

Package ‘LCTMtools’

September 24, 2018

Type Package

Title Latent Class Trajectory Models: Tools for checking adequacy

Version 0.1.2

Author Hannah Lennon

Maintainer Hannah Lennon <hannah.lennon.manchester@gmail.com>

Description A selection of model adequacy tests for Latent Class Trajectory Models (LCTMs) which include the APPA (average posterior probability of assignment), the OCC (odds of correct classification), entropy, Relative entropy.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.0

R topics documented:

actual_proportions	2
appa	2
bmi	3
bmi_long	3
class_assignment	4
confusion_matrix	4
entropy	5
kappa_matrix	5
LCTMcompare	6
LCTMtoolkit	7
LCTMtools	7
mismatch	8
mmlcr_to_lctm	8
occ	9
relative_entropy	9
sastraj_to_lctm	10
Index	11

actual_proportions	<i>The proportion within each class AFTER class assignment (using max posterior rule)</i>
--------------------	---

Description

The proportion within each class AFTER class assignment (using max posterior rule)

Usage

```
actual_proportions(p)
```

Arguments

p is the posterior probabilities of assignment of dimensions, K columns and N rows

Value

The proportion within each class AFTER class assignment (using max posterior rule)

Examples

```
## Not run: actual_proportions(p)
```

appa	<i>Computes the Average Posterior Probability Assignment (APPA) for a K latent class trajectory model.</i>
------	--

Description

Computes the Average Posterior Probability Assignment (APPA) for a K latent class trajectory model.

Usage

```
appa(p)
```

Arguments

p is the posterior probabilities of assignment of dimensions, K columns and N rows

Value

The Average Posterior Probability Assignment (APPA) for each class

Examples

```
## Not run: appa(p)
```

bmi	<i>Body Mass Index (BMI) longitudinal measures of 10,000 individuals.</i>
-----	---

Description

A list containing a long and wide format dataset of BMI values of 10,000 individuals.

Usage

bmi

Format

A list of two formats data frame with 53940 rows and 10 variables:

ID Individual ID

BMI Body mass index of the individual at times T1,T2, T3 and T4, in kg/m²

T Time of BMI measure, in years

true_class Tag to identify the class the individual BMI data was simulated from

bmi_long	<i>Body Mass Index (BMI) longitudinal measures of 10,000 individuals.</i>
----------	---

Description

A list containing a long and wide format dataset of BMI values of 10,000 individuals.

Usage

bmi_long

Format

A list of two formats data frame with 53940 rows and 10 variables:

ID Individual ID

Age Age of BMI measure, in years

BMI Body mass index of the individual at times T1,T2, T3 and T4, in kg/m²

true_class Tag to identify the class the individual BMI data was simulated from

class_assignment	<i>Maximum Class assignment</i>
------------------	---------------------------------

Description

Maximum Class assignment

Usage

```
class_assignment(p)
```

Arguments

p is the posterior probabilities of assignment of dimensions, K columns and N rows

Value

This function computes the Average Posterior Probability Assignment (APPA) for a K latent class trajectory model.

Examples

```
## Not run: class_assignment(p)
```

confusion_matrix	<i>A confusion matrix</i>
------------------	---------------------------

Description

A confusion matrix

Usage

```
confusion_matrix(model1, model2)
```

Arguments

model1 A fitted model from the lcmm R package (or from SAS passed through the SASmodelbuilder() function)

model2 is the posterior probabilities of assignment of dimensions, K columns and N rows

name1 optional paramter to pre-specify name of model

name2 optional paramter to pre-specify name of model

Value

A confusion matrix between two models with the same number of classes

Examples

```
## Not run:
data(bmi_long, package="LCTMtools")
require(lcmm)
model1 <- hlme(BMI ~Age, mixture= ~Age, random= ~Age, nwg=T, ng=2, subject="ID", data=bmi_long[1:500, ])
model2 <- hlme(BMI ~Age, mixture= ~Age, random= ~1, nwg=F, ng=2, subject="ID", data=bmi_long[1:500, ])
confusion_matrix(model1, model2)
## End(Not run)
```

