Package ‘DIMORA’

January 20, 2025

Date 2023-10-07

Type Package

Title Diffusion Models R Analysis

Version 0.3.6

Author Zanghi Federico, Savio Andrea, Filippo Ziliotto, Bessi Alessandro

Depends R (>= 3.2.1), minpack.lm, numDeriv, stats, forecast, reshape2, deSolve

Maintainer Savio Andrea `<svandr97@gmail.com>`

Description
The implemented methods are: Standard Bass model, Generalized Bass model (with rectangular shock, exponential shock, and mixed shock. You can choose to add from 1 to 3 shocks), Guseo-Guidolin model and Variable Potential Market model, and UCRCD model. The Bass model consists of a simple differential equation that describes the process of how new products get adopted in a population, the Generalized Bass model is a generalization of the Bass model in which there is a ``carrier'' function x(t) that allows to change the speed of time sliding. In some real processes the reachable potential of the resource available in a temporal instant may appear to be not constant over time, because of this we use Variable Potential Market model, in which the Guseo-Guidolin has a particular specification for the market function. The UCRCD model (Unbalanced Competition and Regime Change Diachronic) is a diffusion model used to capture the dynamics of the competitive or collaborative transition.

Encoding UTF-8

License GPL (>= 3)

RoxygenNote 6.1.0.9000

NeedsCompilation no

Repository CRAN

Date/Publication 2023-10-07 12:00:02 UTC

Contents

`BM` *Standard Bass model*

Description

Function that estimates the standard Bass model. Fitted values for cumulative and instantaneous data are displayed (if `display = T`). Out-of-sample prediction is performed based on estimated parameters.

Usage

```
BM(series, method = "nls", prelimestimates = c(sum(series) + 100, 0.01, 0.1),
   oos = round(length(series)*0.25), alpha = 0.05 ,display = T)
```

Arguments

`series`	the instantaneous observed data.
`method`	the estimation method, 'nlm' or 'optim' (see `Details`).
`prelimestimates`	a vector containing the starting values used by the algorithm to estimate the parameters. If no values are specified, the default ones are: • market potential: $m = \sum_t (series) + 100$; • innovation coefficient: $p = 0.01$; • imitation coefficient: $q = 0.1$.
`alpha`	the significance level for the confidence intervals.
`oos`	positive integer value: number of predictions after the last observed one. Default setting to 25% of the length of the data.
`display`	if `TRUE` returns the fitted values for cumulative and instantaneous observed data. If 'oos' is specified, it also returns the predicted fit values.

Details

The `optim` method provides only the parameter estimates. It does not provide the standard error and the p-value estimates.

Value

`BM` returns an object of class "`Dimora`".

The function `summary` is used to obtain and print a summary table of the results. The generic accessor functions `coefficients`, `fitted` and `residuals` extract various useful features of the value returned by `BM`.

An object of class "`Dimora`" is a list containing at least the following components:

`model`	the model formula used.
`type`	the model frame used.
`Estimate`	a summary table of estimates.
`coefficients`	a named vector of coefficients.
`Rsquared`	the statistical measure R-squared.
`RSS`	the residual sum of squares.
`residuals`	the residuals (observed cumulative data - fitted cumulative data).
`fitted`	the cumulative fitted values.
`data`	the cumulative observed series.
`call`	the matched call.

Author(s)

- Zanghi Federico: `<federico.zanghi.11@gmail.com>`
  - `https://www.linkedin.com/in/federico-zanghi-75512b165/`
  - `https://github.com/federicozanghi`
- Savio Andrea: `<svandr97@gmail.com>`
  - `https://www.linkedin.com/in/andrea-savio-51861b191/`
- Ziliotto Filippo: `<filippo.ziliotto1996@gmail.com>`
  - `https://www.linkedin.com/in/FilippoZiliottoDev/`
- Bessi Alessandro: `<alessandrobessi92@gmail.com>`
  - `https://www.linkedin.com/in/alessandro-bessi-419812194/`

References

Guidolin, M. (2023). Innovation Diffusion Models: Theory and Practice, First Edition. John Wiley & Sons Ltd.

Bass, F.M. (1969). A new product growth for model consumer durables. Management science, 15 (5), 215-227.

