Package ‘Compositional’

February 16, 2016

Version 1.1

Date 2016-02-15

Title Compositional Data Analysis

Author Michail Tsagris [aut, cre], Giorgos Athineou [aut]

Maintainer Michail Tsagris <mtsagris@yahoo.gr>

Depends R (>= 3.2.2)

Imports sn, quantreg, mixture, BB, doParallel, parallel, foreach, MASS

Description A collection of R functions for compositional data analysis.

License GPL (>= 2)

URL

NeedsCompilation no

Repository CRAN

Date/Publication 2016-02-16 13:48:04

R topics documented:

Compositional-package 3
alfa 4
alfa.pcr 5
alfa.profile 6
alfa.reg 7
alfa.ridge 9
alfadist 10
alfainv 11
alfapcr.tune 12
alfareg.tune 14
alfaridge.plot 16
alfaridge.tune 17
bic.mixcompnorm 19
bivt.contour 20
comp.den 21

comp.kerncontour 22
comp.reg 24
diri.contour 25
diri.est 26
diri.nr 27
diri.reg 28
esov.compreg 30
fast.alfa 31
frechet 32
glm.pcr 33
glmpcr.tune 35
helm 36
kl.compreg 37
kl.diri 38
mix.compnorm 40
mixnorm.contour 41
mkde 42
mkde.tune_2 44
multivreg 45
multivt 46
norm.contour 47
ols.compreg 48
pcr 49
pcr.tune 51
rcompnorm 52
rcompsn 53
rcompt 54
rdiri 56
ridge.plot 57
ridge.reg 58
ridge.tune 59
rmixcomp 61
rmvnorm 62
rmvt 63
skewnorm.contour 64
spat.med 65
spatmed.reg 66
sscov 67
sym.test 68
ternary 69

Index **71**

`Compositional-package` *This is an R package that provides methods for statistical analysis of compositional data*

Description

Many types of regression and discriminant analysis, contour plots, parameter estimation and more functions are included. As a general rule, keep in mind that whenever the log-ratio transformation (additive or isometric) is to be used the compositional data must have no zero values. For all other cases, feel free to have zeros.

Details

Package:	Compositional
Type:	Package
Version:	1
Date:	2016-02-05
License:	GPL-2

Maintainers

Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

Note

Acknowledgments:

Professor Andy Wood and Dr Simon Preston from the university of Nottingham are highly appreciated for being my supervisors during my PhD in compositional data analysis.

Author(s)

Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

alfa *The α-transformation*

Description

The α-transformation.

Usage

```
alfa(x, a, h = TRUE)
```

Arguments

x	A matrix with the compositional data.
a	The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied.
h	A boolean variable. If is TRUE (default value) the multiplication with the Helmert sub-matrix will take place. When $\alpha = 0$ and h = FALSE, the result is the centred log-ratio transformation (Aitchison, 1986). In general, when h = FALSE the resulting transformation maps the data onto a singualr space. The sum of the vectors is equal to 0. Hence, from the simplex constraint the data go to another constraint.

Details

The α-transformation is applied to the compositional data.

Value

A list including:

sa	The logarithm of the Jacobian determinant of the α-transformation. This is used in the "profile" function to speed up the computations.
aff	The α-transformed data.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

Aitchison J. (1986). The statistical analysis of compositional data. Chapman \& Hall.

See Also

alfainv, profile

Examples

```
library(MASS)
x <- fgl[, 2:9]
y1 <- alfa(x, 0.2)$aff
y2 <- alfa(x, 1)$aff
rbind( colMeans(y1), colMeans(y2) )
y3 <- alfa(x, 0.2)$aff
dim(y1)  ;  dim(y3)
rowSums(y1)
rowSums(y3)
```

---

alfa.pcr — *Multivariate or univariate regression with compositional data in the covariates side using the α-transformation*

Description

Multivariate or univariate regression with compositional data in the covariates side using the α-transformation.

Usage

```
alfa.pcr(y, x, a, k, oiko = "normal", xnew = NULL)
```

Arguments

y	A numerical vector containing the response variable values.
x	The predictor variables, the compositional data.
a	The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied.
k	A number at least equal to 1. How many principal components to use.
oiko	The family of distributions. IT can be either, "normal" for continuous response and hence normal distribution, "binomial" corresponding to binary response and hence logistic regression or "poisson" for count response and poisson regression.
xnew	A matrix containing the new compositional data whose response is to be predicted. If you have no new data, leave this NULL as is by default.

Details

The α-transformation is applied to the compositional data first and then the principal components regression is performed.

Value

The output of the pcr or glm.pcr depending on the type of the response.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M. (2015). Regression analysis with compositional data containing zero values. Chilean Journal of Statistics, 6(2): 47-57. http://arxiv.org/pdf/1508.01913v1.pdf

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

See Also

pcr, glm.pcr, alfapcr.tune

Examples

```
library(MASS)
y <- fgl[, 1]
x <- fgl[, 2:9]
mod <- alfa.pcr(y = y, x = x, 0.7, 1, oiko = "normal")
mod
```

---

alfa.profile — *Estimation of the value of α via the alfa.profile log-likelihood*

Description

Estimation of the value of α via the alfa.profile log-likelihood.

Usage

```
alfa.profile(x, a = seq(-1, 1, by = 0.01))
```

Arguments

x	A matrix with the compositional data. Zero values are not allowed.
a	A grid of values of α.

Details

For every value of α the normal likelihood (see the refernece) is computed. At the end, the plot of the values is constructed.

Value

A list including:

`res`	The chosen value of α, the corresponding log-likelihood value and the log-likelihood when $\alpha = 0$.
`ci`	An asympotic 95% confidence interval computed from the log-likelihood ratio test.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain.

See Also

`fast.alfa`, `alfa`, `alfainv`

Examples

```
library(MASS)
x <- iris[, 1:4]
fast.alfa(x)
alfa.profile(x)
```

`alfa.reg` *Regression with compositional data using the α-transformation*

Description

Regression with compositional data using the α-transformation.

Usage

```
alfa.reg(y, x, a, xnew = NULL)
```

Arguments

y	A matrix with the compositional data.
x	The predictor variable(s), they have to be continuous.
a	The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied and the solution exists in a closed form, since it the classical mutivariate regression.
xnew	If you have new data use it, otherwise leave it NULL.

Details

The α-transformation is applied to the compositional data first and then multivariate regression is applied. This involves numerical optimisation.

Value

A list including:

beta	The beta coefficients.
seb	The standard error of the beta coefficients.
est	The fitted or the predicted values (if xnew is not NULL).

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M. (2015). Regression analysis with compositional data containing zero values. Chilean Journal of Statistics, 6(2): 47-57. http://arxiv.org/pdf/1508.01913v1.pdf

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

Mardia K.V., Kent J.T., and Bibby J.M. (1979). Multivariate analysis. Academic press.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman \& Hall.

See Also

diri.reg, esov.compreg, kl.compreg, ols.compreg, comp.reg

Examples

```
library(MASS)
x <- fgl[1:50, 1]
y <- fgl[1:50, 2:9]
mod1 <- alfa.reg(y, x, 0.2)
mod2 <- alfa.reg(y, x, 1)
```

---

alfa.ridge *Ridge regression with compositional data in the covariates side using the α-transformation*

Description

Ridge regression with compositional data in the covariates side using the α-transformation.

Usage

```
alfa.ridge(y, x, a, lambda, B = 1, xnew = NULL)
```

Arguments

y	A numerical vector containing the response variable values. If they are percentages, they are mapped onto R using the logit transformation.
x	The predictor variables, the compositional data.
a	The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied.
lambda	The value of the regularisation parameter, λ.
B	If B > 1 bootstrap estimation of the standard errors is implemented.
xnew	A matrix containing the new compositional data whose response is to be predicted. If you have no new data, leave this NULL as is by default.

Details

The α-transformation is applied to the compositional data first and then ridge components regression is performed.

Value

The output of the ridge.reg.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M. (2015). Regression analysis with compositional data containing zero values. Chilean Journal of Statistics, 6(2): 47-57. http://arxiv.org/pdf/1508.01913v1.pdf

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

See Also

`ridge.reg`, `alfaridge.tune`, `alfaridge.plot`

Examples

```
library(MASS)
y <- fgl[, 1]
x <- fgl[, 2:9]
mod1 <- alfa.ridge(y, x, a = 0.5, lambda = 0.1, B = 1, xnew = NULL)
mod2 <- alfa.ridge(y, x, a = 0.5, lambda = 1, B = 1, xnew = NULL)
```

`alfadist` *The α-distance*

Description

This is the Euclidean distance after the α-transformation has been applied.

Usage

```
alfadist(x, a)
```

Arguments

`x`	A matrix with the compositional data.
`a`	The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied and the solution exists in a closed form, since it the classical mutivariate regression.

Details

The α-transformation is applied to the compositional data first and then the Euclidean distance is calculated.

Value

A matrix including: The pairwise distances of all observations.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M.T., Preston S. and Wood A.T.A. (2016). Improved classification for compositional data using the α-transformation. Journal of Classification (to appear). http://arxiv.org/pdf/1506.04976v2.pdf

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

See Also

alfa, alfainv, alfa.reg

Examples

```
library(MASS)
x <- fgl[1:20, 2:9]
alfadist(x, 0.1)
alfadist(x, 1)
```

---

alfainv *The inverse of the α-transformation*

Description

The inverse of the α-transformation.

Usage

```
alfainv(x, a, h = TRUE)
```

Arguments

x	A matrix with Euclidean data. However, they must lie within the feasible, acceptable space. See references for more information.
a	The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$, the inverse of the isometric log-ratio transformation is applied.
h	If h = TRUE this means that the multiplication with the Helmer sub-matrix will take place. It is set to TRUe by default.

Details

The inverse of the α-transformation is applied to the data. If the data lie outside the α-space, NAs will be returned for some values.

