Package: gbmt (via r-universe)

October 21, 2024

Type Package

Title Group-Based Multivariate Trajectory Modeling

Version 0.1.3

Date 2022-03-05

Author Alessandro Magrini

Maintainer Alessandro Magrini <alessandro.magrini@unifi.it>

Description Estimation and analysis of group-based multivariate trajectory models (Nagin, 2018 <doi:10.1177/0962280216673085>; Magrini, 2022 <doi:10.1007/s10182-022-00437-9>). The package implements an Expectation-Maximization (EM) algorithm allowing unbalanced panel and missing values, and provides several functionalities for prediction and graphical representation.

Depends R (>= 3.5.0), Matrix

Imports methods

License GPL-2

NeedsCompilation no

Date/Publication 2022-03-05 17:00:02 UTC

Repository https://alessandromagrini.r-universe.dev

RemoteUrl https://github.com/cran/gbmt

RemoteRef HEAD

RemoteSha 724f053e83ea31dda531eb1ca6945dad403032f6

Contents

`gbmt-package` *Group-Based Multivariate Trajectory Modeling*

Description

Estimation and analysis of group-based multivariate trajectory models.

Details

Package:	gbmt
Type:	Package
Version:	0.1.3
Date:	2022-03-05
License:	GPL-2

Group-based trajectory modeling is a statistical method to determine groups of units based on the trend of a multivariate time series. It is a special case of latent class growth curves where the units in the same group have the same trajectory (Nagin, 2005), but it assumes a multivariate polynomial regression on time within each group, instead of a univariate one, to account for multiple indicators (Nagin et al., 2018; Magrini, 2022). A group-based multivariate trajectory model is estimated through the Expectation-Maximization (EM) algorithm, which allows unbalanced panel and missing values. The main functions currently implemented in the package are:

- gbmt: to estimate a group-based multivariate trajectory model;
- predict.gbmt: to perform prediction on trajectories;
- plot.gbmt: to display estimated and predicted trajectories;
- posterior: to compute posterior probabilities for new units.

Author(s)

Alessandro Magrini <alessandro.magrini@unifi.it>

References

A. Magrini (2022). Assessment of agricultural sustainability in European Union countries: A group-based multivariate trajectory approach. *Advances in Statistical Analysis*, published online: March 2022. DOI: 10.1007/s10182-022-00437-9

D. S. Nagin, B. L. Jones, V. L. Passos and R. E. Tremblay (2018). Group-based multi-trajectory modeling. *Statistical Methods in Medical Research*, 27(7): 2015-2023. DOI: 10.1177/0962280216673085

D. S. Nagin (2005). Group-based modeling of development. Harvard University Press, Cambridge, US-MA.

agrisus *EU agricultural sustainability data*

Description

Data on several indicators covering the economic, social and environmental dimensions of agricultural sustainability for 26 EU countries in the period 2004-2018.

Usage

```
data(agrisus)
```

Format

A data.frame with a total of 390 observations on the following 16 variables:

`Country` Country name.

`Country_code` Country code.

`Year` Time of measurement (year).

`Date` Time of measurement (date).

`TFP_2005` Total Factor Productivity (TFP) index of agriculture (2005=100). Source: CMEF.

`NetCapital_GVA` Net capital stocks in agriculture (2015 US dollars) to gross value added of agriculture (2015 US dollars). Source: Faostat.

`Manager_ratio` Ratio young/elderly for farm managers (number of managers with less than 35 years by 100 managers with 55 years and over). Source: CMEF.

`FactorIncome_paid_2010` Real income of agricultural factors per paid annual work unit (index 2010=100). Source: Eurostat.

`EntrIncome_unpaid_2010` Net entrepreneurial income of agriculture per unpaid annual work unit (index 2010=100). Source: Eurostat.

`Income_rur` Median equivalised net income in rural areas (power purchasing standard). Source: Eurostat.

`Unempl_rur` At-risk-of-poverty rate in rural areas (%). Source: Eurostat.

`Poverty_rur` Unemployment rate in rural areas (%). Source: Eurostat.

`RenewProd` Production of renewable energy from agriculture (share of total production of renewable energy, %). Source: CMEF.

`Organic_p` Area under organic cultivation (share of utilized agricultural area, %). Source: Faostat.

`GHG_UAA` Greenhouse gas emissions due to agriculture (million CO2 equivalent grams per hectare of utilized agricultural area). Source: Faostat.

