Package ‘randomLCA’

February 20, 2015

Type Package
Title Random Effects Latent Class Analysis
Version 1.0-2
Date 2015-01-09
Author Ken Beath <ken.beath@mq.edu.au>
Maintainer Ken Beath <ken.beath@mq.edu.au>
Description Fits random effects latent class models, as well as standard latent class models.
Depends R(>= 3.0.0), lattice, boot, fastGHQuad
Suggests xtable
LazyLoad yes
LazyData yes
NeedsCompilation yes
License GPL (>= 2)
Repository CRAN
Date/Publication 2015-01-09 21:57:48

R topics documented:

AIC ................................................................. 2
BIC ................................................................. 3
calc.cond.prob ................................................... 3
calc.cond2.prob ................................................. 4
calc.marg.prob ................................................. 5
class.probs ...................................................... 5
dentistry ......................................................... 6
fitted .............................................................. 7
genderrole ....................................................... 8
hivtests ......................................................... 9
logLik .......................................................... 10
myocardial ...................................................... 11
outcome.probs ................................................ 12
Description

Returns AIC for a randomLCA object.

Usage

```r
## S3 method for class 'randomLCA'
AIC(object, ...)
```

Arguments

- `object` randomLCA object
- `...` additional argument; currently none is used.

Value

AIC.

Author(s)

Ken Beath
BIC

BIC for randomLCA object

Description

Returns BIC for a randomLCA object.

Usage

```r
## S3 method for class 'randomLCA'
BIC(object, ...)
```

Arguments

- `object`: randomLCA object
- `...`: additional argument; currently none is used.

Value

BIC

Author(s)

Ken Beath

---

calc.cond.prob

Calculate Conditional Outcome Probabilities

Description

Calculates the conditional outcome probabilities for random effects models or for standard latent class returns the outcome probabilities. For random effects, the outcome probabilities may be calculated for various percentiles of the random effect.

Usage

```r
calc.cond.prob(object, conditionalp = 0.5)
```

Arguments

- `object`: RandomLCA object
- `conditionalp`: the percentiles for the random effect
calc.cond2.prob

Value

Returns a data frame containing class, block, outcome, outcomep (outcome probability) and perc (percentiles of the random effect) if conditionalp is specified. For example a conditionalp of 0.5 is the 50th percentile or the median corresponding to a random effect of zero. 0.025 and 0.975 correspond to the 2.5th and 97.5th percential, so the region between them if 95% of the variation in the data.

Author(s)

Ken Beath <ken.beath@mq.edu.au>

calc.cond2.prob  Calculate Conditional Outcome Probabilities for 2 Level Models

Description

The conditional probabilities are obtained integrating over the period random effect.

Usage

calc.cond2.prob(object, conditionalp = 0.5)

Arguments

object RandomLCA object
conditionalp the percentiles for the random effect

Value

Returns a data frame containing class, block, outcome, outcomep (outcome probability) and perc (percentiles of the random effect) if conditionalp is specified. For example a conditionalp of 0.5 is the 50th percentile or the median corresponding to a random effect of zero. 0.025 and 0.975 correspond to the 2.5th and 97.5th percential, so the region between them if 95% of the variation in the data.

Author(s)

Ken Beath <kenbeath@mq.edu.au>
**calc.marg.prob**

*Calculates Marginal Outcome Probabilities*

**Description**

Calculates the marginal outcome probabilities for a random effects latent class model, by integrating the outcome probability over the random effect. This is performed using Gauss-Hermite quadrature with the number of quadrature points specified for the model fitting.

**Usage**

```r
calc.marg.prob(object)
```

**Arguments**

- `object` randomLCA object

**Value**

Returns a data frame containing class, block, outcome, outcomep (outcome probability).

**Author(s)**

Ken Beath

---

**class.probs**

*Determines class probabilities for fitted model*

**Description**

The class probabilities for the model are returned.

**Usage**

```r
class.probs(object)
```

**Arguments**

- `object` randomLCA object

**Details**

Simply extracts the corresponding variable from the randomLCA object.

