Package 'easyreg'

September 13, 2019

Type Package

Title Easy Regression

Version 4.0

Date 2019-10-13

Author Emmanuel Arnhold

Maintainer Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

Description Performs analysis of regression in simple designs with quantitative treatments, including mixed models and non linear models.

Depends R (>= 3.0.0)

Imports nlme

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2019-09-13 13:30:02 UTC

R topics documented:

easyreg-package *Easy Regression*

Description

Performs analysis of regression in simple designs with quantitative treatments, including mixed models ans non linear models

Details

Package:	easyreg
Type:	Package
Version:	4.0
Date:	2019-10-13
License:	GPL (>= 2)

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

References

KAPS, M. and LAMBERSON, W. R. Biostatistics for Animal Science: an introductory text. 2nd Edition. CABI Publishing, Wallingford, Oxfordshire, UK, 2009. 504p.

SAMPAIO, I. B. M. Estatistica aplicada a experimentacao animal. 3nd Edition. Belo Horizonte: Editora FEPMVZ, Fundacao de Ensino e Pesquisa em Medicina Veterinaria e Zootecnia, 2010. 264p.

Examples

```
# analysis in completely randomized design
data(data1)
r1=er2(data1)
names(r1)
r1
r1[1]

# analysis in randomized block design
data(data2)
r2=er2(data2, design=2)
r2

# analysis in latin square design
data(data3)
```

```
r3=er2(data3, design=3)
r3

# analysis in several latin squares
data(data4)
r4=er2(data4, design=4)
r4

# the growth of Zagorje turkeys (Kaps and Lamberson, 2009)

weight=c(44,66,100,150,265,370,455,605,770)
age=c(1,7,14,21,28,35,42,49,56)

data2=data.frame(age,weight)

# two linear
regplot(data2, model=5, start=c(25,6,10,20))

regplot(data2, model=5, start=c(25,6,10,20), digits=2)

# in other function
bl(data2)
```

---

bl *Analysis of broken line regression*

Description

The function performs analysis of broken line regression

Usage

```
bl(data, model=1, alpha=0.05, xlab = "Explanatory Variable", ylab = "Response Variable",
    position = 1, digits = 6, mean = TRUE, sd=FALSE, legend = TRUE, lty=2,
col="dark blue", pch=20, xlim="default.x",ylim="default.y", ...)
```

Arguments

`data`	data is a data.frame The first column contain the treatments (explanatory variable) and the second column the response variable
`model`	model for analysis: 1=two linear; 2=linear plateau (LRP); 3= model 1 with blocks random; 4 = model 2 with blocks random
`alpha`	significant level for cofidence intervals (parameters estimated)
`xlab`	name of explanatory variable
`ylab`	name of response variable

`position`	position of equation in the graph top=1 bottomright=2 bottom=3 bottomleft=4 left=5 topleft=6 (default) topright=7 right=8 center=9
`digits`	number of digits (default=6)
`mean`	mean=TRUE (plot mean of data) mean=FALSE (plot all data)
`sd`	sd=FALSE (plot without standard deviation) sd=TRUE (plot with standard deviation)
`legend`	legend=TRUE (plot legend) legend=FALSE (not plot legend)
`lty`	line type
`col`	line color
`pch`	point type
`xlim`	limits for x
`ylim`	limits for y
`...`	others graphical parameters (see par)

Value

Returns coefficients of the models, t test for coefficients, knot (break point), R squared, adjusted R squared, AIC, BIC, residuals and shapiro-wilk test for residuals.

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

References

KAPS, M. and LAMBERSON, W. R. Biostatistics for Animal Science: an introductory text. 2nd Edition. CABI Publishing, Wallingford, Oxfordshire, UK, 2009. 504p.