`GNB_UAA` Gross nitrogen balance (tonnes of nutrient per hectare of utilized agricultural area). Source: Eurostat.

Note

This is the dataset employed in Magrini (2022).

References

European Commission (2022). Eurostat database. https://ec.europa.eu/eurostat/data/database

European Commission (2020). Common Monitoring and Evaluation Framework (CMEF) for the CAP 2014-2020. https://agridata.ec.europa.eu/extensions/DataPortal/cmef_indicators.html

Food and Agriculture Organization (2022). FAOSTAT statistical database. https://www.fao.org/faostat/en/#home

A. Magrini (2022). Assessment of agricultural sustainability in European Union countries: A group-based multivariate trajectory approach. *Advances in Statistical Analysis*, published online: March 2022. DOI: 10.1007/s10182-022-00437-9

See Also

agrisus2

`agrisus2` *EU agricultural sustainability data (after imputation of missing values)*

Description

Data on several indicators covering the economic, social and environmental dimensions of agricultural sustainability for 26 EU countries in the period 2004-2018. Missing values have been imputed.

Usage

```
data(agrisus2)
```

Format

A data.frame with a total of 390 observations on the following 16 variables:

`Country` Country name.

`Country_code` Country code.

`Year` Time of measurement (year).

`Date` Time of measurement (date).

`TFP_2005` Total Factor Productivity (TFP) index of agriculture (2005=100). Source: CMEF.

`NetCapital_GVA` Net capital stocks in agriculture (2015 US dollars) to gross value added of agriculture (2015 US dollars). Source: Faostat.

`Manager_ratio` Ratio young/elderly for farm managers (number of managers with less than 35 years by 100 managers with 55 years and over). Source: CMEF.

`FactorIncome_paid_2010` Real income of agricultural factors per paid annual work unit (index 2010=100). Source: Eurostat.

`EntrIncome_unpaid_2010` Net entrepreneurial income of agriculture per unpaid annual work unit (index 2010=100). Source: Eurostat.

`Income_rur` Median equivalised net income in rural areas (power purchasing standard). Source: Eurostat.

`Unempl_rur` At-risk-of-poverty rate in rural areas (%). Source: Eurostat.

`Poverty_rur` Unemployment rate in rural areas (%). Source: Eurostat.

`RenewProd` Production of renewable energy from agriculture (share of total production of renewable energy, %). Source: CMEF.

`Organic_p` Area under organic cultivation (share of utilized agricultural area, %). Source: Faostat.

`GHG_UAA` Greenhouse gas emissions due to agriculture (million CO2 equivalent grams per hectare of utilized agricultural area). Source: Faostat.

`GNB_UAA` Gross nitrogen balance (tonnes of nutrient per hectare of utilized agricultural area). Source: Eurostat.

Note

This is the dataset employed in Magrini (2022) after imputation of missing values according to a group-based multivariate trajectory model with three groups and three polynomial degrees.

References

European Commission (2022). Eurostat database. https://ec.europa.eu/eurostat/data/database

European Commission (2020). Common Monitoring and Evaluation Framework (CMEF) for the CAP 2014-2020. https://agridata.ec.europa.eu/extensions/DataPortal/cmef_indicators.html

Food and Agriculture Organization (2022). FAOSTAT statistical database. https://www.fao.org/faostat/en/#home

A. Magrini (2022). Assessment of agricultural sustainability in European Union countries: A group-based multivariate trajectory approach. *Advances in Statistical Analysis*, published online: March 2022. DOI: 10.1007/s10182-022-00437-9

See Also

agrisus

---

gbmt *Estimation of a group-based multivariate trajectory model*

Description

Estimation of a group-based multivariate trajectory model through the Expectation-Maximization (EM) algorithm. Missing values are allowed and the panel may be unbalanced.

