

Package ‘dynamicalSystemsNotes’

June 29, 2023

Title Dynamical Systems Notes

Version 0.9.1

Description HDFS 538 Dynamical Systems Methods and Applications personal study notes.

URL <https://github.com/ijapesigan/dynamicalSystemsNotes>,
<https://ijapesigan.github.io/dynamicalSystemsNotes/>

BugReports <https://github.com/ijapesigan/dynamicalSystemsNotes/issues>

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

VignetteBuilder knitr

Depends R (>= 3.0.0)

Suggests knitr, rmarkdown, testthat

RoxygenNote 7.2.3

NeedsCompilation no

Author Ivan Jacob Agaloos Pesigan [aut, cre, cph]
(<https://orcid.org/0000-0003-4818-8420>)

Maintainer Ivan Jacob Agaloos Pesigan <ijapesigan@gmail.com>

R topics documented:

Cobweb	2
FixedPoint	2
LinearConstructor	3
LogisticConstructor	4
UnivSeries	5
Index	6

Cobweb

*Cobweb Plot***Description**

Cobweb Plot

Usage

```
Cobweb(y0, func, tol = sqrt(.Machine$double.eps), max_iter = 1000L)
```

Arguments

<code>y0</code>	Numeric. Initial condition.
<code>func</code>	Function. The input is $y_{i(t-1)}$ and the output is y_{it} .
<code>tol</code>	Small numeric value. Convergence criteria.
<code>max_iter</code>	Large positive integer. Maximum number of iterations.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# linear
func <- LinearConstructor(alpha = 8.0, beta = 0.8)
Cobweb(y0 = 0.01, func = func)

# logistic
func <- LogisticConstructor(r = 1.5, K = 10)
Cobweb(y0 = 0.01, func = func)
```

FixedPoint

*Fixed Point***Description**

Fixed Point

Usage

```
FixedPoint(y0, func, tol = sqrt(.Machine$double.eps), max_iter = 1000L)
```

Arguments

<code>y0</code>	Numeric. Initial condition.
<code>func</code>	Function. The input is $y_{i(t-1)}$ and the output is y_{it} .
<code>tol</code>	Small numeric value. Convergence criteria.
<code>max_iter</code>	Large positive integer. Maximum number of iterations.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# linear
func <- LinearConstructor(alpha = 8.0, beta = 0.8)
FixedPoint(y0 = 0.01, func = func)

# logistic
func <- LogisticConstructor(r = 1.5, K = 10)
FixedPoint(y0 = 0.01, func = func)
```

LinearConstructor	<i>Simple Linear System Function Constructor</i>
-------------------	--

Description

Simple Linear System Function Constructor

Usage

```
LinearConstructor(alpha, beta)
```

Arguments

<code>alpha</code>	Numeric. Intercept α .
<code>beta</code>	Numeric. Slope β .

Details

The simple linear system is given by

$$y_{it} = \alpha + \beta y_{i(t-1)}$$

where α is the intercept and β is the slope.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
func <- LinearConstructor(alpha = 8.0, beta = 0.8)
func(0.1)
```

LogisticConstructor	<i>Simple Logistic System Function Constructor</i>
---------------------	--

Description

Simple Logistic System Function Constructor

Usage

```
LogisticConstructor(r, K)
```

Arguments

<code>r</code>	Numeric. Growth rate.
<code>K</code>	Numeric Carrying capacity.

Details

The logistic system is given by

$$y_{it} = ry_{i(t-1)} \left(1 - \frac{y_{i(t-1)}}{K} \right)$$

where r is the growth rate and K is the carrying capacity.

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
func <- LogisticConstructor(r = 1.5, K = 10)
func(0.1)
```

UnivSeries

Univariate Time Series

Description

Univariate Time Series

Usage

```
UnivSeries(y0, func, nt)
```

Arguments

y0	Numeric. Initial condition.
func	Function. The input is $y_{i(t-1)}$ and the output is y_{it} .
nt	Positive integer. Maximum discrete time points.

Details

The univariate time series is generated using the following equation

$$y_{it} = f(y_{i(t-1)}).$$

Author(s)

Ivan Jacob Agaloos Pesigan

Examples

```
# linear
func <- LinearConstructor(alpha = 8.0, beta = 0.8)
y <- UnivSeries(y0 = 0.01, func = func, nt = 100)
plot(y)

# logistic
func <- LogisticConstructor(r = 1.5, K = 10)
y <- UnivSeries(y0 = 0.01, func = func, nt = 100)
plot(y)
```

Index

Cobweb, [2](#)

FixedPoint, [2](#)

LinearConstructor, [3](#)

LogisticConstructor, [4](#)

UnivSeries, [5](#)