Package 'ITRLearn'

October 12, 2022

Type Package

Title Statistical Learning for Individualized Treatment Regime

Version 1.0-1

Date 2018-11-14

Author Chengchun Shi, Rui Song, Wenbin Lu and Bo Fu

Maintainer Chengchun Shi <cshi4@ncsu.edu>

Description Maximin-projection learning (MPL, Shi, et al., 2018) is implemented for recommending a meaningful and reliable individualized treatment regime for future groups of patients based on the observed data from different populations with heterogeneity in individualized decision making. Q-learning and A-learning are implemented for estimating the groupwise contrast function that shares the same marginal treatment effects. The packages contains classical Q-learning and A-learning algorithms for a single stage study as a byproduct. More functions will be added at later versions.

Imports Formula, kernlab

License GPL-2

NeedsCompilation yes

Repository CRAN

Date/Publication 2018-11-15 05:10:03 UTC

R topics documented:

ITRLearn-package *Statistical Learing for Individualized Treatment Regime*

Description

Maximin-projection learning (MPL, Shi, et al., 2018) is implemented for recommending a meaningful and reliable individualized treatment regime for future groups of patients based on the observed data from different populations with heterogeneity in individualized decision making. Q-learning and A-learning are implemented for estimating the groupwise contrast function that shares the same marginal treatment effects. The packages contains classical Q-learning and A-learning algorithms for a single stage study as a byproduct.

Details

Package:	ITRLearn
Type:	Package
Version:	1.0-1
Date:	2018-11-14
License:	GPL-2

Author(s)

Chengchun Shi, Rui Song, Wenbin Lu and Bo Fu

Maintainer: Chengchun Shi <cshi4@ncsu.edu>

References

Shi, C., Song, R., Lu, W., and Fu, B. (2018). Maximin Projection Learning for Optimal Treatment Decision with Heterogeneous Individualized Treatment Effects. *Journal of the Royal Statistical Society, Series B,* **80:** 681-702.

maximin *Maixmin projection learning for optimal individualized treatment regime*

Description

Derives a meaningful and reliable individualized treatment regime for future patients based on estimated groupwise contrast function.

Usage

```
maximin(B, c0)
```

Arguments

B	An p_1*G matrix containing parameters in the groupwise contast function. Here p_1 is the dimension of `x.tau` and G is the number of subgroups. It does not contain the intercept term. It can be computed by `MPL`.
c0	The common marginal treatment effect shared by all subgroups. It can be computed by `MPL`. `maximin` to compute the maximin effects.

Details

Denoted by β_g the g-th column of `B`. This function computes

$$\arg \max_{\|(\beta^T,c)^T\|=1} \min_{g\in\{1,\ldots,G\}} (\beta_g^T \beta + c_0 c).$$

The above optimaization problem can be efficiently computed based on quadratic programming.

Value

A vector of maximin effects.

Author(s)

Chengchun Shi

References

Shi, C., Song, R., Lu, W., and Fu, B. (2018). Maximin Projection Learning for Optimal Treatment Decision with Heterogeneous Individualized Treatment Effects. *Journal of the Royal Statistical Society, Series B,* **80:** 681-702.

See Also

`MPL`

Examples

```
set.seed(12345)
X <- matrix(rnorm(1600), 800, 2)
A <- rbinom(800, 1, 0.5)
h <- 1+sin(0.5*pi*X[,1]+0.5*pi*X[,2])
tau <- rep(0, 800)
B <- matrix(0, 2, 4)
B[,1] <- c(2,0)
B[,2] <- 2*c(cos(15*pi/180), sin(15*pi/180))
B[,3] <- 2*c(cos(70*pi/180), sin(70*pi/180))
B[,4] <- c(0,2)
for (g in 1:4){
```

```
    tau[((g-1)*200+1):(g*200)] <- X[((g-1)*200+1):(g*200),]%*%B[,g]
}
## mean and scale of the subgroup covariates are allowed to be different
X[1:200,1] <- X[1:200,1]+1
X[201:400,2] <- 2*X[201:400,2]-1
X[601:800,] <- X[601:800,]/2
Y <- h+A*tau+0.5*rnorm(800)
G <- c(rep(1,200), rep(2,200), rep(3,200), rep(4,200))
result <- MPL(Y~X|A|G)
maximin(result$B, result$c0)
```

MPL *Maixmin projection learning for optimal individualized treatment regime*

Description

Derives a meaningful and reliable individualized treatment regime based on the observed dataset from different subgroups with heterogeneity in optimal individualized treatment decision making. When patients are coming from the same group, it implements the classical Q learning and A learning algorithm.