Usage

```
gbmt(x.names, unit, time, ng=1, d=2, data, scaling=2, pruning=TRUE, nstart=NULL,
  tol=1e-4, maxit=1000, quiet=FALSE)
```

Arguments

x.names	Character vector including the names of the indicators.
unit	Character indicating the name of the variable identifying the units.
time	Character indicating the name of the variable identifying the time points.
ng	Positive integer value indicating the number of groups to create. Default is 1.
d	Positive integer value indicating the polynomial degree of group trajectories. Default is 2.
data	Object of class data.frame containing the variables indicated in arguments x.names, unit and time. The variable indicated in argument unit must be of type 'character' or 'factor' and cannot contain missing values. The variable indicated in argument time must be of type 'numeric' or 'Date' and cannot contain missing values. Variables indicated in argument x.names must be of type 'numeric' and may contain missing values. Variables indicated in argument x.names which are completely missing or not present in data will be ignored. The time points may differ across units (unbalanced panel) but they must be unique within units.
scaling	Normalisation method, that should be indicated as: 0 (no normalisation), 1 (centering), 2 (standardization), 3 (ratio to the mean) and 4 (logarithmic ratio to the mean). Default is 2 (standardization). See 'Details'.
pruning	Logical value indicating whether non-significant polynomial terms should be dropped. Default is TRUE. See 'Details'.
nstart	Positive integer value indicating the number of random restarts of the EM algorithm. If NULL (the default), the EM algorithm is started from the solution of a hierarchical cluster with Ward's linkage.
tol	Positive value indicating the tolerance of the EM algorithm. Default is 1e-4.
maxit	Positive integer value indicating the maximum number of iterations of the EM algorithm. Default is 1000.
quiet	Logical value indicating whether prompt messages should be suppressed. Default is FALSE.

Details

Let $Y_1, \ldots, Y_k, \ldots, Y_K$ be the considered indicators and $\mathrm{y}_{i,t} = (y_{i,t,1}, \ldots, y_{i,t,k}, \ldots, y_{i,t,K})'$ denote their observation on unit i ($i = 1, \ldots, n$) at time t ($t = 1, \ldots, T$). Also, let $\bar{y}_{i,k}$ and $s_{i,k}$ be, respectively, sample mean and sample standard deviation of indicator Y_k for unit i across the whole period of observation. Each indicator is normalized within units according to one among the following normalisation methods:

0) no normalisation:

$$y^*_{i,t,k} = y_{i,t,k}$$

1) centering:

$$y^*_{i,t,k} = y_{i,t,k} - \bar{y}_{i,k}$$

2) standardization:

$$y^*_{i,t,k} = \frac{y_{i,t,k} - \bar{y}_{i,k}}{s_{i,k}}$$

3) ratio to the mean:

$$y^*_{i,t,k} = \frac{y_{i,t,k}}{\bar{y}_{i,k}}$$

4) logarithmic ratio to the mean:

$$y^*_{i,t,k} = \log\left(\frac{y_{i,t,k}}{\bar{y}_{i,k}}\right) \approx \frac{y_{i,t,k} - \bar{y}_{i,k}}{\bar{y}_{i,k}}$$

Normalisation is required if the trajectories have different levels across units. When indicators have different scales of measurement, standardization is needed to compare the measurements of different indicators. Ratio to the mean and logaritmic ratio to the mean allow comparisons among different indicators as well, but they can be applied only in case of strictly positive measurements.

Denote the hypothesized groups as $j = 1, \ldots, J$ and let G_i be a latent variable taking value j if unit i belongs to group j. A group-based multivariate trajectory model with polynomial degree d is defined as:

$$\mathrm{y}^*_{i,t} \mid G_i = j \sim \mathrm{MVN}\left(\mu_j, \Sigma_j\right) \qquad j = 1, \ldots, J$$

$$\mu_j = \mathrm{B}'_j \left(1, t, t^2, \ldots, t^d\right)'$$

where B_j is the $(d+1) \times K$ matrix of regression coefficients in group j, and Σ_j is the $K \times K$ covariance matrix of the indicators in group j. The likelihood of the model is:

$$\mathcal{L}(\mathrm{B}_1, \ldots, \mathrm{B}_J, \Sigma_1, \ldots, \Sigma_J, \pi_1, \ldots, \pi_J) = \prod_{i=1}^{n} \left[\sum_{j=1}^{J} \pi_j \prod_{t=1}^{T} \phi(\mathrm{y}^*_{i,t} \mid \mathrm{B}_j, \Sigma_j) \right]$$

where $\phi(\mathrm{y}^*_{i,t} \mid \mathrm{B}_j, \Sigma_j)$ is the multivariate Normal density of $\mathrm{y}^*_{i,t}$ in group j, and π_j is the prior probability of group j. The posterior probability of group j for unit i is computed as:

$$\Pr(G_i = j \mid \mathrm{y}^*_i) \equiv \pi_{i,j} = \frac{\widehat{\pi}_j \prod_{t=1}^{T} \phi(\mathrm{y}^*_{i,t} \mid \widehat{\mathrm{B}}_j, \widehat{\Sigma}_j)}{\sum_{j=1}^{J} \widehat{\pi}_j \prod_{t=1}^{T} \phi(\mathrm{y}^*_{i,t} \mid \widehat{\mathrm{B}}_j, \widehat{\Sigma}_j)}$$

where the hat symbol above a parameter denotes the estimated value for that parameter. See the vignette of the package and Magrini (2022) for details on maximum likelihood estimation through the EM algorithm.