See Also

The Dimora models: `GBM`, `GGM`, `UCRCD`.

`summary.Dimora` for summaries.

`plot.Dimora` for graphics and residuals analysis.

`predict.Dimora` for prediction.

`make.instantaneous` to create instantaneous series from the cumulative one.

Examples

```
data(DBdimora)
iphone <- DBdimora$iPhone[7:52]

## Example 1
M1 <- BM(iphone)
summary(M1)
plot.Dimora(M1)
plot.Dimora(M1, oos=25)
# 25 predictions


## Example 2
M2 <- BM(iphone, prelimestimates = c(2000, 0.001, 0.1), method = "optim", oos = 100)
summary(M2)
```

`DBdimora` *Database DIMORA*

Description

New DIMORA dataset which refers to the examples reported on the help files of the functions.

Details

The database contains six time series with the relative time variables: sales units sold for Apple iPhone and iMac, and the annual consumption, in Exajoule, of Denmark gas and renewables and Australian coal and renewables.

Author(s)

- Zanghi Federico: <`federico.zanghi.11@gmail.com`>
    - https://www.linkedin.com/in/federico-zanghi-75512b165/
    - https://github.com/federicozanghi
- Savio Andrea: <`svandr97@gmail.com`>
    - https://www.linkedin.com/in/andrea-savio-51861b191/
- Ziliotto Filippo: <`filippo.ziliotto1996@gmail.com`>
    - https://www.linkedin.com/in/FilippoZiliottoDev/
- Bessi Alessandro: <`alessandrobessi92@gmail.com`>
    - https://www.linkedin.com/in/alessandro-bessi-419812194/

See Also

The Dimora models: `BM`, `GBM`, `GGM`, `UCRCD`.

`summary.Dimora` for summaries.

`plot.Dimora` for graphics and residuals analysis.

`predict.Dimora` for prediction.

`make.instantaneous` to create instantaneous series from the cumulative one.

`DIMORA` *DIMORA Package*

Description

The main functions of the package are `BM`, `GBM`, `GGM`, and `UCRCD`. See the full documentation for more details. At the end of this page there are links to the functions.

Note 1: the names of the model functions have been changed! `BM` refers to the previous `BASS.standard`, `GMB` to the `BASS.generalized`, and `GGM` to the `GG.model`.

Note 2: the SARMAX.refinement has been deleted from the previous version. Function `Arima()` of the `forecast` library works to implement the SARMAX refinement.

Note 3: there is a new DIMORA dataset (DBdimora) which refers to the examples reported on the help files of the functions.

Author(s)

- Zanghi Federico: `<federico.zanghi.11@gmail.com>`
  - `https://www.linkedin.com/in/federico-zanghi-75512b165/`
  - `https://github.com/federicozanghi`
- Savio Andrea: `<svandr97@gmail.com>`
  - `https://www.linkedin.com/in/andrea-savio-51861b191/`
- Ziliotto Filippo: `<filippo.ziliotto1996@gmail.com>`
  - `https://www.linkedin.com/in/FilippoZiliottoDev/`
- Bessi Alessandro: `<alessandrobessi92@gmail.com>`
  - `https://www.linkedin.com/in/alessandro-bessi-419812194/`

See Also

The Dimora models: `BM`, `GBM`, `GGM`, `UCRCD`.

`summary.Dimora` for summaries.

`plot.Dimora` for graphics and residuals analysis.

`predict.Dimora` for prediction.

`make.instantaneous` to create instantaneous series from the cumulative one.

`GBM` *Generalized Bass model*

Description

Function that estimates the Generalized Bass model with Exponential, Rectangular, or Mixed shock. Fitted values for cumulative and instantaneous data are displayed (if `display = T`). Out-of-sample prediction is performed based on estimated parameters.

