Package ‘SIDES’

May 16, 2021

Type Package

Title Subgroup Identification Based on Differential Effect Search

Version 1.16

Date 2021-05-16

Author Marie-Karelle Riviere

Maintainer Marie-Karelle Riviere <eldamjh@gmail.com>

Copyright All files are copyright Sanofi Aventis

Description Provides function to apply “Subgroup Identification based on Differential Effect Search” (SIDES) method proposed by Lipkovich et al. (2011) <doi:10.1002/sim.4289>.

License GPL-3

Imports memoise (>= 1.0.0), nnet (>= 7.3-12), multicool (>= 0.1-9), survival (>= 2.37-7), doParallel (>= 1.0.10), foreach (>= 1.4.3), MASS

Depends R (>= 3.1.2)

LinkingTo

NeedsCompilation no

Repository CRAN

Date/Publication 2021-05-16 21:40:03 UTC

R topics documented:

SIDES-package .................................................. 2
SIDES ............................................................... 2
simulation_SIDES ................................................ 5

Index 9
Description

Provides function to apply "Subgroup Identification based on Differential Effect Search" (SIDES) method proposed by Lipkovich et al. (2011) <doi:10.1002/sim.4289>.

Details

Package: SIDES
Type: Package
Version: 1.16
Date: 2021-05-16
License: GPL-3

Author(s)

Marie-Karelle Riviere-Jourdan <eldamjh@gmail.com>

References


Usage

SIDES(all_set, type_var, type_outcome, level_control, D=0, L=3, S, M=5, gamma=rep(1,3), H=1, pct_rand=0.5, prop_gpe=c(1), alloc_high_prob=TRUE, num_crit=1, step=0.5, nb_sub_cross=5, alpha, nsim=500, nsim_cv=500, ord.bin=10, M_per_covar=FALSE, upper_best=TRUE, selec=FALSE, seed=42, modified=TRUE)
Arguments

**all_set**  
Data frame representing the global data set. The first column must be the outcome (if the outcome is survival, this column should contain a data frame with the time-to-event in the first column and the indicator status in the second column), the second column must be the treatment variable, and other columns are for covariates.

**type_var**  
A vector of length the number of covariates giving for each of them their type. Must be either "continuous", "ordinal" or "nominal".

**type_outcome**  
Type of outcome. Are implementing "continuous", "binary", "survival" and "count".

**level_control**  
Value representing the control in the data set.

**D**  
Minimum desired difference to be demonstrated between the treatment and the control.

**L**  
Maximum number of covariates used to define a subgroup (= depth of the tree). The default value is set at 3.

**S**  
Minimum subgroup size desired. (Subgroups that do not meet this requirement will be excluded).

**M**  
Maximum number of best promising subgroups selected at each step of the algorithm. The default value is set at 5.

**gamma**  
Vector of length L representing the relative improvement parameter. Each element must be between 0 and 1. Smaller values indicate more selective procedure. If any improvement is desired, it is recommended to set all elements to 1. Default values are set at 1.

**H**  
Number of data sets the global data set is split into. There will be 1 training data set and H-1 validation sets. The default value is set at 1.

**pct_rand**  
Proportion of the global data set that is randomly allocated between training and validation sets. The default value is set at 0.5.

**prop_gpe**  
Vector of size H containing the proportion of patients for each data sets (training and validation).

**alloc_high_prob**  
Boolean with value TRUE indicating that patients are allocated to the set the minimizing the imbalanced score, or FALSE indicated that patients are randomized into those sets inversely proportional to their imbalanced score.

**num_crit**  
Integer representing the splitting criterion used. Value equal to 1 stands for criterion maximizing the differential effect between the two child subgroups, while value equal to 2 stands for criterion maximizing the treatment effect in at least one of the two child subgroups. The default value is set at 1.

**step**  
When gamma is not specified, step into which to cut the interval [0,1] to determine gamma by cross-validation. Warning, this process is highly time-consuming and several ties are obtained, thus it is more recommended to provide gamma after thinking about what is desired. The default value is set at 0.5.

**nb_sub_cross**  
Number of folds for cross-validation to determine gamma. The default value is set at 5.
alpha: Overall type I error rate.

nsim: Number of permutations for the resampling-based method used to protect the overall Type I error rate in a weak sense.

nsim_cv: Number of permutations for the resampling-based method used to protect the overall Type I error rate in the cross-validation part to determine gamma. The default value is set at 500.

ord.bin: Number of classes continuous covariates will be discretized into.

M_per_covar: Boolean indicating if the M best promising child subgroups are selected by covariate (TRUE) or across all remaining covariates. The default value is set at FALSE.

upper_best: Boolean indicating if greater values of the outcome mean better responses.

selec: Boolean indicating if in addition of the validated subgroups, the output should also contain subgroups that were selected (before validation).

seed: Seed. The default value is set at 42.

modified: Boolean indicating if modified or original Sidak correction is used for over-representation of covariates with more than 2 levels. Default value is TRUE.