**Value**

A vector of class probabilities for each class.
Six dentists evaluated dental x-rays for incipient caries in Handelman et al (1986), data consists of 5 of the dentists analysed by Espeland and Handelman (1989) using a latent class model. Further analysis incorporating a random effects latent class model by Qu et al (1996), and by Albert and Dodd (2004).

Usage
data(dentistry)

Format
A data frame with 32 observations on the following 6 variables.

v1 Dentist 1
v2 Dentist 2
v3 Dentist 3
v4 Dentist 4
v5 Dentist 5
freq Number of subjects

Source
Espeland and Handelman (1989)

References


Examples

```r
## Not run:
data(dentistry)
dentistry.lca <- randomLCA(dentistry[,1:5], freq=dentistry$freq)
# start with constant loading
dentistry.lcarandom <- randomLCA(dentistry[,1:5], freq=dentistry$freq, random=TRUE, probit=TRUE)
# allow loading to vary by dentist
dentistry.lcarandomunequal <- randomLCA(dentistry[,1:5], freq=dentistry$freq, random=TRUE, constload=FALSE, probit=TRUE)

## End(Not run)
```

---

### fitted

<table>
<thead>
<tr>
<th>fitted values</th>
</tr>
</thead>
</table>

Description

Extract fitted values for randomLCA object

Usage

```r
## S3 method for class 'randomLCA'
fitted(object, ...)
```

Arguments

- `object` randomLCA object
- `...` additional argument; currently none is used.

Value

A data frame. The first columns of the data frame correspond to the patterns, followed by the frequency of each pattern, and then the fitted number for each pattern.

Author(s)

Ken Beath <ken.beath@mq.edu.au>
Gender Role Opinion Items

Description

Opinions collected on gender roles in a study by Felling et al (1987). This was originally published in Heinen (1996) and subsequently in Galindo Garre and Vermunt (2006).

Usage

data(genderrole)

Format

A data frame with 16 observations on the following 5 variables.

Q1 Women’s liberation sets women against men.
Q2 It’s better for a wife not to have a job because that always poses problems in the household, especially if there are children.
Q3 The most natural situation occurs when the man is the breadwinner and the woman runs the household and takes care of the children.
Q4 It isn’t really as important for a girl to get a good education as it is for a boy.
Q5 A woman is better suited to raise small children than a man.
Freq Number of subjects

Source

Galindo Garre and Vermunt (2006)

References


Examples

```r
# Not run:
data(genderrole)
# standard latent class
genderrole.lca1 <- randomLCA(genderrole[,1:5],freq=genderrole$Freq,nclass=1)
genderrole.lca2 <- randomLCA(genderrole[,1:5],freq=genderrole$Freq)
genderrole.lca3 <- randomLCA(genderrole[,1:5],freq=genderrole$Freq,nclass=3)
# repeat with random effect with constant loading
```
hivtests

HIV testing data

Description

Serum samples are tested for HIV by 4 different biossays in Alvord et al (1988) and sensitivity and specificity determined using latent class analysis. Qu et al (1996) repeat the analysis using a model incorporating a random effect.

Usage

data(hivtests)

Format

A data frame with 16 observations on the following 5 variables.

V1 Test 1
V2 Test 2
V3 Test 3
V4 Test 4
freq Number of subjects

Source

Qu, Tan and Kutner (1989)
References


Examples

data(hivtests)
# fit standard latent class
hivtests.lca2 <- randomLCA(hivtests[,1:4], freq=hivtests$freq)
# with random effect model is not identifiable

---

logLik

*log Likelihood for randomLCA object*

Description

Returns log Likelihood for a randomLCA object.

Usage

```r
## S3 method for class 'randomLCA'
logLik(object, ...)
```

Arguments

- `object` randomLCA object
- `...` additional argument; currently none is used.

Value

The loglikelihood.

Author(s)

Ken Beath
**Description**

Four tests were performed on hospital patients to determine if a myocardial infarction had occurred.