See Also

lm, ea1(easyanova package), er1

Examples

```
# the growth of Zagorje turkeys (Kaps and Lamberson, 2009)

weight=c(44,66,100,150,265,370,455,605)
age=c(1,7,14,21,28,35,42,49)

data2=data.frame(age,weight)

# two linear
regplot(data2, model=5, start=c(25,6,10,20))

bl(data2, digits=2)


#linear and quadratic plateau
x=c(0,1,2,3,4,5,6)
y=c(1,2,3,6.1,5.9,6,6.1)

data=data.frame(x,y)

bl(data,model=2, lty=1, col=1, digits=2, position=8)


# effect os blocks
x=c(1,1,2,2,3,3,4,4,5,5,6,6,7,7,8,8)
y=c(4,12,9,20,16,25,21,31,28,42,33,46,33,46,34,44)
blocks=rep(c(1,2),8)

dat=data.frame(x,blocks,y)

bl(dat, 3)

bl(dat,4, sd=TRUE)

bl(dat,4, mean=FALSE)
```

---

data1 *data1: Sampaio (2010): page 134*

Description

Quantitative treatments in completely randomized design.

Usage

```
data(data1)
```

Format

A data frame with 24 observations on the following 2 variables.

`treatment` a numeric vector

`gain` a numeric vector

References

SAMPAIO, I. B. M. Estatistica aplicada a experimentacao animal. 3nd Edition. Belo Horizonte: Editora FEPMVZ, Fundacao de Ensino e Pesquisa em Medicina Veterinaria e Zootecnia, 2010. 264p.

Examples

```
data(data1)
summary(data1)
```

---

data2 *data2: Kaps and Lamberson (2009): page 434*

Description

Quantitative treatments in randomizad block design.

Usage

```
data(data2)
```

Format

A data frame with 25 observations on the following 3 variables.

`protein_level` a numeric vector

`litter` a factor with levels `l1` `l2` `l3` `l4` `l5`

`feed_conversion` a numeric vector

References

KAPS, M. and LAMBERSON, W. R. Biostatistics for Animal Science: an introductory text. 2nd Edition. CABI Publishing, Wallingford, Oxfordshire, UK, 2009. 504p.

Examples

```
data(data2)
summary(data2)
```

`data3` *data3: fictional example*

Description

Quantitative treatments in latin square design.

Usage

```
data(data3)
```

Format

A data frame with 25 observations on the following 4 variables.

`treatment` a numeric vector

`animal` a factor with levels `a1` `a2` `a3` `a4` `a5`

`period` a factor with levels `p1` `p2` `p3` `p4` `p5`

`milk_fat` a numeric vector

Examples

```
data(data3)
summary(data3)
```

`data4` *data4: fictional example*

Description

Quantitative treatments in several latin squares design.

Usage

```
data(data4)
```

Format

A data frame with 50 observations on the following 5 variables.

`treatment` a numeric vector

`square` a numeric vector

`animal` a factor with levels `a1` `a2` `a3` `a4` `a5`

`period` a factor with levels `p1` `p2` `p3` `p4` `p5`

`milk_fat` a numeric vector

Examples

```
data(data4)
summary(data4)
```

---

data5 *data5: fictional example*

Description

Quantitative treatments and three response variable.

Usage

```
data(data5)
```

Format

A data frame with 24 observations on the following 4 variables.

`treatments` a numeric vector

`variable1` a numeric vector

`variable2` a numeric vector

`variable3` a numeric vector

Examples

```
data(data5)
summary(data5)
```

---

er1 *Analysis of regression*

Description

The function performs analysis of some linear and nonlinear models

Usage

```
er1(data, model = 1, start = c(a = 1, b = 1, c = 1, d = 1, e = 1),
mixed=FALSE, digits=6, alpha=0.05)
```

Arguments

`data` data is a data.frame

The first column should contain the treatments (explanatory variable) and the remaining columns the response variables.

`model` define the model

1 = "y~a+b*x" linear

2 = "y~a+b*x+c*x^2" quadratic

3 = "y ~ a + b * (x - c) * (x <= c)" linear plateau

4 = "y ~ (a + b * x + c * I(x^2)) * (x <= -0.5 * b/c) + (a + I(-b^2/(4 * c))) * (x > -0.5 * b/c)" quadratic plateau

5 = "ifelse(x>=d,(a-c*d)+(b+c)*x, a+b*x)" two linear

6 = "y~a*exp(b*x)" exponential

7 = "y~a*(1+b*(exp(-c*x)))^-1" logistic

8 = "y~a*(1-b*(exp(-c*x)))^3" van bertalanffy

9 = "y~a*(1-b*(exp(-c*x)))" brody

10 = "y~a*exp(-b*exp(-c*x)" gompertz

11 = "y~(a*x^b)*exp(-c*x)" lactation curve

12 = "y ~ a + b * (1 - exp(-c * x))" ruminal degradation curve

13 = "y~(a/(1+exp(2-4*c*(x-e))))+(b/(1+exp(2-4*d*(x-e))))" logistic bi-compartmental