Usage

```
GBM(series, shock = c("exp", "rett", "mixed"), nshock,
                    prelimestimates, alpha = 0.05,
                    oos = round(length(series)*0.25), display = T)
```

Arguments

`series`	the instantaneous observed data.
`shock`	the parameters which define the shocks. The available options are: • `"exp"` = `Exponential`; • `"rett"` = `Rectangular`; • `"mixed"` = `Mixed` (only when nshock = 2).
`nshock`	the number of shocks (from 1 to 3).
`prelimestimates`	a vector containing the preliminary estimates of the parameters (see `Details` and `Examples`).
`alpha`	the significance level for confidence intervals.
`oos`	positive integer value: number of predictions after the last observed one. Default setting to 25% of the length of the data.
`display`	if `TRUE` returns the fitted values for cumulative and instantaneous observed data. If 'oos' is specified, it also returns the predicted fit values.

Details

Each type of shock is characterized by specific parameters.

The analyst has to set both the preliminary estimates of the parameters `m, p, q` and the ones related to the shock(s) `a, b, c`. The parameters related to each shock have to be defined as follows:

- Exponential:
  - `a` = starting time of the shock
  - `b` = memory of the effect (typically negative, suggesting an exponentially decaying behavior)
  - `c` = intensity of the shock (may be either positive or negative)

In case of more than one shock, preliminary estimates need to be specified as follows: `prelimestimates = c(m, p, q, a1, b1, c1, a2, b2, c2, a3, b3, c3)`

- Rectangular:
  - `a` = starting time of the shock
  - `b` = ending time of the shock
  - `c` = intensity of the shock (may be either positive or negative)

  In case of more than one shock, preliminary estimates need to be specified as follows: `prelimestimates = c(m, p, q, a1, b1, c1, a2, b2, c2, a3, b3, c3)`
- Mixed: when the series has one exponential and one rectangular shock (you always have to specify the exponential shock before the rectangular one, even if the exponential one occurs later)
  - `a1` = starting time of the exponential shock
  - `b1` = memory of the effect (typically negative, suggesting an exponentially decaying behavior)
  - `c1` = intensity of the exponential shock (may be either positive or negative)
  - `a2` = starting time of the rectangular shock
  - `b2` = ending time of the rectangular shock
  - `c2` = intensity of the rectangular shock (may be either positive or negative)

Value

`GBM` returns an object of class `"Dimora"`.

The function `summary` is used to obtain and print a summary table of the results. The generic accessor functions `coefficients`, `fitted` and `residuals` extract various useful features of the value returned by `GBM`.

An object of class `"Dimora"` is a list containing at least the following components:

`model`	the model formula used.
`type`	the model frame used.
`Estimate`	a summary table of estimates.
`coefficients`	a named vector of coefficients.
`Rsquared`	the statistical measure R-squared.
`RSS`	the residual sum of squares.
`residuals`	the residuals (observed cumulative data - fitted cumulative data).
`fitted`	the cumulative fitted values.
`data`	the cumulative observed series.
`call`	the matched call.

Author(s)

- Zanghi Federico: <federico.zanghi.11@gmail.com>
  - https://www.linkedin.com/in/federico-zanghi-75512b165/
  - https://github.com/federicozanghi
- Savio Andrea: <svandr97@gmail.com>
  - https://www.linkedin.com/in/andrea-savio-51861b191/
- Ziliotto Filippo: <filippo.ziliotto1996@gmail.com>
  - https://www.linkedin.com/in/FilippoZiliottoDev/
- Bessi Alessandro: <alessandrobessi92@gmail.com>
  - https://www.linkedin.com/in/alessandro-bessi-419812194/

References

Guidolin, M. (2023). Innovation Diffusion Models: Theory and Practice, First Edition. John Wiley & Sons Ltd.

Bass, F.M., Krishnan, T.V., & Jain, D.C. (1994). Why the Bass model fits without decision variables. Marketing science, 13 (3), 203-223.

See Also

The Dimora models: `BM`, `GGM`, `UCRCD`.

`summary.Dimora` for summaries.

`plot.Dimora` for graphics and residuals analysis.

`predict.Dimora` for prediction.

`make.instantaneous` to create instantaneous series from the cumulative one.

