Rdocumentation

April 29, 2021

`Group_specific_Delta_AUC_estimation`
Difference of AUC of Two Group-Specific Polynomial Marginal Dynamics

---

Description

This function estimates the difference of area under the curve of marginal dynamics from two groups when marginal dynamics are modeled by group-structured polynomials or B-spline curves.

Usage

```
Group_specific_Delta_AUC_estimation(
  MEM_Pol_group,Group1,Group2,
  time.G1,time.G2,common.interval = TRUE,
  method = "trapezoid",Averaged = FALSE
)
```

Arguments

`MEM_Pol_group` A list with similar structure than the output provided by the function MEM_Polynomial_Group_struct

A list containing:

- `Model_estimation`: a list containing at least 2 elements:
  1. the vector of the marginal (fixed) parameters estimates (at least for the groups whose AUC is to estimate), labeled *'beta'*.
  2. the variance-covariance matrix of these parameters, labeled *'varFix'* (see MEM_Polynomial_Group_structure for details about the parameter order).
- `Model_features`: a list of at least 2 elements:
  1. `Groups`: a vector indicating the names of the groups whose fixed parameters are given.
  2. `Marginal.dyn.feature`: a list summarizing the features of the marginal dynamics defined in the model:
     - `dynamic.type`: a character scalar indicating the chosen type of marginal dynamics. Options are 'polynomial' or 'spline'.

- `intercept`: a logical vector summarizing choices about global and group-specific intercepts (Number of groups + 1) elements whose elements are named as ('global.intercept','group.intercept1', ..., 'group.interceptG') if G Groups are defined in `MEM_Pol_group`. For each element of the vector, if TRUE, the considered intercept is considered as included in the model (see *Examples*).

If `dynamic.type` is defined as 'polynomial':

- `polynomial.degree`: an integer vector indicating the degree of polynomial functions, one value for each group.

If `dynamic.type` is defined as 'spline':

- `spline.degree`: an integer vector indicating the degree of B-spline curves, one for each group.
- `knots`: a list of group-specific internal knots used to build B-spline basis (one numerical vector for each group) (see bs for more details).
- `df`: a numerical vector of group-specific degrees of freedom used to build B-spline basis, (one for each group).
- `boundary.knots`: a list of group-specific boundary knots used to build B-spline basis (one vector for each group) (see bs for more details).

`Group1` a character scalar indicating the name of the first group whose marginal dynamics must be considered. This group name must belong to the set of groups involved in the MEM (see `Groups` vector in `MEM_Pol_group`).

`Group2` a character scalar indicating the name of the second group whose marginal dynamics must be considered. This group name must belong to the set of groups involved in the MEM (see `Groups` vector in `MEM_Pol_group`).

`time.G1` a numerical vector of time points (x-axis coordinates) to use for the Group1 AUC calculation.

`time.G2` a numerical vector of time points (x-axis coordinates) to use for the Group2 AUC calculation.

`common.interval` a logical scalar. If FALSE, the difference of AUC is calculated as the difference of AUCs where the AUC of each group is calculated on its specific interval of time. If TRUE (default), the difference of AUC is estimated on a common interval of time defined as the intersect of the two group-specific interval (see @details for more details).

`method` a character scalar indicating the interpolation method to use to estimate the AUC. Options are 'trapezoid' (default), 'lagrange' and 'spline'. In this version, the 'spline' interpolation is implemented with "not-a-knot" spline boundary conditions.

`Averaged` a logical scalar. If TRUE, the function return the difference of normalized AUC (nAUC) where nAUC is computated as the AUC divided by the range of time of calculation. If FALSE (default), the classic AUC is calculated.

Details

The difference of area under the curve between the two groups of interest is calculated as an approximation of the difference of the integrals of the expected value of the estimated outcome Y specific to the two groups g1 and g2. Assuming a time interval $[0, T_{g1}]$ for the group 1 and $[0, T_{g2}]$ for the group 2, the difference of AUC is then calculated as

$$\Delta AUC = \int_0^{T_{g_2}} E(\hat{Y_{g_2}})(t)dt - \int_0^{T_{g_1}} E(\hat{Y_{g_1}})(t)dt$$

Similarly, the difference of the normalized AUC ($\Delta nAUC$) for these two same groups is then defined as

$$\Delta nAUC = \frac{1}{T_{g_2}} \int_0^{T_{g_2}} E(\hat{Y_{g_2}})(t)dt - \frac{1}{T_{g_1}} \int_0^{T_{g_1}} E(\hat{Y_{g_1}})(t)dt$$

When we want to estimate the difference of AUC (or nAUC) on the same time interval for the two groups (`common.interval = TRUE`), we define $T = \min(T_{g_1}, T_{g_2}))$ and we calculate the difference of AUC as

$$\Delta AUC = \int_0^{T} E(\hat{Y_{g_2}})(t)dt - \int_0^{T} E(\hat{Y_{g_1}})(t)dt$$

$$\Delta nAUC = \frac{1}{T} \int_0^{T} E(\hat{Y_{g_2}})(t)dt - \frac{1}{T} \int_0^{T} E(\hat{Y_{g_1}})(t)dt$$

Value

A numerical scalar defined as $\Delta AUC = AUC_2 - AUC_1$ (or $\Delta nAUC = nAUC_2 - nAUC_1$) with AUC_1 (or $nAUC_1$) and AUC_2 (or $nAUC_2$) being respectively estimated as the AUC (or nAUC) for the Group1 and for the Group2.

See Also

MEM_Polynomial_Group_structure, Group_specific_AUC_estimation

Examples

```
# Download of data
data("HIV_Simu_Dataset_Delta01_cens")
data <- HIV_Simu_Dataset_Delta01_cens

# Change factors in character vectors
data$id <- as.character(data$id) ; data$Group <- as.character(data$Group)

# Example 1: We consider the variable \code{MEM_Pol_Group} as the output of our function
# \link[AUCcomparison]{MEM_Polynomial_Group_structure}
MEM_estimation_1 <- MEM_Polynomial_Group_structure(y=data$VL,x=data$time,Group=data$Group,
                                                   Id=data$id,Cens=data$cens)
time_group1 <- unique(data$time[which(data$Group=="Group1")])
time_group2 <- unique(data$time[which(data$Group=="Group2")])

Delta_AUC_1 <- Group_specific_Delta_AUC_estimation(MEM_Pol_group=MEM_estimation_1,
                                                   Group1="Group1",Group2="Group2",
                                                time.G1=time_group1,time.G2=time_group2)

# Example 2: We consider results of MEM estimation from another source.
# We have to give build the variable 'MEM_Pol_group' with the good structure
# We build the variable 'MEM_estimation_2' with the results of MEM estimation obtained for 2 groups
MEM_estimation_2 <- list(Model_estimation=c(1.077,0.858,-0.061,0.0013,0.887,-0.066,0.0014),
                         Model_features=list(Groups=c("Group1","Group2"),
                                             Marginal.dyn.feature=list(dynamic.type="polynomial",
                                                   intercept=c(global.intercept=TRUE,
                                                group.intercept1=FALSE,group.intercept2=FALSE),
                                                   polynomial.degree=c(3,3))))

Delta_AUC_2 <- Group_specific_Delta_AUC_estimation(MEM_Pol_group=MEM_estimation_2,
                                                   Group1="Group1",Group2="Group2",
```

```
time.G1=time_group1,time.G2=time_group2)
```

Index