Package 'swissMrP'

May 30, 2018

Type Package

Title Multilevel Regression with Post-Stratification (MrP) for Switzerland

Version 0.62

Date 2018-05-25

Author Lucas Leemann

Maintainer Lucas Leemann <lleemann@gmail.com>

Depends R (>= 3.4.0), arm, lme4, maptools, blme, sp, utils

Description

Provides a number of useful functions to employ MrP for small area prediction in Switzerland. Based on a hierarchical model and survey data one can derive cantonal preference measures. The package allows to automatize the prediction and post-stratification steps. It further provides adequate print, summary, map, and plot functions for objects of its class.

License GPL-2

LazyData true

LazyLoad true

NeedsCompilation no

Repository CRAN

Date/Publication 2018-05-30 12:35:42 UTC

R topics documented:

swissMrP-package *Multilevel Regression and Post-Stratification (MrP) for Switzerland*

Description

This package provides a number of useful functions to employ MrP for small area prediction in Switzerland. Based on a hierarchical model (based itself on survey data) one can derive cantonal preference measures. The package allows to automatize the prediction and post-stratification steps. It further provides adequate print, summary, map, and plot functions for objects of its class.

Details

Package:	swissMrP
Type:	Package
Version:	0.1
Date:	2018-26-05
License:	GLP-2

Note

This package was originally written for students of the University of Zurich (Fall 2013). The functions and data make this package a useful resource for anybody implementing MrP on Swiss data.

Author(s)

Lucas Leemann <lleemann@gmail.com>

References

Gelman, Andrew and Thomas C. Little. 1997. Poststratification Into Many Categories Using Hierarchical Logistic Regression. *Survey Research* 23:127-135.

Jeffrey Lax and Justin Phillips. 2009. How Should We Estimate Public Opinion in The States? *American Journal of Political Science* 53 (1), 107-121.

Leemann, Lucas and Fabio Wasserfallen. 2016. The Democratic Effect of Direct Democracy. *American Political Science Review* 110(4): 750-762.

census *Census File, Switzerland 2000*

Description

The file contains the cenus information which was retrieved from the BfS (Federal Statistical Office). It consists of three variables; age (4 groups), gender, and education (6 groups) - see `details` for more information.

Usage

```
data(census)
```

Format

The format is: List of 2 Elements

$ census48.MAZH2013: Matrix 48*26 (48 ideal types, 6*4*2) and 26 cantons. The 26 columns represent the 26 cantons, the 48 rows represent all possible combinations.

group	educ	age	gender
row 1:	1	1	1
row 2:	1	1	2
row 3:	1	2	1
row 4:	1	2	2
row 5:	1	3	1
row 6:	1	3	2
row 7:	1	4	1
row 8:	1	4	2
row 9:	2	1	1
....			
row 46:	6	3	2
row 47:	6	4	1
row 48:	6	4	2

$ INFO : see `census$INFO`

Details

Education: (1 - if only mandatory school or NA), (2 - Apprenticeship), (3 - High School Diploma or teachers' school), (4 - Higher education, germ: hoehere Fachausbildung), (5 - higher education 2, germ: hoehere Fachschule), (6 - Federal technical Institutes or University of apllied sciences), (7 - University). Age (1: 0-34, 2: 35-49, 3: 50-64, 4: 64-). Woman (0 man, 1 woman).

Source

BfS. Bundesamt fuer Statistik.

References

Bundesamt fuer Statistik (BfS). 2000. Eidgenoessische Volkszaehlung 2000; Abschlussbericht zur Volkszaehlung.

Examples

```
# How many young men with no higher education than mandatory schooling live in
# the cantons  ZH, BE, and LU?
data(census)
census$census48.MAZH2013[1:3,1] # 1st column is ZH
```

`map.MrP` *Map for MrP Estimates*

Description

This is a simple wrapper that reads in shape files and creates an ordinary map. The displayed map will show the intensity of support for each canton.