S3 methods available for class `gbmt` include:

- `print`: to see the estimated regression coefficients for each group;
- `summary`: to obtain the summary of the linear regressions (a list with one component for each group and each indicator);
- `plot`: to display estimated and predicted trajectories. See plot.gbmt for details;
- `coef`: to see the estimated coefficients (a list with one component for each group);
- `fitted`: to obtain the fitted values, equating to the estimated group trajectories (a list with one component for each group);
- `residuals`: to obtain the residuals (a list with one component for each group);
- `predict`: to perform prediction on trajectories. See predict.gbmt for details.
- `logLik`: to get the log likelihood;
- `AIC`, `extractAIC`: to get the Akaike information criterion;
- `BIC`: to get the Bayesian information criterion.

Value

An object of class gbmt, including the following components:

- call: list including details on the call.
- prior: vector including the estimated prior probabilities.
- beta: list of matrices, one for each group, including the estimated regression coefficients.
- Sigma: list of matrices, one for each group, including the estimated covariance matrix of the indicators.
- posterior: matrix including posterior probabilities.
- Xmat: the model matrix employed for each indicator in each group.
- fitted: list of matrices, one for each group, including the estimated group trajectories.
- reg: list of objects of class lm, one for each group and each indicator, including the fitted regressions.
- assign: vector indicating the assignement of the units to the groups.
- assign.list: list indicating the assignement of the units to the groups.
- logLik: log-likelihood of the model.
- npar: total number of free parameters in the model.
- ic: information criteria for the model (see Magrini, 2022 for details.
- appa: average posterior probability of assignments (APPA) for the model.
- data.orig: data provided to argument data.
- data.scaled: data after normalization.
- data.imputed: data after imputation of missing values, equal to data.orig if there are no missing data.
- em: matrix with one row for each run of the EM algorithm, including log-likelihood, number of iterations and convergence status (1=yes, 0=no).

References

A. Magrini (2022). Assessment of agricultural sustainability in European Union countries: A group-based multivariate trajectory approach. *Advances in Statistical Analysis*, published online: March 2022. DOI: 10.1007/s10182-022-00437-9

See Also

plot.gbmt; predict.gbmt.

Examples

```
data(agrisus2)

# names of indicators (just a subset for illustration)
varNames <- c("TFP_2005", "NetCapital_GVA",
  "Income_rur", "Unempl_rur", "GHG_UAA", "GNB_N_UAA")
```

```
# model with 2 degrees and 3 groups using the imputed dataset
# - log ratio to the mean is used as normalisation (scaling=4), thus values
#   represent relative changes with respect to country averages (see Magrini, 2022)
# - by default, standardization (scaling=2) is used.
m3_2 <- gbmt(x.names=varNames, unit="Country", time="Year", d=2, ng=3, data=agrisus2, scaling=4)

## NOT RUN: same model with multiple random restarts
#m3_2r <- gbmt(x.names=varNames, unit="Country", time="Year", d=2, ng=3, data=agrisus2,
#  scaling=4, nstart=10)

# resulting groups
m3_2$assign.list

# estimated group trajectories
m3_2$fitted

# summary of regressions by group
summary(m3_2)

# fit a model with 4 groups
m4_2 <- gbmt(x.names=varNames, unit="Country", time="Year", d=2, ng=4, data=agrisus2,
  scaling=4)
rbind(m3_2$ic, m4_2$ic)  ## comparison
```

`plot.gbmt` *Graphics for a group-based multivariate trajectory model*

Description

Visualization of estimated and predicted trajectories.