Usage

```
MPL(formula, data, subset, na.action, method = c("Q", "A"), bootstrap = FALSE,
 control = MPL.control(...), model = TRUE, y = TRUE, a = TRUE,
 g = TRUE, x.tau = TRUE, x.h = TRUE, x.pi = TRUE, random = FALSE, ...)

MPL.fit(y, x.tau, a, g=NULL, x.h=NULL, x.pi=NULL, method=c("Q", "A"),
bootstrap=FALSE, random=FALSE, control=MPL.control())
```

Arguments

formula	A symbolic description of the model to be fitted(of type y ~ x.tau | a, or y ~ x.tau | a | g, or y ~ x.tau | a | g | x.h, or y ~ x.tau | a | g | x.h | x.pi, or y ~ x.tau | a | g | | x.pi. Details are given in 'Details').
data	An optional list or environment containing variables in formula.
subset, na.action	Arguments controlling formula processing via model.frame.
method	Method used for estimating the parameter in the groupwise contrast function. See 'Details'.
bootstrap	A logical value indicating whether bootstrap will be used. Default is FALSE. See 'Details'.
control	A list of control argument via MPL.control.
model	A logical value indicating whether *model frame* should be included as a component of the return value.

`y, a, g, x.tau, x.h, x.pi`
: For `MPL`: logical values indicating whether the response, the treatment, the subgroup indicator, covariates used to fit the contrast function, covariates used to fit the baseline function and covariates used to fit the propensity score function.

  For `MPL.fit`: y is the response vector (the larger the better), a is the treatment vector denoting the treatment patients receive, g is the group indicator indicating which group each patient belongs to, x.tau, x.h, x.pi are the design matrices used to fit the contrast, the baseline and the propensity score function.

`random`
: A logical value indicating whether using a constant to fit the propensity score function or not. In randomized studies, the propensity score is usually a constant function independent of baseline covariates. When `random=TRUE`, `MPL` uses a constant to fit the propensity score. Otherwise, it uses a logistic regression function based on covariates in `x.pi`.

`...`
: Argument passed to `MPL.control`.

Details

A saline feature of data from clinical trials and medical studies is inhomogeneity. Patients not only differ in baseline characteristics, but also the way they respond to the treatment. Individualized treatment regimes are developed to select effective treatments based on patient's heterogeneity. Formally speaking, an individualized treatment regime (ITR) is a function that maps patients' baseline covariates to the space of available treatment options. The goal in precision medicine is to identify the optimal ITR to reach the best clinical outcomes.

However, the optimal ITR might also vary for patients across different subgroups. This function implements the maximin projection learning method that derives a meanful and reliable ITR for future patients based on the observed data from different populations with heterogeneity in optimal individualized decision making.

The means and covariance matrices of patients baseline covariates are allowed to vary across different subgroups. `MPL` will first standardize the groupwise baseline covariates to have zero mean and indentity covariance matrix (based on Gram-Schmidt Orthonormalization) and then recommends an ITR for future groups of patients. Notice that the resulting ITR cannot be directly applied to future patients. We need to standardize future patients baseline covariates (based on the same procedure) first and apply the transformed covariates to the ITR. This is implemented by the `TR` function.

When the group indicator g is omitted (or it is a constant vector) in the formula, `MPL` assumes all the patients are coming from the same group and implements the classical Q-learning and A-learning algorithm. Otherwise, g should be a numeric vector that has the same length of `y`, indicating which group each patient belongs to.

When `x.h` is omitted and the baseline `h.est` in `MPL.control` is not specified, `MPL` sets `x.h=x.tau`. When `x.pi` is omitted, the propensity score `pi.est` in `MPL.control` is not specified, and `random=FALSE`, `MPL` sets `x.pi=x.tau`.

