Package ‘slgf’

February 17, 2021

Type Package

Title Bayesian Model Selection with Suspected Latent Grouping Factors

Version 0.1.0

Date 2021-02-15

Author Thomas A. Metzger and Christopher T. Franck

Maintainer Thomas A. Metzger <metzger.181@osu.edu>

Description Implements the Bayesian model selection method with suspected latent grouping factor methodology of Metzger and Franck (2020), <doi:10.1080/00401706.2020.1739561>. SLGF detects latent heteroscedasticity or group-based regression effects based on the levels of a user-specified categorical predictor. We encourage you to review examples in vignette(``slgf_vignette'', ``slgf'').

License GPL (>= 2)

Encoding UTF-8

Imports Rdpack, numDeriv, utils

RdMacros Rdpack

LazyData true

Depends R (>= 3.5.0)

Suggests knitr, captioner, formatR, rcrossref, rmarkdown

VignetteBuilder knitr

RoxygenNote 7.1.1

NeedsCompilation no

Repository CRAN

Date/Publication 2021-02-17 18:50:02 UTC

R topics documented:

`chips` *Chips data on breaking strength by starch type and chip thickness*

Description

Flurry (1939) analyzes the breaking strength of a starch chip as a function of the chip's thickness (measured in 10^-4 inches) and the type of plant from which the starch was derived (corn, canna, or potato).

Usage

```
data(chips)
```

Format

A data frame with 49 rows and 3 variables:

strength the response, the breaking strength.

film the chip's film thickness, measured in 10^-4 inches.

starch the chip's starch component: canna, corn, or potato

References

Flurry MS (1939). "Breaking strength, elongation and folding endurance of films of starches and gelatin used in sizing." *Technical Bulletin (United States Department of Agriculture)*, **674**, 1–36.

`column_centerer` *Column centerer for a design matrix.*

Description

`column_centerer` Centers the columns of a matrix by the column mean.

Usage

```
column_centerer(mm)
```

Arguments

`mm`	a model matrix.

Value

`column_centerer` centers the columns of a design matrix by each column's mean.

Examples

```
set.seed(314159)
test.data <- data.frame("y"=c(rnorm(10,0,1), rnorm(10,3,1), rnorm(10,5,3)),
                        "x1"=c(rep("A",10), rep("B",10), rep("C",10)),
                        "x2"=rnorm(30,0,1))
m <- lm(y~x1+x2, data=test.data)
mm <- model.matrix(m)
column_centerer(mm)
```

`extract.hats` *Obtain the concentrated maximum likelihood estimators from a specified model.*

Description

`extract.hats` Returns the concentrated maximum likelihood estimators from a specified model.

Usage

```
extract.hats(slgf.obj, model.index = NULL)
```

Arguments

`slgf.obj`	output from `ms.slgf`
`model.index`	the model index of the model for which estimates are desired.

Value

`extract.hats` returns a list with the following elements:
1) `model`, the model desired
2) `scheme`, the scheme associated with the model desired
3) `coef`, the regression coefficients associated with the model desired
4) `sigsq`, the error variance(s) assicoated with the model desired
5) `g`, the g estimate, if `prior="zs"`

Examples

```
# Obtain the concentrated maximum likelihood estimates
# for the second-most probable model.

library(numDeriv)

set.seed(314159)
test.data <- data.frame("y"=c(rnorm(10,0,1), rnorm(10,3,1), rnorm(10,5,3)),
                        "x"=c(rep("A",10), rep("B",10), rep("C",10)))
test.models <- list("y~1", "y~x", "y~group")
test.models
test.out <- ms.slgf(dataf=test.data, response="y", lgf="x",
                    usermodels=test.models,
                    prior="flat", het=c(1,1,1), min.levels=1)
extract.hats(test.out, 2)
```

`groupings` *Groupings finder for two-way layouts.*

Description

`groupings` Computes the possible grouping schemes for a two-way layout with r rows and c columns.

Usage

```
groupings(data_matrix)
```

Arguments

`data_matrix`	an r by c data matrix.