Examples

```
data(DBdimora)
iphone<- DBdimora$iPhone[7:52]
imac<- DBdimora$iMac[1:52]

## Example 1: exponential shock
M3 <- GBM(iphone, shock = "exp", nshock = 1,
          prelimestimates = c(BM(iphone, display=FALSE)$Estimate[1,1],
                              BM(iphone, display=FALSE)$Estimate[2,1],
                              BM(iphone, display=FALSE)$Estimate[3,1],
                              17,-0.1,0.1))
summary(M3)
plot.Dimora(M3, oos=25)
# 25 predictions

## Example 2: rectangular shock
M4 <- GBM(imac,shock = "rett",nshock = 1,
           prelimestimates = c(BM(imac, display=FALSE)$Estimate[1,1],
                               BM(imac, display=FALSE)$Estimate[2,1],
```

```
                                BM(imac, display=FALSE)$Estimate[3,1],
                                20,30,0.1), oos=20)
summary(M4)

## Example 3: mixed shock
## The prelimestimates of m, p, q are their relative values estimated through M4.

M5 <- GBM(imac,shock = "mixed",nshock = 2,
           prelimestimates = c(M4$Estimate[1,1],
                                M4$Estimate[2,1],
                                M4$Estimate[3,1],
                                6,-0.1,0.1, 20,30,0.1), oos=0)
summary(M5)
```

`GGM` *Guseo-Guidolin model*

Description

Function that estimates the dynamic market potential model. Fitted values for cumulative and instantaneous data are displayed (if `display = T`). Out-of-sample prediction is performed based on estimated parameters. Function `m(t)` is defined, as default, as the standard Guseo-Guidolin model, but it can also be defined as a general function with some constraints (see `Details` and `exmaples`).

Usage

```
GGM(series, prelimestimates = NULL, mt = 'base', alpha = 0.05,
      oos = round(length(series)*0.25), display = T)
```

Arguments

`series`	the instantaneous observed data.
`prelimestimates`	a vector containing the preliminary estimates of the parameters (see `Details` and `Examples`).
`mt`	function that define the dynamic market potential.
`alpha`	the significance level for the confidence intervals.
`oos`	positive integer value: number of predictions after the last observed one. Default setting to 25% of the length of the data.
`display`	if `TRUE` returns the fitted values for cumulative and instantaneous observed data. If 'oos' is specified, it also returns the predicted fit values.

Details

The `GGM` function allows to define the parameter `m(t)` in two ways:

- use the standard GGM model. Note: define the `prelimestimates = c(K, qc, pc, qs, ps)` (see Example 1 below).
- define function `m(t)` according to modeling needs. Note: function `m(t)` must be positive and is essential a c.d.f.; default values for prelimestimates are based on the BM function (see Example 2 below).

Value

`GGM` returns an object of class `"Dimora"`. The function `summary` is used to obtain and print a summary table of the results. The generic accessor functions `coefficients`, `fitted` and `residuals` extract various useful features of the value returned by `GGM`.

An object of class `"Dimora"` is a list containing at least the following components:

`model`	the model formula used.
`type`	the model frame used.
`Estimate`	a summary table of estimates.
`coefficients`	a named vector of coefficients.
`Rsquared`	the statistical measure R-squared.
`RSS`	the residual sum of squares.
`residuals`	the residuals (observed cumulative data - fitted cumulative data).
`fitted`	the cumulative fitted values.
`data`	the cumulative observed series.
`call`	the matched call.

Author(s)

- Zanghi Federico: <`federico.zanghi.11@gmail.com`>
    - `https://www.linkedin.com/in/federico-zanghi-75512b165/`
    - `https://github.com/federicozanghi`
- Savio Andrea: <`svandr97@gmail.com`>
    - `https://www.linkedin.com/in/andrea-savio-51861b191/`
- Ziliotto Filippo: <`filippo.ziliotto1996@gmail.com`>
    - `https://www.linkedin.com/in/FilippoZiliottoDev/`
- Bessi Alessandro: <`alessandrobessi92@gmail.com`>
    - `https://www.linkedin.com/in/alessandro-bessi-419812194/`

References

Guidolin, M. (2023). Innovation Diffusion Models: Theory and Practice, First Edition. John Wiley & Sons Ltd.

Guseo, R., & Guidolin, M. (2009). Modelling a dynamic market potential: A class of automata networks for diffusion of innovations. Technological Forecasting and Social Change, 76(6), 806-820.

See Also

The Dimora models: `BM`, `GBM`, `UCRCD`.

`summary.Dimora` for summaries.