**Usage**

```r
data(myocardial)
```

**Format**

A data frame with 32 observations on the following 6 variables.

- `Q.wave`: result from ECG test
- `History`: clinical history
- `LDH`: flipped, enzyme related to tissue breakdown
- `CPK`: high, creatine kinase or creatine phosphokinase, related to muscle damage
- `freq`: Number of subjects

**Source**

Rindskopf and Rindskopf (1986)

**References**


**Examples**

```r
data(myocardial)
# fit 2 class model from Rindskopf and Rindskopf (1986)
myocardial.lca2 <- randomLCA(myocardial[,1:4], freq=myocardial$freq)
```
outcome.probs

Extract outcome probabilities for randomLCA object

Description

Extract outcome probabilities and confidence intervals for a randomLCA object.

Usage

outcome.probs(object, level = 0.95, boot = FALSE, type = "norm",
               R = ifelse(type == "norm", 199, 999), ...)

Arguments

- **object**: randomLCA object
- **level**: confidence interval
- **boot**: use parametric bootstrap to obtain confidence interval
- **type**: type of bootstrap confidence intervals to use, with "perc" or "norm" valid, see boot.ci for description. It seems reasonable to use the normal approximation.
- **R**: replications for parametric bootstrap
- ... additional argument; currently none is used.

Details

Confidence intervals are calculated based on asymptotic normality of the estimates transformed by either the inverse of the probit or logistic, or using parametric bootstrap. The asymptotic confidence intervals are currently only available for models without random effects. For the confidence intervals obtained from the parametric bootstrap, the bootstrap is performed on the data that has been transformed to the logit or probit scale, as appropriate. The samples are close to normal allowing for the use of confidence intervals based on the normal approximation. About 199 replications gives similar accuracy to percentile with 999.

Value

Data frame consisting of outcome probabilities and confidence intervals. One for each class.

Author(s)

Ken Beath

Examples

```r
# dentist data
data(dentistry)
# standard latent class with 2 classes
dentistry.lca2 <- randomLCA(dentistry[,1:5], freq=dentistry$freq, nclass=2)
print(outcome.probs(dentistry.lca2))
```
plot

Plot a randomLCA object

Description

Plots the outcome probabilities for a randomLCA object, for random effects objects this can be either marginal or conditional or both. For a 2 level random effects model conditional2 will condition on the subject random effect and integrate over the period random effects. Note that plot is based on the xyplot function.

Usage

```r
## S3 method for class 'randomLCA'
plot(x, ..., graphtype = ifelse(x$rrandom, "marginal","conditional"),
     conditionalp = 0.5, classhorizontal = TRUE)
```

Arguments

- `x`: randomLCA object
- `graphtype`: Type of graph
- `conditionalp`: For a conditional graph the percentile corresponding to the random effect at which the outcome probability is to be calculated
- `classhorizontal`: classes to be plotted across the page
- `...`: additional parameters to xyplot

Author(s)

Ken Beath <ken.beath@mq.edu.au>

See Also

calc.cond.prob, calc.marg.prob

Examples

```r
## Not run:
data(uterinecarcinoma)
# standard latent class with 2 classes
uterinecarcinoma.lca2 <- randomLCA(uterinecarcinoma[,1:7], freq=uterinecarcinoma$freq)
plot(uterinecarcinoma.lca2)
uterinecarcinoma.lcarandom2 <- randomLCA(uterinecarcinoma[,1:7],
                                          freq=uterinecarcinoma$freq, random=TRUE, probit=TRUE, quadpoints=41)
# default for random effects models is marginal
plot(uterinecarcinoma.lcarandom2)
# default for random effects models conditional is p=0.5 i.e. median
plot(uterinecarcinoma.lcarandom2, graphtype = "conditional")
```
# look at variability by plotting conditional probabilities at 0.05, 0.5 and 0.95
plot(uterinecarcinoma.lcarandom2, grplotype="conditional", conditionalp=c(0.05, 0.5, 0.95))

## End(Not run)

---

**post.class.probs**  
*Determines posterior class probabilities for fitted model*

### Description
The posterior class probabilities for each observed pattern and class is determined. These are returned as a data frame together with the patterns for each observation. If `class=0` is requested then all classes are returned, otherwise only the selected class.

