Package 'CAMAN'

September 26, 2016

Type Package

Version 0.74

Date 2016-09-24

Title Finite Mixture Models and Meta-Analysis Tools - Based on C.A.MAN

Author Peter Schlattmann <peter.schlattmann@mti.uni-jena.de>, Johannes Hoehne, Maryna Verba

Maintainer Philipp Doebler <philipp.doebler@googlemail.com>

Description Tools for the analysis of finite semiparametric mixtures. These are useful when data is heterogeneous, e.g. in pharmacokinetics or meta-analysis. The NPMLE and VEM algorithms (flexible support size) and EM algorithms (fixed support size) are provided for univariate and bivariate data.

Depends R (>= 2.10), methods, sp, mvtnorm

License GPL (>= 2)

NeedsCompilation yes

Repository CRAN

Date/Publication 2016-09-26 09:34:15

R topics documented:

`anova.CAMAN.object` *ANOVA for finite mixture models*

Description

A common problem in the estimation of mixture models is to determine the number of components. This may be done using a parametric bootstrap. This function simulates from a mixture model under the null hypothesis with `k0` components. A mixture model with k0 and usually k0 +1 components is fit to the data and then the likelihood ratio statistic (LRS) is computed.

Based on the bootstrap the distribution of the LRS is obtained which allows to obtain an approximation to the achieved level of significance corresponding to the value of $-2\log\xi$ obtained from the original sample.

Usage

```
## S3 method for class 'CAMAN.object'
anova(object, object1, nboot=2500, limit=0.01, acc=10^(-7),
     numiter=5000, giveBootstrapData=FALSE, giveLikelihood=FALSE, ...)
```

Arguments

`object`	A CAMAN-object which quantifies a finite mixture model under null hypothesis.
`object1`	A CAMAN-object which quantifies another finite mixture model under the alterative hypothesis.
`nboot`	Number of bootstrap samples.
`limit`	parameter to control the limit of union several components. Default is 0.01.
`acc`	convergence criterion. VEM and EM loops stop when deltaLL<acc. Default is 10^(-7).

numiter	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.
giveBootstrapData	A Boolean that indicates whether the bootstrapped data should be returned or not
giveLikelihood	Return the likelihood-values of both models for each generated dataset.
...	Arguments to be passed on to other methods; currently none.

Details

The parameters limit, acc and numiter are used for the VEM algorithm in each bootstrap sample.

Value

The function returns a list with components

- overview: comparison of the models, including BIC, LL and LL-ratio
- `LL ratios in bootstrap-data`: 90, 95, 97.5 and 99 percentiles of LL-ratios
- `simulated p-value`: p-value, quantifying the null model

Author(s)

Peter Schlattmann and Johannes Hoehne

References

McLachlan, G. and Peel, D. (2000). *Finite Mixture Models*, Chichester: Wiley.

Schlattmann, P. (2009). *Medical Applications of Finite Mixture Models*. Berlin: Springer.

Examples

```
data(thai_cohort)
mix0 <- mixalg(obs="counts", weights="frequency", family="poisson", data=thai_cohort,
               numiter=18000, acc=0.00001,startk=25)
em0<-mixalg.EM(mix0,p=c(1),t=c(1))
em1<-mixalg.EM(mix0,p=c(0.7,0.3),t=c(2,9))
## Not run: ll<-anova(em0,em1,nboot=250) #might take some minutes
```

aspirin *Aspirin use and breast cancer risk*

Description

This data set can be used to examine the recent epidemiological studies on aspirin use and breast cancer risk published from 2001 to 2007 within a meta-analysis, and to investigate reasons for heterogeneity between the individual studies.

We systematically searched for cohort-studies and case-control-studies from 2001-2007, which evaluated the association between aspirin and breast cancer risk. A total of 15 studies (seven cohort studies and eight case-control studies) met the inclusion criteria.

Usage

```
data("aspirin")
```

Format

A data frame consisting of 15 data sets (rows) and 11 attributes (columns)

References

Schlattmann, P.(2009) *Medical Applications of Finite Mixture Models.* Berlin: Springer.

Examples

```
#Example
#Homogeneous Metaregression adjusting for study type and year of publication

data(aspirin)
wgt <- 1/aspirin$var# calculate weights
m0 <- mixcov(dep="logrr", fixed=c("type","yearc"), random="", weight=wgt,
             k=1,family="gaussian",data=aspirin)
```

betaplasma *Determinants of Beta-Carotene Levels*

Description

Observational studies have suggested that low dietary intake or low plasma concentrations of retinol, beta-carotene, or other carotenoids might be associated with increased risk of developing certain types of cancer. However, relatively few studies have investigated the determinants of plasma concentrations of these micronutrients. Nierenberg et al. designed a cross-sectional study to investigate the relationship between personal characteristics and dietary factors, and plasma concentrations of retinol, beta-carotene and other carotenoids. Study subjects (N = 315) were patients who had an elective surgical procedure during a three-year period to biopsy or remove a lesion of the lung, colon, breast, skin, ovary or uterus that was found to be non-cancerous.

Usage

```
data("betaplasma")
```

Format

datafile contains 315 observations on 15 variables. This data set can be used to demonstrate multiple regression, transformations, categorical variables, outliers, pooled tests of significance and model building strategies.