Usage

```
map.MrP(x, colors.m, threshold, main, labels = TRUE, legend.text1, legend.text2, ...)
```

Arguments

`x`	The output object from `swissMrP` whether it is a vector or a matrix (the latter if simulations were ran in `swissMrP`).
`colors.m`	Vector of colors to be used in increasing order. If the user does not supply any colors, the plot will use a default option (from red to green).
`threshold`	Vector. By default the map breaks support in ten equal-sized intervals. If one supplies colors, the support will be broken into `length(colors.m)` equal intervals. If the user wishes to change the intervals this is done by supplying a vector with interval boundaries starting with 0. For *n* intervals one needs *n+1* bondaries.
`main`	User may specify a title for the plot.
`labels`	Logical. User may specify whether labels should be written automatically or if they should be omitted.
`legend.text1`	User may specify a legend, e.g. *Support Smoking Ban* to declare what the colors indicate.
`legend.text2`	Like `legend.text1`, offers a second line.
`...`	additional arguments to be passed to the low level plotting functions.

Note

The shape files for this command are from the Swiss government (Swiss Federal Statistics Office, 2013.). While this function only needs the cantonal boundaries one can also download the shape files for municipality boundaries at the above mentioned URL. The actual mapping is done with the plot command based on elements from the `maptools` package.

Author(s)

Lucas Leemann

See Also

`zip1` `swissMrP`

Examples

```
# Vanilla example
fake.pref <- runif(26)
class(fake.pref) <- "swissMrP"

map.MrP(fake.pref, main="This Map Shows Random Data",
        legend.text1="Support for Anything")

## changing intervals

map.MrP(fake.pref,
        threshold=c(0,0.3,0.45,0.48,0.49,0.5,0.51,0.52,0.55,0.7,1))

## no labels

map.MrP(fake.pref, main="This Map Shows Random Data",
        legend.text1="Support for Anything", labels=FALSE)

## specify different colors and less groups

map.MrP(fake.pref, main="This Map Shows Random Data",
        legend.text1="Support for Anything",
        colors.m=c("skyblue","skyblue1","skyblue2","skyblue3",
        "slateblue","slateblue1","slateblue2"))
```

`plot` *Plots* `swissMrP` *Objects*

Description

This function plots the results of `swissMrP`.

Usage

```
## S3 method for class 'swissMrP'
plot(x, conf.int = 0.95, col, design.dot = TRUE, lab, ...)
```

Arguments

x	Object returned from swissMrP().
conf.int	Confidence interval that should be displayed. Only meaningful if x is a matrix with dimensions 26*Number.sim.
col	Color for plot.
design.dot	Logical. If design.dot=TRUE the plot will display dots. If design.dot=FALSE the plot will display the cantonal abbreviations (ZH, BE, LU, ...).
lab	Label for x-axis. If there is no user specified label the default is Degree of Estimated Support.
...	Further graphical arguments can be used, e.g. main to specify a title.

Details

The design, when design.dot=FALSE, was added for Fabio Wasserfallen.

Author(s)

Lucas Leemann, <lleemann@gmail.com>

See Also

swissMrP, map.MrP

Examples

```
# fake data example
vec <- pnorm(rnorm(26))
class(vec) <- "swissMrP"
plot(vec, design.dot=FALSE, main="Simulated Data",
             lab="Heterogeneous Preferences", col="green")

plot(vec, design.dot=TRUE, main="Simulated Data",
             lab="Heterogeneous Preferences", col="blue")
```

print *Print Function for* swissMrP *Objects*

Description

Prints.

Usage

```
## S3 method for class 'swissMrP'
print(x, ...)
```

Arguments

x	An object of class swissMrP.
...	additional arguments to be passed to print function.

Author(s)

Lucas Leemann

See Also

swissMrP

Examples

```
## fake data
fake.est <- runif(26)
class(fake.est) <- "swissMrP"
print(fake.est)
```

summary *Summary Function for* swissMrP *Objects*

Description

Provides the lowest, the two median, and the highest estimates of public support based on the MrP results.

Usage

```
## S3 method for class 'swissMrP'
summary(object,...)
```

Arguments

object	An object generated by swissMrP.
...	additional arguments to be passed to summary function.

Author(s)

Lucas Leemann <lleemann@gmail.com>

See Also

swissMrP

Examples

```
## fake data example
fake <- runif(26)
class(fake) <- "swissMrP"
summary(fake)
```

---

swissMrP *Multilevel Regression with Poststratification for the 26 Swiss Cantons*

Description

This function provides cantonal estimates of public support based on the MrP procedure. Specifically: this function caries out the third (prediction) and forth step (post-stratification) of MrP.