### Usage

```r
post.class.probs(object, class=0)
```

### Arguments
- **object**: randomLCA object
- **class**: class to be returned. Zero returns all classes.

### Details
Extracts the corresponding data from the randomLCA object.

### Value
A data frame. The first columns of the data frame correspond to the patterns, followed by the frequency of each pattern, and then the posterior class probabilities for either the selected class or for all classes.

### Author(s)
Ken Beath
**randomLCA**

*Fits a Latent Class Model including a Random Effect*

---

**Description**

Fit latent class models, which may include a random effect.

**Usage**

```r
classrandomLCA(patterns,freq,nclass=2,calcSE=TRUE,notrials=20,
random=FALSE,byclass=FALSE,quadpoints=21,constload=TRUE,blocksize=dim(patterns)[2],
level2=FALSE,probit=FALSE,level2size=blocksize,
qniterations=5,penalty=0.0001,verbose=FALSE,
seed = as.integer(runif(1, 0, .Machine$integer.max)))
```

**Arguments**

- `patterns`: Data frame or matrix of 0 and 1 defining the outcome patterns
- `freq`: Frequency for each outcome pattern, if missing this is calculated from the patterns, and the patterns are summarised to remove duplicate values.
- `nclass`: Number of classes to be fitted
- `calcSE`: Calculate standard errors for parameters
- `notrials`: For a standard latent class model, the number of random starting values used
- `random`: Random effect
- `byclass`: Random effect by class
- `quadpoints`: Number of quadrature points for adaptive quadrature
- `constload`: Outcome loadings are constant for random effects model
- `blocksize`: Where a random effects (single level) model is broken into blocks, that is the loadings are repeated, this defines the size of the blocks?
- `probit`: Probit model for random effect
- `level2`: Fit 2 level random effects model (further details to follow)
- `level2size`: Size of level 2 blocks if fitting 2 level models
- `qniterations`: Number of Quasi-Newton iterations within each EM/adaptive cycle. Decrease if there is a failure to converge
- `penalty`: penalty applied to likelihood for outcome probabilities. Shrinks outcome probabilities in slightly and can prevent extreme values. Setting penalty to 0 will produce an unpenalized fit.
- `verbose`: Prints fit progress if true
- `seed`: Initial random seed for generating starting values. This can be set to guarantee that the fit is the same each time, including the order of the classes.
Details

The structure of the patterns is assumed to be a number of blocks of different outcomes each of level2size, allowing outcomes to be repeated. Each outcome is assumed to have its own loading. An example is the width of the patterns is n and the level2size is n, resulting in n outcomes and therefore n loadings. Alternatively if the level2size is 1, then there are n repeats of the same outcome (but with different probabilities) with the same loading. In practice they may not be the same type of outcome, but usually will be.

The algorithm used is EM for the standard latent class and adaptive (in the sense of moving the location of the quadrature points) Gauss-Hermite quadrature for the random effects models. The number of quadrature points defaults to 21.

NOTE: in the returned object there are fields for patterns and frequencies. If frequencies are not supplied then the patterns and frequencies are constructed. If frequencies are supplied then zero rows are removed. When frequencies are supplied it is assumed that the data has been simplified. The returned class probabilities etc, all correspond to the simplified patterns, not to the original data.