Variable Names in order from left to right:

item participant id

age Age (years)

sex Sex, Factor with 2 levels (M=Male, F=Female).

smokestat Smoking status, Factor with 3 levels (Never, Former, Current Smoker)

bmi body mass index (weight/(height^2))

vituse Vitamin Use, Factor with 3 levels (Yes (fairly often), Yes (not often), No)

calories Number of calories consumed per day.

fat Grams of fat consumed per day.

fiber Grams of fiber consumed per day.

alcohol Number of alcoholic drinks consumed per week.

chol Cholesterol consumed (mg per day).

betadiet Dietary beta-carotene consumed (mcg per day).

retdiet Dietary retinol consumed (mcg per day)

betacaro Plasma beta-carotene (ng/ml)

retplasma Plasma Retinol (ng/ml)

References

Schlattmann, P.(2009) *Medical Applications of Finite Mixture Models.* Berlin: Springer.

These data have not been published yet but a related reference is

Nierenberg DW, Stukel TA, Baron JA, Dain BJ, Greenberg ER (1989) "Determinants of plasma levels of beta-carotene and retinol." *American Journal of Epidemiology*, **130**, 511–521.

The use of the data is kindly permitted Dr. Therese Stukel, Dartmouth Hitchcock Medical Center, USA

The data can also be found at StatLib: http://lib.stat.cmu.edu/index.php

Examples

```
data(betaplasma)
beta4 <- mixcov(dep="betacaro", fixed=c("chol","sex","bmi"), random="betadiet",
                data=betaplasma,k=4,family="gaussian")
```

`bivariate.EM` *EM-algorithm for bivariate normally distributed data*

Description

Function

Usage

```
bivariate.EM(obs1, obs2, type, data = NULL,
             var1, var2,
             corr, lambda1, lambda2, p,
             numiter=5000,acc=1.e-7,class)
```

Arguments

`obs1`	the first column of the observations
`obs2`	the second column of the observations
`type`	kind of data

data	an optional data frame. If not NULL, obs1, obs2, var1, var2 and corr will be looked for in data
var1	Variance of the first column of the observations(except meta-analysis)
var2	Variance of the second column of the observations (except meta-analysis)
corr	correlation coefficient
lambda1	Means of the first column of the observations
lambda2	Means of the second column of the observations
p	Mixing weight
numiter	parameter to control the maximal number of iterations in the EM loops. Default is 5000.
acc	convergence criterion. Default is 1.e-7
class	classification of studies?

Examples

```
## Not run:
# 1.EM and classification for bivariate data with starting values
data(rs12363681)
lambda1<-c(1540.97, 837.12, 945.40, 1053.69)
lambda2<-c(906.66, 1371.81 ,1106.01,973.11)
p<-c(0.05,0.15,0.6,0.2)
test<-bivariate.EM(obs1=x, obs2=y, type="bi", lambda1=lambda1,lambda2=lambda2,
                   p=p,data=rs12363681,class="TRUE")
# scatter plot with ellipse
plot(test, ellipse=TRUE)
# scatter plot without ellipse
plot(test, ellipse=FALSE)

## End(Not run)
# 2. EM-algorithm for a diagnostic meta-analysis with bivariate
#    normally distributed data and study specific fixed variances
data(CT)
p2<-c(0.4,0.6)
lamlog12<-c(2.93,3.22)
lamlog22<-c(2.5,1.5)
ct.m1 <- bivariate.EM(obs1=logitTPR, obs2=logitTNR,
                      var1=varlogitTPR, var2=varlogitTNR,
                      type="meta", lambda1=lamlog12, lambda2=lamlog22,
                      p=p2,data=CT,class="TRUE")
```

`bivariate.mixalg` *EM algorithm and classification for univariate data, for bivariate data and for meta data*

Description

Function

Usage

```
bivariate.mixalg(obs1, obs2, type, data = NULL,
                 var1, var2, corr, lambda1, lambda2,
                 p,startk, numiter=5000, acc=1.e-7, class)
```

Arguments

`obs1`	the first column of the observations
`obs2`	the second column of the observations
`type`	kind of data
`data`	an optional data frame
`var1`	Variance of the first column of the observations(except meta-analysis)
`var2`	Variance of the second column of the observations (except meta-analysis)
`corr`	correlation coefficient
`lambda1`	Means of the first column of the observations
`lambda2`	Means of the second column of the observations
`p`	Probability
`startk`	starting/maximal number of components. This number will be used to compute the grid in the VEM. Default is 20.
`numiter`	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.
`acc`	convergence criterion. Default is 1.e-7

class	classification of studies

Examples

```
## Not run:
#1.EM and classification for bivariate data
#Examples
data(rs12363681)
test <- bivariate.mixalg(obs1=x, obs2=y, type="bi",
                         lambda1=0, lambda2=0, p=0,
                         data=rs12363681, startk=20, class="TRUE")
#scatter plot with ellipse
plot(test)
#scatter plot without ellipse
plot(test, ellipse = FALSE)
#2.EM and classification for meta data
#Examples
data(CT)
bivariate.mixalg(obs1=logitTPR, obs2=logitTNR,
                 var1=varlogitTPR, var2=varlogitTNR,
                 type="meta", lambda1=0, lambda2=0,
                 p=0,data=CT,startk=20,class="TRUE")

## End(Not run)
```

---

bivariate.VEM *VEM algorithm for univariate data, for bivariate data and for meta data*

Description

VEM algorithm for univariate data, for bivariate data and for meta data

Usage

```
bivariate.VEM(obs1, obs2, type, data = NULL, var1, var2,
              lambda1, lambda2, p, startk, numiter=5000,
              acc=1.e-7)
```

Arguments

obs1	the first column of the observations
obs2	the second column of the observations
type	kind of data

data	an optional data frame. If not NULL, obs1, obs2, var1 and var2 will be looked for in data
lambda1	Means of the first column of the observations
lambda2	Means of the second column of the observations
p	Mixing weight
var1	Variance of the first column of the observations(only for meta-analysis)
var2	Variance of the second column of the observations (only for meta-analysis)
startk	starting/maximal number of components. This number will be used to compute the grid in the VEM. Default is 20.)
numiter	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.
acc	convergence criterion. Default is 1.e-7