Value

A matrix with the pairwise distances.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M.T., Preston S. and Wood A.T.A. (2016). Improved classification for compositional data using the α-transformation. Journal of Classification (to appear). http://arxiv.org/pdf/1506.04976v2.pdf

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

See Also

`alfa`, `alfadist`

Examples

```
library(MASS)
x <- fgl[1:10, 2:9]
y <- alfa(x, 0.5)$aff
alfainv(y, 0.5)
```

---

alfapcr.tune — *Tuning the number of PCs in the PCR with compositional data using the α-transformation*

Description

This is a cross-validation procedure to decide on the number of principal components when using regression with compositional data (as predictor variables) using the α-transformation.

Usage

```
alfapcr.tune(y, x, M = 10, maxk = 50, a = seq(-1, 1, by = 0.1), oiko = "normal",
seed = FALSE, ncores = 2, graph = TRUE, col.nu = 15)
```

Arguments

`y`	A vector with either continuous, binary or count data.
`x`	The predictor variables, the compositional data. Zero values are allowed.
`M`	The number of folds for the K-fold cross validation, set to 10 by default.
`maxk`	The maximum number of principal components to check.
`a`	The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied and the solution exists in a closed form, since it the classical mutivariate regression. The estimated bias correction via the (Tibshirani and Tibshirani (2009) criterion is applied.
`oiko`	The family to be used, "normal" for a continuous variable, "binomial" for a binary or "poisson" for a count vaiable.
`seed`	If seed is TRUE the same folds will always be generated.
`ncores`	How many cores to use. If you have heavy computations or do not want to wait for long time more than 1 core (if available) is suggested.
`graph`	If graph is TRUE (default value) a filled contour plot will appear.
`col.nu`	A number parameter for the filled contour plot, taken into account only if graph is TRUE.

Details

The α-transformation is applied to the compositional data first and the function "pcr.tune" or "glmpcr.tune" is called. The estimated bias correction via the (Tibshirani and Tibshirani (2009) criterion is applied.

Value

If graph is TRUE a fileld contour a filled contour will appear. A list including:

`mspe`	The MSPE where rows correspond to the α values and the columns to the number of principal components.
`best.par`	The best pair of α and number of principal components.
`performance`	The minimum mean squared error of prediction (bias corrected) and the estimated bias.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M. (2015). Regression analysis with compositional data containing zero values. Chilean Journal of Statistics, 6(2): 47-57. http://arxiv.org/pdf/1508.01913v1.pdf

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

Jolliffe I.T. (2002). Principal Component Analysis.

Tibshirani and Tibshirani (2009). A bias correction for the minimum error rate in cross-validation. The Annals of Applied Statistics, 3(1):822-829.

See Also

alfa, profile, alfa.pcr, pcr.tune, glmpcr.tune, glm

Examples

```
library(MASS)
y <- fgl[, 1]
x <- fgl[, 2:9]
mod <- alfapcr.tune(y, x, M = 10, maxk = 50, a = seq(-1, 1, by = 0.1), oiko = "normal",
seed = FALSE, ncores = 1)
```

---

alfareg.tune — *Tuning the value of α in the α-regression*

Description

Tuning the value of α in the α-regression.

Usage

```
alfareg.tune(y, x, a = seq(0.1, 1, by = 0.1), K = 10, nc = 2, graph = TRUE)
```

Arguments

y	A matrix with the compositional data. zero values are allowed.
x	A matrix with the continuous predictor variables.
a	The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied.
K	The number of folds to split the data.
nc	The number of cores to use. IF you have a multicore computer it is advisable to use more than 1. It makes the procedure faster.
graph	If grpah is TRUE a plot of the performance for each fold along the values of α will appear.

Details

The α-transformation is applied to the compositional data and the numerical otpimiation is performed for the regression, unless $\alpha = 0$, where the coefficients are available in closed form. The estimated bias correction via the Tibshirani and Tibshirani (2009) criterion is applied.

Value

A plot of the estimated Kullback-Leibler divergences (multiplied by 2) along the values of α (if graph is set to TRUE). A list including:

`kl`	Twice the Kullback-Leibler divergence of the observed from the fitted values.
`opt`	The optimal value of α.
`value`	The minimum value of twice the Kullback-Leibler with the estimated bias added.
`bias`	The stimated bias.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris Michail (2015). Regression analysis with compositional data containing zero values. Chilean Journal of Statistics, 6(2): 47-57. http://arxiv.org/pdf/1508.01913v1.pdf

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

Tibshirani and Tibshirani (2009). A bias correction for the minimum error rate in cross-validation. The Annals of Applied Statistics, 3(1):822-829.

See Also

`alfa`, `alfa.reg`

Examples

```
library(MASS)
y <- fgl[1:30, 2:4]
x <- fgl[1:30, 1]
mod <- alfareg.tune(y, x, a = seq(0.2, 0.3), K = 5, nc = 1)
```

alfaridge.plot *Ridge regression plot*

Description

A plot of the regularised regression coefficients is shown.

Usage

```
alfaridge.plot(y, x, a, lambda = seq(0, 5, by = 0.1) )
```

Arguments

`y`	A numeric vector containing the values of the target variable. If the values are proportions or percentages, i.e. strictly within 0 and 1 they are mapped into R using the logit transformation. In any case, they must be continuous only.
`x`	A numeric matrix containing the continuous variables. Rows are samples and columns are features.
`a`	The value of the α-transformation. It has to be between -1 and 1. If there are zero values in the data, you must use a strictly positive value.
`lambda`	A grid of values of the regularisation parameter λ.

Details

For every value of λ the coefficients are obtained. They are plotted versus the λ values.

Value

A plot with the values of the coefficients as a function of λ.

Author(s)

Michail Tsagris

R implementation and documentation: Giorgos Athineou <athineou@csd.uoc.gr> and Michail Tsagris <mtsagris@yahoo.gr>

References

Hoerl A.E. and R.W. Kennard (1970). Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, 12(1): 55-67.

Brown P. J. (1994). Measurement, Regression and Calibration. Oxford Science Publications.

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

See Also

ridge.plot, alfa.ridge

Examples

```
library(MASS)
y <- fgl[, 1]
x <- fgl[, 2:9]
alfaridge.plot(y, x, a = 0.5, lambda = seq(0, 5, by = 0.1) )
```

---

alfaridge.tune *Cross validation for the ridge regression with compositional data as predictor using the α-transformation*

Description

Cross validation for the ridge regression is performed using the TT estimate of bias (Tibshirani and Tibshirani, 2009). There is an option for the GCV criterion which is automatic. The predictor variables are compositional data and the α-transformation is applied first.

Usage

```
alfaridge.tune(y, x, M = 10, a = seq(-1, 1, by = 0.1), lambda = seq(0, 2, by = 0.1),
seed = FALSE, ncores = 2, graph = TRUE, col.nu = 15)
```

Arguments

y	A numeric vector containing the values of the target variable. If the values are proportions or percentages, i.e. strictly within 0 and 1 they are mapped into R using the logit transformation.
x	A numeric matrix containing the compositional data, i.e. the predictor variables.
M	The number of folds. Set to 10 by default.
a	A vector with the a grid of values of α to be used.
lambda	A vector with the a grid of values of λ to be used.
seed	A boolean variable. If it is TRUE the results will always be the same.
ncores	The number of cores to use. If it is more than 1 parallel computing is performed.
graph	If graph is TRUE (default value) a filled contour plot will appear.
col.nu	A number parameter for the filled contour plot, taken into account only if graph is TRUE.

Details

A k-fold cross validation is performed and the estimated performance is bias corrected as suggested by Tibshirani and Tibshirani (2009).

Value

If graph is TRUE a fileld contour a filled contour will appear. A list including:

mspe
: The MSPE where rows correspond to the α values and the columns to the number of principal components.

best.par
: The best pair of α and λ.

performance
: The minimum mean squared error of prediction (bias corrected) and the estimated bias.

Author(s)

Michail Tsagris

R implementation and documentation: Giorgos Athineou <athineou@csd.uoc.gr> and Michail Tsagris <mtsagris@yahoo.gr>

References

Hoerl A.E. and R.W. Kennard (1970). Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, 12(1):55-67.

Brown P. J. (1994). Measurement, Regression and Calibration. Oxford Science Publications.

Tibshirani R.J., and Tibshirani R. (2009). A bias correction for the minimum error rate in cross-validation. The Annals of Applied Statistics 3(2): 822-829.

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

See Also

alfa.ridge, ridge.tune

Examples

```
library(MASS)
y <- fgl[, 1]
x <- fgl[, 2:9]
alfaridge.tune(y, x, M = 10, a = seq(0.1, 0.5, by = 0.1), lambda = seq(0.1, 1, by = 0.2),
seed = FALSE, ncores = 1, graph = TRUE, col.nu = 15)
```

bic.mixcompnorm *Mixture model selection via BIC*

Description

Mixture model selection via BIC.

Usage

```
bic.mixcompnorm(x, A, type = "alr")
```

Arguments

x	A matrix with the compositional data.
A	The maximum number of components, clusters, to be considered.
type	The type of trasformation to be used, either additive log-ratio ("alr") or the isometric log-ratio ("ilr").

Details

The alr or the ilr-transformation is applied to the compositional data first and then mixtures of multivariate Gaussian distributions are fitted. BIC is used to decide on the optimal model and number of components.

Value

a plot with the BIC of the best model for each number of components versus the number of components. A list including:

mod	A message informing the user about the best model.
BIC	The BIC values for every possible model and number of components.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Ryan P. Browne, Aisha ElSherbiny and Paul D. McNicholas (2015). mixture: Mixture Models for Clustering and Classification. R package version 1.4.

Ryan P. Browne and Paul D. McNicholas (2014). Estimating Common Principal Components in High Dimensions. Advances in Data Analysis and Classification, 8(2), 217-226.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

mix.compnorm, mixnorm.contour, rmixcomp

Examples

```
library(MASS)
x <- iris[, 1:4]
bic.mixcompnorm(x, 6, type = "alr")
bic.mixcompnorm(x, 6, type = "ilr")
```

---

bivt.contour *Contour plot of the t distribution in S^2*

Description

Contour plot of the t distribution in S^2.