Usage

```
swissMrP(response.model, augment.data = NA, augment.row = 0, uncertainty = FALSE,
         Number.sim = 1000, region)
```

Arguments

response.model	The output object of glmer or blmer where one estimates a hierarchical model with random effects for: age, education, gender, canton, and region. One can include any level 2 predictors.
augment.data	In case one uses survey data for the generation of response.model which does not have any respondents from one or more cantons there is a problem. The function does not know the values of the level 2 explanatory variables in the response.model and would hence produce NA's or wrong results. Users can supply the missing information as augment.data. If a canton is missing it will be a vector, if more than one is missing it is a matrix where the number of rows equals the number of missing cantons. For each level 2 variable one adds a column (see example) and a preceeding column of 1's.

`augment.row` Integer, between 1 and 26. Indicates the row or rows where the `augment.data` will be inserted. If data is missing for BE (second canton) and LU (third canton) one will have to supply a vector `c(2,3)`.

`uncertainty` Logical. If `uncertainty=TRUE` the function will run `Number.sim` simulations to obtain a measure of uncertainty for each cantonal prediction.

`Number.sim` Integer. Number of simulations to be run for uncertainty measure. Default is set to 1000.

`region` Vector. The regional grouping is based on the large regions in Switzerland (see Leemann and Wasserfallen, 2013). But for certain questions and models alternative groupings could be beneficial. The user can supply a vector (length 26) with integers from 1 to the maximal number of distict regions whereas the first element indicates the regional group of ZH, the second element implies the regional group of BE, ... etc. (it is the standard order of cantons used by the Federal Statistics Office.)

Value

Creates an object of class `swissMrP`. A list where the first element is a vector of length 26 (an estimate for each of the 26 cantons).

Author(s)

Lucas Leemann <lleemann@gmail.com>

References

Gelman, Andrew and Thomas C. Little. 1997. Poststratification Into Many Categories Using Hierarchical Logistic Regression. *Survey Research* 23:127-135.

Jeffrey Lax and Justin Phillips. 2009. How Should We Estimate Public Opinion in The States? *American Journal of Political Science* 53 (1), 107-121.

Leemann, Lucas and Fabio Wasserfallen. 2016. The Democratic Effect of Direct Democracy. *American Political Science Review* 110(4): 750-762.

Examples

```
library(lme4)
### Fake data
err.ind <- rnorm(1000,sd=4)
woman <- sample(c(0,1),replace=TRUE,size=1000)
age <- sample(c(1:4),replace=TRUE,size=1000)
education <- sample(c(1:6),replace=TRUE,size=1000)
cantonnr <- sample(c(1:26),replace=TRUE,size=1000)
region <- sample(c(1:7),replace=TRUE,size=1000)
x <- cbind(rnorm(26),rnorm(26)); err.con <- rnorm(26,sd=4); X <- matrix(NA,1000,2)
for (q in 1:1000){ X[q,] <- c(x[cantonnr[q]],err.con[cantonnr[q]])}
y.fake <- X[,1] +X[,2] + woman+age+education+cantonnr+region + err.ind
y <- rep(0,length(y.fake))
y[y.fake>mean(y.fake)]<-1
```

```
model1 <- glmer(y ~ X[,1] +X[,2] + (1|woman)  + (1|education) + (1|age) + (1|cantonnr)
                + (1|region), family=binomial(probit))
# use the MrP function
mrp1 <- swissMrP(model1)

## Here is an example if two cantons (10,22; FR & VD) are missing
mrp2 <- swissMrP(model1,
augment.data=matrix(c(c(1,2,-1),c(1,1,-5)),2,3, byrow=TRUE),
augment.row=c(10,22))
```

`swissMrP-internal` *Internal swissMrP Functions*

Description

Internal swissMrP functions

Usage

```
...
```

Details

These are not to be called by the user (or in some cases are just waiting for proper documentation to be written). This is a test.

`zip1` *Shape Files for Swiss Cantons*

Description

The shape files are from the Swiss government (Swiss Federal Statistics Office, 2013).

Usage

```
data(zip1)
```

Format

@ students: shape files are slightly more involved as they are `SpatialPolygonsDataFrame` objects. You should most likely not have to work with the files themselves.

Details

If you are looking at this help file then chance is that something went wrong :-(

Source

Swiss Federal Statistics Office, 2013.

Examples

```
# A map of the 26 Swiss cantons:
plot(zip1)
```

Index