Examples

```
## Not run:
# 1. VEM-algorithm  for bivariate normally distributed data
#Examples
data(rs12363681)
bivariate.VEM(obs1=x,obs2=y,type="bi", data=rs12363681,startk=20)
# 2.VEM for metadata
data(CT)
bivariate.VEM(obs1=logitTPR, obs2=logitTNR,
              var1=varlogitTPR, var2= varlogitTNR,
              type="meta", data=CT, startk=20)

## End(Not run)
```

CAMANboot *Parametric bootstrap*

Description

Parametric bootstrap for bivariate normally distributed data

Usage

```
CAMANboot(obs1, obs2, var1, var2, lambda11, lambda12,
     prob1, lambda21, lambda22, prob2, rep,
     data,numiter=10000,acc=1.e-7)
```

Arguments

obs1	the first column of the observations
obs2	the second column of the observations
data	a data frame
var1	Variance of the first column of the observations(except meta-analysis)
var2	Variance of the second column of the observations (except meta-analysis)
lambda11	first means of the first column of the observations
lambda12	first means of the second column of the observations
prob1	first mixing weight
lambda21	second means of the first column of the observations
lambda22	second means of the second column of the observations
prob2	second mixing weight
numiter	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.
acc	convergence criterion. Default is 1.e-7
rep	number of repetitions

Examples

```
# Parametric bootstrap for bivariate normally distributed data
data(CT)
library(mvtnorm)
hom1<-c(3.142442)
hom2<-c(-1.842393)
p1<-c(1)
```

```
start1<-c(2.961984,3.226141)
start2<-c(-2.578836, -1.500823)
pvem<-c(0.317,0.683)
CAMANboot(obs1=logitTPR, obs2=logitTNR, var1=varlogitTPR, var2=varlogitTNR,
          lambda11=hom1, lambda12=hom2, prob1=p1,
          lambda21=start1, lambda22=start2, prob2=pvem,rep=3,data=CT)
```

CT *Meta-anaysis: noninvasive coronary angiography using computed tomography (CT)*

Description

CT for ruling out clinically significant coronary artery disease (CAD) in adults with suspected or known CAD. The accuracy and clinical value of CT was assessed in this meta-analysis.

MEDLINE, EMBASE, and ISI Web of Science searches from inception through 2 June 2009 and bibliographies of reviews. Prospective English- or German-language studies that compared CT or MRI with conventional coronary angiography in all patients and included sufficient data for compilation of 2 x 2 tables. Two investigators independently extracted patient and study characteristics; differences were resolved by consensus. 89 studies comprising 7516 assessed the diagnostic value of CT.

Usage

```
data("CT")
```

Format

A data frame consisting of 91 data sets (rows) and 10 attributes (columns)

Variable Names in order from left to right:

Author Author

Year Year

TP true positive

FP False positive

FN False negative

TN True negative

logitTPR logit-true positive rate

logitTNR logit-true negative rate

varlogitTPR Variance of logit TPR

varlogitTNR Variance of logit TPR

References

Schuetz GM, Zacharopoulou NM, Schlattmann P, Dewey M. Meta-analysis: noninvasive coronary angiography using computed tomography versus magnetic resonance imaging. Ann Intern Med. 2010 Feb 2;152(3):167-77. doi: 10.7326/0003-4819-152-3-201002020-00008.

Examples

```
#Use the EM-algorithm for a diagnostic meta-analysis based on a mixture
#of bivariate  normal densities.
#Here fixed study specific variances are calculated based on logit
#transformed sensitivity and specificity.
data(CT)
p2 <- c(0.4,0.6)
lamlog12 <- c(2.93,3.22)
lamlog22 <- c(2.5,1.5)

m0 <- bivariate.EM(obs1=logitTPR,obs2=logitTNR,
                   var1=varlogitTPR,var2=varlogitTNR,
                   type="meta",lambda1=lamlog12,lambda2=lamlog22,
                   p=p2,data=CT,class="FALSE")
```

---

getFDR *Compute false discovery rates and related statistics*

Description

This function is especially useful in the context of genetic data.

Usage

```
getFDR(dat, threshold = 0.7, idxNotDiff = 1)
```

Arguments

dat	A mixture, i.e. a CAMAN.object.
threshold	numeric, should be between 0 and 1 to be meaningful.
idxNotDiff	integer, index of component for which computations are required.

Details

See Schlattmann (2009) for more details.

Value

A list with components FDR, FNDR, FPR and FNR.

Author(s)

Peter Schlattmann and Johannes Hoehne

References

Schlattmann, P. (2009). *Medical Applications of Finite Mixture Models.* Berlin: Springer.

---

golubMerge *Data from the Golub et al (1999) Paper*

Description

The data are from Golub et al. These are the combined training samples and test samples. There are 47 patients with acute lymphoblastic leukemia (ALL) and 25 patients with acute myeloid leukemia (AML). The samples were assayed using Affymetrix Hgu6800 chips and data on the expression of 7129 genes (Affymetrix probes) are available. The data were obtained from the Web site listed below and transformed slightly. Two objects are in the workspace when the data is loaded: golubMerge.exprs and sample.labels.

Usage

```
data("golubMerge")
```

Format

A matrix with 7129 rows (for the genes) and 72 columns (for the patients).

Note

The data also appear in the Bioconductor package golubEsets in a different format. See Schlattmann (2009) for details on how to handle this type of data.

Source

This data is a variant of the data from the Bioconductor golubEsets package.

