Package ‘doebioresearch’

July 22, 2025

Type Package

Title Analysis of Design of Experiments for Biological Research

Version 0.1.0

Description
Performs analysis of popular experimental designs used in the field of biological research. The designs covered are completely randomized design, randomized complete block design, factorial completely randomized design, factorial randomized complete block design, split plot design, strip plot design and latin square design. The analysis include analysis of variance, coefficient of determination, normality test of residuals, standard error of mean, standard error of difference and multiple comparison test of means. The package has functions for transformation of data and yield data conversion. Some datasets are also added in order to facilitate examples.

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Imports agricolae (>= 1.3.3), stats (>= 4.0.2)

Depends R (>= 2.10)

NeedsCompilation no

Author Raj Popat [aut, cre],
Kanthesh Banakara [aut]

Maintainer Raj Popat <popatrajc@gmail.com>

Repository CRAN

Date/Publication 2020-07-08 12:20:03 UTC

Contents

`arcsineretransform` *Re-transform the Arc sine transformed data*

Description

Re-transform the arc sine transformed data. When arc sine transformation is done, the mean of the treatments needs to be re-transformed for comparison.

Usage

```
arcsineretransform(mean.vector, type)
```

Arguments

`mean.vector`	vector of mean which needs to be re-transformed
`type`	0 if data was in proportion prior to re-transformation, 1 if data was in percentage prior to re-transformation

Value

Arc sine re-transformed vector

Examples

```
data<-c(60,63.43495,71.56505,78.46304)
#If data was in percentage prior to re-transformation
arcsineretransform(data,1)
#If data was in proportion prior to re-transformation
arcsineretransform(data,0)
```

arcsinetransform *Arc sine transformation of the numeric vector*

Description

The function divide values by 100, does square root and than sin inverse of each values of vector. If any of the values of a vector is 0 or 100, it is replaced by 1/4n or 100-(1/4n), respectively.

Usage

```
arcsinetransform(numeric.vector, type, n)
```

Arguments

`numeric.vector`	data vector to be transformed
`type`	0 if data is in percentage and 1 if data is in proportion
`n`	is the number of units upon which the percentage/proportion data is based

Value

Arc sine transformed data

Examples

```
vector<-c(23,0,29.6,35.6,33,35.6,10.5,100)
# Arc sine trnasformation for percentage data and n=10
arcsinetransform(vector,0,10)
```

convert *Convert the data frame into list of numeric nature*

Description

Convert the data frame into list of numeric nature

Usage

```
convert(data1)
```

Arguments

`data1`	data-frame to be converted into list

Value

list of numeric vectors

`crd` *Analysis of Completely Randomized Design*

Description

The function gives ANOVA, R-square of the model, normality testing of residuals, SEm (standard error of mean), SEd (standard error of difference), interpretation of ANOVA results and multiple comparison test for means

Usage

```
crd(data, trt.vector, MultipleComparisonTest)
```

Arguments

`data` dependent variables

`trt.vector` vector containing treatments

`MultipleComparisonTest`
0 for no test, 1 for LSD test, 2 for Duncan test and 3 for HSD test

Value

ANOVA, interpretation of ANOVA, R-square, normality test result, SEm, SEd and multiple comparison test result

Examples

```
data<-data.frame(Treatments=c("T1","T2","T3","T4","T5","T6","T7","T1","T2","T3","T4","T5","T6",
"T7","T1","T2","T3","T4","T5","T6","T7"),
yield=c(25,21,21,18,25,28,24,25,24,24,16,21,20,17,16,19,14,15,13,11,25),
height=c(130,120,125,135,139,140,145,136,129,135,150,152,140,148,130,135,145,160,145,130,160))
#CRD analysis with LSD test for yield only
crd(data[2],data$Treatments,1)
#CRD analysis with LSD test for both yield and height
crd(data[2:3],data$Treatments,1)
```

`factorialdata` *Data of Factorial Experiment*

Description

The data consists of three factors nitrogen, phosphorus and Potassium, replication and two dependent variables yield and plant height. The data is generated manually.

Usage

```
factorialdata
```

Format

The data has 6 columns and 36 rows

Nitrogen Consist sequence of two nitrogen levels n0 and n1

Phosphorus Consist sequence of two phosphorus levels p0 and p1

Potassium Consist sequence of two potassium levels k0 and k1

Replication Contains replication which has three levels

Yield Yield as dependent variable

Plant Height Plant height as dependent variable

---

fcrd2fact *Analysis of Factorial Completely Randomized Design for 2 factors*

Description

The function gives ANOVA, R-square of the model, Normality testing of residuals, SEm (standard error of mean), SEd (standard error of difference), interpretation of ANOVA results and multiple comparison test for means

Usage

```
fcrd2fact(data, fact.A, fact.B, Multiple.comparison.test)
```

Arguments

`data`	dependent variables
`fact.A`	vector containing levels of first factor
`fact.B`	vector containing levels of second factor
`Multiple.comparison.test`	0 for no test, 1 for LSD test, 2 for Duncan test and 3 for HSD test

Value

ANOVA, interpretation of ANOVA, R-square, normality test result, SEm, SEd and multiple comparison test result for both the factors as well as interaction.