Usage

```
bivt.contour(x, type = "alr", n = 100, appear = TRUE)
```

Arguments

x	A matrix with the compositional data. It has to be a 3 column matrix.
type	This is either "alr" or "ilr", corresponding to the additive and the isometric log-ratio transformation respectively.
n	The number of grid points to consider over which the density is calculated.
appear	Should the available data appear on the ternary plot (TRUE) or not (FALSE)?

Details

The alr or the ilr transformation is applied to the compositional data at first and the location, scatter and degrees of freedom of the bivariate t distribution are computed. Then for a grid of points within the 2-dimensional simplex the bivariate t density is calculated and the contours are plotted along with the points.

Value

A ternary diagram with the points (if appear = TRUE) and the bivariate t contour lines.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

diri.contour, mixnorm.contour, norm.contour, skewnorm.contour

Examples

```
x <- iris[, 1:3]
x <- x / rowSums(x)
bivt.contour(x)
dev.new()
bivt.contour(x, type = "ilr")
```

---

comp.den *Estimating location and scatter parameters*

Description

Estimating location and scatter parameters in a robust and non robust way.

Usage

```
comp.den(x, type = "alr", dist = "normal")
```

Arguments

x	A matrix containing the compositional data. No zero values are allowed.
type	A boolean variable indicating the transformation to be used. Either "alr" or "ilr" corresponding to the additive or the isometric log-ratio transformation respectively.
dist	Takes values "normal", "t", "skewnorm", "rob" and "spatial". They first three options correspond to the parameters of the normal, t and skew normal distribution respectively. If it set to "rob" the MCD estimates are computed and if set to "spatial" the spatial median and spatial sign covariance matrix are computed.

Details

This function calculates robust and non robust estimates of location and scatter.

Value

A list including: The mean vector and covariance matrix mainly. Other parameters are also returned depending on the value of the argument "dist".

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

P. J. Rousseeuw and K. van Driessen (1999) A fast algorithm for the minimum covariance determinant estimator. Technometrics 41, 212-223.

Mardia K.V., Kent J.T., and Bibby J.M. (1979). Multivariate analysis. Academic press.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

T. Karkkaminen and S. Ayramo (2005). On computation of spatial median for robust data mining. Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems EUROGEN 2005.

A Durre, D Vogel, DE Tyler (2014). The spatial sign covariance matrix with unknown location. Journal of Multivariate Analysis, 130: 107-117.

J. T. Kent, D. E. Tyler and Y. Vardi (1994) A curious likelihood identity for the multivariate t-distribution. Communications in Statistics-Simulation and Computation 23, 441-453.

Azzalini A. and Dalla Valle A. (1996). The multivariate skew-normal distribution. Biometrika 83(4): 715-726.

See Also

spat.med, sscov, multivt

Examples

```
library(MASS)
x <- iris[, 1:4]
x <- x / rowSums(x)
comp.den(x)
comp.den(x, type = "alr", dist = "t")
comp.den(x, type = "alr", dist = "spatial")
```

---

comp.kerncontour *Contour plot of the kernel density estimate in $S\hat{}2$*

Description

Contour plot of the kernel density estimate in S^2.

Usage

```
comp.kerncontour(x, type = "alr", n = 100)
```

Arguments

x	A matrix with the compositional data. It has to be a 3 column matrix.
type	This is either "alr" or "ilr", corresponding to the additive and the isometric log-ratio transformation respectively.
n	The number of grid points to consider, over which the density is calculated.

Details

The alr or the ilr transformation are applied to the compositional data. Then, the optimal bandwidth using maximum likelihood cross-validation is chosen. The multivariate normal kernel density is calculated for a grid of points. Those points are the points on the 2-dimensional simplex. Finally the contours are plotted.

Value

A ternary diagram with the points and the kernel contour lines.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

M.P. Wand and M.C. Jones (1995). Kernel smoothing, CrC Press.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

diri.contour, mixnorm.contour, bivt.contour, norm.contour

Examples

```
x <- iris[, 1:3]
x <- x / rowSums(x)
comp.kerncontour(x, type = "alr", n = 20)
dev.new()
comp.kerncontour(x, type = "ilr", n = 20)
```

comp.reg *Regression with compositional data*

Description

Regression with compositional data.

Usage

```
comp.reg(y, x, type = "classical", xnew = NULL)
```

Arguments

y	A matrix with the compsitional data. Zero values are not allowed.
x	The predictor variable(s), they have to be continuous.
type	The type of regression to be used, "classical" for standard multivariate regression, "spatial" for spatial median regression, which is also robust.
xnew	This is by default set to NULL. If you have new data whose compositional data values you want to predict, put them here.

Details

The additive log-ratio transformation is applied and then the chosen multivariate regression is implemented. The alr is easier to explain than the ilr and that is why the latter is avoided here.

Value

A list including:

beta	The beta coefficients.
seb	The standard error of the beta coefficients.
est	The fitted values if xnew is NULL, or the predicted values otherwise.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Mardia K.V., Kent J.T., and Bibby J.M. (1979). Multivariate analysis. Academic press.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

multivreg, spatmed.reg, esov.compreg, diri.reg

Examples

```
library(MASS)
y <- iris[, 1:3]
x <- iris[, 4]
mod1 <- comp.reg(y, x)
mod2 <- comp.reg(y, x, type = "spatial")
mod1
mod2
```

---

diri.contour *Contour plot of a Dirichlet distribution in $S\hat{}2$*

Description

Contour plot of a Dirichlet distribution in S^2.

Usage

```
diri.contour(a, n = 100, x = NULL)
```

Arguments

a	A matrix with the compositional data. It has to be a 3 column matrix.
n	The number of grid points to consider over which the density is calculated.
x	This is either NULL (no data) or contains a 3 column matrix with compositional data.

Details

The user can plot only the contour lines of a Dirichlet with a given vector of parameters, or can also add the relevant data should he/she wish to.

Value

A ternary diagram with the points and the Dirichlet contour lines.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Ng Kai Wang, Guo-Liang Tian and Man-Lai Tang (2011). Dirichlet and related distributions: Theory, methods and applications. John Wiley \& Sons.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

norm.contour, bivt.contour, comp.kerncontour, mixnorm.contour

Examples

```
x <- iris[, 1:3]
x <- x / rowSums(x)
diri.contour( a = c(3, 4, 2) )
```

---

diri.est *Fiiting a Dirichlet distribution*

Description

Estimation of the parameters of a fitted Dirichlet distribution.

Usage

```
diri.est(x, type = "mle")
```

Arguments

x	A matrix containing the compositional data.
type	If you want to estimate the parameters use type="mle". If you want to estimate the mean vector along with the precision parameter, the second parametrisation of the Dirichlet, use type="prec". If you want to estimate the parameters via the entropy, then use type="ent".

Details

Maximum likelihood estimation of the parameters of a Dirichlet distribution is performed.

Value

A list including:

loglik	The value of the log-likelihood.
param	The estimated parameters.
phi	The estimated precision parameter, if type = "prec".
a	The estimated mean vector, if type = "prec".

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Ng Kai Wang, Guo-Liang Tian and Man-Lai Tang (2011). Dirichlet and related distributions: Theory, methods and applications. John Wiley \& Sons.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

diri.nr, diri.contour

Examples

```
x = rdiri( 100, c(5, 7, 1, 3, 10, 2, 4) )
diri.est(x)
diri.est(x, type = "prec")
diri.est(x, type = "ent")
```

---

diri.nr *Fiiting a Dirichlet distribution via Newton-Rapshon*

Description

Fiiting a Dirichlet distribution via Newton-Rapshon.

Usage

```
diri.nr(x)
```

Arguments

x	A matrix containing the compositional data.

Details

Maximum likelihood estimation of the parameters of a Dirichlet distribution is performed via Newton-Raphson. Initial values suggested by Minka (2003) are used. The esimtation is much faster than "diri.est" when the sample size and or the dimensions increase.

Value

A list including:

loglik	The value of the log-likelihood.
param	The estimated parameters.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Thomas P. Minka (2003). Estimating a Dirichlet distribution. http://research.microsoft.com/en-us/um/people/minka/papers/dirichlet/minka-dirichlet.pdf

See Also

`diri.est`, `diri.contour` `rdiri`

Examples

```
x = rdiri( 100, c(5, 7, 1, 3, 10, 2, 4) )
diri.nr(x)
diri.est(x)
```

---

diri.reg *Dirichlet regression*

Description

Dirichlet regression

Usage

```
diri.reg(y, x, plot = TRUE, xnew = NULL)
```

Arguments

`y`	A matrix with the compositional data (dependent variable). Zero values are not allowed.
`x`	The predictor variable(s), they have to be continuous.
`plot`	A boolean variable specifying whether to plot the leverage values of the observations or not. This is taken into account only when xnew = NULL.
`xnew`	If you have new data use it, otherwise leave it NULL.

Details

A Dirichlet distribution is assumed for the regression. This involves numerical optimisation.

Value

A list including:

loglik	The value of the log-likelihood.
phi	The precision parameter.
log.phi	The logarithm of the precision parameter.
std.logphi	The standard error of the logarithm of the precision parameter.
beta	The beta coefficients.
seb	The standard error of the beta coefficients.
lev	The leverage values.
est	The fitted or the predicted values (if xnew is not NULL).

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Maier, Marco J. (2014) DirichletReg: Dirichlet Regression for Compositional Data in R. Research Report Series/Department of Statistics and Mathematics, 125. WU Vienna University of Economics and Business, Vienna. http://epub.wu.ac.at/4077/1/Report125.pdf

Gueorguieva, Ralitza, Robert Rosenheck, and Daniel Zelterman (2008). Dirichlet component regression and its applications to psychiatric data. Computational statistics & data analysis 52(12): 5344-5355.

See Also

esov.compreg, kl.compreg, ols.compreg, comp.reg, alfa.reg

Examples

```
x <- iris[, 4]
y <- iris[, 1:3]
mod1 <- diri.reg(y, x)
mod2 <- comp.reg(y, x)
mod1
mod2
```

esov.compreg *ESOV regression*

Description

Regression based on the ESOV divergence.