Value

An object of class "SIDES" is returned, consisting of:

candidates: A list containing selected candidates subgroups (before validation step) and their associated p-values.

confirmed: A list containing confirmed/validated subgroups and their associated p-values.

Author(s)

Marie-Karelle Riviere-Jourdan <eldamjh@gmail.com>

References


Examples

```r
n=500
x=data.frame(matrix(rnorm(n*10,10,5),n,10),matrix(rbinom(n*10,1,0.5),n,10))
colnames(x)=paste("x",c(1:10),sep="\"")
rownames(x)=1:n
trt=rbinom(n,1,0.5)
I1=(x$x1>10);n1=sum(I1)
I6=(x$x6==0);n6=sum(I6)
I7=(x$x7==0);n7=sum(I7)
y=trt*(I1*(n-n1)-(1-I1)*n1+I6*(n-n6)-(1-I6)*n6+I7*(n-n7)-(1-I7)*n7)/n+rnorm(n)
data=cbind(y,trt,x)
head(data)
```
# REAL EXAMPLES TO UNCOMMENT
#s1 = SIDES(all_set=data,
#type_var=c(rep("continuous",5),rep("ordinal",5)), type_outcome="continuous",
#level_control=0, D=0, L=3, S=30, M=5, gamma=c(1,1,1), H=1, num_crit=1,
#alpha=0.10, nsim=1000, ord.bin=10, upper_best=TRUE, seed=42)

#s1 = SIDES(all_set=data,
#type_var=c(rep("continuous",5),rep("ordinal",5)), type_outcome="continuous",
#level_control=0, D=0, L=3, S=30, M=5, gamma=c(1,1,1), H=2, pct_rand=0.5,
#prop_gpe=c(0.7,0.3), numCrit=1, alpha=0.10, nsim=1000, ord.bin=10,
#upper_best=TRUE, seed=42)

#Example on how to enter data for survival
#n=200
#data=data.frame(rep(NA,n), rbinom(n,1,0.5), matrix(rbinom(n*5,1,0.5),n,5))
#colnames(data)=c("y", "trt", paste("x",c(1:5),sep="/quotesingle.Var/quotesingle.Var"))
#rownames(data)=1:n
#data$y = matrix(NA,ncol=2,nrow=n)
#data$y[,1] = rexp(n)
#data$y[,2] = rbinom(n,1,0.5)
#head(data)

---

### Description

simulation_SIDES is used to perform simulations of SIDES algorithm on a data set for binary, continuous, survival or count outcome.

### Usage

```r
simulation_SIDES(all_set, type_var, type_outcome, level_control, D=0, L=3, S, M=5, num_crit=1, gamma=rep(1,3), alpha, nsim=500, ord.bin=10, nrep=100, seed=42, H=1, pct_rand=0.5, prop_gpe=c(1), alloc_high_prob=TRUE, step=0.5, nb_sub_cross=5, nsim_cv=500, M_per_covar=FALSE, upper_best=TRUE, nb_cores=NA, ideal=NA, modified=TRUE)
```

### Arguments

- **all_set**: Data frame representing the global data set. The first column must be the outcome (if the outcome is survival, this column should contain a data frame with the time-to-event in the first column and the indicator status in the second column), the second column must be the treatment variable, and other columns are for covariates.
- **type_var**: A vector of length the number of covariates giving for each of them their type. Must be either "continuous", "ordinal" or "nominal".
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type_outcome</td>
<td>Type of outcome. Are implementing &quot;continuous&quot;, &quot;binary&quot;, &quot;survival&quot; and &quot;count&quot;.</td>
</tr>
<tr>
<td>level_control</td>
<td>Value representing the control in the data set.</td>
</tr>
<tr>
<td>D</td>
<td>Minimum desired difference to be demonstrate between the treatment and the control.</td>
</tr>
<tr>
<td>L</td>
<td>Maximum number of covariates used to define a subgroup (= depth of the tree). The default value is set at 3.</td>
</tr>
<tr>
<td>S</td>
<td>Minimum subgroup size desired. (Subgroups that do not meet this requirement will be excluded).</td>
</tr>
<tr>
<td>M</td>
<td>Maximum number of best promising subgroups selected at each step of the algorithm. The default value is set at 5.</td>
</tr>
<tr>
<td>num_crit</td>
<td>Integer representing the splitting criterion used. Value equal to 1 stands for criterion maximizing the differential effect between the two child subgroups, while value equal to 2 stands for criterion maximizing the treatment effect in at least one of the two child subgroups. The default value is set at 1.</td>
</tr>
<tr>
<td>gamma</td>
<td>Vector of length L representing the relative improvement parameter. Each element must be between 0 and 1. Smaller values indicates more selective procedure. If any improvement is desired, it is recommended to set all elements to 1. Default values are set at 1.</td>
</tr>
<tr>
<td>alpha</td>
<td>Overall type I error rate.</td>
</tr>
<tr>
<td>nsim</td>
<td>Number of permutations for the resampling-based method used to protect the overall Type I error rate in a weak sense.</td>
</tr>
<tr>
<td>ord.bin</td>
<td>Number of classes continuous covariates will be discretized into.</td>
</tr>
<tr>
<td>nrep</td>
<td>Number of simulation replicates.</td>
</tr>
<tr>
<td>seed</td>
<td>Seed. The default value is set at 42.</td>
</tr>
<tr>
<td>H</td>
<td>Number of data sets the global data set is split into. There will be 1 training data set and H-1 validation sets. The default value is set at 1.</td>
</tr>
<tr>
<td>pct_rand</td>
<td>Proportion of the global data set that is randomly allocated between training and validation sets. The default value is set at 0.5.</td>
</tr>
<tr>
<td>prop_gpe</td>
<td>Vector of size H containing the proportion of patients for each data sets (training and validation).</td>
</tr>
<tr>
<td>alloc_high_prob</td>
<td>Boolean with value TRUE indicating that patients are allocated to the set the minimizing the imbalanced score, or FALSE indicated that patients are randomized into those sets inversely proportional to their imbalanced score.</td>
</tr>
<tr>
<td>step</td>
<td>When gamma is not specified, step into which to cut the interval [0,1] to determine gamma by cross-validation. Warning, this process is highly time-consuming and several ties are obtained, thus it is more recommended to provide gamma after thinking about what is desired. The default value is set at 0.5.</td>
</tr>
<tr>
<td>nb_sub_cross</td>
<td>Number of folds for cross-validation to determine gamma. The default value is set at 5.</td>
</tr>
<tr>
<td>nsim_cv</td>
<td>Number of permutations for the resampling-based method used to protect the overall Type I error rate in the cross-validation part to determine gamma. The default value is set at 500.</td>
</tr>
</tbody>
</table>
**M_per_covar** Boolean indicating if the M best promising child subgroups are selected by covariate (TRUE) or across all remaining covariates. The default value is set at FALSE.