Value

randomLCA object This contains

- fit: Fit object from optim
- nclass: Number of classes
- classp: Class probabilities
- outcomep: Outcome probability
- lambdacoef: Loadings
- se: Standard errors corresponding to results returned by optim
- np: Number of parameters
- nobs: Number of observations in total
- logLik: log likelihood for fitted model
- penlogLik: Penalised log likelihood for fitted model
- observed: Observed numbers corresponding to each pattern
- fitted: Fitted number corresponding to each pattern
- deviance: Deviance
- classprob: Posterior class probability for each pattern
- bics: BIC obtained for each trial when fitting initial latent class models
- call: call to randomLCA
- random: random parameter to randomLCA
- constload: constload parameter to randomLCA
- level2: level2 parameter to randomLCA
- level2size: level2size parameter to randomLCA
- byclass: byclass parameter to randomLCA
ranef

probit probit parameter to randomLCA
quadpoints quadpoints parameter to randomLCA
blocksize blocksize parameter to randomLCA
freq frequency of each pattern
qiterations qiterations parameter to randomLCA
penalty penalty parameter to randomLCA

Author(s)
Ken Beath

Examples

## Not run:
# dentist data
data(dentistry)
# standard latent class with 2 classes
dentistry.lca2 <- randomLCA(dentistry[,1:5],freq=dentistry$freq,nclass=2)
# random effects model with constant random effect loading
dentistry.lca2random <- randomLCA(dentistry[,1:5],freq=dentistry$freq,
nclass=2,random=TRUE,constload=TRUE,probit=TRUE)
# allow loading to vary by dentist
# this is the ZLCR model from Qu et al (1996)
dentistry.lca2randomT <- randomLCA(dentistry[,1:5],freq=dentistry$freq,
nclass=2,random=TRUE,constload=FALSE,probit=TRUE)

## End(Not run)

---

**ranef**

*Extract random effects from a randomLCA object*

**Description**

Extracts the Empirical Bayes estimates of the random effects.

**Usage**

```r
## S3 method for class 'randomLCA'
ranef(object, ...)
```

**Arguments**

- **object** randomLCA object with a random effect
- **...** additional argument; currently none is used.
Value

A matrix with the first column containing the random effects and the second column the standard error of the random effects.

Author(s)

Ken Beath

refit

Refit an randomLCA object

Description

Refits an randomLCA object using new data.

Usage

## S3 method for class 'randomLCA'
refit(object, newpatterns, newfreq, ...)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>object</td>
<td>randomLCA object</td>
</tr>
<tr>
<td>newpatterns</td>
<td>the new patterns that are to be fitted using the existing model</td>
</tr>
<tr>
<td>newfreq</td>
<td>the frequencies corresponding to the patterns if required</td>
</tr>
<tr>
<td>...</td>
<td>additional argument; currently none is used.</td>
</tr>
</tbody>
</table>

Value

The fitted model to the new data.

Author(s)

Ken Beath
simulate

Simulate data from a fitted randomLCA model

Usage

```r
## S3 method for class 'randomLCA'
simulate(object, nsim, seed, ...)
```

Arguments

- `object`: randomLCA object
- `nsim`: number of data sets to be simulated
- `seed`: random seed
- `...`: additional optional arguments.

Details

Generates random data from the supplied object.

Value

A simulated data frame.

Author(s)

Ken Beath

summary

Summary for randomLCA object

Description

Summarises the fit of a randomLCA object.

Usage

```r
## S3 method for class 'randomLCA'
summary(object, ...)
Arguments

object  randomLCA object
...  additional argument; currently none is used.

Value

logLik  Log Likelihood
AIC  AIC
BIC  BIC
nclass  no of classes
probit  link is probit
classp  class probabilities
outcomep  outcome probabilities (conditional)
margoutcomep  outcome probabilities (marginal), if model contains random effects
random  model includes random effects
level2  model has 2 level hierarchy
constload  loadings are constant by outcome
byclass  lambda and tau vary by class
lambdacoef  lambda coefficients
taucoef  tau coefficients

Author(s)

Ken Beath

symptoms  Symptoms data

Description

This is the data for Beath and Heller (2009).
Allergy and respiratory symptoms for infants 0 to 2 years in six month periods. Outcome is presence or absence of symptom in the six months. Original data was collected at Visits 1-7 over the 2 year period which were summarised to six month periods.

Note that these models can be slow to fit, with the "symptoms.lca2random2" model taking about 1-2 hours.

Thanks to the investigators of the CAPS study for making the data available.

Usage

data(sympotms)
**Format**

A data frame with 444 observations on the following 17 variables.

- Nightcough.13 Night cough in visits 1-3
- Wheeze.13 Wheeze in visits 1-3
- Itchy rash.13 Itchy rash in visits 1-3
- FlexDerma.13 Flexural Dermatitis in visits 1-3
- Nightcough.45 Night cough in visits 1-3
- Wheeze.45 Wheeze in visits 4-5
- Itchy rash.45 Itchy rash in visits 4-5
- FlexDerma.45 Flexural Dermatitis in visits 4-5
- Nightcough.6 Night cough in visit 6
- Wheeze.6 Wheeze in visit 6
- Itchy rash.6 Itchy rash in visit 6
- FlexDerma.6 Flexural Dermatitis in visits 1-3
- Nightcough.7 Night cough in visit 7
- Wheeze.7 Wheeze in visit 7
- Itchy rash.7 Itchy rash in visit 7
- FlexDerma.7 Flexural Dermatitis in visit 7

**Freq** Number of subjects

**Source**

Mihrshai et al (2001)

**References**


**Examples**