Usage

```
esov.compreg(y, x, B = 1000, ncores = 1, xnew = NULL)
```

Arguments

y	A matrix with the compositional data (dependent variable). Zero values are allowed.
x	The predictor variable(s), they have to be continuous.
B	If B is greater than 1 bootstrap estimates of the standard error are returned. If B=1, no standard errors are returned.
ncores	If ncores is 2 or more parallel computing is performed. This is to be used for the case of bootstrap. If B=1, this is not taken into consideration.
xnew	If you have new data use it, otherwise leave it NULL.

Details

The ESOV metric is adopted as the objective function. This involves numerical optimisation. There is no log-likelihood.

Value

A list including:

beta	The beta coefficients.
seb	The standard error of the beta coefficients, if bootstrap is chosen, i.e. if B > 1.
est	The fitted or the predicted values (if xnew is not NULL).

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Michail Tsagris (2015). A novel, divergence based, regression for compositional data. Proceedings of the 28th Panhellenic Statistics Conference. http://arxiv.org/pdf/1511.07600v1.pdf

Endres, D. M. and Schindelin, J. E. (2003). A new metric for probability distributions. Information Theory, IEEE Transactions on, 49(7):1858-1860.

Osterreicher, F. and Vajda, I. (2003). A new class of metric divergences on probability spaces and its applicability in statistics. Annals of the Institute of Statistical Mathematics, 55(3):639-653.

See Also

`diri.reg`, `kl.compreg`, `ols.compreg`, `comp.reg`, `alfa.reg`

Examples

```
library(MASS)
x <- fgl[, 1]
y <- fgl[, 2:9]
mod1 <- esov.compreg(y, x, B = 1, ncores = 1)
mod2 <- kl.compreg(y, x, B = 1, ncores = 1)
mod1
mod2
```

`fast.alfa` *Fast estimation of the value of α*

Description

Fast estimation of the value of α.

Usage

```
fast.alfa(x, B = 1, ncores = 1)
```

Arguments

`x`	A matrix with the compositional data. No zero vaues are allowed.
`B`	If no (bootstrap based) confidence intervals should be returned this should be 1 and more than 1 otherwise.
`ncores`	If ncores is greater than 1 parallel computing is performed.

Details

This is a faster function than `profile` for choosing the value of α.

Value

A vector with the best alpha, the maximised log-likelihood and the log-likelihood at $\alpha = 0$, when B = 1 (no bootstrap) or a list including:

- `param` The best alpha and the value of the log-likelihod, along with the 95% bootstrap based confidence intervals if B is greater than 1.
- `message` A message with some information about the histogram if B is greater than 1.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

See Also

`alfa.profile`, `alfa`, `alfainv`

Examples

```
library(MASS)
x <- iris[, 1:4]
fast.alfa(x)
alfa.profile(x)
```

---

frechet *The Frechet mean*

Description

Mean vector using the α-transformation.

Usage

```
frechet(x, a)
```

Arguments

- `x` A matrix with the compositional data.
- `a` The value of the power transformation, it has to be between -1 and 1. If zero values are present it has to be greater than 0. If $\alpha = 0$ the isometric log-ratio transformation is applied and the closed geometric mean is calculated.

Details

The power transformation is applied to the compositional data and the mean vector is calculated. Then the inverse of it is calculated and the inverse of the power transformation applied to the last vector is the Frechet mean.

Value

A vector with the Frechet mean for the given value of α.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain.

See Also

alfa, alfainv, profile

Examples

```
library(MASS)
x <- fgl[, 2:9]
frechet(x, 0.2)
frechet(x, 1)
```

---

glm.pcr *Principal component generalised linear models*

Description

Principal component generalised linear models.

Usage

```
glm.pcr(y, x, k, oiko = "binomial", xnew = NULL)
```

Arguments

`y`	A numerical vector, either 0 and 1 (binary) or discrete (count) data.
`x`	The predictor variable(s), they have to be continuous.
`k`	A number at least equal to 1. How many principal components to use.
`oiko`	The type of regression to be used, "binomial" for binary response or "poisson" for count response.
`xnew`	If you have new data use it, otherwise leave it NULL.

Details

Principal component regression is performed with binary logistic or Poisson regression.

Value

A list including:

`model`	The summary of the glm model
`per`	The percentage of variance of the predictor variables retained by the k principal components.
`est`	The fitted or the predicted values (if xnew is not NULL).

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Jolliffe I.T. (2002). Principal Component Analysis.

See Also

`pcr`, `alfa.pcr`, `alfapcr.tune`, `glm`

Examples

```
library(MASS)
x<- iris[, 1:4]
y<- rbinom(150, 1, 0.6)
mod<- glm.pcr(y, x, k = 1, oiko = binomial)
```

glmpcr.tune *Tuning the principal components in the GLM*

Description

Tuning the number of principal components in the generalised linear models.

Usage

```
glmpcr.tune(y, x, M = 10, maxk = 10, oiko = "binomial", seed = FALSE, ncores = 2,
graph = TRUE)
```

Arguments

y	A real valued vector.
x	The predictor variables, they have to be continuous.
M	The number of folds in the cross validation.
maxk	The maximum number of principal components to check.
oiko	The type of regression to be used, "binomial" if you have binary data or "poisson" if you have count data.
seed	If seed is TRUE the folds will always be the same.
ncores	The number of cores to use. If more than 1, parallel computing takes place.
graph	If grpah is TRUE a plot of the performance for each fold along the values of α will appear.

Details

Cross validation is performed to select the optimal number of principal components in the GLM. This is used by `alfapcr.tune`.

Value

If graph is TRUE a plot of the performance versus the number of principal components will appear. A list including:

msp	A matrix with the mean squared error of prediction (MSPE) for every fold.
mpd	A vector with the mean squared error of prediction (MSPE), each value corresponds to a number of principal components.
k	The number of principal components which minimizes the MSPE.
performance	The bias corrected lowest value of the MSPE along with the estimated bias via the Tibshirani and Tibshirani (2009) suggestion.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Jolliffe I.T. (2002). Principal Component Analysis.

Tibshirani and Tibshirani (2009). A bias correction for the minimum error rate in cross-validation. The Annals of Applied Statistics, 3(1):822-829.

See Also

`pcr.tune`, `glm.pcr`, `alfa.pcr`, `alfapcr.tune`

Examples

```
library(MASS)
x <- fgl[, 2:9]
y <- rpois(214, 10)
glmpcr.tune(y, x, M=10, maxk = 20, oiko = "poisson", seed = FALSE, ncores = 1)
```

---

helm *The Helmert sub-matrix*

Description

The Helmert sub-matrix.

Usage

```
helm(n)
```

Arguments

n	A number grater than or equal to 2.

Details

The Helmert sub-matrix is returned. It is an orthogonal matrix without the first row.

Value

A $(n-1) \times n$ matrix.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Tsagris M.T., Preston S. and Wood A.T.A. (2011). A data-based power transformation for compositional data. In Proceedings of the 4th Compositional Data Analysis Workshop, Girona, Spain. http://arxiv.org/pdf/1106.1451.pdf

John Aitchison (2003). The Statistical Analysis of Compositional Data p. 99 Blackburn Press.

Lancaster H. O. (1965). The Helmert matrices. The American Mathematical Monthly 72(1): 4-12.

See Also

`alfa`, `alfainv`

Examples

```
helm(3)
helm(5)
```

`kl.compreg` *Kullback-Leibler regression*

Description

Regression based on the Kullback-Leibler divergence.

Usage

```
kl.compreg(y, x, B = 1000, ncores = 4, xnew = NULL)
```

Arguments

`y`	A matrix with the compositional data (dependent variable). Zero values are allowed.
`x`	The predictor variable(s), they have to be continuous.
`B`	If B is greater than 1 bootstrap estimates of the standard error are returned. If B=1, no standard errors are returned.
`ncores`	If ncores is 2 or more parallel computing is performed. This is to be used for the case of bootstrap. If B=1, this is not taken into consideration.
`xnew`	If you have new data use it, otherwise leave it NULL.

Details

The Kullback-Leibler divergence is adopted as the objective function. This involves numerical optimisation. There is no log-likelihood.

Value

A list including:

beta	The beta coefficients.
seb	The standard error of the beta coefficients, if bootstrap is chosen, i.e. if B > 1.
est	The fitted or the predicted values (if xnew is not NULL).

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Murteira, Jose MR, and Joaquim JS Ramalho. Regression analysis of multivariate fractional data. Econometric Reviews (To appear).

See Also

diri.reg, esov.compreg, ols.compreg, comp.reg

Examples

```
library(MASS)
x <- fgl[, 1]
y <- fgl[, 2:9]
mod1<- kl.compreg(y, x, B = 1, ncores = 1)
mod2 <- esov.compreg(y, x, B = 1, ncores = 1)
mod1
mod2
```

---

kl.diri *Kullback-Leibler divergence and Bhattacharyya distance between two Dirichlet distributions*

Description

Kullback-Leibler divergence and Bhattacharyya distance between two Dirichlet distributions.

Usage

```
kl.diri(a, b, type = "KL")
```

Arguments

a	A vector with the parameters of the first Dirichlet distribution.
b	A vector with the parameters of the second Dirichlet distribution.
type	A variable indicating whether the Kullback-Leibler divergence ("KL") or the Bhattacharyya distance ("bhatt") is to be computed.

Details

Note that the order is important in the Kullback-Leibler divergence, since this is asymetric, but not in the Bhattacharyya distance, since it is a metric.

Value

The value of the Kullback-Leibler divergence or the Bhattacharyya distance.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Ng Kai Wang, Guo-Liang Tian and Man-Lai Tang (2011). Dirichlet and related distributions: Theory, methods and applications. John Wiley \& Sons.

See Also

diri.est, diri.nr

Examples

```
library(MASS)
a <- runif(10, 0, 20)
b <- runif(10, 1, 10)
kl.diri(a, b)
kl.diri(b, a)
kl.diri(a, b, type = "bhatt")
kl.diri(b, a, type = "bhatt")
```

mix.compnorm *Gaussian mixture models for compositional data*

Description

Gaussian mixture models for compositional data.