Value

`groupings` returns the unique possible row-wise groupings of the input two-way layout.

Examples

```
# Determine the possible row-wise groupings for an 8 by 5 matrix.
groupings(matrix(NA, nrow=8, ncol=4))
```

`labeler` *Labels observations according to group membership.*

Description

`labeler` Returns a Boolean indicator for each observation's group membership for a two-way layout.

Usage

```
labeler(nrows, ncols, combo.iteration)

labeler(nrows, ncols, combo.iteration)
```

Arguments

`nrows`	the number of rows in the data matrix.
`ncols`	the number of columns in the data matrix.
`combo.iteration`	the index of the grouping scheme under consideration.

Value

`labeler` returns a vector of 1s and 0s corresponding to the input vector's group membership by index.

`locknut` *Locknut data on torque required to tighten a fixture by plating method*

Description

Meek and Ozgur (1991) analyzes the torque required to strengthen a fixture (bolt or mandrel) as a function of the fixture's plating method (cadmium and wax, heat treating, and phosphate and oil, denoted CW, HT, and PO, respectively).

Usage

```
data(locknut)
```

Format

A data frame with 60 rows and 3 variables:

Torque the response, the torque required to tighten the fixture.

Fixture the type of fixture, bolt or mandrel.

Plating the plating treatment, CW, HT, or PO.

References

Meek GE, Ozgur CO (1991). "Torque Variation Analysis." *Journal of the Industrial Mathematics Society*, **41**, 1–16.

`lymphoma` *Lymphoma data on genomic hybridizaiton signal from six dogs with normal and tumor tissue samples taken.*

Description

Franck et. al. (2013) analyzes the genomic hybridization signal measured from normal and tumor tissue samples taken from six dogs.

Usage

```
data(lymphoma)
```

Format

A data frame with 6 rows and 2 variables:

V1 the signals from the normal tissue samples.

V2 the signals from the tumor tissue samples.

References

Franck CT, Nielsen DM, Osborne JA (2013). "A method for detecting hidden additivity in two-factor unreplicated experiments." *Computational Statistics \& Data Analysis*, **67**(Supplement C), 95–104. ISSN 0167-9473, doi: 10.1016/j.csda.2013.05.002, `https://doi.org/https://doi.org/10.1016/j.csda.2013.05.002`.

maketall *Converts a two-way layout into tall format with row and column index labels.*

Description

maketall Converts a two-way layout into tall format with row and column index labels.

Usage

```
maketall(data_matrix)
```

Arguments

data_matrix	an r by c data matrix.

Value

maketall returns a data frame containing the original observations, row labels, and column labels.

Examples

```
library(slgf)
maketall(lymphoma)
```

ms.slgf *Bayesian Model Selection with Latent Group-Based Regression Effects and Heteroscedasticity*

Description

ms.slgf Implements the model selection method proposed by (Metzger and Franck 2019).

Usage

```
ms.slgf(
  dataf,
  response,
  lgf = NA,
  usermodels,
  prior = "flat",
  het = rep(0, length(usermodels)),
  min.levels = 1
)
```

Arguments

`dataf` A data frame containing a continuous response, at least one categorical predictor, and any other covariates of interest. This data frame should not contain column names with the character string `group`.

`response` A character string indicating the column of `dataf` that contains the response.

`lgf` A character string indicating the column of ‘dataf‘ that contains the suspected latent grouping factor (SLGF).

`usermodels` A list of length `M` where each element contains a string of *R* class `formula` or `character` indicating the models to consider. The term `group` should be used to replace the name of the slgf in models with group-based regression effects. This list must contain at least one model with group-based regression effects.

`prior` A character string `"flat"` or `"zs"` indicating whether to implement the flat or Zellner-Siow mixture g-prior on regression effects, respectively. Defaults to `"flat"`.

`het` A vector of 0s and 1s of length `M`. If the mth element of `het` is 0, then the mth model of `usermodels` is considered in a homoscedastic context only; if the mth element of `het` is 1, the mth model of `usermodels` is considered in both homoscedastic and heteroscedastic contexts.