Examples

```
data(factorialdata)
#Analysis of Factorial Completely Randomized design along with Dunccan test for Yield only
fcrd2fact(factorialdata[5],factorialdata$Nitrogen,factorialdata$Phosphorus,2)
#Analysis of Factorial Completely Randomized design along with Dunccan test for Yield & Plant Height
fcrd2fact(factorialdata[5:6],factorialdata$Nitrogen,factorialdata$Phosphorus,2)
```

fcrd3fact *Analysis of Factorial Completely Randomized Design for 3 factors*

Description

The function gives ANOVA, R-square of the model, normality testing of residuals, SEm (standard error of mean), SEd (standard error of difference), interpretation of ANOVA results and multiple comparison test for means.

Usage

```
fcrd3fact(data, fact.A, fact.B, fact.C, Multiple.comparison.test)
```

Arguments

data	dependent variables
fact.A	vector containing levels of first factor
fact.B	vector containing levels of second factor
fact.C	vector containing levels of third factor
Multiple.comparison.test	0 for no test, 1 for LSD test, 2 for Duncan test and 3 for HSD test

Value

ANOVA, interpretation of ANOVA, R-square, normality test result, SEm, SEd and multiple comparison test result for both the factors as well as interaction.

Examples

```
data(factorialdata)
#FCRD analysis along with dunccan test for two dependent var.
fcrd3fact(factorialdata[5:6],factorialdata$Nitrogen,
factorialdata$Phosphorus,factorialdata$Potassium,2)
```

frbd2fact *Analysis of Factorial Randomized Block Design for 2 factors*

Description

The function gives ANOVA, R-square of the model, normality testing of residuals, SEm (standard error of mean), SEd (standard error of difference), interpretation of ANOVA results and multiple comparison test for means.

Usage

```
frbd2fact(data, replicationvector, fact.A, fact.B, Multiple.comparison.test)
```

Arguments

`data` dependent variables

`replicationvector`
vector containing replications

`fact.A` vector containing levels of first factor

`fact.B` vector containing levels of second factor

`Multiple.comparison.test`
0 for no test, 1 for LSD test, 2 for Duncan test and 3 for HSD test

Value

ANOVA, interpretation of ANOVA, R-square, normality test result, SEm, SEd and multiple comparison test results for both the factors as well as interaction.

Examples

```
data(factorialdata)
#FRBD analysis along with dunccan test for two dependent var.
frbd2fact(factorialdata[5:6],factorialdata$Replication,
factorialdata$Nitrogen,factorialdata$Phosphorus,2)
```

`frbd3fact` *Analysis of Factorial Randomized Block Design for 3 factors*

Description

The function gives ANOVA, R-square of the model, normality testing of residuals, SEm (standard error of mean), SEd (standard error of difference), interpretation of ANOVA results and multiple comparison test for means.

Usage

```
frbd3fact(
  data,
  replicationvector,
  fact.A,
  fact.B,
  fact.C,
  Multiple.comparison.test
)
```

Arguments

`data` dependent variables

`replicationvector`
vector containing replications

`fact.A` vector containing levels of first factor

`fact.B` vector containing levels of second factor

`fact.C` vector containing levels of third factor

`Multiple.comparison.test`
0 for no test, 1 for LSD test, 2 for Duncan test and 3 for HSD test

Value

ANOVA, interpretation of ANOVA, R-square, normality test result, SEm, SEd and multiple comparison test result for the factors as well as the interaction.

Examples

```
data(factorialdata)
#FRBD analysis along with dunccan test for two dependent var.
frbd3fact(factorialdata[5:6],factorialdata$Replication,factorialdata$Nitrogen,
factorialdata$Phosphorus,factorialdata$Potassium,2)
```

logretransform *Re-transform the log transformed data*

Description

Re-transform the log transformed data. When log transformation is done, the mean of the treatments needs to be re-transformed for comparison.

Usage

```
logretransform(transformed.mean, if.zero.present)
```

Arguments

`transformed.mean`
vector of mean which needs to be re-transformed

`if.zero.present`
0 if zero was present in the data prior to transformation of data. 1 if zero was absent in the data prior to transformation

Value

Log re-transformed values

Examples

```
vector<-c(0,2.004,1.114,1.491,1.431,1.415,1.845)
#Re-transformation of data with zero present in data prior to transformation
logretransform(vector,0)
```

`logtransform` *Log transformation of the numeric vector*

Description

The function carries out log with base 10 transformation of each values of vector. If one of values of a vector is 0, 1 is added to each observation. Log transformation is carried out for the data when variance is proportional to square of the mean and treatment effects are multiplicative in nature.

