

Package ‘LKT’

May 25, 2023

Title Logistic Knowledge Tracing

Version 1.5.0

Description Computes Logistic Knowledge Tracing ('LKT') which is a general method for tracking human learning in an educational software system. Please see Pavlik, Eglington, and Harrel-Williams (2021) <<https://ieeexplore.ieee.org/document/9616435>>. 'LKT' is a method to compute features of student data that are used as predictors of subsequent performance. 'LKT' allows great flexibility in the choice of predictive components and features computed for these predictive components. The system is built on top of 'LiblineaR', which enables extremely fast solutions compared to base glm() in R.

License GPL-3

Encoding UTF-8

LazyData true

VignetteBuilder knitr

RoxygenNote 7.2.3

Depends R (>= 3.5.0), SparseM (>= 1.78), methods, Matrix, data.table (>= 1.13.2), LiblineaR (>= 2.10-8)

Imports glmnet (>= 4.0-2), glmnetUtils (>= 1.1.8), lme4 (>= 1.1-23), cluster (>= 2.1.3), pROC (>= 1.16.2), crayon, HDInterval (>= 0.2.2)

Suggests rmarkdown, knitr, utils, caret, ggplot2

NeedsCompilation no

Author Philip I. Pavlik Jr. [aut, ctb, cre]
(<<https://orcid.org/0000-0001-6467-9452>>),
Luke G. Eglington [aut, ctb] (<<https://orcid.org/0000-0002-8432-9203>>)

Maintainer Philip I. Pavlik Jr. <imrryr@gmail.com>

Repository CRAN

Date/Publication 2023-05-25 07:20:14 UTC

R topics documented:

buildLKTModel	2
computefeatures	3
computeSpacingPredictors	4
countOutcomeold	5
largerawsample	5
LASSOLKTData	6
LASSOLKTModel	7
LKT	8
LKT_HDI	9
samplelkt	10
smallSet	11
ViewExcel	11
Index	12

buildLKTModel	<i>buildLKTModel</i>
---------------	----------------------

Description

Forward and backwards stepwise search for a set of features and components with tracking of nonlinear parameters.

Usage

```
buildLKTModel(
  data,
  allcomponents,
  allfeatures,
  currentcomponents = c(),
  specialcomponents = c(),
  specialfeatures = c(),
  forv,
  bacv,
  preset = NA,
  presetint = T,
  currentfeatures = c(),
  verbose = FALSE,
  traceCV = TRUE,
  currentfixedpars = c(),
  maxitv = 10,
  interc = FALSE,
  forward = TRUE,
  backward = TRUE,
  metric = "BIC"
)
```

Arguments

- data is a dataset with Anon.Student.Id and CF..ansbin.
- allcomponents is search space for LKT components
- allfeatures is search space for LKT features
- currentcomponents components to start search from
- specialcomponents add special components (not crossed with features, only paired with special features 1 for 1)
- specialfeatures features for each special component (not crossed during search)
- forv the minimum amount of improvement needed for the addition of a new term
- bacv the maximum amount of loss for a term to be removed
- preset One of "static", "AFM", "PFA", "advanced", "AFMLLTM", "PFALLTM", "advancedLLTM"
- presetint should the intercepts be included for preset components
- currentfeatures features to start search from
- verbose passed to LKT
- traceCV produce a CV from the LKT method at the beginning of each cycle
- currentfixedpars used for current features as an option to start
- maxitv passed to LKT
- interc passed to LKT
- forward TRUE or FALSE
- backward TRUE or FALSE
- metric One of "BIC", "AUC", "AIC", and "RMSE"

Value

list of values "tracetable" and "currentfit"

computefeatures *computefeatures*

Description

Compute feature describing prior practice effect.

Usage

computefeatures(data, feat, par1, par2, index, index2, par3, par4, par5, fcomp)

Arguments

data	copy of main data frame.
feat	is the feature to be computed.
par1	nonlinear parameters used for nonlinear features.
par2	nonlinear parameters used for nonlinear features.
index	a student by component levels index
index2	a component levels index
par3	nonlinear parameters used for nonlinear features.
par4	nonlinear parameters used for nonlinear features.
par5	nonlinear parameters used for nonlinear features.
fcomp	the component name.