14 = "y~a*(x^b)" exponential (allometric model)

15 = "y~a+b*x+c*x^2+d*x^3" cubic

16 = "y~a/(1+b*(exp(-c*x)))^d" richards

17 = "y~(a^d+ ((b^d)-(a^d))*((1-exp(-c*(x-t1)))/ (1-exp(-c*(t2-t1)))))^(1/d)" schnute

`start` start values of the iteration process

`mixed` FALSE/defalt for fixed model or TRUE for mixed model

`digits` number of digits in results (default=6)

`alpha` significant level of the confident intervals for parameters in the models

Value

Returns coefficients of the models, t test for coefficients, R squared, adjusted R squared, AIC, BIC, and residuals of the model

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

References

KAPS, M. and LAMBERSON, W. R. Biostatistics for Animal Science: an introductory text. 2nd Edition. CABI Publishing, Wallingford, Oxfordshire, UK, 2009. 504p.

TERRANCE J. QUINN II and RICHARD B. DERISO. Quantitative Fish Dynamics, New York, Oxford, Oxford University Press, 1999.

See Also

nls, nls2

Examples

```
# weights of an Angus cow at ages from 8 to 108 months (Kaps and Lamberson, 2009)

weight=c(280,340,430,480,550,580,590,600,590,600)
age=c(8,12,24,36,48,60,72,84,96,108)

data1=data.frame(age, weight)

# linear
er1(data1, model=1)

# quadratic
er1(data1, model=2)

# linear plateau
er1(data1, model=3)

# quadratic plateau
er1(data1, model=4)

# two linear
er1(data1, model=5, start=c(250,6,2,50))

# exponential
er1(data1, model=6, start=c(250,0.05))

# logistic
er1(data1, model=7, start=c(600,4,0.05))

# van bertalanffy
er1(data1, model=8, start=c(600,2,0.05))

# brody
er1(data1, model=9, start=c(600,4,0.05))

# gompertz
er1(data1, model=10, start=c(600,4,0.05))

# richards
er1(data1, model=16, start=c(600,2,0.05,1.4))

# allometric
er1(data1, model=14)

# cubic
er1(data1, model=15)
```

```
# growth of Zagorje turkeys (Kaps and Lamberson, 2009)


weight=c(44,66,100,150,265,370,455,605,770)
age=c(1,7,14,21,28,35,42,49,56)

data2=data.frame(age,weight)

# two linear
er1(data2, model=5, start=c(25,6,10,20))

# gain weight measurements of turkey poults (Kaps and Lamberson, 2009)

methionine=c(80,85,90,95,100,105,110,115,120)
gain=c(102,115,125,133,140,141,142,140,142)

data3=data.frame(methionine, gain)

# linear
er1(data3, model=1)

# quadratic
er1(data3, model=2)

# linear plateau
er1(data3, model=3)

# quadratic plateau
er1(data3, model=4)

# lactation curve
 milk=c(25,24,26,28,30,31,27,26,25,24,23,24,22,21,22,
20,21,19,18,17,18,18,16,17,15,16,14)
 days=c(15,15,15,75,75,75,135,135,135,195,
195,195,255,255,255,315,315,315,375,375,375,435,435,435,495,495,495)

 data4=data.frame(days,milk)


er1(data4, model=11, start=c(16,0.25,0.004))

# ruminal degradation
time=c(2,6,9,24,48,72,96)
deg=c(20,33,46,55,66,72,76)

data5=data.frame(time,deg)

er1(data5, model=12)

# logistic bi-compartmental (gas production)
time=c(0,12,24,36,48,60,72,84,96,108,120,144,168,192)
gas=c(0.002,3.8,8,14.5,16,16.5,17,17.4,17.9,18.1,18.8,19,19.2,19.3)
```

```
data6=data.frame(time,gas)

er1(data6, model=13, start=c(19,4,0.025,0.004,5))

# Schnute model
#pacific halibut weight-age data of females (Terrance and Richard, 1999)

age=c(4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,
19,20,21,22,23,24,28)
weight=c(1.7,2,3.9, 4.2,6.4,7.6,10.9,14.9,18.2,21.6,
25.4,28.8,30.9,35.6,37.9,34.7,44.8,52.6,49.1,56.7,58.6,54.1)

halibut=data.frame(age,weight)


t1=min(halibut[,2])
t2=max(halibut[,2])

er1(halibut,model=17, start=c(a=t1,b=t2,c=0.15,d=-0.50))
```