Q-learning fits the entire Q function (the conditional mean of response given baseline covariates and treatment) to derive the optimal ITR. A-learning is a more robust method that focuses directly on the contrast function (the difference between two Q functions). It requires to specify both the baseline and the propensity score function and the resulting estimator for the contrast function is consistent when either of the function is correctly specified. This is referred to as the doubly robustness property of A-learning. `MPL` uses Q-learning or A-learning to estimate the groupwise contrast function that shares the same marginal treatment effects across different subgroups. These

estimators are further used to derive a ITR for future groups of patients. By default, `method="A"` and A-learning is implemented.

Inference for the maximin effects and the parameters in the groupwise contrast functions are conducted based on bootstrap. By default, `bootstrap=FALSE` and Bootstrap will not be conducted.

Value

`Theta.tau.est` An $(p_1+1)\times G$ matrix containing estimated parameters in the groupwise contrast function. Here p_1 is the dimension of `x.tau` and G is the number of subgroups. The first row contains the intercept term.

`Theta.h.est` An $(p_2+1)\times G$ matrix containing estimated parameters in the groupwise baseline function. Here p_2 is the dimension of `x.h` and G is the number of subgroups. The first row contains the intercept term. It equals `NULL` when `h.est` in `MPL.control` is prespecified.

`Theta.pi.est` An $(p_3+1)\times G$ matrix containing estimated parameters in the groupwise propensity score function. Here p_3 is the dimension of `x.pi` and G is the number of subgroups. The first row contains the intercept term. It equals `NULL` when `pi.est` in `MPL.control` is prespecified.

`h.est` Estimated baseline function.

`pi.est` Estimated propensity score function.

`B` An $p_1\times G$ matrix containing estimated parameters in the groupwise contast function. Here p_1 is the dimension of `x.tau` and G is the number of subgroups. It does not contain the intercept term. These parameters are the corresponding coefficients of the transformed covariates and are thus different from `Theta.tau.est`. It can be used as the input of the function `maximin` to compute the maximin effects.

`c0` The common marginal treatment effect shared by all subgroups. It can be used as the input of the function `maximin` to compute the maximin effects.

`beta.est` The estimated maximin effects used to contruct ITR for future patients.

`Theta.tau.boot` An $(p_1+1)\times G\times B_0$ array containing bootstrap samples for the estimated parameters in the groupwise contrast function. Here p_1 is the dimension of `x.tau`, G is the number of subgroups and B_0 is the number of bootstrap samples. It equals `NULL` when `bootstrap=FALSE`.

`Theta.h.boot` An $(p_2+1)\times G\times B_0$ array containing bootstrap samples for the estimated parameters in the groupwise baseline function. Here p_2 is the dimension of `x.h`, G is the number of subgroups and B_0 is the number of bootstrap samples. It equals `NULL` when `bootstrap=FALSE` or `h.est` in `MPL.control` is prespecified.

`Theta.pi.boot` An $(p_3+1)\times G\times B_0$ array containing bootstrap samples for the estimated parameters in the groupwise propensity score function. Here p_3 is the dimension of `x.pi`, G is the number of subgroups and B_0 is the number of bootstrap samples. It equals `NULL` when `bootstrap=FALSE` or `pi.est` in `MPL.control` is prespecified.

`beta.boot` An $p_1\times B_0$ matrix containing bootstrap sample for the estimated maximin effects. Here p_1 is the dimension of `x.tau` and B_0 is the number of bootstrap samples. It equals `NULL` when `bootstrap=FALSE`

standardize	A logical value indcating whether future patients covariates should be standardized first to be applied to the ITR constructed by the maximin effects. TRUE if there are multiple subgroups and FALSE otherwise.
model	The full model frame (if model = TRUE).
y	Response vector (if y = TRUE).
x.tau	Covariates used to model the contrast function (if x.tau = TRUE).
a	Treatment vector (if a = TRUE).
g	Group Indicator (if g = TRUE).
x.h	Covariates used to model the baseline function (if x.h = TRUE).
x.pi	Covariates used to model the propensity score function (if x.pi = TRUE).