`plot.Dimora` for graphics and residuals analysis.

`predict.Dimora` for prediction.

`make.instantaneous` to create instantaneous series from the cumulative one.

Examples

```
data(DBdimora)
iphone <- DBdimora$iPhone[7:52]

## Example 1
M6 <- GGM(iphone, prelimestimates=c(1823, 0.001, 0.1, 0.001, 0.1), oos=100)
# 1823 refers to the estimated parameter m of the standard Bass model on iphone
summary(M6)
plot.Dimora(M6, oos=25)
# 25 predictions

## Example 2
M7 <- GGM(iphone, mt = function(x) pchisq(x,10))
summary(M7)
```

`make.instantaneous` *Function that transforms your data*

Description

Function that transforms cumulative data into instantaneous; this function can be useful because all the model functions in this package require the istantaneous data as input.

Usage

```
make.instantaneous(cumulate.data)
```

Arguments

`cumulate.data`	the cumulative data.

Author(s)

- Zanghi Federico: `<federico.zanghi.11@gmail.com>`
  - `https://www.linkedin.com/in/federico-zanghi-75512b165/`
  - `https://github.com/federicozanghi`
- Savio Andrea: `<svandr97@gmail.com>`
  - `https://www.linkedin.com/in/andrea-savio-51861b191/`

- Ziliotto Filippo: `<filippo.ziliotto1996@gmail.com>`
    - `https://www.linkedin.com/in/FilippoZiliottoDev/`
- Bessi Alessandro: `<alessandrobessi92@gmail.com>`
    - `https://www.linkedin.com/in/alessandro-bessi-419812194/`

See Also

The Dimora models: `BM`, `GBM`, `GGM`, `UCRCD`.

`summary.Dimora` for summaries.

`plot.Dimora` for graphics and residuals analysis.

`predict.Dimora` for prediction.

Examples

```
cumulate.data = c(1,2,3,6,12)
data.inst = make.instantaneous(cumulate.data)
```

`plot.Dimora` *Plot method for Dimora models*

Description

Plot method for the DIMORA package models.

Usage

```
## S3 method for class 'Dimora'
plot(x,..., type = c('all','res','fit'), oos = 0)
```

Arguments

`x`	an object of class `Dimora`.
`type`	specifies the type of plot. Default setting is `'all'`: • `'fit'` returns the cumulative and instantaneous fit of the model; • `'res'` returns the 'Residuals VS Time' graphic and the ACF of the residuals of the model; • `'all'` returns the two types before combined in an unique graphical window.
`oos`	positive integer value: number of predictions after the last observed one. If specified, it also determines the xlim for the plots. It only works for univariate models.
`...`	other graphics parameters.

Author(s)

- Zanghi Federico: <federico.zanghi.11@gmail.com>
  - https://www.linkedin.com/in/federico-zanghi-75512b165/
  - https://github.com/federicozanghi
- Savio Andrea: <svandr97@gmail.com>
  - https://www.linkedin.com/in/andrea-savio-51861b191/
- Ziliotto Filippo: <filippo.ziliotto1996@gmail.com>
  - https://www.linkedin.com/in/FilippoZiliottoDev/
- Bessi Alessandro: <alessandrobessi92@gmail.com>
  - https://www.linkedin.com/in/alessandro-bessi-419812194/

See Also

The Dimora models: `BM`, `GBM`, `GGM`, `UCRCD`.

`summary.Dimora` for summaries.

`predict.Dimora` for prediction.

`make.instantaneous` to create instantaneous series from the cumulative one.

Examples

```
data(DBdimora)
iphone <- DBdimora$iPhone[7:52]
Gd <- DBdimora$Gdenmark[1:36]
Rd <- DBdimora$Rdenmark[13:36]

## Plot with univariate model
M1 <- BM(iphone,display = FALSE)
plot.Dimora(M1,type='all',oos = 20)

## Plot with multivariate model
M8 <- UCRCD(Gd,Rd)
plot.Dimora(M8,type='fit')
```

`predict.Dimora` *Predict method for Dimora objects*

Description

Prediction of test data using any model of the package.