Usage

```
## S3 method for class 'gbmt'
plot(x, group=NULL, unit=NULL, x.names=NULL, n.ahead=0, bands=TRUE, conf=0.95,
 observed=TRUE, equal.scale=FALSE, trim=0, ylim=NULL, xlab="", ylab="", titles=NULL,
  add.grid=TRUE, col=NULL, transparency=-1, add.legend=TRUE, pos.legend=c(0,0),
  cex.legend=0.6, mar=c(5.1,4.1,4.1,2.1), ...)
```

Arguments

`x`	Object of class `gbmt`.
`group`	Numerical value indicating the group for which the estimated trajectories should be displayed. If `NULL` (the default), the estimated trajectories for each group will be overlapped. Ignored if `unit` is not `NULL`.
`unit`	Character indicating the name of the unit for which estimated trajectories should be displayed. If `NULL` (the default), estimated group trajectories are displayed.

x.names	Character including the names of the indicators for which the estimated trajectory should be displayed. If NULL (the default), estimated trajectories of all indicators are displayed.
n.ahead	Non-negative integer value indicating the number of steps ahead to perform prediction. Default is 0, meaning no prediction.
bands	Logical value indicating whether the prediction bands for should be drawn. Default is TRUE.
conf	Numerical value indicating the confidence level for the prediction bands. Default is 0.05. Ignored if bands is FALSE.
observed	Logical indicating whether observed trajectories should be drawn. Default is TRUE. Ignored if both group and unit are NULL.
equal.scale	Logical indicating whether indicators should have the same scale across all groups. Default is FALSE. Ignored if ylim is not NULL or if unit is not NULL.
trim	Numerical value indicating the proportion of extreme values to trim when either equal.scale is TRUE. Ignored if observed is FALSE or both group and unit are NULL. Default is 0, meaning no trim.
ylim	vector of length 2 indicating the limits of the y-axis, which will be applied to all indicators. If NULL (the default), it will be determined independently for each indicator based on data, unless equal.scale is TRUE.
xlab	label for the x-axis, which will be applied to all indicators. Default is empty string.
ylab	label for the y-axis, which will be applied to all indicators. Default is empty strings.
titles	vector of titles for the indicators. If NULL, the name of the indicators is used as title.
add.grid	Logical value indicating whether the grid should be added. Default is TRUE.
col	Character or numerical vector indicating the color of group trajectories. If group is not NULL, only the first valid color is considered. If group is NULL and there are more than ng valid colors, only the first ng valid colors are considered, otherwise valid colors are recycled to achieve a total number equal to ng. If NULL (the default), colors of group trajectories will be determined automatically.
transparency	Numerical value between 0 and 100 indicating the trasparency of prediction regions. Value 75 is suggested. If negative, only prediction bands are displayed. Default is -1. Ignored if group is not NULL or bands is FALSE.
add.legend	Logical value indicating whether the legend for groups should be added. Default is TRUE.
pos.legend	Numerical vector of length 2 indicating the horizontal-vertical shift of the legend for groups with respect to the position 'topleft'. Default is c(0,0). Ignored if group is not NULL or add.legend is FALSE.
cex.legend	Expansion factor relative to the legend for groups. Default is 0.6. Ignored if group is not NULL or add.legend is FALSE.
mar	Numerical vector of length 4 indicating the margin size in the order: bottom, left, top, right, which will be applied to all indicators. Default is c(5.1,4.1,4.1,2.1).
...	Further graphical parameters.

Value

No return value.

Note

If `unit` is not `NULL`, values are back transformed to the original scales of indicators.

See Also

gbmt.

Examples

```
data(agrisus2)

# names of indicators (just a subset for illustration)
varNames <- c("TFP_2005", "NetCapital_GVA",
  "Income_rur", "Unempl_rur", "GHG_UAA", "GNB_N_UAA")

# model with 2 polynomial degrees and 3 groups
m3_2 <- gbmt(x.names=varNames, unit="Country", time="Year", d=2, ng=3, data=agrisus2, scaling=4)

# group trajectories including 3 steps ahead prediction
mar0 <- c(3.1,2.55,3.1,1.2)
plot(m3_2, n.ahead=3, mar=mar0)  ## overlapped groups
plot(m3_2, group=1, n.ahead=3, mar=mar0)  ## group 1
plot(m3_2, group=2, n.ahead=3, mar=mar0)  ## group 2
plot(m3_2, group=3, n.ahead=3, mar=mar0)  ## group 3

# same scale to ease comparisons
plot(m3_2, n.ahead=3, mar=mar0, equal.scale=TRUE)
plot(m3_2, group=1, n.ahead=3, mar=mar0, equal.scale=TRUE, trim=0.05)
plot(m3_2, group=2, n.ahead=3, mar=mar0, equal.scale=TRUE, trim=0.05)
plot(m3_2, group=3, n.ahead=3, mar=mar0, equal.scale=TRUE, trim=0.05)

# overlapped groups with transparency
plot(m3_2, group=1, n.ahead=3, mar=mar0, equal.scale=TRUE, trim=0.05,
  transparency=80)

# trajectories including 3 steps ahead prediction for unit 'Italy'
plot(m3_2, unit="Italy", n.ahead=3, transparency=80)
```

---

posterior — *Posterior probabilities based on a group-based multivariate trajectory model*

Description

Computation of posterior probabilities for new units.