`er2` *Analysis of polynomial regression*

Description

The function performs analysis of polynomial regression in simple designs with quantitative treatments.

Usage

```
er2(data, design = 1, list = FALSE, type = 2)
```

Arguments

`data`	data is a data.frame data frame with two columns, treatments and response (completely randomized design) data frame with three columns, treatments, blocks and response (randomized block design) data frame with four columns, treatments, rows, cols and response (latin square design) data frame with five columns, treatments, square, rows, cols and response (several latin squares)

`design` 1 = completely randomized design
2 = randomized block design
3 = latin square design
4 = several latin squares

`list` FALSE = a single response variable
TRUE = multivariable response

`type` type is form of obtain sum of squares
1 = a sequential sum of squares
2 = a partial sum of squares

Details

The response and the treatments must be numeric. Other variables can be numeric or factors.

Value

Returns analysis of variance, models, t test for coefficients and R squared and adjusted R squared.

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

References

KAPS, M. and LAMBERSON, W. R. Biostatistics for Animal Science: an introductory text. 2nd Edition. CABI Publishing, Wallingford, Oxfordshire, UK, 2009. 504p.

SAMPAIO, I. B. M. Estatistica aplicada a experimentacao animal. 3nd Edition. Belo Horizonte: Editora FEPMVZ, Fundacao de Ensino e Pesquisa em Medicina Veterinaria e Zootecnia, 2010. 264p.

See Also

lm, lme(package nlme), ea1(package easyanova), er1

Examples

```
# analysis in completely randomized design
data(data1)
r1=er2(data1)
names(r1)
r1
r1[1]

# analysis in randomized block design
data(data2)
r2=er2(data2, design=2)
r2

# analysis in latin square design
```

```
data(data3)
r3=er2(data3, design=3)
r3

# analysis in several latin squares
data(data4)
r4=er2(data4, design=4)
r4

# data
treatments=rep(c(0.5,1,1.5,2,2.5,3), c(3,3,3,3,3,3))
r1=rnorm(18,60,3)
r2=r1*1:18
r3=r1*18:1
r4=r1*c(c(1:10),10,10,10,10,10,10,10,10)
data6=data.frame(treatments,r1,r2,r3, r4)

# use the argument list = TRUE
er2(data6, design=1, list=TRUE)
```

---

regplot *Plot data and equation*

Description

The function plot data and equation

Usage

```
regplot(data, model=1, start=c(a=1,b=1,c=1,d=1,e=1), xlab="Explanatory Variable",
ylab="Response Variable", position=1, digits=6, mean=TRUE, sd=FALSE,
legend = TRUE, lty=2, col="dark blue", pch=20, xlim="defalt.x",ylim="defalt.y",...)
```

Arguments

`data` data is a data.frame The first column contain the treatments (explanatory variable) and the remaining column the response variable

`model` define the model

1 = "y~a+b*x" linear

2 = "y~a+b*x+c*x^2" quadratic

3 = "y ~ a + b * (x - c) * (x <= c)" linear plateau

4 = "y ~ (a + b * x + c * I(x^2)) * (x <= -0.5 * b/c) + (a + I(-b^2/(4 * c))) * (x > -0.5 * b/c)" quadratic plateau

