Package 'MNLpred'

September 2, 2020

Title Simulated Predicted Probabilities for Multinomial Logit Models

Version 0.0.4

Depends R (>= 3.5.0)

Description Functions to easily return simulated predicted probabilities and first differences for multinomial logit models. It takes a specified scenario and a multinomial model to predict probabilities with a set of coefficients, drawn from a simulated sampling distribution. The simulated predictions allow for meaningful plots with means and confidence intervals. The methodological approach is based on the principles laid out by King, Tomz, and Wittenberg (2000) <doi:10.2307/2669316> and Hanmer and Ozan Kalkan (2016) <doi:10.1111/j.1540-5907.2012.00602.x>.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Suggests knitr, rmarkdown, testthat, nnet, magrittr, ggplot2, scales

VignetteBuilder knitr

Imports MASS, stats

NeedsCompilation no

Author Manuel Neumann [aut, cre] (<https://orcid.org/0000-0002-7953-3939>)

Maintainer Manuel Neumann <manuel.neumann@mzes.uni-mannheim.de>

Repository CRAN

Date/Publication 2020-09-02 06:50:43 UTC

R topics documented:

`gles` *German Longitudinal Election Study*

Description

A sample of 1,000 respondents in the Rolling Cross Sectional study in the German Longitudinal Election Study in 2017.

Usage

```
data(gles)
```

Format

An data frame with 1,000 observations and 6 variables:

vote Voting decision for party

egoposition_immigration Ego-position toward immigration (0 = very open to 10 = very restrictive)

ostwest Dummy for respondents from Eastern Germany (= 1)

political_interest Measurement for political interst (0 = low, 4 = high)

income Self-reported income satisfaction (0 = low, 4 = high)

gender Self-reported gender (binary coding with 1 = female)

Source

GESIS Datenarchiv

References

Roßteutscher, Sigrid et al. 2019. "Rolling Cross-Section-Wahlkampfstudie mit Nachwahl-Panelwelle (GLES 2017)." ZA6803 Datenfile Version 4.0.1. (GESIS Datenarchiv).

Examples

```
data(gles)
table(gles$vote)
```

`mnl_fd2_ova` *Multinomial First Differences Predictions For Two Values (Observed Value Approach)*

Description

Multinomial First Differences Predictions For Two Values (Observed Value Approach)

Usage

```
mnl_fd2_ova(
  model,
  data,
  xvari,
  value1,
  value2,
  nsim = 1000,
  seed = "random",
  probs = c(0.025, 0.975)
)
```

Arguments

`model`	the multinomial model, from a `multinom`()-function call (see the `nnet` package)
`data`	the data with which the model was estimated
`xvari`	the name of the variable that should be varied
`value1`	first value for the difference
`value2`	second value for the difference
`nsim`	numbers of simulations
`seed`	set a seed for replication purposes.
`probs`	a vector with two numbers, defining the significance levels. Default to 5% significance level: `c(0.025,0.975)`

Value

The function returns a list with several elements. Most importantly the list includes the simulated draws 'S', the simulated predictions 'P', the first differences of the predictions 'P_fd', a data set for plotting 'plotdata' the predicted probabilities, and one for the first differences 'plotdata_fd'.

Examples

```
library(nnet)
library(MASS)

dataset <- data.frame(y = c(rep("a", 10), rep("b", 10), rep("c", 10)),
                      x1 = rnorm(30),
```

```
                      x2 = rnorm(30, mean = 1),
                      x3 = sample(1:10, 30, replace = TRUE))

mod <- multinom(y ~ x1 + x2 + x3, data = dataset, Hess = TRUE)

fdi1 <- mnl_fd2_ova(model = mod, data = dataset,
                    xvari = "x1",
                    value1 = min(dataset$x1),
                    value2 = max(dataset$x1),
                    nsim = 10)
```

`mnl_fd_ova` *Multinomial First Differences Prediction (Observed Value Approach)*

Description

This function predicts values for two different scenarios over a range of values. It then takes the differences between the different simulations to return first differences for each value.

Usage

```
mnl_fd_ova(
  model,
  data,
  xvari,
  scenname,
  scenvalues,
  by = NULL,
  nsim = 1000,
  seed = "random",
  probs = c(0.025, 0.975)
)
```

Arguments

`model`	the multinomial model, from a `multinom`()-function call (see the `nnet` package)
`data`	the data with which the model was estimated
`xvari`	the name of the variable that should be varied (the x-axis variable in prediction plots)
`scenname`	if you want to hold a specific variable stable over all scenarios, you can name it here (optional).
`scenvalues`	determine the two values at which value you want to fix the scenario (`scenname`). The first differences will be computed by subtracting the values of the first supplied scenario from the second one.

by	define the steps of the xvari.
nsim	numbers of simulations
seed	set a seed for replication purposes.
probs	a vector with two numbers, defining the significance levels. Default to 5% significance level: c(0.025,0.975)

Details

The function uses the mnl_pred_ova function for each scenario. The results of these predictions are also returned and can therefore be easily accessed. If you need predictions for multiple scenarios, you can use this function to both plot the predictions for each scenario and the differences between them.

Value

The function returns a list with several elements. Most importantly the list includes the simulated draws 'S', the simulated predictions 'P', and a data set for plotting 'plotdata'.

Examples

```
library(nnet)
library(MASS)

dataset <- data.frame(y = c(rep("a", 10), rep("b", 10), rep("c", 10)),
                      x1 = rnorm(30),
                      x2 = rnorm(30, mean = 1),
                      x3 = sample(1:10, 30, replace = TRUE))

mod <- multinom(y ~ x1 + x2 + x3, data = dataset, Hess = TRUE)

fdif <- mnl_fd_ova(model = mod, data = dataset,
                   xvari = "x1", scenname = "x3",
                   scenvalues = c(min(dataset$x3), max(dataset$x3)),
                   nsim = 10)
```

---

mnl_pred_ova *Multinomial Prediction Function (Observed Value Approach)*

Description

This function predicts probabilities for all choices of a multinomial logit model over a specified span of values.

Usage

```
mnl_pred_ova(
  model,
  data,
  xvari,
  by = NULL,
  scenname = NULL,
  scenvalue = NULL,
  nsim = 1000,
  seed = "random",
  probs = c(0.025, 0.975)
)
```

Arguments

model	the multinomial model, from a multinom()-function call (see the nnet package)
data	the data with which the model was estimated
xvari	the name of the variable that should be varied (the x-axis variable in prediction plots)
by	define the steps of the xvari.
scenname	if you want to hold a specific variable stable over all scenarios, you can name it here (optional).
scenvalue	determine at which value you want to fix the scenname.
nsim	numbers of simulations
seed	set a seed for replication purposes.
probs	a vector with two numbers, defining the significance levels. Default to 5% significance level: c(0.025,0.975)

Value

The function returns a list with several elements. Most importantly the list includes the simulated draws 'S', the simulated predictions 'P', and a data set for plotting 'plotdata'.

Examples

```
library(nnet)
library(MASS)

dataset <- data.frame(y = c(rep("a", 10), rep("b", 10), rep("c", 10)),
                      x1 = rnorm(30),
                      x2 = rnorm(30, mean = 1),
                      x3 = sample(1:10, 30, replace = TRUE))

mod <- multinom(y ~ x1 + x2 + x3, data = dataset, Hess = TRUE)

pred <- mnl_pred_ova(model = mod, data = dataset,
                     xvari = "x1",
```

```
                        nsim = 10)
```

Index