Usage

```
mix.compnorm(x, g, model, type = "alr")
```

Arguments

- `x` A matrix with the compositional data.
- `g` How many clusters to create.
- `model` The type of model to be used.
  1. "EII": All groups have the same diagonal covariance matrix, with the same variance for all variables.
  2. "VII": Different diagonal covariance matrices, with the same variance for all variables within each group.
  3. "EEI": All groups have the same diagonal covariance matrix.
  4. "VEI": Different diagonal covariance matrices. If we make all covariance matrices have determinant 1, (divide the matrix with the p-th root of its determinant) then all covariance matrices will be the same.
  5. "EVI": Different diagonal covariance matrices with the same determinant.
  6. "VVI": Different diagonal covariance matrices, with nothing in common.
  7. "EEE": All covariance matrices are the same.
  8. "EEV": Different covariance matrices, but with the same determinant and in addition, if we make them have determinant 1, they will have the same trace.
  9. "VEV": Different covariance matrices but if we make the matrices have determinant 1, then they will have the same trace.
  10. "VVV": Different covariance matrices with nothing in common.
  11. "EVE": Different covariance matrices, but with the same determinant. In addition, calculate the eigenvectors for each covariance matrix and you will see the extra similarities.
  12. "VVE": Different covariance matrices, but they have something in common with their directions. Calculate the eigenvectors of each covariance matrix and you will see the similarities.
  13. "VEE": Different covariance matrices, but if we make the matrices have determinant 1, then they will have the same trace. In addition, calculate the eigenvectors for each covariance matrix and you will see the extra similarities.
  14. "EVV": Different covariance matrices, but with the same determinant.
- `type` Either the additive ("alr") or the isometric log-ratio transformation is to be used ("ilr").

Details

A log-ratio transformation is applied and then a Gaussian mixtued model is constructed.

Value

A list including:

mu	A matrix where each row corresponds to the mean vector of eahc cluster.
su	An array containing the covariance matrix of each cluster.
prob	The estimated mixing probabilities.
est	The estimated cluster membership values.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Ryan P. Browne, Aisha ElSherbiny and Paul D. McNicholas (2015). R package mixture: Mixture Models for Clustering and Classification.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

bic.mixcompnorm, rmixcomp, mixnorm.contour

Examples

```
x <- iris[, 1:4]
mod1 <- mix.compnorm(x, 3, model = "EII" )
mod2 <- mix.compnorm(x, 4, model = "VII")
mod1
mod2
```

---

mixnorm.contour *Contour plot of a Gaussian mixture model in Sˆ2*

Description

Contour plot of a Gaussian mixture model in S^2.

Usage

```
mixnorm.contour(x, mod)
```

Arguments

x A matrix with the compositional data.

mod An object containing the output of a mix.compnorm model.

Details

The contour plot of a Gaussian mixture model is plotted. For this you need the data and the fitted model.

Value

A ternary plot with the data and the contour lines of the fitted Gaussian mixture model.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Ryan P. Browne, Aisha ElSherbiny and Paul D. McNicholas (2015). R package mixture: Mixture Models for Clustering and Classification

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

mix.compnorm, bic.mixcompnorm, diri.contour

Examples

```
x <- iris[, 1:3]
x <- x / rowSums(x)
mod <- mix.compnorm(x, 3, model = "EII")
mixnorm.contour(x, mod)
```

---

mkde *Multivariate kernel density estimation*

Description

Multivariate kernel density estimation.

Usage

```
mkde(x, h, thumb = "none")
```

Arguments

x	A matrix with Euclidean (continuous) data.
h	The bandwidh value. It can be a single value, which is turned into a vector and then into a diagonal matrix, or a vector which is turned into a diagonal matrix.
thumb	Do you want to use a rule of thumb for the bandwidth parameter? If no, leave it "none", or else type "scott" or "silverman".

Details

The multivariate kernel density estimate is calculated with a (not necssarily given) bandwidth value. It is used a wrapper for the funciton `comp.kerncontour`.

Value

A vector with the density estimates calculated for every vector.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Arsalane Chouaib Guidoum (2015). Kernel Estimator and Bandwidth Selection for Density and its Derivatives. The kedd package. http://cran.r-project.org/web/packages/kedd/vignettes/kedd.pdf

M.P. Wand and M.C. Jones (1995). Kernel smoothing, pages 91-92.

B.W. Silverman (1986). Density estimation for statistics and data analysis, pages 76-78.

See Also

`mkde.tune_2`, `comp.kerncontour`

Examples

```
mkde( iris[, 1:4], thumb = "scott" )
mkde( iris[, 1:4], thumb = "silverman" )
```

mkde.tune_2 *Tuning of the bandwidth h of the kernel using the maximum likelihood cross validation*

Description

Tuning of the bandwidth h of the kernel using the maximum likelihood cross validation.

Usage

```
mkde.tune_2( x, low = 0.1, up = 3, s = cov(x) )
```

Arguments

`x`	A matrix with Euclidean (continuous) data.
`low`	The minimum value to search for the optimal bandwidth value.
`up`	The maximum value to search for the optimal bandwidth value.
`s`	A covariance matrix. By default it is equal to the covariance matrix of the data, but can change to a robust covariance matrix, MCD for example.

Details

Maximum likelihood cross validation is applied in order to choose the optimal value of the bandwidth parameter. No plot is produced.

Value

A list including:

`hopt`	The optimal bandwidth value.
`maximum`	The value of the pseudo-log-likelihood at that given bandwidth value.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Arsalane Chouaib Guidoum (2015). Kernel Estimator and Bandwidth Selection for Density and its Derivatives. The kedd package. http://cran.r-project.org/web/packages/kedd/vignettes/kedd.pdf

M.P. Wand and M.C. Jones (1995). Kernel smoothing, pages 91-92.

See Also

`mkde`, `comp.kerncontour`

Examples

```
library(MASS)
mkde.tune_2(iris[, 1:4], c(0.1, 3) )
```

---

multivreg — *Multivariate linear regression*

Description

Multivariate linear regression.

Usage

```
multivreg(y, x, plot = TRUE, xnew = NULL)
```

Arguments

`y`	A matrix with the Eucldidean (continuous) data.
`x`	The predictor variable(s), they have to be continuous.
`plot`	Should a plot appear or not?
`xnew`	If you have new data use it, otherwise leave it NULL.

Details

The classical multivariate linear regression model is obtained.

Value

A list including:

`suma`	A summary as produced by `lm`, which inlcudes the coefficients, their standard error, t-values, p-values.
`r.squared`	The value of the R^2 for each univariate regression.
`resid.outliers`	A vector with number indicating which vectors are potential residual outliers.
`x.leverage`	A vector with number indicating which vectors are potential outliers in the predicto variables space.
`x.leverage`	A vector with number indicating which vectors are potential outliers in the residuals and in the predictor variables space.
`est`	The fitted values if xnew is NULL, or the predicted values otherwise.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

K.V. Mardia, J.T. Kent and J.M. Bibby (1979). Multivariate Analysis. Academic Press.

See Also

diri.reg, esov.compreg, kl.compreg, ols.compreg, comp.reg

Examples

```
library(MASS)
x <- iris[, 1:2]
y<- iris[, 3:4]
multivreg(y, x, plot = TRUE)
```

---

multivt *MLE for the multivarite t distribution*

Description

MLE of the parameters of a multivarite t distribution.

Usage

```
multivt(y, plot = FALSE)
```

Arguments

y	A matrix with continuous data.
plot	If plot is TRUE the value of the maximum log-likelihood as a function of the degres of freedom is presented.

Details

The parameters of a multivariate t distribution are estimated. This is used by the functions comp.den and bivt.contour.

Value

A list including:

center	The location estimate.
scatter	The scatter matrix estimate.
df	The estimated degrees of freedom.
loglik	The loglikelihood value.
mesos	The classical mean vector.
covariance	The classical covariance matrix.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Nadarajah, S. and Kotz, S. (2008). Estimation methods for the multivariate t distribution. Acta Applicandae Mathematicae, 102(1):99-118.

See Also

`bivt.contour`, `comp.den`, `rmvt`

Examples

```
x <- iris[, 1:4]
multivt(x)
```

`norm.contour` *Contour plot of the normal distribution in S^2*

Description

Contour plot of the normal distribution in S^2.

Usage

```
norm.contour(x, type = "alr", n = 100, appear = TRUE)
```

Arguments

`x`	A matrix with the compositional data. It has to be a 3 column matrix.
`type`	This is either "alr" or "ilr", corresponding to the additive and the isometric log-ratio transformation respectively.
`n`	The number of grid points to consider over which the density is calculated.
`appear`	Should the available data appear on the ternary plot (TRUE) or not (FALSE)?

Details

The alr or the ilr transformation is applied to the compositional data at first. Then for a grid of points within the 2-dimensional simplex the bivariate normal density is calculated and the contours are plotted along with the points.

Value

A ternary diagram with the points (if appear = TRUE) and the bivariate normal contour lines.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

diri.contour, mixnorm.contour, bivt.contour, skewnorm.contour

Examples

```
x <- iris[, 1:3]
x <- x / rowSums(x)
norm.contour(x)
dev.new()
norm.contour(x, type = "ilr")
```

---

ols.compreg *Non linear least squares regression*

Description

Non linear least squares regression.

Usage

```
ols.compreg(y, x, B = 1000, ncores = 4, xnew = NULL)
```

Arguments

y	A matrix with the compositional data (dependent variable). Zero values are allowed.
x	The predictor variable(s), they have to be continuous.
B	If B is greater than 1 bootstrap estimates of the standard error are returned. If B=1, no standard errors are returned.
ncores	If ncores is 2 or more parallel computing is performed. This is to be used for the case of bootstrap. If B=1, this is not taken into consideration.
xnew	If you have new data use it, otherwise leave it NULL.

Details

The ordinary least squares between the observed and the fitted compositional data is adopted as the objective function. This involves numerical optimisation since the relationship is non linear. There is no log-likelihood.

Value

A list including:

beta	The beta coefficients.
seb	The standard error of the beta coefficients, if bootstrap is chosen, i.e. if B > 1.
est	The fitted or the predicted values (if xnew is not NULL).

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Murteira, Jose MR, and Joaquim JS Ramalho. Regression analysis of multivariate fractional data. Econometric Reviews (To appear).