References

Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring, Science, 531-537, 1999, T. R. Golub and D. K. Slonim and P. Tamayo and C. Huard and M. Gaasenbeek and J. P. Mesirov and H. Coller and M.L. Loh and J. R. Downing and M. A. Caligiuri and C. D. Bloomfield and E. S. Lander

Schlattmann, P. (2009). *Medical Applications of Finite Mixture Models.* Berlin: Springer.

Examples

```
## Not run:
## microarray analysis example
data(golubMerge)
idxALL <- which(sample.labels== "ALL")
idxAML <- which(sample.labels == "AML")
pvals <- apply(golubMerge.exprs, 1, function(x){t.test(x[idxAML],x[idxALL])[[3]]})
zvals <- qnorm(1-pvals)
hist(zvals,100)
### Z-values are gaussian distributed, mix identifies a mixture of gaussians.
mix <- mixalg(obs=zvals, family="gaussian", startk=25)
hist(mix)

### get False discovery rate (Not-differential expressed genes are in component 1)
getFDR(mix, threshold=.4)

## End(Not run)
```

---

heage *Northeast Thailand malnourishment data.*

Description

When screening for subclinical malnourishment standardized scores like the HE/AGE (based on an international reference population) are recommended. HE/AGE is computed as the difference of the childs height minus the median height in the reference population divided by the standadard deviation. Measurements of 708 preschool children in northeast Thailand are contained in this data set.

Usage

```
data(heage)
```

Format

A data frame with 708 observations on the following variable.

`value` a numeric vector

References

B\"ohning, D. and Schlattmann, P. and Lindsay, B. (1992). "Computer-Assisted Analysis of Mixtures (C.A.MAN): Statistical Algorithms". *Biometrics*. **48**, 283–303.

`hepab` *Hepatitis B data from Schlattmann and B\"ohning (1993).*

Description

This data set appears as an example for an application of mixture modelling in disease mapping. The data is on hepatitis B cases in Berlin 1989.

Usage

```
data(hepab)
```

Format

A data frame with 23 observations on the following 2 variables.

`observations` a numeric vector

`expected` a numeric vector

References

Schlattmann, P. and B\"ohning, D. (1993). "Mixture models and disease mapping". *Statistics in Medicine*. **12**, 1943–1950.

`hist.CAMAN.object` *Histograms for finite mixture models*

Description

Graphical display of the data and the mixture. Intended for larger data sets.

Usage

```
## S3 method for class 'CAMAN.object'
hist(x, nbreaks=NULL, mixdens=TRUE, mixdens.col="red",
                return.mixdens=FALSE, data.plot=NULL, singleDistr=TRUE,
                main="", xlab="", plotlegend=TRUE, ...)
```

Arguments

x	A CAMAN.object.
nbreaks	Number of breaks in histogram. Defaults to NULL. In this case it is set to min(60,max(30,x@num.obs)). Passed on to hist.
mixdens	logical; plot mixture density?
mixdens.col	color of mixture density.
return.mixdens	logical; return the mixture density?
data.plot	data to plot. Defaults to NULL. In this case it is taken from the data of the mixture density.
singleDistr	logical; should the distributions of the single components be plotted?
main	character; heading of the plot.
xlab	character; label of x-axis.
plotlegend	logical; should a legend be plotted.
...	arguments to be passed on to hist.

Value

The function plots a histogram with the mixture density and its components overlaid. If return.mixdens is set to TRUE, the density at the breaks will also be returned.

Author(s)

Peter Schlattmann and Johannes Hoehne

References

Schlattmann, P. (2009). *Medical Applications of Finite Mixture Models*. Berlin: Springer.

---

leukDat *Childhood leukemia in the former GDR*

Description

In this example we present data from the former East Germany within the time period from 1980 to 1989. The data are taken from the cancer atlas of the former East Germany (Moehner et al. 1994). The data provided the regional distribution of childhood leukemia in the former East Germany.

Usage

```
data("leukDat")
```

References

Schlattmann, P.(2009) *Medical Applications of Finite Mixture Models.* Berlin: Springer.

Examples

```
## disease mapping example
data(leukDat)
data(GDRmap) # this is the map of the former GDR
mix.leuk <- mixalg(obs = "oleuk", pop.at.risk = "eleuk",
                   data = leukDat, family = "poisson")
mix.leuk
plot(GDRmap, col = mix.leuk@classification)
```

`mixalg` *Fitting Finite Mixture Models*

Description

This hybrid mixture algorithm combines the VEM algorithm for flexible support size and the EM algorithm for a fixed number of components. The solution of the VEM algorithm provides starting values for the EM algorithm. By the NPMLE theorem the EM algorithm thus starts very close to the global maximum and proper convergence of the EM algorithm to a global maximum is ensured.

The algorithm proceeds as follows

Step 1: Define an approximating grid `lambda[1], ..., lambda[L]`

Step 2: Use the VEM algorithm to maximize L(P) in the simplex Ω and identify grid points with positive support. Here positive support is defined as `p[j] >= epsilon` (often `epsilon = 10^-2`). This gives an initial estimate of `k`.

Step 3: Use these `k` points and corresponding mixing weights p[j] as starting values for the EM algorithm

Step 4: Collapse identical components if `| lambda[j]- lambda[i] | < delta` (often delta=0.05) for i != j

Step 5: Obtain the final number of components `k`

This sequential algorithm leads to an initial estimate of the NPMLE and a proper solution for the subsequent EM algorithm. Crucial points are the definitions of δ and ϵ. Depending on these settings different solutions could result from this algorithm.