```r
## Not run:
data(symptoms)
symptoms.lca2 <- randomLCA(symptoms[,1:16], freq=symptoms$Freq, nclass=2)
symptoms.lca2random <- randomLCA(symptoms[,1:16], freq=symptoms$Freq, random=TRUE, nclass=2, blocksize=4, constload=FALSE)
symptoms.lca2random2 <- randomLCA(symptoms[,1:16], freq=symptoms$Freq, random=TRUE, level2=TRUE, nclass=2, level2size=4, constload=FALSE)

## End(Not run)
```
uterinecarcinoma  Uterine Carcinoma Data

Description
Classification of 118 histology samples by 118 pathologists. Original classification in Holmquist et al (1967) was to one of five categories, this has been reduced to two. Analysed by a number of authors, with a random effects in Qu et al (1996).

Usage

data(uterinecarcinoma)

Format
A data frame with 20 observations on the following 8 variables.

V1  Pathologist 1
V2  Pathologist 2
V3  Pathologist 3
V4  Pathologist 4
V5  Pathologist 5
V6  Pathologist 6
V7  Pathologist 7
freq  Number of observed pattern

Source
Qu et al (1996)

References


Examples

```r
## Not run:
data(uterinecarcinoma)

uterinecarcinoma.lcarandom2 <- randomLCA(uterinecarcinoma[,1:7],
   freq=uterinecarcinoma$freq,random=TRUE,probit=TRUE,quadpoints=41)
# LCR1 model of Que et al. This is fairly unstable and
# is also slow and doesn't improve the model fit
```
uterinecarcinoma

uterinecarcinoma.lcarandom2by <- randomLCA(uterinecarcinoma[,1:7], freq=uterinecarcinoma$freq, byclass=TRUE, random=TRUE, probit=TRUE, quadpoints=61)
## End(Not run)
Index

*Topic **datasets**
- dentistry, 6
- genderrole, 8
- hivtests, 9
- myocardial, 11
- symptoms, 20
- uterinecarcinoma, 22

*Topic **methods**
- AIC, 2
- BIC, 3
- calc.cond.prob, 3
- calc.cond2.prob, 4
- calc.marg.prob, 5
- class.probs, 5
- fitted, 7
- logLik, 10
- outcome.probs, 12
- plot, 13
- post.class.probs, 14
- ranef, 17
- refit, 18
- simulate, 19
- summary, 19

*Topic **multivariate**
- randomLCA, 15

AIC, 2

BIC, 3

calc.cond.prob, 3, 13
calc.cond2.prob, 4
calc.marg.prob, 5, 13
class.probs, 5

dentistry, 6

fitted, 7

genderrole, 8

hivtests, 9

logLik, 10

myocardial, 11

outcome.probs, 12

plot, 13

post.class.probs, 14

randomLCA, 15

ranef, 17

refit, 18

simulate, 19

summary, 19

symptoms, 20

uterinecarcinoma, 22

24