Value

a vector suitable for regression input.

computeSpacingPredictors
computeSpacingPredictors

Description

Compute repetition spacing time based features from input data CF.Time. and/or CF.reltime. which will be automatically computed from Duration.sec. if not present themselves.

Usage

```
computeSpacingPredictors(data, KCs)
```

Arguments

data	is a dataset with Anon.Student.Id and CF.ansbin.
KCs	are the components for which spaced features will be specified in LKT

Value

data which is the same frame with the added spacing relevant columns.

countOutcomeold	<i>countOutcome</i>
-----------------	---------------------

Description

Compute the prior sum of the response appearing in the outcome column for the index

Usage

```
countOutcomeold(data, index, response)
```

Arguments

data	the dataset to compute an outcome vector for
index	the subsets to count over
response	the actually response value being counted

Value

the vector of the lagged cumulative sum.

largerawsample	<i>Trial sequences for practice participants.</i>
----------------	---

Description

A dataset containing a raw sample from the Memphis Datashop.

Usage

```
largerawsample
```

Format

A data frame please see the DataShop for more info.
It has many columns.

Source

<https://psl1cdatashop.web.cmu.edu/Export?datasetId=5513>

LASSOLKTData

LASSOLKTData

Description

Forward and backwards stepwise search for a set of features and components with tracking of nonlinear parameters.

Usage

```
LASSOLKTData(
  data,
  gridpars,
  allcomponents,
  allfeatures,
  preset = NA,
  presetint = T,
  specialcomponents = c(),
  specialfeatures = c(),
  specialpars = c()
)
```

Arguments

<code>data</code>	is a dataset with Anon.Student.Id and CF.ansbin.
<code>gridpars</code>	a vector of parameters to create each feature at
<code>allcomponents</code>	is search space for LKT components
<code>allfeatures</code>	is search space for LKT features
<code>preset</code>	One of "static", "AFM", "PFA", "advanced", "AFMLLTM", "PFALLTM", "advancedLLTM"
<code>presetint</code>	should the intercepts be included for preset components
<code>specialcomponents</code>	add special components (not crossed with features, only paired with special features 1 for 1)
<code>specialfeatures</code>	features for each special component (not crossed during search)
<code>specialpars</code>	parameters for the special features (if needed)

Value

data which is the same frame with the added spacing relevant columns.
list of values "tracetable" and "currentfit"

LASSOLKTModel	<i>LASSOLKTModel</i>
---------------	----------------------

Description

runs LASSO search on the data

Usage

```
LASSOLKTModel(
  data,
  gridpars,
  allcomponents,
  preset = NA,
  presetint = T,
  allfeatures,
  specialcomponents = c(),
  specialfeatures = c(),
  specialpars = c(),
  target_n
)
```

Arguments

data	is a dataset with Anon.Student.Id and CF.ansbin.
gridpars	a vector of parameters to create each feature at
allcomponents	is search space for LKT components
preset	One of "static", "AFM", "PFA", "advanced", "AFMLLTM", "PFALLTM", "advancedLLTM"
presetint	should the intercepts be included for preset components
allfeatures	is search space for LKT features
specialcomponents	add special components (not crossed with features, only paired with special features 1 for 1)
specialfeatures	features for each special component (not crossed during search)
specialpars	parameters for the special features (if needed)
target_n	chosen number of features in model

Value

list of values "dropped 1se", "retained 1se", "target features", "target dropped", "target pseudo R2", "best pseudo R2", "target mod rmse", "target mod auc", and "target_mod_bic"

LKT

*LKT***Description**

Compute a logistic regression model of learning for input data.