entropy	<i>Entropy</i>
---------	----------------

Description

A global measure of uncertainty with values close to zero implying a good model. Entropy is a global measure of classification uncertainty, which takes into account all $N \times K$ posterior probabilities. The entropy of a model is defined as which takes values from $[0, \text{infinity})$, with higher values indicating a larger amount of uncertainty. Entropy values closest to 0 correspond to models with least classification uncertainty.

Usage

```
entropy(p)
```

Arguments

`p` is the posterior probabilities of assignment of dimensions, K columns and N rows

Value

Entropy value between (0, infinity)

Examples

```
## Not run: entropy(p)
```

kappa_matrix	<i>Kappa matrix</i>
--------------	---------------------

Description

Kappa matrix of cohen's kappa values

Usage

```
kappa_matrix(ConfMatrix, acc = 2)
```

Arguments

ConfMatrix	A confusion matrix made using confusion_matrix(model1, model2)
acc	The accuracy of the results, defaults to 2 decimal places

Value

Unweighted and weighted Kappa value computed using the cohen.kappa() function from the psych R package

Examples

```
data(bmi_long, package="LCTMtools")
require(lcmm)
require(psych)
model1 <- hlme(BMI ~Age, mixture= ~Age, random= ~Age, nwg=T, ng=2, subject="ID", data=bmi_long[1:500, ])
model2 <- hlme(BMI ~Age, mixture= ~Age, random= ~1, nwg=F, ng=2, subject="ID", data=bmi_long[1:500, ])
ConfMatrix <- confusion_matrix(model1, model2)
kappa_matrix(ConfMatrix)
```

LCTMcompare

A model comparison toolkit. LCTMcompare

Description

The function LCTMcompare gives a summary of comparison between fitted LCTM models.

Usage

```
LCTMcompare(modelA, modelB)
```

Arguments

model	is the output from hlme() R model or model is the output of SASmodelbuilder(oe, os, op, of) passed through it
-------	---

Value

A selection of model adequacy tests, including the APPA (average posterior probability of assignment), the OCC (odds of correct classification), entropy \$E\$, Relative entropy (\$E_k\$),

Examples

```
data(bmi_long, package="LCTMtools")
require(lcmm)
modelA <- hlme(BMI ~Age, mixture= ~Age, random= ~Age,
nwg=TRUE, ng=2, subject="ID", data=bmi_long[1:500, ])
postprob(model2class)
modelB <- hlme(BMI ~Age, mixture= ~1, random= ~1,
nwg=FALSE, ng=2, subject="ID", data=bmi_long[1:500, ])
postprob(modelA, modelB)
LCTMcompare(modelA, modelB)
```

LCTMtoolkit	<i>A toolkit which computes a selection of model adequacy tests</i>
LCTMtoolkit	

Description

The function LCTMtoolkit computes a selection of model adequacy tests, including the APPA (average posterior probability of assignment), the OCC (odds of correct classification), entropy E, Relative entropy (E_k), odds of correct classification is the ratio of the odds of classification based on the maximum posterior probability classification rule and the estimated class membership proportions (pi_k)

Usage

```
LCTMtoolkit(model)
```

Arguments

model	is the output from hlme() R model or model is the output of SASmodelbuilder(oe, os, op, of) passed through it
-------	---

Value

A selection of model adequacy tests, including the APPA (average posterior probability of assignment), the OCC (odds of correct classification), entropy \$E\$, Relative entropy (\$E_k\$),

Examples

```
data(bmi_long, package="LCTMtools")
require(lcmm)
model2class <- hlme(BMI ~Age, mixture= ~Age, random= ~Age,
  nwg=TRUE, ng=2, subject="ID", data=bmi_long[1:500, ])
postprob(model2class)
LCTMtoolkit(model2class)
```

LCTMtools	<i>LCTMtools: A package for computing a number of Latent Class Trajectory Model tools for a given hlme() object or SAS model.</i>
-----------	---

Description

The LCTMtools package provides two categories of important functions: LCTMtools (to test a models adequacy) and LCTMcompare (to aid model selection).