Usage

```
posterior(x, newdata=NULL)
```

Arguments

x	Object of class gbmt.
newdata	Object of class data.frame containing the multivariate time series of the indicators for the new units. If NULL (the default), posterior probabilities of the sample units are returned. If newdata is not NULL, it must include the variable identifying the time points. If newdata does not include the variable identifying the units, it is assumed that all observations refer to the same unit.

Value

An object of class data.frame with one entry for each unit, containing the posterior probability of each group for that unit.

Note

Data in newdata must be expressed on the original scale of the indicators. Normalisation is applied internally.

See Also

gbmt.

Examples

```
data(agrisus2)

# names of indicators (just a subset for illustration)
varNames <- c("TFP_2005", "NetCapital_GVA",
  "Income_rur", "Unempl_rur", "GHG_UAA", "GNB_N_UAA")

# model with 2 polynomial degrees and 3 groups
m3_2 <- gbmt(x.names=varNames, unit="Country", time="Year", d=2, ng=3, data=agrisus2, scaling=4)

# pretend that 'Italy' is a new unit
posterior(m3_2, newdata=agrisus2[which(agrisus2$Country=="Italy"),])

# consider only the last 3 years
posterior(m3_2, newdata=
  agrisus2[which(agrisus2$Country=="Italy"&agrisus2$Year>=2016),]
  )

# provide more than one new unit
posterior(m3_2, newdata=
  agrisus2[which(agrisus2$Country%in%c("Italy","Austria","Greece")),]
  )
```

predict.gbmt *Prediction based on a group-based multivariate trajectory model*

Description

Computation of in-sample and/or out-of-sample prediction of trajectories.

Usage

```
## S3 method for class 'gbmt'
predict(object, unit=NULL, n.ahead=0, bands=TRUE, conf=0.95, in.sample=FALSE, ...)
```

Arguments

object	Object of class gbmt.
unit	Character indicating the name of the unit for which prediction should be performed. If NULL (the default), group trajectories are predicted.
n.ahead	Non-negative integer value indicating the number of steps ahead for prediction. If a numerical vector is provided, only the maximum value is considered. If 0 (the default), in-sample prediction is returned.
bands	Logical value indicating whether the prediction bands should be computed.
conf	Numerical value indicating the confidence level for the prediction bands. Default is 0.05. Ignored if bands is FALSE.
in.sample	Logical value indicating whether in-sample prediction should be returned along with out-of-sample one. If FALSE (the default) and n.ahead is greater than 0, out-of-sample prediction is returned. Ignored if n.ahead is 0.
...	Further arguments for the generic predict method.

Value

If unit is NULL, a list with one component for each group, including a list with one object of class data.frame for each indicator. Otherwise, a list with one object of class data.frame for each indicator. Each of these dataframes has one column containing point predictions if bands=FALSE, otherwise three columns containing point predictions and their respective predictive bands.

Note

If unit is not NULL, values are back transformed to the original scales of indicators.

See Also

gbmt.

Examples

```
data(agrisus2)

# names of indicators (just a subset for illustration)
varNames <- c("TFP_2005", "NetCapital_GVA",
  "Income_rur", "Unempl_rur", "GHG_UAA", "GNB_N_UAA")

# model with 2 polynomial degrees and 3 groups
m3_2 <- gbmt(x.names=varNames, unit="Country", time="Year", d=2, ng=3, data=agrisus2, scaling=4)

# 3 steps ahead prediction of group trajectories
predict(m3_2, n.ahead=3)
predict(m3_2, n.ahead=3, in.sample=TRUE)  ## include in-sample prediction

# 3 steps ahead prediction for unit 'Italy'
predict(m3_2, unit="Italy", n.ahead=3)
predict(m3_2, unit="Italy", n.ahead=3, in.sample=TRUE)  ## include in-sample prediction
```

Index