

Package ‘TheSFACE’

July 21, 2025

Type Package

Title The Subtype Free Average Causal Effect

Version 0.1.0

Author Amit Sasson [aut, cre]

Maintainer Amit Sasson <amitsasson1@gmail.com>

Description

Estimation of the SF-ACE, a Causal Inference estimand proposed in the paper “The Subtype-Free Average Causal Effect For Heterogeneous Disease Etiology” (soon on arXiv).

Imports nnet, gridExtra, dplyr, purrr, tidyr, ggplot2, tibble, stats

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.1.2

NeedsCompilation no

Repository CRAN

Date/Publication 2022-06-01 16:00:08 UTC

Contents

max_lambda	2
plot.sface	3
print.sface	4
sface	5

Index	7
--------------	----------

max_lambda	<i>Max lambda1 and lambda2 values</i>
------------	---------------------------------------

Description

In order to perform the sensitivity analysis, the researcher must choose which values of lambda1 and lambda2 to explore. We give here an upper limit for lambda1 and lambda2, estimated from the data.

Usage

```
max_lambda(stand_formula, exposure, df, weight = 1)
```

Arguments

stand_formula	A formula for standartization and DR, $y \sim A + X$, the outcome as a function of the exposure and covariates
exposure	The treatment/exposure vector pf length n. Must be encoded 1 for treated and 0 for untreated.
df	a data frame with columns for the outcome, exposure and covariates.
weight	A numerical vector of length n, holding weights to adjust for missing subtypes, Default: 1

Value

a vector of two numeric values, the first element is the maximum possible lambda1 value and the second element is the maximum possible lambda2 value.

See Also

[multinom](#)

Examples

```
A <- rbinom(n = 1000, size = 1, prob = 0.5)
X1 <- rbinom(n = 1000, size = 1, prob = 0.5)
X2 <- rnorm(n = 1000, mean = 0, sd = 1)
y <- sample(c(0,1,2), 1000, replace=TRUE, prob=c(0.8, 0.1, 0.1) )
weight <- rep(1, n = 1000)
df <- data.frame(y, A, X1, X2, weight)

max_lambda(stand_formula = y ~ A + X1 + X2,
           exposure = "A",
           df = df,
           weight = weight)
```

plot.sface	<i>Plot SF-ACE</i>
------------	--------------------

Description

this function plots lists of the class "sface"

Usage

```
## S3 method for class 'sface'  
plot(x, ...)
```

Arguments

x	a list of class "sface", usually the output of the function sface
...	not used

Value

a plot of the estimated SF-ACEs as a function of lambda1, lambda2 or both. The plot is separated into panels based on method and scale, and shows each subtype in a different color.

Examples

```
A <- rbinom(n = 1000, size = 1, prob = 0.5)  
X1 <- rbinom(n = 1000, size = 1, prob = 0.5)  
X2 <- rnorm(n = 1000, mean = 0, sd = 1)  
y <- sample(c(0,1,2), 1000, replace=TRUE, prob=c(0.8, 0.1, 0.1) )  
weight <- runif(n = 1000, 0,1)  
df <- data.frame(y, A, X1, X2, weight)  
  
lst <- sface(stand_formula = y ~ A + X1 + X2,  
iptw_formula = A ~ X1 + X2,  
exposure = "A",  
outcome = "y",  
df = df,  
weight = "weight",  
lambda1 = c(0.3, 0.5, 0.7))  
  
plot(lst)
```

`print.sface`*Print SF-ACE*

Description

this function prints lists of the class "sface"

Usage

```
## S3 method for class 'sface'  
print(x, digits = 4, ...)
```

Arguments

<code>x</code>	a list of class "sface", usually the output of the function <code>sface</code>
<code>digits</code>	number of digits chosen for the print. Default: 4.
<code>...</code>	not used

Value

This function does not return a value, it prints the estimated SF-ACEs in the chosen methods and scales.

Examples

```
A <- rbinom(n = 1000, size = 1, prob = 0.5)  
X1 <- rbinom(n = 1000, size = 1, prob = 0.5)  
X2 <- rnorm(n = 1000, mean = 0, sd = 1)  
y <- sample(c(0,1,2), 1000, replace=TRUE, prob=c(0.8, 0.1, 0.1) )  
weight <- runif(n = 1000, 0,1)  
df <- data.frame(y, A, X1, X2, weight)  
  
lst <- sface(stand_formula = y ~ A + X1 + X2,  
iptw_formula = A ~ X1 + X2,  
exposure = "A",  
outcome = "y",  
df = df,  
weight = "weight",  
lambda1 = c(0.3, 0.5, 0.7))  
  
print(lst)
```

sface

*Subtype Free Average Causal Effect***Description**

A function to estimate the Subtype Free Average Causal Effect.

Usage

```
sface(
  stand_formula,
  iptw_formula,
  exposure,
  outcome,
  df,
  subtype = c(1, 2),
  scale = c("diff", "RR"),
  method = c("stand", "IPTW", "DR"),
  lambda1 = 0,
  lambda2 = 0,
  weight = 1,
  MultPer = 1
)
```

Arguments

stand_formula	A formula for standartization and DR, $y \sim A + X$, the outcome as a function of the exposure and covariates
iptw_formula	A formula for IPTW and DR, $A \sim X$, the exposure as a function of the covariates.
exposure	The treatment/exposure vector pf length n. Must be encoded 1 for treated and 0 for untreated.
outcome	The categorical outcome vector of length n. Must be encoded 0 for disease-free, 1 for the first subtype and 2 for the second subtype.
df	a data frame with columns for the outcome, expousre and covariates.
subtype	Should the SF-ACE be estimated for subtype 1 or subtype 2
scale	Should the SF-ACE be estimated on the difference or risk ratio scale.
method	Which method to use when adjusting for covariates, possibilities include standardization ("stand"), Inverse Probability Treatment Weighting ("IPTW"), and doubly robust estimation ("DR")
lambda1	sensitivity parameter for subtype 1. Can range between 0 (S-Monotonicity for subtype 1) and 1 (D-Monotonicity for subtype 1), Default: 0
lambda2	sensitivity parameter for subtype 2. Can range between 0 (S-Monotonicity for subtype 2) and 1 (D-Monotonicity for subtype 2), Default: 0

weight	A numerical vector of length n, holding weights to adjust for missing subtypes, Default: 1
MultiPer	A numeric value indicating per how many people the effect should be calculated on the difference scale, Default: 1

Details

DETAILIS

Value

a list of class "sface". In the cell "sface", there is a list holding the estimated SF_ACEs in the chosen methods and scales. In the cell "additional info", there is with additional info regarding the params used.

See Also

[multinom](#)

Examples

```
A <- rbinom(n = 1000, size = 1, prob = 0.5)
X1 <- rbinom(n = 1000, size = 1, prob = 0.5)
X2 <- rnorm(n = 1000, mean = 0, sd = 1)
y <- sample(c(0,1,2), 1000, replace=TRUE, prob=c(0.8, 0.1, 0.1) )
weight <- runif(n = 1000, 0,1)
df <- data.frame(y, A, X1, X2, weight)

sface(stand_formula = y ~ A + X1 + X2,
iptw_formula = A ~ X1 + X2,
exposure = "A",
outcome = "y",
df = df,
scale = c("diff", "RR"),
method = c("stand", "IPTW"),
weight = "weight")
```

Index

max_lambda, 2
multinom, 2, 6
plot.sface, 3
print.sface, 4
sface, 5