Usage

```
## S3 method for class 'Dimora'
predict(object, ..., newx)
```

Arguments

`object`	an object of class `Dimora`, as that created by `BM`, `GBM` or `GGM`.
`newx`	a number or a vector of numeric values rapresenting the time window (see `Example` below).
`...`	additional arguments affecting the predictions produced.

Details

It works only for univariate models; specifically, for the models `BM`, `GBM`, `GGM`, it returns a vector of the cumulative predicted values.

Value

The method returns a vector of predicted values.

Author(s)

- Zanghi Federico: `<federico.zanghi.11@gmail.com>`
  - `https://www.linkedin.com/in/federico-zanghi-75512b165/`
  - `https://github.com/federicozanghi`
- Savio Andrea: `<svandr97@gmail.com>`
  - `https://www.linkedin.com/in/andrea-savio-51861b191/`
- Ziliotto Filippo: `<filippo.ziliotto1996@gmail.com>`
  - `https://www.linkedin.com/in/FilippoZiliottoDev/`
- Bessi Alessandro: `<alessandrobessi92@gmail.com>`
  - `https://www.linkedin.com/in/alessandro-bessi-419812194/`

See Also

The Dimora models: `BM`, `GBM`, `GGM`, `UCRCD`.

`summary.Dimora` for summaries.

`plot.Dimora` for graphics and residuals analysis.

`make.instantaneous` to create instantaneous series from the cumulative one.

Examples

```
data(DBdimora)
iphone <- DBdimora$iPhone[7:52]
M1 <- BM(iphone)

# Predict the values of the observed series (lenght(iphone)=46)
predict.Dimora(M1, newx = c(1:46))


# Predict the values of the observed series and other 34 future values
predict.Dimora(M1, newx = c(1:80))
```

`summary.Dimora` *Summary method for Dimora objects*

Description

Summary method for the `Dimora` objects.

Usage

```
## S3 method for class 'Dimora'
summary(object,...)
```

Arguments

`object`	an object of class `Dimora`, as that created by `BM`, `GBM`, `GGM` or `UCRCD`.
`...`	additional arguments affecting the summary produced.

Value

`summary` is a generic function used to produce summaries of the results of various model fitting functions. The function invokes particular methods which depend on the class of the first argument; in this case, the class is `Dimora`.

Author(s)

- Zanghi Federico: `<federico.zanghi.11@gmail.com>`
  - `https://www.linkedin.com/in/federico-zanghi-75512b165/`
  - `https://github.com/federicozanghi`
- Savio Andrea: `<svandr97@gmail.com>`
  - `https://www.linkedin.com/in/andrea-savio-51861b191/`
- Ziliotto Filippo: `<filippo.ziliotto1996@gmail.com>`
  - `https://www.linkedin.com/in/FilippoZiliottoDev/`
- Bessi Alessandro: `<alessandrobessi92@gmail.com>`
  - `https://www.linkedin.com/in/alessandro-bessi-419812194/`

See Also

The Dimora models: `BM`, `GBM`, `GGM`, `UCRCD`.

`plot.Dimora` for graphics and residuals analysis.

`predict.Dimora` for prediction.

`make.instantaneous` to create instantaneous series from the cumulative one.

Examples

```
data(DBdimora)
iphone <- DBdimora$iPhone[7:52]

M1 <- BM(iphone)
summary(M1)
```

`UCRCD` *UCRCD model*

Description

Function that estimates the Unbalanced Competition Regime Change Diachronic model. Fitted values for cumulative and instantaneous data are displayed (if `display = T`).

Usage

```
UCRCD(series1, series2, display=T, alpha=0.05,
      delta=0.01, gamma=0.01, par="double",
      m1  = BM(series1,display = F)$Estimate[1,1],
      m2  = BM(series2,display = F)$Estimate[1,1],
      p1c = BM(series1,display = F)$Estimate[2,1],
      q1c = BM(series1,display = F)$Estimate[3,1],
      p2  = BM(series2,display = F)$Estimate[2,1],
      q2  = BM(series2,display = F)$Estimate[3,1])
```