Usage

```
mixalg(obs, weights=NULL, family="gaussian", data=NULL, pop.at.risk=NULL,
       var.lnOR=NULL, limit=0.01, acc=10^(-7), numiter=5000, startk=50)
```

Arguments

`obs`	observed / dependent variable. Vector or colname of `data`. Must be specified!
`weights`	weights of the data. Vector or colname of `data`. Default is `NULL`.
`family`	the underlying type density function as a character ("gaussian", "poisson" or "binomial")!

data	an optional data frame. obs, weights, pop.at.risk and var.lnOR can be specified as column name of the data frame.
pop.at.risk	population at risk: These data could be used to determine a mixture model for Poisson data. Vector or colname of data. Default isNULL.
var.lnOR	variances of the data: These variances might be given when working with meta analyses! Vector or colname of data. Default is NULL.
limit	parameter to control the limit of union several components. Default is 0.01.
acc	convergence criterion. VEM and EM loops stop when deltaLL<acc. Default is 10^(-7).
numiter	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.
startk	starting/maximal number of components. This number will be used to compute the grid in the VEM. Default is 50.

Details

The documentation of leukDat contains a disease mapping example using mixalg and the documentation of golubMerge contains a microarray analysis example.

Value

The function returns a CAMAN.object, describing a finite mixture model. The main information about the mixture model is printed by just typing the <object>. Additional information is given in summary(object) (summary.CAMAN.object). Single attributes can be accessed using the @, e.g. mix@LL.

dat	(input) data
family	underlying type density function
LL	Likelihood of the final (best) iteration
BIC	Likelihood of the final (best) iteration
num.k	number of components obtained
p	probability of each component
t	parameter of distribution (normal distr. -> mean, poisson distr. -> lambda, binomial distr. -> prob)
component.var	variance of each component (ONLY if family == "gaussian")
prob	probabilies, belonging to each component
classification	classification labels for each observation (which.max of @prob).
steps	number of steps performed (EM, VEM).
VEM_result	result of VEM algorithm.
cl	the matched call.
is_metaAnalysis	parameter specifying, whether a meta analysis was performed.
VEM_result	Outcome of the VEM-algorithm, which was run before the EM.
finalacc	deltaLL of the final iteration (for VEM and EM)

Author(s)

Peter Schlattmann and Johannes Hoehne

References

D. B\"ohning, P. Schlattmann, B.G. Lindsay: C.A.MAN - Computer Assisted Analysis of Mixtures: Statistical Algorithms.*Biometrics*, 1992, 48, 283-303

P. Schlattmann: On bootstrapping the unknown number of components in finite mixtures of Poisson distributions. *Statistics and Computing*, 2005, 15, 179-188

Schlattmann, P. (2009). *Medical Applications of Finite Mixture Models.* Berlin: Springer.

See Also

mixalg.EM, mixalg.VEM, anova.CAMAN.object, mixcov, mixalg.boot

Examples

```
### POISSON data with weights: thai_cohort
data(thai_cohort)
mix <- mixalg(obs="counts", weights="frequency", family="poisson",
              data=thai_cohort, numiter=18000, acc=0.00001, startk=25)


# meta analysis
data(aspirin)
mix <- mixalg(obs="logrr", var.lnOR="var", data=aspirin)


## See the documentation of golub.Merge for a
## microarray analysis example using mixalg

## See the documentation of leukDat for a disease
## mapping example using mixalg
```

---

mixalg.boot *bootstrap replication / validation of finite mixture models*

Description

This function may be used to estimate the number of components based on a nonparametric bootstrap approach.A bootstrap sample is obtained from the original sample with replacement. Corresponding to the bootstrap data set we obtain an estimate of the number of components k applying a combination of the VEM- and EM algorithm. The bootstrap algorithm involves drawing B independent bootstrap samples and estimating k using the hybrid mixture algorithm. The result is the bootstrap distribution of the number of components k. The mode of this distribution is taken as an estimate of the number of components.

Usage

```
mixalg.boot(mix, nboot=500, limit=0.01, acc=10^(-5), numiter=5000,
            startk=50, returnBootstrapRep=FALSE)
```

Arguments

	the parameters `limit, acc, numiter` and `startk` are used for the VEM algorithm in each bootstrap sample.
	A CAMAN-object which quantifies a finite mixture model.
`nboot`	number of bootstrap replications
`limit`	parameter to control the limit of union several components. Default is 0.01.
`acc`	convergence criterion. VEM and EM loops stop when deltaLL<acc. Default is 10^(-7).
`numiter`	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.
`startk`	starting/maximal number of components for the VEM algorithm in each bootstrap sample. This number will be used to compute the grid in the VEM. Default is 50.
`returnBootstrapRep`	A Boolean that indicates whether the bootstrapped data should be returned or not

Value

The function returns a list, describing the bootstrap replications:

`$dat.bootstrap`	(used) sampled data
`$LL`	Likelihood of the final solutions of each bootstrap replication
`$LL_k1`	vector with LL for each bootstrap replication using a homogeneous model (k=1)
`$numk.boot`	number of components obtained in replications.

Note

`mixalg.Boot` and `mixboot` are deprecated names for the `mixalg.boot` function.

Author(s)

Peter Schlattmann and Johannes Hoehne

Examples

```
### POISSON data with weights: thai_cohort
data(thai_cohort)
m.thai <- mixalg(obs="counts", weights="frequency",
                 family="poisson", data=thai_cohort,
                 acc = 0.00003)
```

```
## Not run: boot <- mixalg.boot(m.thai, nboot=1000) #may take a few minutes

### POISSON data with observed and expected data: hepab
data(hepab)
mix <- mixalg(obs="observations",pop.at.risk="expected",family= "poisson",data=hepab)
## Not run: boot <- mixalg.boot(mix, nboot=250) #may take some time
table(boot$numk.boot)
## End(Not run)
```

`mixalg.EM` *EM algorithm*

Description

In the fixed support size case the number of components `k` is assumed to be known. Here the unknown parameters are the mixing weights `p[j]` and the parameters `lambda[j]` of the subpopulation.

