec, yamm.nprojs = 2000,
             PmedMCInt.nprojs = 20000, no.subinterval = c(18,36),
             opt.method = "BFGS", xlab = "Component1",
             ylab = "Component2", zlab = "Component3")
```

Arguments

data	The data as a matrix or data frame, with each row being viewed as one multivariate observation.
xvec	A numeric vector containing the maximum and minimum values you desire for the x-axis.
yvec	A numeric vector containing the maximum and minimum values you desire for the y-axis.
zvec	A numeric vector containing the maximum and minimum values you desire for the z-axis.

`yamm.nprojs` The number of projections for the dataset when computing `yamm`. The default value is 2000.

`PmedMCInt.nprojs` The number of projections for the dataset when computing `PmedMCInt`. The default value is 20000, since `PmedMCInt` requires large number of projections while doing the Monte Carlo integration to ensure accuracy.

`no.subinterval` A numeric vector of two entries which represents the number of subintervals chosen while using the trapezoidal rule to approximate the projection median with `PmedTrapz` function. The default vector is $c(36, 36)$. Note small values (e.g. less than 10) for each entry of `no.subinterval` should not be used, to safeguard accuracy.

`opt.method` The method chosen for the optimiser when computing the `yamm`, with default function "`BFGS`". `optim` is used to minimise the objective function `yamm.obj`. Apart from `BFGS`, other functions in `optim` like "`Nelder-Mead`", "`CG`", "`L-BFGS-B`", and "`SANN`" can also be used.

`xlab` Title for x-axis. Must be a character string.

`ylab` Title for y-axis. Must be a character string.

`zlab` Title for z-axis. Must be a character string.

Details

The Spatial median is obtained using `L1median` in the Rpackage **robustX**. The Component-wise (CWmed), and Tukey's median are produced using function `med` in the Rpackage **depth**. Oja's median is no longer computed here. Liu's median is not available in higher dimensions (> 2), so it is not shown here. When computing the projection median, three approximations are implemented and diplayed in the plot, where `PmedMCInt` uses Monte Carlo method, `PmedTrapz` is computed by the trapezoidal rule, and `yamm` uses an optimiser.

The argument `xvec`, `yvec` and `zvec` are useful when there are outliers in the dataset, which are not expected to be shown in the figure in some cases. Determining the x-axis y-axis, and z-axis allows you to zoom in the plot and see the difference between multivariate medians and mean value.

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

See Also

`PmedMCInt`, `PmedTrapz` `yamm`, `yamm.obj`, `optim`.

Examples

```
# Load a data frame with 105 rows and 3 columns.
# The last five rows of the data are the outliers.
data(clusters3d)
#
# Remove the outliers of the dataset.
```

```
cluster_without_outlier <- clusters3d[c(1:100),]
myxvec <- c(min(cluster_without_outlier[,1]),
          max(cluster_without_outlier[,1]))
myyvec <- c(min(cluster_without_outlier[,2]),
          max(cluster_without_outlier[,2]))
myzvec <- c(min(cluster_without_outlier[,3]),
          max(cluster_without_outlier[,3]))
#
# Plot the figure.
set.seed(15)
Plot3dMedian(cluster_without_outlier, myxvec, myyvec, myzvec,
             yamm.nprojs = 2000, PmedMCInt.nprojs = 15000,
             no.subinterval = c(18,36),opt.method = "BFGS",
             xlab = "Component1",ylab = "Component2",
             zlab = "Component3")
```

PmedMCInt *Projection Median Approximated by Monte Carlo Integration*

Description

This function approximates the projection median using Monte Carlo integration, which can be used for any dimensions. PmedMCInt is implemented internally using C code CPmedMCInt and hence is much faster than coding with R only.

Usage