Arguments

`series1`	the istantaneous observed data of the first product.
`series2`	the istantaneous observed data of the second product.
`alpha`	the significance level for confidence intervals.
`par`	default is `'double'`. Instead `'unique'`, meaning the costraint delta equal to gamma has been selected.
`display`	if `TRUE` returns the predicted values for cumulative and instantaneous observed data.
`delta`	preliminary estimate of delta.
`gamma`	preliminary estimate of gamma.
`m1`	preliminary estimate of the first product's market potential under competition
`m2`	preliminary estimate of the second product's market potential
`p1c`	preliminary estimate of the first product's innovation coefficient under competition
`p2`	preliminary estimate of the second product's innovation coefficient
`q1c`	preliminary estimate of the first product's imitation coefficient under competition
`q2`	preliminary estimate of the second product's imitation coefficient

Details

The model estimates several parameters that refer both to the stand-alone phase and the competition phase. The description of the parameters is reported below: (wom = "Word Of Mouth")

- `ma` : Market Potential 1 (stand-alone phase)
- `p1a` : Innovation coefficient 1 (stand-alone phase)
- `q1a` : Imitation coefficient 1 (stand-alone phase)
- `mc` : Market Potential 1 (competition phase)
- `p1c` : Innovation coefficient 1 (competition phase)
- `q1c+delta` : Within-product wom 1 (competition phase)
- `q1c` : Cross-product wom 1 (competition phase)
- `p2` : Innovation coefficient 2
- `q2` : Within-product wom 2
- `q2-gamma` : Cross-product wom 2

Value

`UCRCD` returns an object of class "`Dimora`".

The function `summary` is used to obtain and print a summary table of the results. The generic accessor functions `coefficients`, `fitted` and `residuals` extract various useful features of the value returned by `UCRCD`.

An object of class "`Dimora`" is a list containing at least the following components:

`model`	the model formula used.
`type`	the model frame used.
`Estimate`	a summary table of estimates.
`coefficients`	a named vector of coefficients.
`Rsquared`	the statistical measure R-squared (on the istantaneous data).
`RSS`	the residual sum of squares (on the istantaneous data).
`residuals`	the residuals (observed cumulative data - fitted cumulative data).
`fitted`	the cumulative fitted values.
`data`	the cumlative observed data.
`call`	the matched call.

The UCRCD model is estimated on the instantaneous data. For this reason, the UCRCD function also returns the following elements:

`data.i`	the instantaneous observed data.
`fitted.i`	the instantaneous fitted values.
`residuals.i`	the residuals (observed instantaneous data - fitted instantaneous data).

Author(s)

- Zanghi Federico: `<federico.zanghi.11@gmail.com>`
  - `https://www.linkedin.com/in/federico-zanghi-75512b165/`
  - `https://github.com/federicozanghi`
- Savio Andrea: `<svandr97@gmail.com>`
  - `https://www.linkedin.com/in/andrea-savio-51861b191/`
- Ziliotto Filippo: `<filippo.ziliotto1996@gmail.com>`
  - `https://www.linkedin.com/in/FilippoZiliottoDev/`
- Bessi Alessandro: `<alessandrobessi92@gmail.com>`
  - `https://www.linkedin.com/in/alessandro-bessi-419812194/`

References

Guidolin, M. (2023). Innovation Diffusion Models: Theory and Practice, First Edition. John Wiley & Sons Ltd.

Guseo, R., & Mortarino, C. (2014). Within-brand and cross-brand word-of-mouth for sequential multi-innovation diffusions. IMA Journal of Management Mathematics, 25(3), 287-311.

See Also

The Dimora models: `BM`, `GBM`, `GGM`.

`summary.Dimora` for summaries.

`plot.Dimora` for graphics and residuals analysis.

`predict.Dimora` for prediction.

`make.instantaneous` to create instantaneous series from the cumulative one.

Examples

```
data(DBdimora)
Gd <- DBdimora$Gdenmark[1:36]
Rd <- DBdimora$Rdenmark[13:36]
Ca <- DBdimora$Caustralia
Ra <- DBdimora$Raustralia[26:56]

## Example 1
M8 <- UCRCD(Gd,Rd)
summary(M8)
plot.Dimora(M8,type="fit")

## Example 2
M9 <- UCRCD(Ca,Ra,par = "unique",display=FALSE)
summary(M9)
plot.Dimora(M9)
```

Index