LCTMtools functions

The LCTMtools functions arw a selection of model adequacy tests for Latent Class Trajectory Models (LCTMs) which include the APPA (average posterior probability of assignment), the OCC (odds of correct classification), entropy, Relative entropy.

mismatch	<i>Computes the mismatch of the posterior probabilities (mismatch=actual-estimated)</i>
----------	---

Description

Computes the mismatch of the posterior probabilities (mismatch=actual-estimated)

Usage

```
mismatch(p, pi)
```

Arguments

p	is the posterior probabilities of assignment of dimensions, K columns and N rows
pi	is the estimated proportion of class membership of length K

Value

The mismatch of posterior probabilities

Examples

```
## Not run: mismatch(p, pi)
```

mmlcr_to_lctm	<i>Converts an R mmlcr model output to the format of R's hlme class</i>
---------------	---

Description

Converts an R mmlcr model output to the format of R's hlme class

Usage

```
mmlcr_to_lctm(model)
```

Arguments

model	contains model parameter estimates and maximised likelihood, AIC, BIC values
-------	--

Value

A format to feed into the LCTMtoolkit() R function

Examples

```
## Not run: mmlcr_to_lctm(model)
```


occ

The odds of correct classification is the ratio of the odds of classification based on the maximum posterior probability classification rule and the estimated class membership proportions (π_k).

Description

The odds of correct classification is the ratio of the odds of classification based on the maximum posterior probability classification rule and the estimated class membership proportions (π_k).

Usage

```
occ(p, pi)
```

Arguments

p is the posterior probabilities of assignment of dimensions, K columns and N rows

pi is the estimated proportion of class membership of length K

Value

The odds of correct classification

Examples

```
## Not run: occ(p, pi)
```

relative_entropy

The Relative Entropy

Description

The Relative Entropy

Usage

```
relative_entropy(p)
```

Arguments

p is the posterior probabilities of assignment of dimensions, K columns and N rows

Value

Relative Entropy - where values close to 1 indicate lowest classification uncertainty. In the special case when there is most uncertainty and each individual has equal probability of belonging to each class, $E_K=0$. Jedidi et al., describes relative entropy as a relative measure of ‘fuzziness’, and suggested cause concern when close to zero, as this implies that the latent class centroids are not sufficiently separated.

Examples

```
## Not run: relative_entropy(p)
```

sastraj_to_lctm

Converts a SAS proc traj model to the format of R's hlme class

Description

Converts a SAS proc traj model to the format of R's hlme class

Usage

```
sastraj_to_lctm(oe, of, op, os)
```

Arguments

oe	contains model parameter estimates and maximised likelihood, AIC, BIC values
of	contains posterior probabilities
op	contains predictors
os	contains fixed effect and class membership parameter estimates

Value

A format to feed into the LCTMtoolkit() R function

Examples

```
## Not run: sastraj_to_lctm(oe, of, op, os)
```

Index

*Topic **datasets**

bmi, [3](#)

bmi_long, [3](#)

actual_proportions, [2](#)

appa, [2](#)

bmi, [3](#)

bmi_long, [3](#)

class_assignment, [4](#)

confusion_matrix, [4](#)

entropy, [5](#)

kappa_matrix, [5](#)

LCTMcompare, [6](#)

LCTMtoolkit, [7](#)

LCTMtools, [7](#)

LCTMtools-package (LCTMtools), [7](#)

mismatch, [8](#)

mmlcr_to_lctm, [8](#)

occ, [9](#)

relative_entropy, [9](#)

sastraj_to_lctm, [10](#)