5 = "ifelse(x>=d,(a-c*d)+(b+c)*x, a+b*x)" two linear

6 = "y~a*exp(b*x)" exponential

7 = "y~a*(1+b*(exp(-c*x)))^-1" logistic

8 = "y~a*(1-b*(exp(-c*x)))^3" van bertalanffy
9 = "y~a*(1-b*(exp(-c*x)))" brody
10 = "y~a*exp(-b*exp(-c*x)" gompertz
11 = "y~(a*x^b)*exp(-c*x)" lactation curve
12 = "y ~ a + b * (1 - exp(-c * x))" ruminal degradation curve
13 = "y~(a/(1+exp(2-4*c*(x-e))))+(b/(1+exp(2-4*d*(x-e))))" logistic bi-compartmental
14 = "y~a*(x^b)" exponential (allometric model)
15 = "y~a+b*x+c*x^2+d*x^3" cubic
16 = "y~a/(1+b*(exp(-c*x)))^d" richards
17 = "y~(a^d+ ((b^d)-(a^d))*((1-exp(-c*(x-t1)))/ (1-exp(-c*(t2-t1)))))^(1/d)" schnute

`start` start (iterations) values of model

`xlab` names of variable x

`ylab` names of variable y

`position` position of equation in the graph
top=1
bottomright=2
bottom=3
bottomleft=4
left=5
topleft=6 (default)
topright=7
right=8
center=9

`digits` number of digits (defalt=6)

`mean` mean=TRUE (plot mean of data) mean=FALSE (plot all data)

`sd` sd=FALSE (plot without standard deviation) sd=TRUE (plot with standard deviation)

`legend` legend=TRUE (plot legend) legend=FALSE (not plot legend)

`lty` line type

`col` line color

`pch` point type

`xlim` limits for x

`ylim` limits for y

`...` others graphical parameters (see par)

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

References

KAPS, M. and LAMBERSON, W. R. Biostatistics for Animal Science: an introductory text. 2nd Edition. CABI Publishing, Wallingford, Oxfordshire, UK, 2009. 504p.

TERRANCE J. QUINN II and RICHARD B. DERISO. Quantitative Fish Dynamics, New York, Oxford, Oxford University Press, 1999.

See Also

nls,er1,er2,bl

Examples

```
# weights of Angus cow at ages from 8 to 108 months (Kaps and Lamberson, 2009)

weight=c(280,340,430,480,550,580,590,600,590,600)
age=c(8,12,24,36,48,60,72,84,96,108)

data1=data.frame(age, weight)

# linear
regplot(data1, model=1, digits=3, position=3, ylab="weight", xlab="age")

# quadratic
regplot(data1, model=2, digits=3, position=3, col=1, ylim=c(200,700))

# linear plateau
regplot(data1, model=3,ylab="weight", xlab="age", lty=5, col="dark green",
position=3, ylim=c(200,700), xlim=c(0,150), lwd=2)

# quadratic plateau
regplot(data1, model=4,ylab="weight", xlab="age")

# two linear
regplot(data1, model=5, start=c(250,6,2,50),digits=3, position=3 )

# exponential
regplot(data1, model=6, start=c(250,0.05))

# logistic
regplot(data1, model=7, start=c(600,4,0.05))

# van bertalanffy
regplot(data1, model=8, start=c(600,2,0.05))

# brody
regplot(data1, model=9, start=c(600,4,0.05))

# gompertz
regplot(data1, model=10, start=c(600,4,0.05))

# richards
```

```
regplot(data1, model=16, start=c(600,2,0.05,1.4))

# allometric
regplot(data1, model=14)

# cubic
regplot(data1, model=15)



# growth of Zagorje turkeys (Kaps and Lamberson, 2009)


weight=c(44,66,100,150,265,370,455,605,770)
age=c(1,7,14,21,28,35,42,49,56)

data2=data.frame(age,weight)

# two linear
regplot(data2, model=5, start=c(25,6,10,20))


# weight gain measurements of turkey poults (Kaps and Lamberson, 2009)

methionine=c(80,85,90,95,100,105,110,115,120)
gain=c(102,115,125,133,140,141,142,140,142)

data3=data.frame(methionine, gain)

# linear
regplot(data3, model=1)

# quadratic
regplot(data3, model=2)

# linear plateau
regplot(data3, model=3)

# quadratic plateau
regplot(data3, model=4)

# lactation curve
 milk=c(25,24,26,28,30,31,27,26,25,24,23,24,22,21,22,20,21,19,
18,17,18,18,16,17,15,16,14)
 days=c(15,15,15,75,75,75,135,135,135,195,195,195,255,255,255,315,
315,315,375,375,375,435,435,435,495,495,495)

data4=data.frame(days,milk)


regplot(data4, model=11, start=c(16,0.25,0.004))

# ruminal degradation
```

```
time=c(2,6,9,24,48,72,96)
deg=c(20,33,46,55,66,72,76)

data5=data.frame(time,deg)

regplot(data5, model=12)

# logistic bi-compartmental (gas production)
time=c(0,12,24,36,48,60,72,84,96,108,120,144,168,192)
gas=c(0.002,3.8,8,14.5,16,16.5,17,17.4,17.9,18.1,18.8,19,19.2,19.3)

data6=data.frame(time,gas)

regplot(data6, model=13, start=c(19,4,0.025,0.004,5))

# multiple curves
time=c(0,12,24,48,64,72,96)
t1=c(36,48,59,72,85,86,87)
t2=c(14,25,36,49,59,65,72)
t3=c(55,78,86,87,86,87,88)

data=data.frame(time,t1,t2,t3)

regplot(data, model=12)
regplot(data, model=4)

# include standard deviation in graph
data(data1)

regplot(data1, sd=TRUE)

# Schnute model
#pacific halibut weight-age data of females (Terrance and Richard, 1999)

age=c(4,5,6,7,8,9,10,11,12,13,14,15,16,17,
18,19,20,21,22,23,24,28)
weight=c(1.7,2,3.9, 4.2,6.4,7.6,10.9,14.9,18.2,21.6,25.4,28.8,
30.9,35.6,37.9,34.7,44.8,52.6,49.1,56.7,58.6,54.1)

halibut=data.frame(age,weight)


t1=min(halibut[,2])
t2=max(halibut[,2])

regplot(halibut,model=17,start=c(t1,t2,0.22,-0.63), ylim=c(0,100))
```