Estimation of these models' parameters is usually achieved by application of the EM algorithm.

Usage

```
mixalg.EM (mix = NULL, p, t, obs = NULL, weights = NULL, family = "gaussian",
    data = NULL, pop.at.risk = NULL, var.lnOR = NULL, limit = 0.01,
    acc = 10^(-7), numiter = 5000)
```

Arguments

`mix`	A CAMAN-object which quantifies a finite mixture model and the input data.
`p`	vector containing the starting values for p
`t`	vector containing the starting values for the distribution specific parameters (poisson->λ, gaussian-> μ)
`obs`	observed / dependent variable. Vector or colname of `data`. Must be specified!
`weights`	weights of the data. Vector or colname of `data`. Default is `NULL`.
`family`	the underlying type density function as a character ("gaussian", "poisson" or "binomial")!
`data`	an optional data frame. `obs`, `weights`, `pop.at.risk` and `var.lnOR` can be specified as column name of the data frame.
`pop.at.risk`	population at risk: These data could be used to determine a mixture model for Poisson data. Vector or colname of `data`. Default is`NULL`.
`var.lnOR`	variances of the data: These variances might be given when working with meta analyses! Vector or colname of `data`. Default is `NULL`.
`limit`	parameter to control the limit of union several components. Default is 0.01.
`acc`	convergence criterion. VEM and EM loops stop when deltaLL<acc. Default is 10^(-7).
`numiter`	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.

Value

The function returns a CAMAN.object object, thus the same as mixalg!

Author(s)

Peter Schlattmann and Johannes Hoehne

Examples

```
data(vitA)
m1<-mixalg.EM(obs="logrr",var.lnOR="var" ,family="gaussian",
              p=c(1),t=c(0),data=vitA)
m2<-mixalg.EM(obs="logrr",var.lnOR="var" ,family="gaussian",
              p=c(0.5,0.5),t=c(-0.3,0.2),data=vitA)

# apply EM-algorithm on an existing CAMAN.object:
data(thai_cohort)
mix0 <- mixalg(obs="counts", weights="frequency", family="poisson",
               data=thai_cohort, numiter=18000, acc=0.00001,startk=25)
em0<-mixalg.EM(mix0,p=c(1),t=c(1))
em1<-mixalg.EM(mix0,p=c(0.7,0.3),t=c(2,9))
```

---

mixalg.VEM *VEM algorithm*

Description

When fitting finite mixture models two cases must be distinguished. The flexible support size case, where no assumption about the number of components k is made in advance and the fixed support size case. For the flexible support size case the VEM-algorithm can be used.

The algorithm proceeds as follows:
Step 1: Define an approximating grid lambda[1], . . . , lambda[L]
Step 2: Use the VEM algorithm to maximize L(P) in the simplex Ω and identify grid points with positive support.

Usage

```
mixalg.VEM(mix = NULL, obs=NULL, weights=NULL, data=NULL, pop.at.risk=NULL,
           var.lnOR=NULL, family="gaussian", limit=0.01, acc=10^(-7),
           numiter=5000, startk=50)
```

Arguments

mix	A CAMAN-object which quantifies a finite mixture model and the input data.
obs	observed / dependent variable. Vector or colname of data. Must be specified!
weights	weights of the data. Vector or colname of data. Default is NULL.
family	the underlying type density function as a character ("gaussian", "poisson" or "binomial")!
data	an optional data frame. obs, weights, pop.at.risk and var.lnOR can be specified as column name of the data frame.
pop.at.risk	population at risk: These data could be used to determine a mixture model for Poisson data. Vector or colname of data. Default isNULL.
var.lnOR	variances of the data: These variances might be given when working with meta analyses! Vector or colname of data. Default is NULL.
limit	parameter to control the limit of union several components. Default is 0.01.
acc	convergence criterion. VEM and EM loops stop when deltaLL<acc. Default is 10^(-7).
numiter	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.
startk	starting/maximal number of components. This number will be used to compute the grid in the VEM. Default is 50.

Value

The function returns a CAMAN.VEM.object object.

Author(s)

Peter Schlattmann and Johannes Hoehne

Examples

```
data(vitA)
m0<-mixalg.VEM(obs="logrr",var.lnOR="var",family="gaussian", data=vitA,startk=20)
plot(m0@totalgrid[,2],m0@totalgrid[,3],type="l",xlab="parameter",ylab="gradient")
m1<-mixalg.EM(obs="logrr",var.lnOR="var" ,family="gaussian",p=c(1),t=c(0),data=vitA)
```

---

mixcov *Fitting mixture models with covariates*

Description

The function mixcov can be used to estimate the parameters of covariate adjusted mixture models. This is done using the EM algorithm. The function first performs the necessary data augmentation and then applies the EM-algorithm. Covariates may be included as fixed and random effects into the model.

Thus, the EM algorithm for covariate adjusted mixture models implies to perform first the necessary data augmentation, and then based on starting values for `p[j]` and `beta[j]` the computation of posterior probabilities `e[ij]`. This is the E-step. In the M-step new mixing weights `p[j]` and regression coefficients `beta[j]` are computed.

Then the algorithm performs as follows:
\ Step 0: Let `P` be any vector of starting values
Step 1: Compute the E-step, that is estimate the probability of component membership for each observation
Step 2: Compute the M-step, that is compute new mixing weights and model coefficients
Proceed until convergence is met.