Author(s)

Chengchun Shi

References

Shi, C., Song, R., Lu, W., and Fu, B. (2018). Maximin Projection Learning for Optimal Treatment Decision with Heterogeneous Individualized Treatment Effects. *Journal of the Royal Statistical Society, Series B,* **80:** 681-702.

See Also

MPL.control, TR, maximin

Examples

```
set.seed(12345)
X <- matrix(rnorm(1600), 800, 2)
A <- rbinom(800, 1, 0.5)
h <- 1+sin(0.5*pi*X[,1]+0.5*pi*X[,2])
tau <- rep(0, 800)
B <- matrix(0, 2, 4)
B[,1] <- c(2,0)
B[,2] <- 2*c(cos(15*pi/180), sin(15*pi/180))
B[,3] <- 2*c(cos(70*pi/180), sin(70*pi/180))
B[,4] <- c(0,2)
for (g in 1:4){
    tau[((g-1)*200+1):(g*200)] <- X[((g-1)*200+1):(g*200),]%*%B[,g]
}
## mean and scale of the subgroup covariates are allowed to be different
X[1:200,1] <- X[1:200,1]+1
X[201:400,2] <- 2*X[201:400,2]-1
X[601:800,] <- X[601:800,]/2
Y <- h+A*tau+0.5*rnorm(800)
G <- c(rep(1,200), rep(2,200), rep(3,200), rep(4,200))

## Q-learning
result <- MPL(Y~X|A|G, method="Q")
```

```
## A-learning
result <- MPL(Y~X|A|G)

## treating as homogeneous
result <- MPL(Y~X|A)


result <- MPL(Y~X|A|G, bootstrap=TRUE)
```

`MPL.control` *Control parameters for maximin projection learning*

Description

Parameters that control fitting of maximin projection learning.

Usage

```
MPL.control(pi.est = NULL, h.est = NULL, boot.sample = 600)
```

Arguments

`pi.est`	Estimated propentisy score for each patient. If not specified, a logistic regression model is fitted to estimate the propensity score.
`h.est`	Estimated baseline function for each patient. If not specified, a linear regression model is fitted to estimate the baseline function.
`boot.sample`	Number of bootstrap samples used for inference of the maximin effects and the subgroup parameter. Default is 600.

Value

A list with the arguments specified.

See Also

`MPL`, `MPL.fit`

TR *Individualized treatment regime based on MPL.*

Description

Recommend individualized treatment regime for future patients, based on the maximin projection learning method.

Usage

```
TR(object, x)
```

Arguments

object	Fitted object of class "MPL".
x	A matrix consisting of future patients baseline covariates. If there's only one group of patients in the observed data fitted by "MPL", then future patients are assuming to coming from this group and there is no restrictions on the number of observations in x. Otherwise, we require the number of observations in x to be larger than or equal to the dimension of x.

Value

A vector of individualized treatments tailored for future patients.

Author(s)

Chengchun Shi

See Also

MPL

Examples

```
X <- matrix(rnorm(1600), 800, 2)
A <- rbinom(800, 1, 0.5)
h <- 1+sin(0.5*pi*X[,1]+0.5*pi*X[,2])
tau <- rep(0, 800)
B <- matrix(0, 2, 4)
B[,1] <- c(2,0)
B[,2] <- 2*c(cos(15*pi/180), sin(15*pi/180))
B[,3] <- 2*c(cos(70*pi/180), sin(70*pi/180))
B[,4] <- c(0,2)
for (g in 1:4){
    tau[((g-1)*200+1):(g*200)] <- X[((g-1)*200+1):(g*200),]%*%B[,g]
}
## mean and scale of the subgroup covariates are allowed to be different
X[1:200,1] <- X[1:200,1]+1
```

```
X[201:400,2] <- 2*X[201:400,2]-1
X[601:800,] <- X[601:800,]/2
Y <- h+A*tau+0.5*rnorm(800)
G <- c(rep(1,200), rep(2,200), rep(3,200), rep(4,200))
result <- MPL(Y~X|A|G)
ITR <- TR(result, matrix(rnorm(200), 100, 2))
```

Index