See Also

diri.reg, esov.compreg, kl.compreg, comp.reg, comp.reg, alfa.reg

Examples

```
library(MASS)
x <- fgl[, 1]
y <- fgl[, 2:9]
mod1 <- ols.compreg(y, x, B = 1, ncores = 1)
mod2 <- esov.compreg(y, x, B = 1, ncores = 1)
mod1
mod2
```

---

pcr *Principal components regression*

Description

Principal components regression.

Usage

```
pcr(y, x, k = 1, xnew = NULL)
```

Arguments

y	A real values vector.
x	The predictor variable(s), they have to be continuous.
k	The number of principal components to use.
xnew	If you have new data use it, otherwise leave it NULL.

Details

The principal components of the cross product of the independent variables are obtained and classical regression is performed. This is use din the function `alfa.pcr`.

Value

A list including:

beta	The beta coefficients.
parameters	The beta coefficients and their standard eror.
mse	The MSE of the linear regression, if xnew is NULL, becuase it needs the fitted values.
adj.rsq	The value of the adusted R^2 if xnew is NULL.
per	The percentage of variance of the cross product of the independent variables explained by the k components.
est	The fitted or the predicted values (if xnew is not NULL).

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Jolliffe I.T. (2002). Principal Component Analysis.

See Also

`pcr.tune`, `alfa.pcr`, `glm.pcr`

Examples

```
library(MASS)
x <- fgl[, 2:9]
y <- fgl[, 1]
mod1 <- pcr(y, x, 1)
mod2 <- pcr(y, x, 2)
```

pcr.tune *Tuning of the principal components regression*

Description

Tuning the number of principal components in the principal components regression.

Usage

```
pcr.tune(y, x, M = 10, maxk = 50, seed = FALSE, ncores = 2, graph = TRUE)
```

Arguments

y	A real valued vector.
x	The predictor variables, they have to be continuous.
M	The number of folds in the cross validation.
maxk	The maximum number of principal components to check.
seed	If seed is TRUE the folds will always be the same.
ncores	The number of cores to use. If more than 1, parallel computing takes place.
graph	If grpah is TRUE a plot of the performance for each fold along the values of α will appear.

Details

Cross validation is performed to select the optimal number of principal components in the regression. This is used by alfapcr.tune.

Value

A list including: If graph is TRUE a plot of the performance versus the number of principal components will appear.

msp	A matrix with the mean squared error of prediction (MSPE) for every fold.
mspe	A vector with the mean squared error of prediction (MSPE), each value corresponds to a number of principal components.
k	The number of principal components which minimizes the MSPE.
performance	The bias corrected lowest value of the MSPE along with the estimated bias via the Tibshirani and Tibshirani (2009) suggestion.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Jolliffe I.T. (2002). Principal Component Analysis.

Tibshirani and Tibshirani (2009). A bias correction for the minimum error rate in cross-validation. The Annals of Applied Statistics, 3(1):822-829.

See Also

glmpcr.tune, glm.pcr, alfa.pcr, alfapcr.tune

Examples

```
library(MASS)
x <- fgl[, 2:9]
y <- fgl[, 1]
pcr.tune(y, x, M = 10, maxk = 50, seed = FALSE, ncores = 1)
```

---

rcompnorm — *Multivariate normal random values simulation on the simplex*

Description

Multivariate normal random values simulation on the simplex.

Usage

```
rcompnorm(n, m, s, type = "alr")
```

Arguments

n	The sample size, a numerical value.
m	The mean vector in R^d.
s	The covariance matrix in R^d.
type	The alr (type = "alr") or the ilr (type = "ilr") is to be used for closing the Euclidean data onto the simplex.

Details

The algorithm is straightforward, generate random values from a multivariate normal distribution in R^d and brings the values to the simplex S^d using the inverse of a log-ratio transformation.

Value

A matrix with the simulated data.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

comp.den, rdiri, rcompt, rcompsn

Examples

```
x <- as.matrix(iris[, 1:2])
m <- colMeans(x)
s <- var(x)
y <- rcompnorm(100, m, s)
comp.den(y)
ternary(y)
```

---

rcompsn *Multivariate skew normal random values simulation on the simplex*

Description

Multivariate skew normal random values simulation on the simplex.

Usage

```
rcompsn(n, xi, Omega, alpha, dp = NULL, type = "alr")
```

Arguments

n	The sample size, a numerical value.
xi	A numeric vector of length d representing the location parameter of the distribution.
Omega	A $d \times d$ symmetric positive-definite matrix of dimension.
alpha	A numeric vector which regulates the slant of the density.
dp	A list with three elements, corresponding to xi, Omega and alpha described above. The default value is FALSE. If dp is assigned, individual parameters must not be specified.
type	The alr (type = "alr") or the ilr (type = "ilr") is to be used for closing the Euclidean data onto the simplex.

Details

The algorithm is straightforward, generate random values from a multivariate t distribution in R^d and brings the values to the simplex S^d using the inverse of a log-ratio transformation.

Value

A matrix with the simulated data.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Azzalini, A. and Dalla Valle, A. (1996). The multivariate skew-normal distribution. Biometrika, 83(4): 715-726.

Azzalini, A. and Capitanio, A. (1999). Statistical applications of the multivariate skew normal distribution. J.Roy.Statist.Soc. B, 61(3):579-602. Full-length version available at http://arXiv.org/abs/0911.2093

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

comp.den, rdiri, rcompnorm, rmvt

Examples

```
x <- as.matrix(iris[, 1:2])
par <- sn::msn.mle(y = x)$dp
y <- rcompsn(100, dp = par)
comp.den(y, dist = "skewnorm")
ternary(y)
```

---

rcompt *Multivariate t random values simulation on the simplex*

Description

Multivariate t random values simulation on the simplex.

Usage

```
rcompt(n, m, s, dof, type = "alr")
```

Arguments

n	The sample size, a numerical value.
m	The mean vector in R^d.
s	The covariance matrix in R^d.
dof	The degrees of freedom.
type	The alr (type = "alr") or the ilr (type = "ilr") is to be used for closing the Euclidean data onto the simplex.

Details

The algorithm is straightforward, generate random values from a multivariate t distribution in R^d and brings the values to the simplex S^d using the inverse of a log-ratio transformation.

Value

A matrix with the simulated data.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

comp.den, rdiri, rcompnorm, rmvt

Examples

```
x <- as.matrix(iris[, 1:2])
m <- colMeans(x)
s <- var(x)
y <- rcompt(100, m, s, 10)
comp.den(y, dist = "t")
ternary(y)
```

rdiri *Dirichlet random values simulation*

Description

Dirichlet random values simulation.

Usage

```
rdiri(n, a)
```

Arguments

n	The sample size, a numerical value.
a	A numerical vector with the parameter values.

Details

The algorithm is straightforward, for each vector, independent gamma values are generated and then divided by their total sum.

Value

A matrix with the simulated data.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Ng Kai Wang, Guo-Liang Tian and Man-Lai Tang (2011). Dirichlet and related distributions: Theory, methods and applications. John Wiley \& Sons.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

diri.est, diri.nr, diri.contour

Examples

```
x = rdiri( 100, c(5, 7, 1, 3, 10, 2, 4) )
diri.est(x)
```

`ridge.plot` *Ridge regression plot*

Description

A plot of the regularised regression coefficients is shown.

Usage

```
ridge.plot(y, x, lambda = seq(0, 5, by = 0.1) )
```

Arguments

`y`	A numeric vector containing the values of the target variable. If the values are proportions or percentages, i.e. strictly within 0 and 1 they are mapped into R using the logit transformation. In any case, they must be continuous only.
`x`	A numeric matrix containing the continuous variables. Rows are samples and columns are features.
`lambda`	A grid of values of the regularisation parameter λ.

Details

For every value of λ the coefficients are obtained. They are plotted versus the λ values.

Value

A plot with the values of the coefficients as a function of λ.

Author(s)

Michail Tsagris

R implementation and documentation: Giorgos Athineou <athineou@csd.uoc.gr> and Michail Tsagris <mtsagris@yahoo.gr>

References

Hoerl A.E. and R.W. Kennard (1970). Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, 12(1): 55-67.

Brown P. J. (1994). Measurement, Regression and Calibration. Oxford Science Publications.

See Also

`ridge.reg`, `ridge.tune`, `alfa.ridge`, `alfaridge.plot`

Examples

```
y <- iris[, 1]
x <- iris[, 2:4]
ridge.plot(y, x, lambda = seq(0, 2, by = 0.1) )
```

`ridge.reg` *Ridge regression*

Description

Ridge regression.

Usage

```
ridge.reg(y, x, lambda, B = 1, xnew = NULL)
```

Arguments

y	A real valued vector. If it contains percentages, the logit transformation is applied.
x	The predictor variable(s), they have to be continuous.
lambda	The vlaue of the regularisation parameter λ.
B	If B = 1 (default value) no bootstrpa is performed. Otherwise bootstrap standard errors are returned.
xnew	If you have new data whose response value you want to predict put it here, otherwise leave it as is.

Details

This is used in the function `alfa.ridge`. There is also built-in function available from the MASS library, called `lm.ridge`.

Value

A list including:

beta	The beta coefficients.
seb	The standard eror of the coefficiens. If B > 1 the bootstrap standard errors will be returned.
est	The fitted or the predicted values (if xnew is not NULL).

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Hoerl A.E. and R.W. Kennard (1970). Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, 12(1): 55-67.

Brown P. J. (1994). Measurement, Regression and Calibration. Oxford Science Publications.

See Also

`ridge.tune`, `alfa.ridge`, `ridge.plot`

Examples

```
y <- iris[, 1]
x <- iris[, 2:4]
mod1 <- ridge.reg(y, x, lambda = 0.1)
mod2 <- ridge.reg(y, x, lambda = 0)
```

---

ridge.tune *Cross validation for the ridge regression*

Description

Cross validation for the ridge regression is performed using the TT estimate of bias (Tibshirani and Tibshirani, 2009). There is an option for the GCV criterion which is automatic.

