

Package: epr (via r-universe)

October 18, 2024

Type Package

Title Easy Polynomial Regression

Version 3.0

Date 2017-11-14

Author Emmanuel Arnhold

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

Description Performs analysis of polynomial regression in simple designs with quantitative treatments.

Depends R (>= 3.0.0)

Imports car, lme4

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2017-11-16 22:15:46 UTC

Contents

epr-package	2
bl	3
data1	4
data2	5
data3	5
data4	6
data5	6
pr1	7
pr2	8
r.test	10
regplot	11

Index	13
--------------	-----------

epr-package

Easy Polynomial Regression

Description

Performs analysis of polynomial regression in simple designs with quantitative treatments.

Details

Package: epr
Type: Package
Version: 3.0
Date: 2017-11-14
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. Estatística aplicada a experimentação animal. 3rd Edition. Belo Horizonte: Editora FEPMVZ, Fundação de Ensino e Pesquisa em Medicina Veterinária e Zootecnia, 2010. 264p.

Examples

```
# analysis in completely randomized design
data(data1)
r1=pr2(data1)
names(r1)
r1
r1[[1]]

pr1(data1)

# analysis in randomized block design
data(data2)
r2=pr2(data2, design=2)
r2
```

```
# analysis in latin square design
data(data3)
r3=pr2(data3, design=3)
r3

# analysis in several latin squares
data(data4)
r4=pr2(data4, design=4)
r4
```

bl

Analysis of bronken line regression

Description

The function performs analysis of broken line regression.

Usage

```
bl(data, xlab="Explanatory Variable", ylab="Response Variable", position=1)
```

Arguments

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

Value

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

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

See Also

lm, eal(easyanova package), pr2, regplot

Examples

```
x=c(0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08,0.09,0.10)
y=c(5.5,4,3.2,2.1,1,0.1,1.6,2.2,3,5)
y=y/100
data=data.frame(x,y)

### bl(data)
```

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. Estatística aplicada a experimentação animal. 3rd Edition. Belo Horizonte: Editora FEPMVZ, Fundação de Ensino e Pesquisa em Medicina Veterinária e Zootecnia, 2010. 264p.

Examples

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

data2

data2: Kaps and Lamberson (2009): page 434

Description

Quantitative treatments in randomized 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	<i>data4: fictional example</i>
-------	---------------------------------

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	<i>data5: fictional example</i>
-------	---------------------------------

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)
```

pr1 *Analysis of polynomial regression*

Description

The function performs analysis of polynomial regression in simple designs with quantitative treatments. The function also performs with random factor in mixed models.

Usage

```
pr1(data, mixed = FALSE, digits = 6)
```

Arguments

data	data is a data.frame The first column should contain the treatments (explanatory variable) and the remaining columns the response variables (fixed model). The first column should contain the treatments (explanatory variable), second column should contain the random variable and the remaining columns the response variables (mixed model).
mixed	FALSE = fixed model TRUE = mixed model
digits	6 = default (number of digits)

Value

Returns coefficients of the models, t test for coefficients, R squared, adjusted R squared, AIC, BIC and the maximum (or minimum) values of y and critical point of x, residuals and normality test.

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

See Also

lm, eal(easynova package), pr2, regplot

Examples

```
# data
data(data5)

# linear and quadratic models
results1=pr1(data5)
results1

# analysis in completely randomized design
data(data1)
r1=pr2(data1)
names(r1)
r1
r1[1]

pr1(data1)

# analysis in randomized block design
data(data2)
r2=pr2(data2, design=2)
r2

pr1(data2, mixed=TRUE)
```

pr2

Analysis of polynomial regression

Description

The function performs analysis of polynomial regression in simple designs with quantitative treatments. This function performs analysis the lack of fit .

Usage

```
pr2(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. Estatística aplicada a experimentação animal. 3rd Edition. Belo Horizonte: Editora FEPMVZ, Fundação de Ensino e Pesquisa em Medicina Veterinária e Zootecnia, 2010. 264p.

See Also

lm, lme(package nlme), ea1(package easyanova), pr1, regplot

Examples

```
# analysis in completely randomized design
data(data1)
r1=pr2(data1)
names(r1)
r1
r1[[1]]

# analysis in randomized block design
```

```

data(data2)
r2=pr2(data2, design=2)
r2

# analysis in latin square design
data(data3)
r3=pr2(data3, design=3)
r3

# analysis in several latin squares
data(data4)
r4=pr2(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
pr2(data6, design=1, list=TRUE)

```

r.test

Tests for model identity and parameter

Description

The function performs tests of parameters and models.

Usage

```
r.test(data, digits=6)
```

Arguments

data	data is a data.frame The first column should contain the x (explanatory variable) second treatments and the remaining columns the response variables.
digits	number of digits (default = 6)

Value

Returns coefficients of the models, t test for coefficients and tests for parameters and models.

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

See Also

lm, eal(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("a1","a2","a3","a1","a2","a3","a1","a2","a3","a1","a2","a3")
data=data.frame(x,t,y)

r.test(data)
```

 regplot

Graphics of the regression

Description

The function generates the scatter plot with the regression equation.

Usage

```
regplot(data, xlab="Explanatory Variable", ylab="Response Variable",
position=6, mean=TRUE, digits=4)
```

Arguments

data	data is a data.frame the first column contain the explanatory variable the others columns contain the responses variables
xlab	name of variable x
ylab	name 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
mean	TRUE = scatter plots with averages (default) FALSE = scatter plots with all data
digits	number of digits

Value

The function generates the scatter plot with the regression equation.

Author(s)

Emmanuel Arnhold <emmanuelarnhold@yahoo.com.br>

See Also

lm, lme, eal(easyanova package), pr2, pr2, dplot(ds package)

Examples

```
# data
data(data5)

d1=data5[,c(1,2)]
regplot(d1, position=8)

d2=data5[,c(1,3)]
regplot(d2, position=8)

d3=data5[,c(1,4)]
regplot(d3, position=8)
```

Index

[b1](#), [3](#)

[data1](#), [4](#)

[data2](#), [5](#)

[data3](#), [5](#)

[data4](#), [6](#)

[data5](#), [6](#)

[epr \(epr-package\)](#), [2](#)

[epr-package](#), [2](#)

[pr1](#), [7](#)

[pr2](#), [8](#)

[r.test](#), [10](#)

[regplot](#), [11](#)