```
PmedMCInt(x, nprojs = 20000)
```

Arguments

x	The data as a matrix or data frame, with each row being viewed as one multivariate observation.
nprojs	The number of projections when using the Monte Carlo method to approximate the integration. The default value is 20000, since PmedMCInt requires large a number of projections to ensure the accuracy. More projections may increase the accuracy, as well as the running time.

Details

The projection median was introduced by Durocher and Kirkpatrick (2009) and generalised by Basu, Bhattacharya and Talukdar (2012). PmedMCInt produces the projection median using Monte Carlo approximation, which is valid in any multi-dimensional data. However, a large number of projections is sometimes required to ensure accuracy, which will also increase the running time. In this case, PmedTrapz is preferred for the two- or three-dimensional data, which is fast and accurate in general. In higher dimensions, yamm is another alternative for computing the projection median.

Value

A vector of the projection median for n-dimensional data.

References

Durocher, S. and Kirkpatrick, D. (2009), The projection median of a set of points, *Computational Geometry*,**42**, 364-375.

Basu, R., Bhattacharya, B.B., and Talukdar, T. (2012) The projection median of a set of points in Rd *CCCG.*, **47**, 329-346. doi:10.1007/s0045401193806

See Also

PmedTrapz, yamm

Examples

```
# Load a 2-dimensional data set
data(clusters2d)
#
# Set seed for reproduction.
set.seed(5)
#
# Projection median approximated by Monte Carlo Integration.
PmedMCInt(clusters2d, nprojs = 50000)
# [1]  4.3246488 -0.1535201
#
#
# Load a 6-dimensional data set
data(beetle)
#
set.seed(5)
PmedMCInt(beetle,nprojs = 150000)
# [1] 179.92439 125.16939  50.01176 136.55460  13.22277  95.04224
```

PmedTrapz *Projection Median Approximated by Trapezoidal Rule*

Description

This function approximates the projection median using trapezoidal rule, which is only valid for the two- and three-dimensional cases. PmedTrapz is implemented internally using C code CPmedTrapz2D and CPmedTrapz3D and hence is much faster than coding with R only.

Usage

```
PmedTrapz(x, no.subinterval)
```

Arguments

x	The data as a matrix or data frame, with each row being viewed as one multivariate observation.
no.subinterval	A vector of subintervals chosen for implementing the trapezoidal rule. It is a number in the two-dimensional case, and has a length of two for the three-dimensional data, since the trapezoidal rule is only required once in 2D and needs to be applied twice for the double integral in 3D. Small values (e.g. less than 10) for each entry of no.subinterval should not be used, to safeguard the accuracy.

Details

The projection median was introduced by Durocher and Kirkpatrick (2009) and generalised by Basu, Bhattacharya and Talukdar (2012). PmedTrapz produces the projection median directly from the definition using the trapezoidal rule, but current function is only valid in the two-dimensional and three-dimensional case. For more general dimensionalities, you can refer to function PmedMCInt and yamm.

Value

A vector of the projection median in the two or three dimensions.

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

Durocher, S. and Kirkpatrick, D. (2009), The projection median of a set of points, *Computational Geometry*,**42**, 364-375.

Basu, R., Bhattacharya, B.B., and Talukdar, T. (2012) The projection median of a set of points in Rd *CCCG.*, **47**, 329-346. doi:10.1007/s0045401193806

See Also

PmedMCInt

Examples

```
# Load a 2-dimensional dataset
data(clusters2d)
#
# Projection median approximated by the trapezoidal rule.
PmedTrapz(clusters2d,no.subinterval=180)
# [1]  4.1556553 -0.3566614
#
# Load a 3-dimensional dataset
data(clusters3d)
#
PmedTrapz(clusters3d,c(180,360))
```

```
# [1] -0.906680  1.584866  2.695584
```

---

yamm *Yet Another Multivariate Median*

Description

Another method for computing the projection median for any dimensional dataset. Basically, it minimises the objective function yamm.obj over a unit hypersphere and finds the optimal shift vector mu in yamm.obj. optim in the **stats** package is used in this function to minimise yamm.obj.

Usage

