Rdocumentation

April 29, 2021

`ANOVA_AUC_Group_Specific` *One-way analysis of variance of group specific AUC*

Description

This function performs a one-way ANOVA to compare the area under the curves of multiple groups marginal dynamics, modeled by group-structured polynomials or B-spline curve in Mixed-Effects model. Before performing the ANOVA, this function can parform a Bartlett's test to evaluate homoscedasticity. In addition to ANOVA, users can decide to evaluate all the 2 by 2 comparisons.

Usage

```
ANOVA_AUC_Group_Specific(
  MEM_Pol_group,Groups,Time_groups,
  Nb_id_group,common.interval = TRUE,
  method = "trapezoid",Averaged = FALSE,
  conf_level = 0.95,bartlettTest = FALSE,
  data = NULL,twobytwo.comp = TRUE,
  alternative = "two.sided"
)
```

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).

`Groups` a vector indicating the names of the groups belonging to the set of groups involved in `MEM_Pol_group` we want to include in the ANOVA (a subset or the entire set of groups involved in the model can be considered).

`Time_groups` a list of numerical vectors indicating the time points to consider in AUC calculation for each group (as much elements than the number of groups in `Groups`).

`Nb_id_group` a numerical vector indicating the number of individuals belonging to each group (as much elements than the number of groups in `Groups`).

`common.interval`

a logical scalar. If TRUE (default) AUCs of all the compared groups are calculated on the same time interval defined as the intersect of all the time interval defined in `Time_groups`. If FALSE, AUC specific to each group is evaluated on its own interval.

`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, AUC are evaluated as normalized AUC (nAUC) where nAUC is computed as the AUC divided by the range of time of calculation. If FALSE (default), the classic AUC is calculated (see Group_specific_AUC_estimation for more details).

`conf_level` a numerical value (between 0 and 1) indicating the confidence level of the interval. By default, this variable is fixed at 0.95

`bartlettTest` a logial scalar indicating whether a bartettTest must be performed before the ANOVA. Default: FALSE

`data` a dataframe gathering data for the groups involved in the ANOVA that have been fitted by the MEM model summarized in `MEM_Pol_group`. This dataframe has to contain at least 4 columns:

- A column labeled ’Group’ containing the information of the group for each observation (the same Groups than those defined in `Groups`).
- A column labeled ’id’ containing the information of individual identifier.
- A column labeled ’time’ containing information about the time of observations.
- A column labeled ’value’ containing the longitudinal observations.

`twobytwo.comp` a logical scalar indicating whether all the 2 by 2 comparisons must be evaluated after the ANOVA. Default: TRUE.

`alternative` a character scalar specifying the alternative hypothesis for the 2 by 2 comparisons. Options are ’two.sided’ (default), ’greater’ or ’less’.

Value

A list of three elements elements:

- `bartlettTest`: a list of class "htest" corresponding to the Bartlett’s test results (see bartlett.test for more details). If the test is not performed, a character ’Not performed’ is returned.
- `ANOVA_F`: a list containing:
  1. `Followup`: a vector of two numerical values indicating the time interval when `Common.interval` = TRUE. If `common.interval`=FALSE, if variable is not included in the list.
  2. `Estimated_AUCs`: the vector of AUC estimated for each group involved in the ANOVA.
  3. `Estimated_VarAUCs`: the vector of the intra group variance estimated by the MEM (equiv. squared Standard error).
  4. `Between`: a vector gathering the between groups sum of squares, degree of freedom and variance with var=SS/df.
  5. `Within`: a vector gathering the within groups sum of squares, degree of dreedom and variance with var=SS/df.
  6. `ANOVA_F`: the value of the ANOVA statistic F.
  7. `Pvalue`: the value of the Pvalue.
  8. `Reject_H0`: a boolean indicating whether the null hypothesis (all AUC are equals) is rejected.
- `TwobyTwo_Comparison`: a list of K sublists where each sublist k gathers results of the kth comparison. If the 2 by 2 comparison is nor performed, a character ’Not performed’ is returned.
  1. `Groups`: a vector indicating the 2 compared groups.
  2. `Estimated.AUC`: a vector of the 2 estimated values of AUC.
  3. `Delta_AUC`: the value of the difference of AUC between the 2 compared groups.
  4. `Tstat`: the value of the t-statistic.
  5. `Pvalue`: the P-value (without any adjustment on multiple tests).
  6. `Conf.int`: the confidence interval.
  7. `Adjusted.Pvalue`: the P-value adjusted on multiple testing by "bonferroni" method (see p.adjust for more details).

See Also

`AUC_time_weights_estimation` `bartlett.test`

Examples

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

colnames(data) <- c("id","time","Group","value","cens")
# Change factors in character vectors
data$id <- as.character(data$id) ; data$Group <- as.character(data$Group)

MEM_estimation <- MEM_Polynomial_Group_structure(y=data$value,x=data$time,Group=data$Group,
                                                 Id=data$id,Cens=data$cens)

Groups <- unique(data$Group)
Time_groups <- lapply(seq(1,length(Groups)),function(g)
                           return(unique(data$time[which(data$Group == Groups[g])])))
Nb_id_group <- sapply(seq(1,length(Groups)),function(g)
                        return(length(unique(data$id[which(data$Group == Groups[g])]))))

ANOVA_test <- ANOVA_AUC_Group_Specific(MEM_Pol_group=MEM_estimation,Groups=Groups,
                                       Time_groups=Time_groups,Nb_id_group=Nb_id_group,
                                 common.interval=TRUE,method="trapezoid",Averaged=FALSE,
                                       conf_level=0.95,bartlettTest=TRUE,data=data,
                                       twobytwo.comp=TRUE,alternative="two.sided")
```

Index