Usage

```
LKT(
  data,
  components,
  features,
  fixedpars = NA,
  seedpars = NA,
  interacts = NA,
  curvefeats = NA,
  dualfit = FALSE,
  interc = FALSE,
  cv = FALSE,
  verbose = TRUE,
  epsilon = 1e-04,
  cost = 512,
  lowb = 1e-05,
  highb = 0.99999,
  type = 0,
  maketimes = FALSE,
  bias = 0,
  maxitv = 100,
  factrv = 1e+12,
  nosolve = FALSE,
  autoKC = rep(0, length(components)),
  autoKCcont = rep("NA", length(components)),
  connectors = rep("+", max(1, length(components) - 1))
)
```

Arguments

<code>data</code>	A dataset with <code>Anon.Student.Id</code> and <code>CF.ansbin</code> .
<code>components</code>	A vector of factors that can be used to compute each features for each subject.
<code>features</code>	a vector methods to use to compute a feature for the component.
<code>fixedpars</code>	a vector of parameters for all features+components.
<code>seedpars</code>	a vector of parameters for all features+components to seed non-linear parameter search.
<code>interacts</code>	A list of components that interacts with component by feature in the main specification.

curvefeats	vector of columns to use with "diff" functions
dualfit	TRUE or FALSE, fit a simple latency using logit. Requires Duration..sec. column in data.
interc	TRUE or FALSE, include a global intercept.
cv	TRUE or FALSE, if TRUE runs N-fold cv. Requires premade column named 'fold' with integers denoting the N folds
verbose	provides more output in some cases.
epsilon	passed to LiblinearR
cost	passed to LiblinearR
lowb	lower bound for non-linear optimizations
highb	upper bound for non-linear optimizations
type	passed to LiblinearR
maketimes	Boolean indicating whether to create time based features (or may be precomputed)
bias	passed to LiblinearR
maxitv	passed to nonlinear optimization a maxit control
factrv	controls the optim() function
nosolve	causes the function to return a sparse data matrix of the features, rather than a solution
autoKC	a vector to indicate whether to use autoKC for the component (0) or the k for the numebr of clusters
autoKCcont	a vector of text strings set to "rand" for component to make autoKC assignment to cluster is randomized (for comaprison)
connectors	a vector if linear equation R operators including +, * and :

Value

list of values "model", "coefs", "r2", "prediction", "nullmodel", "latencymodel", "optimizedpars", "subjectrmse", "newdata", and "automat"

LKT_HDI

LKT_HDI

Description

Bootstrap credibility intervals to aid in interpreting coefficients.

Usage

```
LKT_HDI(  
  dat,  
  n_boot,  
  n_students,  
  components,  
  features,  
  interacts,  
  fixedpars,  
  get_hdi = TRUE,  
  cred_mass = 0.95  
)
```

Arguments

dat	Dataframe
n_boot	Number of subsamples to fit
n_students	Number of students per subsample
components	components in model
features	features in model
interacts	interacts in model
fixedpars	fixed pars in model
get_hdi	boolean to decide if generating HDI per coefficient
cred_mass	credibility mass parameter to decide width of HDI

Value

list of values "par_reps", "mod_full", "coef_hdi"

samplelkt

Trial sequences for practice participants.

Description

A dataset containing a small sample of participants in a memory experiment.

Usage

samplelkt

Format

A data frame with 2074 rows and many variables:

Anon.Student.Id unique identifier for each student

Duration..sec. unique identifier for each student

KC..Default. unique identifier for each student

Outcome unique identifier for each student ...

Source

<https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=5508>

smallSet	<i>smallSet</i>
----------	-----------------

Description

smallSet

Usage

```
smallSet(data, nSub)
```

Arguments

data	Dataframe of student data
nSub	Number of students

ViewExcel	<i>ViewExcel</i>
-----------	------------------

Description

ViewExcel

Usage

```
ViewExcel(df = .Last.value, file = tempfile(fileext = ".csv"))
```

Arguments

df	Dataframe
file	name of the Excel file

Index

* datasets

- largerawsample, [5](#)
- samplelkt, [10](#)

buildLKTModel, [2](#)

computefeatures, [3](#)

computeSpacingPredictors, [4](#)

countOutcomeold, [5](#)

largerawsample, [5](#)

LASSOLKTData, [6](#)

LASSOLKTModel, [7](#)

LKT, [8](#)

LKT_HDI, [9](#)

samplelkt, [10](#)

smallSet, [11](#)

ViewExcel, [11](#)