`min.levels` A numeric value indicating the minimum number of levels of the SLGF that can comprise a group. Defaults to 1.

Value

`ms.slgf` returns a list of six elements:
1) `results`, an `M` by 11 matrix where columns contain the model selection results and information for each model, including:
- `Model`, the formula associated with each model;
- `Scheme`, the grouping scheme associated with each model;
- `Variance`, a label of whether each model is homoscedastic or heteroscedastic;
- `logFlik`, the fractional log-likelihood associated with each model;
- `Mod.Prior`, the prior assigned to each model;
- `Fmodprob`, the fractional posterior probability associated with each model;
- `Cumulative`, the cumulative fractional posterior probability associated with a given model and the previous models;
- `dataf.Index`, an index indicating which element of `group.datafs` contains the corresponding group `dataframe`;
- `mle.index`, an index indicating which element of `coefficients`, `variances`, and `gs` contains the corresponding estimates;
- `Model.Index`, an index indicating where the model ranks in its posterior model probability;
- `Class`, a label of the model with its group variance specification;

2) `group.datafs`, a list containing dataframes associated with each model class containing the appropriate effects, including group effects;
3) `scheme.Probs`, a `data.frame` containing the total posterior probability for each grouping scheme considered;
4) `class.Probs`, a `data.frame` containing the total posterior probability for each model class considered;

5) coefficients, MLEs for each model's regression effects;
6) variances, MLEs based on concentrated likelihood for each model's variance(s);
7) gs, MLEs based on concentrated likelihood for each model's g; only included if prior="zs".

Author(s)

Thomas A. Metzger and Christopher T. Franck

References

Metzger TA, Franck CT (2019). "Detection of latent heteroscedasticity and group-based regression effects in linear models via Bayesian model selection." *arXiv e-prints*.

Examples

```
# Fit a a heteroscedastic ANOVA example with distinct means by level of the LGF.

library(numDeriv)

set.seed(314159)
test.data <- data.frame("y"=c(rnorm(10,0,1), rnorm(10,3,1), rnorm(10,5,3)),
                        "x"=c(rep("A",10), rep("B",10), rep("C",10)))
test.models <- list("y~1", "y~x", "y~group")
test.out <- ms.slgf(dataf=test.data, response="y", lgf="x",
                    usermodels=test.models,
                    prior="flat", het=c(1,1,1), min.levels=1)
test.out$results[1:3,c(1:4,6,7)]
```

roadwear *Roadwear data on four tires, each comprising three compounds, from a balanced incompleted block design.*

Description

Davies (1954) analyzes the wear on four tires, where each tire comprises three distinct compounds.

Usage

```
data(roadwear)
```

Format

A data frame with 12 rows and 3 variables:

abrasion the measurement of abrasion.

compound the compound from which each measurement was taken, either A, B, C, or D.

tire the tire from which each measurement was taken, either 1, 2, 3, or 4.

References

Davies OL (1954). *The Design and Analysis of Industrial Experiments*. Oliver \& Boyd, London.

`smell` *Smell data on olfactory function by age group*

Description

O'Brien and Heft (1995) studied the University of Pennsylvania Smell Identification Test (UPSIT). 180 subjects of different age groups were asked to describe 40 different odors. Olfactory index was quantified by the Freeman-Tukey modified arcsine transformation on the proportion of correctly identified odors. Subjects were divided into five age groups: group 1 if age 2 or younger; group 2 if between ages 26 and 40; group 3 if between ages 41 and 55; group 4 if between ages 56 and 70; and group 5 if older than 75.

Usage

```
data(smell)
```

Format

A data frame with 180 rows and 2 variables:

agecat age category, from 1 to 5.

olf olfactory function, measured as the Freeman-Tukey modified arcsine transformation on the proportion of correctly identified odors.

Source

SAS/STAT 15.2 User's Guide

References

OBrien RG, Heft MW (1995). "New Discrimination Indexes and Models for Studying Sensory Functioning in Aging." *Journal of Applied Statistics*, **22**, 9–27.

Index