`regtest` *Test of models and parameters*

Description

This function performs test of models and parameters

Usage

```
regtest(data, model = 1, start = c(a = 1, b = 1, c = 1, d = 1,e = 1))
```

Arguments

`data` data is a data.frame The first column contain explanatory variable, second column contain treatments and the third column contain the response variable

`model` define the model

1 = "y~a+b*x" linear

2 = "y~a+b*x+c*x^2" quadratic

3 = "y ~ a + b * (x - c) * (x <= c)" linear plateau

4 = "y ~ (a + b * x + c * I(x^2)) * (x <= -0.5 * b/c) + (a + I(-b^2/(4 * c))) * (x > -0.5 * b/c)" quadratic plateau

5 = "ifelse(x>=d,(a-c*d)+(b+c)*x, a+b*x)" two linear

6 = "y~a*exp(b*x)" exponential

7 = "y~a*(1+b*(exp(-c*x)))^-1" logistic

8 = "y~a*(1-b*(exp(-c*x)))^3" van bertalanffy

9 = "y~a*(1-b*(exp(-c*x)))" brody

10 = "y~a*exp(-b*exp(-c*x)" gompertz

11 = "y~(a*x^b)*exp(-c*x)" lactation curve

12 = "y ~ a + b * (1 - exp(-c * x))" ruminal degradation curve

13 = "y~(a/(1+exp(2-4*c*(x-e))))+(b/(1+exp(2-4*d*(x-e))))" logistic bi-compartmental

14 = "y~a*(x^b)" exponential (allometric model)

15 = "y~a+b*x+c*x^2+d*x^3" cubic

16 = "y~a/(1+b*(exp(-c*x)))^d" richards

17 = "y~(a^d+ ((b^d)-(a^d))*((1-exp(-c*(x-t1)))/ (1-exp(-c*(t2-t1)))))^(1/d)" schnute

`start` start values of iterations

Value

Returns coefficients of the models, test for coefficients, AIC and BIC.

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

See Also

lm, ea1(easyanova package), pr2, regplot

Examples

```
x=c(1,1,1,2,2,2,3,3,3,4,4,4)
y=c(5,5.3,6,8,8.9,12,14,18,25,25,29,32)
t=c("t1","t2","t3","t1","t2","t3","t1","t2","t3","t1","t2","t3")
data=data.frame(x,t,y)
# linear
regtest(data, model=1)
# quadratic
regtest(data, model=2)
# exponential
regtest(data, model=6)
# ... etc
```

Index