Usage

```
mixcov(dep, fixed, random="", data, k, weight=NULL, pop.at.risk=NULL,
       var.lnOR=NULL, family="gaussian", maxiter=50,
       acc=10^-7, returnHomogeneousModel = FALSE)
```

Arguments

`dep`	dependent variable. Vector or colname of `data`. Must be specified!
`fixed`	fixed effects. Vector or colname of `data`. Must be specified!
`random`	random effects. Vector or colname of `data`. Must be specified!
`k`	number of components.
`weight`	weights of the data. Vector or colname of `data`. Default is `NULL`.
`family`	the underlying type density function as a character ("gaussian" or "poisson")!
`data`	an optional data frame. `obs`, `weights`, `pop.at.risk` and `var.lnOR` can be specified as column name of the data frame.
`pop.at.risk`	population at risk: An offset that could be used to determine a mixture model for Poisson data from unequally large populations at risk. Vector or colname of `data`. Default is`NULL`.
`var.lnOR`	variances of the data: These variances might be given when working with meta analyses! Vector or colname of `data`. Default is `NULL`.
`maxiter`	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.
`acc`	convergence criterion. VEM and EM loops stop when deltaLL<acc. Default is 10^(-7).

`returnHomogeneousModel`
: boolean to indicate whether the homogeneous model (simple glm) should be returned too. Default is FALSE.

Value

The function returns a CAMAN.glm.object (S4-class), describing a mixture model with covariates. The main information about the mixture model is printed by just typing the <object>. Additional information is given in `summary(object)`. Single attributes can be accessed using the `@`, e.g. mix@LL.

`dat`	(input) data
`family`	underlying type density function
`LL`	Likelihood of the final (best) iteration
`BIC`	Likelihood of the final (best) iteration
`num.k`	number of components obtained
`p`	probability of each component
`t`	parameter of distribution (normal distr. -> mean, poisson distr. -> lambda, binomial distr. -> prob)
`coefMatrix`	complete coefficient matrix
`prob`	probabilies, belonging to each component
`steps`	number of steps performed (EM).
`commonEffect`	common effects
`hetvar`	heterogeneity variance
`commonEffect`	common effects
`classification`	classification labels for each observation (`which.max` of prob).
`cl`	the matched call.
`fittedObs`	predictions of the fitted model, given the observations

Author(s)

Peter Schlattmann and Johannes Hoehne

Examples

```
### Toy data: simulate subjects with a different relationship between age and salariy
  grps = sample(1:3,70, replace=TRUE) #assign each person to one group
  salary=NULL
  age = round(runif(70) * 47 + 18)
  #random effects: age has a different influence (slope) on the salary
  salary[grps == 1] = 2000 + 12 * age[grps==1]
  salary[grps == 2] = 4000 + 4 * age[grps==2]
  salary[grps == 3] = 3200 + (-15) * age[grps==3]
  salary = salary + rnorm(70)*30  #some noise
  sex =sample(c("m","w"), 70, replace=TRUE)
```

```
salary[sex=="m"] = salary[sex=="m"] * 1.2 #men earn 20 percent more than women
salaryData = data.frame(salary=salary, age=age, sex=sex)
tstSalary <- mixcov(dep="salary", fixed="sex", random="age" ,data=salaryData,
                    k=3,family="gaussian", acc=10^-3)


### POISSON data:
data(NoP)
ames3 <- mixcov(dep="count",fixed=c("dose", "logd"),random="",data=NoP,
                k=3,family="poisson")

### Gaussian data
data(betaplasma)
beta4 <- mixcov(dep="betacaro", fixed=c("chol","sex","bmi"), random="betadiet",
                data=betaplasma, k=4, family="gaussian")
```

NoP *Ames test data: Mutagenicity of 4NoP*

Description

Ames test data where a strain of SalmonellaTA98 was activated with a homogenate of rat liver cells and exposed to 4-nitro-ophenylenediamine (4NoP) with various doses. 4NoP is a nitrated aromatic amine. This chemical is a component of both semipermanent and permanent hair dye formulations. The substance is frequently used as a comparator in mutagenicity tests. The are taken from Margolin et al. (1989) and denote the bacteria count for various doses of4NoP. The primary question of interest is whether 4NoP acts mutagenically in the Ames test.

Usage

```
data("NoP")
```

Format

A data frame with 100 rows and 3 columns (dose, count, logd)

References

Margolin, B. H., B. S. Kim, and K. J. Risko. (1989). "The Ames Salmonella Microsome Mutagenicity Assay: Issues of Inference and Validation." *Journal of the American Statistical Association* **84**, 651–661.

Schlattmann, P.(2009). *Medical Applications of Finite Mixture Models.* Berlin: Springer.

Examples

```
data(NoP)
ames3 <- mixcov(dep="count",fixed=c("dose", "logd"),random="",data=NoP,
                k=3,family="poisson")
```

PCT *Procalcitonin as diagnostic marker for sepsis*

Description

Procalcitonin is a promising marker for identification of bacterial infections. The accuracy and clinical value of procalcitonin for diagnosis of sepsis in critically ill patients was assessed in this meta-analysis.

Medline, Embase, ISI Web of Knowledge, the Cochrane Library, Scopus, BioMed Central, and Science Direct were searched, from inception to Feb 21, 2012, and reference lists of identified primary studies. We included articles written in English, German, or French that investigated procalcitonin for differentiation of septic patients–those with sepsis, severe sepsis, or septic shock–from those with a systemic inflammatory response syndrome of non-infectious origin. Studies of healthy people, patients without probable infection, and children younger than 28 days were excluded. Two independent investigators extracted patient and study characteristics; discrepancies were resolved by consensus.

This search returned 3487 reports, of which 30 fulfilled the inclusion criteria, accounting for 3244 patients.

