Package 'surveynnet'

December 9, 2024

Title Neural Network for Complex Survey Data

Version 1.0.0

Description The goal of 'surveynnet' is to extend the functionality of 'nnet',
which already supports survey weights, by enabling it to handle clustered
and stratified data. It achieves this by incorporating design effects
through the use of effective sample sizes as outlined by Chen
and Rust (2017), <doi:10.1093/jssam/smw036>, and performed by
'deffCR' in the package 'PracTools' (Valliant, Dever, and Kreuter (2018),
<doi:10.1007/978-3-319-93632-1>).

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

URL https://github.com/237triangle/surveynnet

BugReports https://github.com/237triangle/surveynnet/issues

Imports dplyr, nnet, PracTools, stats, survey, survival

Depends R (>= 2.10)

LazyData true

NeedsCompilation no

Author Aaron Cohen [aut, cre] (<https://orcid.org/0009-0003-1971-1072>),
Raul Cruz-Cano [aut] (<https://orcid.org/0000-0001-7715-1198>)

Maintainer Aaron Cohen <cohenaa@iu.edu>

Repository CRAN

Date/Publication 2024-12-09 13:40:05 UTC

Contents

---

body_fat   *Simple body fat example data*

Description

Simple body fat example data

Usage

body_fat

Format

body_fat:
A data frame with 12 rows and 9 columns:

Subject Subject ID
group group
Weight_kg weight
Height_cm height
Age age
pct_body_fat percent body fat
survey_wt survey weight
stratum stratum
cluster cluster

---

nhanes.demo   *Nhanes example*

Description

Nhanes example

Usage

nhanes.demo

Format

nhanes.demo:
A data frame with 6230 rows and 9 columns:

SEQN Respondent sequence number
SDMVPSU Masked variance pseudo-PSU
SDMVSTRA Masked variance pseudo-stratum

    WTMEC2YR Full sample 2 year MEC exam weight

    BMXHT Standing height (cm)

    BMXWT Weight (kg)

    BMXBMI Body maxx index (kg/m**2)

    BPXSY1 Systolic blood pressure

    BPXDI1 Diastolic blood pressure

---

predict.surveynnet   *Predict response from fitted nnet, using new data*

Description

Predict response from fitted nnet, using new data

Usage

```
## S3 method for class 'surveynnet'
predict(object, newdat, ...)
```

Arguments

object	The surveynnet object (returned by surveynnet())
newdat	The matrix or data frame of test examples. Must be of the same structure as the data matrix used to fit the surveynnet object.
...	arguments passed to or from other methods

Value

The matrix/vector of values returned by the trained network. Note: it is possible to pass type = "raw" or type = "class" as appropriate. See predict.nnet() for more details.

Examples

```
# From the example in `surveynnet` help file:
y <- body_fat$pct_body_fat
x <- body_fat[,c("Weight_kg", "Height_cm", "Age")]
weight <- body_fat$survey_wt
strat <- body_fat$stratum
clust <- body_fat$cluster
y[strat==1] <- y[strat==1] + 30*0.00015*rnorm(sum(strat==1))
y[strat==2] <- y[strat==2] + 30*0.15*rnorm(sum(strat==2))
myout <- surveynnet(x,y,weight = weight, strat = strat, clust=clust)
newdat <-  2*x+rnorm(dim(x)[1])
predict(myout, newdat = newdat)
```

surveynnet | *Neural Net for Complex Survey Data*

Description

The surveynnet package extends the functionality of nnet (Venables and Ripley, 2002), which already supports survey weights, by enabling it to handle clustered and stratified data. It achieves this by incorporating design effects through the use of effective sample sizes in the calculations, performed by the package described in Valliant et al. (2023), by following the methods outlined by Chen and Rust (2017) and Valliant et al. (2018).

Usage

```
surveynnet(x, y, weight, strat, clust, comp_cases = FALSE, ...)
```

Arguments

x	Matrix or data frame of predictors. Must not contain any missing values.
y	Vector of targets / response values. Must not contain any missing values.
weight	The weights for each sample.
strat	The stratum for each sample.
clust	The cluster for each sample.
comp_cases	If TRUE, filter out missing values from x, y, weight, strat, and clust. Default FALSE. Note that in either case, the dimensions of all data mentioned above must agree.
...	Additional arguments to be passed into `PracTools::deffCR` or `nnet::nnet`. See documentation of those packages and functions for more details. Note that for the neural net (`nnet`), the default here is set to 3 layers ("size" parameter) and maximum iterations ("maxit" parameter) is set to 2000.

Value

A list containing two objects:

- A dataframe with the fitted values of the neural nets, using: no weights ("fitted"), the user-inputted weights ("fitted_weighted"), and the new method that adjusts the weights by using a design effect incorporating cluster and strata ("fitted_deff").
- The fitted neural network object (from `nnet`), using the novel design-effect based weights; this can be used to predict the outcomes for new observations.

References

- Chen, S., and K. F. Rust. 2017."An Extension of Kish's Formula for Design Effects to Two- and Three-Stage Designs with Stratification.", Journal of Survey Statistics and Methodology,5 (2): 111–30.
- Valliant, R., J. A. Dever, and F. Kreuter. 2018. Practical Tools for Designing and Weighting Survey Samples .2nd ed. New York: Springer-Verlag.

Examples

```
# short example with body fat dataset
y <- body_fat$pct_body_fat
x <- body_fat[,c("Weight_kg", "Height_cm", "Age")]
weight <- body_fat$survey_wt
strat <- body_fat$stratum
clust <- body_fat$cluster
y[strat==1] <- y[strat==1] + 30*0.00015*rnorm(sum(strat==1))
y[strat==2] <- y[strat==2] + 30*0.15*rnorm(sum(strat==2))

myout <- surveynnet(x,y,weight = weight, strat = strat, clust=clust)
myout


# NHANES example
# Predicting Diastolic BP from BMI, Systolic BP and Height
# PLEASE NOTE: for this example, pass "nest=TRUE" into the
# "..." parameters of the main function `surveynnet`

x <- nhanes.demo[,c("BMXBMI", "BPXSY1", "BMXHT")]
weight <- nhanes.demo$WTMEC2YR
strat <- nhanes.demo$SDMVSTRA
clust <- nhanes.demo$SDMVPSU
y <- nhanes.demo$BPXDI1
myout <- surveynnet(x,y,weight = weight, strat = strat, clust=clust, nest=TRUE)
head(myout$results, 15)
```

Index