Usage

```
ridge.tune(y, x, M = 10, lambda = seq(0, 2, by = 0.1), seed = FALSE, ncores = 2,
graph = TRUE)
```

Arguments

y	A numeric vector containing the values of the target variable. If the values are proportions or percentages, i.e. strictly within 0 and 1 they are mapped into R using the logit transformation.
x	A numeric matrix containing the variables. Rows are samples and columns are features.
M	The number of folds. Set to 10 by default.
lambda	A vector with the a grid of values of λ to be used.
seed	A boolean variable. If it is TRUE the results will always be the same.
ncores	The number of cores to use. If it is more than 1 parallel computing is performed.
graph	If graph is set to TRUE the performances for each fold as a function of the λ values will appear.

Details

A k-fold cross validation is performed and the estimated performance is bias corrected as suggested by Tibshirani and Tibshirani (2009). This function is used by `alfaridge.tune`.

Value

A list including:

`msp`	The performance of the ridge regression for every fold.
`mspe`	The values of the mean prediction error for each value of λ.
`lambda`	The value of λ which corresponds to the minimum MSPE.
`performance`	The minimum bias corrected MSPE along with the estimate of bias.

Author(s)

Michail Tsagris

R implementation and documentation: Giorgos Athineou <athineou@csd.uoc.gr> and Michail Tsagris <mtsagris@yahoo.gr>

References

Hoerl A.E. and R.W. Kennard (1970). Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, 12(1):55-67.

Brown P. J. (1994). Measurement, Regression and Calibration. Oxford Science Publications.

Tibshirani R.J., and Tibshirani R. (2009). A bias correction for the minimum error rate in cross-validation. The Annals of Applied Statistics 3(2): 822-829.

See Also

`ridge.reg`, `alfaridge.tune`

Examples

```
y <- iris[, 1]
x <- iris[, 2:4]
ridge.tune(y, x, M = 10, lambda = seq(0, 2, by = 0.1), seed = FALSE, ncores = 1, graph = TRUE)
```

rmixcomp *Simulation of compositional data from Gaussian mixture models*

Description

Simulation of compositional data from Gaussian mixture models.

Usage

```
rmixcomp(n, prob, mu, sigma, type = "alr")
```

Arguments

n	The sample size
prob	A vector with mixing probabilities. ITs length is equal to the number of clusters.
mu	A matrix where each row corresponds to the mean vector of each cluster.
sigma	An array consisting of the covariance matrix of each cluster.
type	Should the additive ("type=alr") or the isometric (type="ilr") log-ration be used? The default value is for the additive log-ratio transformation.

Details

A sample from a multivariate Gaussian mixture model is generated.

Value

A list including:

id	A numeric variable indicating the cluster of simulated vector.
x	A matrix containing the simulated compositional data. The number of dimensions will be + 1.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Ryan P. Browne, Aisha ElSherbiny and Paul D. McNicholas (2015). R package mixture: Mixture Models for Clustering and Classification.

See Also

mix.compnorm, bic.mixcompnorm

Examples

```
p <- c(1/3, 1/3, 1/3)
mu <- matrix(nrow = 3, ncol = 4)
s <- array( dim = c(4, 4, 3) )
x <- as.matrix(iris[, 1:4])
ina <- as.numeric(iris[, 5])
for (i in 1:3) {
  mu[i, ] <- colMeans(x[ina == i, ])
  s[, , i] <- cov(x[ina == i, ])
}
y <- rmixcomp(100, p, mu, s, type = "alr")
```

`rmvnorm` *Multivariate normal random values simulation*

Description

Multivariate normal random values simulation.

Usage

```
rmvnorm(n, mu, sigma)
```

Arguments

`n`	The sample size, a numerical value.
`mu`	The mean vector in R^d.
`sigma`	The covariance matrix in R^d.

Details

The algorithm uses univariate normal random values and transforms them to multivariate via a spectral decomposition. It is slower than the command `mvrnorm` available from MASS, but it allows for singular covariance matrices. This function is used by `rcompnorm`.

Value

A matrix with the simulated data.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

comp.den, rdiri, rmvt

Examples

```
x <- as.matrix(iris[, 1:2])
m <- colMeans(x)
s <- var(x)
y <- rmvnorm(100, m, s)
colMeans(y)
var(y)
```

---

rmvt *Multivariate t random values simulation*

Description

Multivariate t random values simulation.

Usage

```
rmvt(n, mu, sigma, v)
```

Arguments

n	The sample size, a numerical value.
mu	The mean vector in R^d.
sigma	The covariance matrix in R^d.
v	The degrees of freedom.

Details

The algorithm uses univariate normal and chi-square random values and then generates multivariate t values via a spectral decomposition.This function is used by rcompt.

Value

A matrix with the simulated data.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

rdiri, rmvnorm, multivt

Examples

```
x <- as.matrix(iris[, 1:2])
m <- colMeans(x)
s <- var(x)
y <- rmvt(100, m, s, 10)
multivt(y)
```

---

skewnorm.contour *Contour plot of the skew skewnormal distribution in Sˆ2*

Description

Contour plot of the skew skewnormal distribution in S^2.

Usage

```
skewnorm.contour(x, type = "alr", n = 100, appear = TRUE)
```

Arguments

x	A matrix with the compositional data. It has to be a 3 column matrix.
type	This is either "alr" or "ilr", corresponding to the additive and the isometric log-ratio transformation respectively.
n	The number of grid points to consider over which the density is calculated.
appear	Should the available data appear on the ternary plot (TRUE) or not (FALSE)?

Details

The alr or the ilr transformation is applied to the compositional data at first. Then for a grid of points within the 2-dimensional simplex the bivariate skew skewnormal density is calculated and the contours are plotted along with the points.

Value

A ternary diagram with the points (if appear = TRUE) and the bivariate skew skewnormal contour lines.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Azzalini A. and Valle A. D. (1996). The multivariate skew-skewnormal distribution. Biometrika 83(4):715-726.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

diri.contour, mixnorm.contour, bivt.contour, norm.contour

Examples

```
x <- iris[, 1:3]
x <- x / rowSums(x)
skewnorm.contour(x)
dev.new()
skewnorm.contour(x, type = "ilr")
```

---

spat.med *Spatial median*

Description

Spatial median.

Usage

```
spat.med(x)
```

Arguments

x	A matrix with Euclidean data, continuous variables.

Details

The spatial median, using a fixed point iterative algorithm, for Euclidean data is calcualted. It is a robust location estimate. This function is used by comp.den.

Value

A vector with the spatial median.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Jyrki Mottonen, Klaus Nordhausen and Hannu Oja (2010). Asymptotic theory of the spatial median. In Nonparametrics and Robustness in Modern Statistical Inference and Time Series Analysis: A Festschrift in honor of Professor Jana Jureckova

T. Karkkaminen and S. Ayramo (2005). On computation of spatial median for robust data mining. Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems, EUROGEN 2005, R. Schilling, W.Haase, J. Periaux, H. Baier, G. Bugeda (Eds) FLM, Munich. http://users.jyu.fi/~samiayr/pdf/ayramo_eurogen05.pdf

See Also

`sscov`, `comp.den`

Examples

```
library(MASS)
spat.med( iris[, 1:4] )
colMeans( iris[, 1:4] )
```

`spatmed.reg` *Spatial median regression*

Description

Spatial median regression with Euclidean data.

Usage

```
spatmed.reg(y, x, xnew = NULL)
```

Arguments

`y`	A matrix with the compsitional data. Zero values are not allowed.
`x`	The predictor variable(s), they have to be continuous.
`xnew`	If you have new data use it, otherwise leave it NULL.

Details

The objective function is the minimization of the sum of the absolute residuals. It is the multivariate generalisation of the median regression. This function is used by `link{comp.reg}`.

Value

A list including:

`beta`	The beta coefficients.
`seb`	The standard error of the beta coefficients.
`est`	The fitted or the predicted values (if xnew is not NULL).

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Biman Chakraborty (2003) On multivariate quantile regression. Journal of Statistical Planning and Inference http://www.stat.nus.edu.sg/export/sites/dsap/research/documents/tr01_2000.pdf

See Also

multivreg, comp.reg, alfa.reg, esov.compreg, diri.reg

Examples

```
library(MASS)
x <- iris[, 3:4]
y <- iris[, 1:2]
mod1 <- spatmed.reg(y, x)
mod2 <- multivreg(y, x, plot = FALSE)
```

---

sscov *Spatial sign covariance matrix*

Description

Spatial sign covariance matrix.

Usage

```
sscov(x)
```

Arguments

x	A matrix with continuous data.

Details

The spatial median is at firrst computed and then the covariance matrix. This is used in the function comp.den.

Value

The spatial sign covariance matrix.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

A Durre, D Vogel, DE Tyler (2014). The spatial sign covariance matrix with unknown location. Journal of Multivariate Analysis, 130: 107-117. http://arxiv.org/pdf/1307.5706v2.pdf

See Also

spat.med, comp.den

Examples

```
library(MASS)
sscov(iris[, 1:4])
```

---

sym.test *Log-likelihood ratio test for a symmetric Dirichlet distribution*

Description

Log-likelihood ratio test for a symmetric Dirichlet distribution

Usage

```
sym.test(x)
```

Arguments

x	A matrix with the compositional data. No zero values are allowed.

Details

Log-likelihood ratio test is performed for the hypothesis that all Dirichelt parameters are equal.

Value

A list including:

est.par	The estimated parameters under the alternative hypothesis.
one.par	The value of the estimated parameter under the null hypothesis.
res	The loglikelihood under the alternative and the null hypothesis, the value of the test statistic, its relevant p-value and the associated degrees of freedom, which are actually the dimensionality of the simplex, $D-1$, where D is the number of components.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Ng Kai Wang, Guo-Liang Tian and Man-Lai Tang (2011). Dirichlet and related distributions: Theory, methods and applications. John Wiley \& Sons.

See Also

diri.reg, esov.compreg, kl.compreg, ols.compreg, comp.reg

Examples

```
x <- rdiri( 100, c(5, 7, 1, 3, 10, 2, 4) )
sym.test(x)
x <- rdiri( 100, c(5, 5, 5, 5, 5) )
sym.test(x)
```

---

ternary *Ternary diagram*

Description

Ternary diagram.