```
yamm(x, nprojs = 2000, reltol = 1e-6, abstol=-Inf,
     xstart = L1median(x)$estimate,
     opt.method = "BFGS", doabs = 0, full.results=FALSE)
```

Arguments

x	The data as a matrix or data frame, with each row being viewed as one multivariate observation.
nprojs	The number of projections for the shifted data matrix while using the Monte Carlo method to approximate the integration. The default value is 2000, more projections may be required for complicated data to ensure accuracy, which, however, increases the running time.
reltol	The tolerance of the optimisation process gets supplied to control arguments of optim. The default value is $1e-6$. Loosening tolerance will make the running process faster. Generally, $1e-3$ is enough to obtain a good approximation for a short running time.
abstol	The absolute convergence tolerance of the optimisation process gets supplied to control arguments of optim. The default value is negative infinity.
xstart	The starting value for the optimiser. The default value is Spatial median of the data using function L1median. Other multivariate medians or mean values can also be used. Note, you should be aware of the outliers when using the mean values as a starting point, which may slow down the optimisation process or result in a less accurate median.
opt.method	The method chosen for the optimiser when computing the yamm, with default function "BFGS". Apart from "BFGS", other functions in optim like "Nelder-Mead", "CG", "L-BFGS-B", and "SANN" can also be used.
doabs	If 0 (default), the function yamm.obj integrates the square of the univariate median of the projection to the shifted data set over a unit hypersphere; if 1, yamm.obj integrates the absolute value of the univariate median instead.
full.results	Logical. If FALSE (default), the function yamm only returns the best set of yamm location estimator found; if TRUE, a list of full reults from the function optim is displayed.

Value

If `full.results = FALSE`, it returns the best set of yamm location estimator found, otherwise, it returns a list comprising of

`par`	The best set of parameters found, which is the yamm location estimator.
`value`	The value of objective function `yamm.obj` corresponding to `par`.
`counts`	A two-element integer vector giving the number of calls to the objective function and gradient of the function respectively. This excludes those calls needed to compute the Hessian, if requested, and any calls to the objective function to compute a finite-difference approximation to the gradient.
`convergence`	An integer code. 0 indicates successful completion (which is always the case for method "`SANN`" and "`Brent`"). Possible error codes are 1 indicates that the iteration limit had been reached. 10 indicates degeneracy of the `Nelder-Mead` simplex. 51 indicates a warning from the "`L-BFGS-B`" method; see component `message` for further details. 52 indicates an error from the "`L-BFGS-B`" method; see component `message` for further details.
`message`	A character string giving any additional information returned by the optimiser, or `NULL`

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

See Also

`yamm.obj`, `optim`.

Examples

```
data(beetle)
#
# Set seed for reproduction.
set.seed(5)
#
# Yamm approximated using 1000 projections.
yamm(beetle,nprojs = 1000,reltol=1e-3,doabs=0,full.results=TRUE)
#
# $par
# [1] 180.30601 124.23781  50.16349 135.53947  13.45252  95.64742
#
# $value
# [1] 5.704375
#
# $counts
```

```
# function gradient
#      69        4
#
# $convergence
# [1] 0
#
# $message
# NULL
```

---

yamm.obj *Objective Function for Yamm*

Description

The objective function when computing yamm, which is the integral of the squared or absolute value of the univariate median of the projection of the shifted data set over a unit hypersphere. It is implemented internally using C code Cyammobj and hence is much faster than coding with R only.

Usage

```
yamm.obj(x, mu, nprojs = 2000, doabs = 0)
```

Arguments

x	The data as a matrix or data frame, with each row being viewed as one multivariate observation.
mu	A shift vector with length n, where n should equal to the number of columns (variables) of the data matrix. Each row of the data matrix x is shifted by mu to obtain the shifted data matrix.
nprojs	The number of projections for the shifted data matrix while using the Monte Carlo method to approximate the integration. The default value is 2000.
doabs	If 0 (default), function yamm.obj integrates square of the univariate median of the projection to the shifted data set over a unit hypersphere; if 1, yamm.obj integrates absolute value of the univariate median instead.

Value

A univariate integral of the squared or absolute value of the median of the projection of the shifted data set over a unit hypersphere is returned from the .C calling function

References

Chen, F. and Nason, Guy P. (2020) A new method for computing the projection medi an, its influence curve and techniques for the production of projected quantile plots. *PLOS One*, doi:10.1371/journal.pone.0229845

See Also

yamm

Examples

```
data(beetle)
#
# Set seed for reproduction.
set.seed(5)
#
# Objective function for yamm with a chosen shift vector.
#
yamm.obj(beetle, mu=rep(10,6), nprojs=5000, doabs=1)
# [1] 88.38346
```

Index