**upper_best** Boolean indicating if greater values of the outcome mean better responses.

**nb_cores** Number of cores to use as algorithm is parallelized. The default value used all available cores minus 1.

**ideal** When a simulation study is set up and data are generated by the user, the "true" ideal subgroup can be provided by the user to obtain additional results.

**modified** Boolean indicating if modified or original Sidak correction is used for over-representation of covariates with more than 2 levels. Default value is TRUE.

**Value**

An object of class "simulation_SIDES" is returned, consisting of:

- **pct_no_subgroup** Percentage of simulations where no subgroup is identified and validated.
- **mean_size** Mean subgroups size across all simulations (returning at least one subgroup).
- **subgroups** List of subgroups that are validated as responders.
- **pct_selection** Vector containing the percentage of selection and validation of each subgroup in subgroups.

**Author(s)**

Marie-Karelle Riviere-Jourdan <eldamjh@gmail.com>

**References**


**Examples**

```r
n=500
x=data.frame(matrix(rnorm(n*10,10,5),n,10),matrix(rbinom(n*10,1,0.5),n,10))
colnames(x)=paste("x",c(1:10),sep="\"Var\"")
rownames(x)=1:n
trt=rbinom(n,1,0.5)
I1=(x$x1>10);n1=sum(I1)
I6=(x$x6==0);n6=sum(I6)
I7=(x$x7==0);n7=sum(I7)
y=trt*I1*(n-n1)-(1-I1)*n1+I6*(n-n6)-(1-I6)*n6+I7*(n-n7)-(1-I7)*n7)/n+rnorm(n)
data=cbind(y,trt,x)
head(data)

# DUMMY EXAMPLE TO RUN
s1 = simulation_SIDES(all_set=data[,c(1,2,8,9,10)], type_var=rep("ordinal",3),
type_outcome="continuous", level_control=0, D=0, L=1, S=50, M=1, num_crit=1,
```
gamma=c(1), alpha=0.05, nsim=1, ord.bin=10, nrep=1, seed=42,
H=2, pct_rand=1.0, prop_gpe=c(0.7,0.3), upper_best=TRUE, nb_cores=1)

# REAL EXAMPLES TO UNCOMMENT
#s1 = simulation_SIDES(all_set=data,
#type_var=c(rep("continuous",5),rep("ordinal",5)), type_outcome="continuous",
#level_control=0, D=0, L=3, S=30, M=5, num_crit=1, gamma=c(1,1,1), alpha=0.10,
#nsim=1000, ord.bin=10, nrep=1000, seed=42, H=1, upper_best=TRUE)
#s1

#s1 = simulation_SIDES(all_set=data,
#type_var=c(rep("continuous",5),rep("ordinal",5)), type_outcome="continuous",
#level_control=0, D=0, L=3, S=30, M=5, num_crit=1, gamma=c(1,1,1), alpha=0.10,
#nsim=1000, ord.bin=10, nrep=1000, seed=42, H=2, pct_rand=0.5,
#prop_gpe=c(0.7,0.3), upper_best=TRUE)
#s1
Index

print.SIDES(SIDES), 2
print.simulation_SIDES
    (simulation_SIDES), 5

SIDES, 2
SIDES-package, 2
simulation_SIDES, 5