Usage

```
ternary(x, means = TRUE, pca = FALSE)
```

Arguments

x	A matrix with the compositional data.
means	A boolean variable. Should the closed geometric mean and the arithmetic mean appear (TRUE) or not (FALSE)?.
pca	Should the first PCA calculated Aitchison (1983) described appear? If yes, then this should be TRUE, or FALSE otherwise.

Details

The first PCA is calcualte using the centred log-ratio transformation as Aitchison (1983, 1986) suggested. If the data contain zero values, the first PCA will not be plotted. There are two ways to create a ternary graph. The one I used here, where eqch edge is equal to 1 and the one Aitchison (1986) uses. For every given point, the sum of the distances from the edges is equal to 1. Zeros in the data appear with green circles in the triangle and you will also see NaN in the closed geometric mean.

Value

The ternary plot and a matrix with the closed geometric and the simple arithmetic mean vector.

Author(s)

Michail Tsagris

R implementation and documentation: Michail Tsagris <mtsagris@yahoo.gr> and Giorgos Athineou <athineou@csd.uoc.gr>

References

Aitchison, J. (1983). Principal component analysis of compositional data. Biometrika 70(1):57-65.

Aitchison J. (1986). The statistical analysis of compositional data. Chapman & Hall.

See Also

comp.den, alfa, diri.contour, comp.kerncontour

Examples

```
library(MASS)
x <- fgl[, 2:4]
ternary(x, means = FALSE)
x <- iris[, 1:3]
ternary(x, pca = TRUE)
```

Index

∗Topic **-regression**
alfareg.tune, 14
∗Topic **-transformation**
alfa, 4
alfa.pcr, 5
alfa.profile, 6
alfa.reg, 7
alfa.ridge, 9
alfadist, 10
alfainv, 11
alfapcr.tune, 12
alfareg.tune, 14
fast.alfa, 31
frechet, 32
∗Topic **Bhattacharyya distance**
kl.diri, 38
∗Topic **Compositional data**
Compositional-package, 3
∗Topic **Contour plots**
Compositional-package, 3
∗Topic **Dirichlet distribution**
diri.contour, 25
diri.est, 26
diri.nr, 27
kl.diri, 38
rdiri, 56
∗Topic **Dirichlet regression**
diri.reg, 28
∗Topic **Discriminant analysis**
Compositional-package, 3
∗Topic **ESOV metric**
esov.compreg, 30
∗Topic **Euclidean distance**
alfadist, 10
∗Topic **Frechet mean**
frechet, 32
∗Topic **GLM**
glmpcr.tune, 35
∗Topic **Gaussian mixture models**
rmixcomp, 61
∗Topic **Gaussian mixtures**
mix.compnorm, 40
∗Topic **Gaussianmixture model**
mixnorm.contour, 41
∗Topic **Kullback-Lebler divergence**
kl.compreg, 37
∗Topic **Kullback-Leibler divergence**
kl.diri, 38
∗Topic **MLE**
Compositional-package, 3
∗Topic **Newton-Raphson**
diri.nr, 27
∗Topic **Regression**
Compositional-package, 3
∗Topic **Ridge regression**
alfaridge.tune, 17
ridge.tune, 59
∗Topic **bandwidth tuning**
mkde.tune_2, 44
∗Topic **bivariate normal distribution**
norm.contour, 47
∗Topic **bivariate skew skewnormal distribution**
skewnorm.contour, 64
∗Topic **bivariate t distribution**
bivt.contour, 20
∗Topic **contour plot**
bivt.contour, 20
comp.kerncontour, 22
diri.contour, 25
mixnorm.contour, 41
norm.contour, 47
skewnorm.contour, 64
∗Topic **covariance matrix**
sscov, 67
∗Topic **generalised linear models**
glm.pcr, 33
∗Topic **inverse transformation**

alfainv, 11
*Topic **kernel density estimate**
mkde, 42
*Topic **kernel density**
comp.kerncontour, 22
mkde.tune_2, 44
*Topic **linear regression**
pcr, 49
pcr.tune, 51
*Topic **list(list(\alpha))**
alfa, 4
alfa.pcr, 5
alfa.profile, 6
alfa.reg, 7
alfa.ridge, 9
alfadist, 10
alfainv, 11
alfapcr.tune, 12
alfareg.tune, 14
fast.alfa, 31
frechet, 32
*Topic **location and scatter**
comp.den, 21
*Topic **log-likeliihood ratio test**
sym.test, 68
*Topic **maximum likelihood cross validation**
mkde.tune_2, 44
*Topic **maximum likelihood estimation**
diri.est, 26
diri.nr, 27
*Topic **maximum log-likelihood estimation**
fast.alfa, 31
*Topic **mixtures of normal distributions**
bic.mixcompnorm, 19
*Topic **model based clustering**
mix.compnorm, 40
*Topic **model selection**
bic.mixcompnorm, 19
*Topic **multivariate kernel**
mkde, 42
*Topic **multivariate linear regression**
multivreg, 45
*Topic **multivariate normal distribution**
rcompnorm, 52
rmvnorm, 62
*Topic **multivariate regression**
alfa.reg, 7
alfapcr.tune, 12
diri.reg, 28
esov.compreg, 30
kl.compreg, 37
ols.compreg, 48
spatmed.reg, 66
*Topic **multivariate rgression**
comp.reg, 24
*Topic **multivariate skew normal distribution**
rcompsn, 53
*Topic **multivariate t distribution**
multivt, 46
rcompt, 54
rmvt, 63
*Topic **ordinary least squares**
ols.compreg, 48
*Topic **orthogonal matrix**
helm, 36
*Topic **parameters tuning**
alfaridge.tune, 17
*Topic **plot**
alfaridge.plot, 16
ridge.plot, 57
*Topic **principal components regression**
alfa.pcr, 5
glm.pcr, 33
*Topic **principal components**
glmpcr.tune, 35
pcr, 49
pcr.tune, 51
*Topic **profile log-likelihood**
alfa.profile, 6
*Topic **random values simulation**
rcompnorm, 52
rcompsn, 53
rcompt, 54
rdiri, 56
rmvnorm, 62
rmvt, 63
*Topic **random vectors simulation**
rmixcomp, 61
*Topic **regularisation**

ridge.reg, 58
∗Topic **ridge regression**
alfa.ridge, 9
alfaridge.plot, 16
ridge.plot, 57
ridge.reg, 58
∗Topic **robust estimation**
comp.den, 21
∗Topic **robust statistics**
spat.med, 65
sscov, 67
∗Topic **spatial median regression**
comp.reg, 24
spatmed.reg, 66
∗Topic **spatial median**
spat.med, 65
∗Topic **symmetric Dirichlet distribution**
sym.test, 68
∗Topic **ternary plot**
ternary, 69
∗Topic **visualisation**
ternary, 69
∗Topic
alfa, 4
alfa.pcr, 5
alfa.profile, 6
alfa.reg, 7
alfa.ridge, 9
alfadist, 10
alfainv, 11
alfapcr.tune, 12
alfareg.tune, 14
fast.alfa, 31
frechet, 32

alfa, 4, *7*, *11*, *12*, *14*, *15*, *32*, *33*, *37*, *70*
alfa.pcr, 5, *14*, *34*, *36*, *50*, *52*
alfa.profile, 6, *32*
alfa.reg, 7, *11*, *15*, *29*, *31*, *49*, *67*
alfa.ridge, 9, *17*, *18*, *57–59*
alfadist, 10, *12*
alfainv, *5*, *7*, *11*, 11, *32*, *33*, *37*
alfapcr.tune, *6*, 12, *34–36*, *51*, *52*
alfareg.tune, 14
alfaridge.plot, *10*, 16, *57*
alfaridge.tune, *10*, 17, *60*

bic.mixcompnorm, 19, *41*, *42*, *61*
bivt.contour, 20, *23*, *26*, *46–48*, *65*

comp.den, 21, *46*, *47*, *53–55*, *63*, *65–68*, *70*
comp.kerncontour, 22, *26*, *43*, *44*, *70*
comp.reg, *8*, 24, *29*, *31*, *38*, *46*, *49*, *67*, *69*
Compositional-package, 3

diri.contour, *21*, *23*, 25, *27*, *28*, *42*, *48*, *56*, *65*, *70*
diri.est, 26, *28*, *39*, *56*
diri.nr, *27*, 27, *39*, *56*
diri.reg, *8*, *24*, 28, *31*, *38*, *46*, *49*, *67*, *69*

esov.compreg, *8*, *24*, *29*, 30, *38*, *46*, *49*, *67*, *69*

fast.alfa, *7*, 31
frechet, 32

glm, *14*, *34*
glm.pcr, *6*, 33, *36*, *50*, *52*
glmpcr.tune, *14*, 35, *52*

helm, 36

kl.compreg, *8*, *29*, *31*, 37, *46*, *49*, *69*
kl.diri, 38

lm, *45*
lm.ridge, *58*

mix.compnorm, *20*, 40, *42*, *61*
mixnorm.contour, *20*, *21*, *23*, *26*, *41*, 41, *48*, *65*
mkde, 42, *44*
mkde.tune_2, *43*, 44
multivreg, *24*, 45, *67*
multivt, *22*, 46, *64*
mvrnorm, *62*

norm.contour, *21*, *23*, *26*, 47, *65*

ols.compreg, *8*, *29*, *31*, *38*, *46*, 48, *69*

pcr, *6*, *34*, 49
pcr.tune, *14*, *36*, *50*, 51
profile, *5*, *14*, *31*, *33*

rcompnorm, 52, *54*, *55*, *62*
rcompsn, *53*, 53
rcompt, *53*, 54, *63*
rdiri, *28*, *53–55*, 56, *63*, *64*

ridge.plot, *17*, 57, *59*
ridge.reg, *9, 10*, *57*, 58, *60*
ridge.tune, *18*, *57*, *59*, 59
rmixcomp, *20*, *41*, 61
rmvnorm, 62, *64*
rmvt, *47*, *54, 55*, *63*, 63

skewnorm.contour, *21*, *48*, 64
spat.med, *22*, 65, *68*
spatmed.reg, *24*, 66
sscov, *22*, *66*, 67
sym.test, 68

ternary, 69