Usage

```
data("PCT")
```

Format

A data frame consisting of 35 data sets (rows) and 10 attributes (columns)

Variable Names in order from left to right:

Study Study

Year Year

TP True positive

FP False positive

FN False negative

TN True negative

logitTPR logit-true positive rate

logitTNR logit-true negative rate

varlogitTPR Variance of logit TPR

varlogitTNR Variance of logit TPR

References

Wacker C, Prkno A, Brunkhorst FM, Schlattmann P. Procalcitonin as a diagnostic marker for sepsis: a systematic review and meta-analysis. Lancet Infect Dis. 2013 May;13(5):426-35. doi: 10.1016/S1473-3099(12)70323-7

Examples

```
#Use the VEM-algorithm for a diagnostic meta-analysis based on a mixture
#of bivariate  normal densities.
#Study specific fixed variances are based on logit transformed
#sensitivity and specificity.

data(PCT)
names(PCT)

m0 <- bivariate.VEM(obs1 = logitTPR, obs2 = logitTNR,
                    var1 = varlogitTPR, var2 = varlogitTNR,
                    type = "meta", data=PCT, startk=20)
```

---

plot.CAMAN.BIMIXALG.object

Plot ellipses

Description

Graphical display of the EM algorithm and bivariate MIXALG

Usage

```
## S3 method for class 'CAMAN.BIMIXALG.object'
plot(x, ellipse, ...)
## S3 method for class 'CAMAN.BIEM.object'
plot(x, ellipse, ...)
```

Arguments

x	a CAMAN.BIEM.object or a CAMAN.BIMIXALG.object
ellipse	logical. Plot ellipse? Defaults is TRUE
...	further arguments. Currently ignored.

Author(s)

Peter Schlattmann

References

Schlattmann, P. (2009). *Medical Applications of Finite Mixture Models*. Berlin: Springer.

`rs12363681` *Gene calling*

Description

This data set can be used to perform a cluster analyis of bivariate data.

This data set contains SNP of 3680 individuala

Usage

```
data("rs12363681")
```

Format

A data frame consisting of 3680 data sets (rows) and 2 attributes (columns)

References

Schlattmann, P.(2009) *Medical Applications of Finite Mixture Models.* Berlin: Springer.

Examples

```
## Not run:
# Example
# EM and classification for bivariate data
data(rs12363681)
test <- bivariate.mixalg(obs1=x, obs2=y, type="bi",
                         lambda1=0, lambda2=0, p=0,
                         data=rs12363681, startk=20, class="TRUE")
# scatter plot with ellipse
plot(test)
# scatter plot without ellipse
plot(test, ellipse = FALSE)

## End(Not run)
```

thai_cohort *Cohort study in north east Thailand*

Description

In a cohort study in northeast Thailand the health status of 602 preschool children was checked every 2 weeks from June 1982 until September 1985 (Schelp et al. 1990). In this time period it was recorded how often the children showed symptoms of fever, cough, running nose, or these symptoms together. The frequencies of these illness spells are given in the data set.

Usage

```
data("thai_cohort")
```

References

Schelp, F., P. Vivatanasept, P. Sitaputra, S. Sornmani, P. Pongpaew, N. Vudhivai, S. Egormaiphol, and D. B\"ohning. "Relationship of the morbidity of under-fives to anthropometric measurements and community health intervention." *Trop Med Parasitol*, 1990, 41(2), 121–126.

Schlattmann, P. (2009) *Medical Applications of Finite Mixture Models*. Berlin: Springer.

Examples

```
data("thai_cohort")
mix <- mixalg(obs="counts", weights="frequency", family="poisson", data=thai_cohort,
              numiter=18000, acc=0.00001, startk=25)
```

vem_grad *VEM algorithm for univariate data, for bivariate data and for meta data*

Description

This function

Usage

```
vem_grad(obs1, obs2, type, data, var1, var2,
         lambda1, lambda2, p, startk,
         numiter=5000, acc=1.e-7)
```

Arguments

obs1	the first column of the observations
obs2	the second column of the observations
type	kind of data
data	an optional data frame
lambda1	Means of the first column of the observations
lambda2	Means of the second column of the observations
p	Mixing weight
var1	Variance of the first column of the observations(only for meta-analysis)
var2	Variance of the second column of the observations (only for meta-analysis)
startk	starting/maximal number of components. This number will be used to compute the grid in the VEM. Default is 20.)
numiter	parameter to control the maximal number of iterations in the VEM and EM loops. Default is 5000.
acc	convergence criterion. Default is 1.e-7

Examples

```
## Not run:
#1. VEM-algorithm  for bivariate normally distributed data
#Examples
data(rs12363681)
vem_grad(obs1=x,obs2=y,type="bi", data=rs12363681,startk=20)
#2.VEM for metadata
data(CT)
vem_grad(obs1=logitTPR, obs2= logitTNR,
         var1= varlogitTPR, var2= varlogitTNR,
         type="meta", data=CT, startk=20)

## End(Not run)
```

vitamin *Meta-analysis if vitamin A supplementation of childhood mortality*

Description

Supplementation of vitamins is not only supposed to be beneficial for the prevention of cancer as discussed in Sect. 2.1. Fawzi et al. (1993) study the effect of vitamin A supplementation and childhood mortality in preschool children. They concluded that vitamin A supplements are associated with a significant reduction in mortality when given periodically to children at the community level. All studies were community randomized trials from South Asia or Southeast Asia, except the second study, which was from northern Sudan. This data set contains the log relative risk and corresponding variance. The data is supplied in three different formats

Usage

```
data("vitamin")
data("vitA")
data("vitA2")
```

References

Schlattmann, P. (2009) *Medical Applications of Finite Mixture Models.* Berlin: Springer.

Fawzi,W.W., T. C. Chalmers, M. G. Herrera, and F. Mosteller (1993). "Vitamin A supplementation and child mortality. A meta-analysis." *JAMA* **269**(7), 898–903.

Index