Usage

```
logtransform(numeric.vector)
```

Arguments

`numeric.vector`	data vector to be transformed

Value

A list of

- `Ratio`- A ratio of maximum and minimum values of the data
- `LogTransformedVector` - A vector of the transformed data
- `Comment` - A comment about zero being present in data or not

Examples

```
vector<-c(100,0,120,1000,52,30,60)
logtransform(vector)
```

lsd *Analysis of Latin Square Design*

Description

The function gives ANOVA, R-square of the model, normality testing of residuals, SEm (standard error of mean), SEd (standard error of difference), interpretation of ANOVA results and multiple comparison test for means.

Usage

```
lsd(data, treatmentvector, row, column, MultipleComparisonTest)
```

Arguments

`data`	dependent variables
`treatmentvector`	vector containing treatments
`row`	vector for rows
`column`	vector for columns
`MultipleComparisonTest`	0 for no test, 1 for LSD test, 2 for Duncan test and 3 for HSD test

Value

ANOVA, interpretation of ANOVA, R-square, normality test result, SEm, SEd and multiple comparison test result

Examples

```
data(lsddata)
#LSD analysis with LSD test for Yield only
lsd(lsddata[4],lsddata$Treatment,lsddata$Row,lsddata$Column,1)
#LSD analysis with LSD test for Yield and Plant Height
lsd(lsddata[4:5],lsddata$Treatment,lsddata$Row,lsddata$Column,1)
```

lsddata *Data for Latin Square Design*

Description

The data consists of Rows, Columns, Treatments and two dependent variables Yield and Plant Height. The data is generated manually.

Usage

```
lsddata
```

Format

The data has 5 columns and 25 rows

Row Consist sequence of rows. Row consists of 5 levels

Column Consist sequence of column. Column consists of 5 levels

Treatment Consist sequence of treatments. There are 5 treatments A, B, C, D & E

Yield Yield as dependent variable

Plant Height Plant height as dependent variable

`rcbd` *Analysis of Randomized Complete Block Design*

Description

The function gives ANOVA, R-square of the model, normality testing of residuals, SEm (standard error of mean), SEd (standard error of difference), interpretation of ANOVA results and multiple comparison test for means.

Usage

```
rcbd(data, treatmentvector, replicationvector, MultipleComparisonTest)
```

Arguments

`data`	dependent variables
`treatmentvector`	vector containing treatments
`replicationvector`	vector containing replications
`MultipleComparisonTest`	0 for no test, 1 for LSD test, 2 for Duncan test and 3 for HSD test

Value

ANOVA, interpretation of ANOVA, R-square, normality test result, SEm, SEd and multiple comparison test result

Examples

```
data<-data.frame(GFY=c(16,13,14,16,16,17,16,17,16,16,17,16,15,15,15,13,15,14,
16,14,15,14,15,17,18,15,15,15,14,14,14,14,15,15,13,15,14,14,13,13,13,12,15,12,15),
DMY=c(5,5,6,5,6,7,6,8,6,9,8,7,5,5,5,4,6,5,8,5,5,5,4,6,6,5,5,6,6,6,5,5,5,5,5,6,5,5,5,4,5,4,5,5,5),
Rep=rep(c(”R1”,”R2”,”R3”),each=15),
Trt=rep(c(”T1”,”T2”,”T3”,”T4”,”T5”,”T6”,”T7”,”T8”,”T9”,”T10”,”T11”,”T12”,”T13”,”T14”,”T15”),3))
#' #RCBD analysis with duncan test for GFY only
rcbd(data[1],data$Trt,data$Rep,2)
#RCBD analysis with duncan test for both GFY and DMY
rcbd(data[1:2],data$Trt,data$Rep,2)
```

`splitdata` *Data for Split plot Design*

Description

The data consists of replication, date of sowing (as main-plot), varieties (as sub-plot) and two dependent variables yield and plant height. The data is generated manually.

Usage

```
splitdata
```

Format

The data has 5 columns and 36 rows

Replication Consist sequence of replications. Replications consists of 3 levels

Date of Sowing Consist sequence of levels of date of sowing as Main-plot. Date of sowing consists of 2 levels

Varities Consist sequence of levels of varities as Sub-plot. Varities consist of 6 levels

Yield Yield as dependent variable

Plant Height Plant height as dependent variable

`splitplot` *Analysis of Split plot design*

Description

The function gives ANOVA, R-square of the model, normality testing of residuals, SEm (standard error of mean), SEd (standard error of difference), interpretation of ANOVA results and multiple comparison test for means.

Usage

```
splitplot(data, block, main.plot, sub.plot, mean.comparison.test)
```

Arguments

`data`	dependent variables
`block`	vector containing replications
`main.plot`	vector containing main-plot levels
`sub.plot`	vector containing sub-plot levels
`mean.comparison.test`	0 for no test, 1 for LSD test, 2 for Dunccan test and 3 for HSD test

Value

ANOVA, interpretation of ANOVA, R-square, normality test result, SEm, SEd and multiple comparison test result

Examples

```
data(splitdata)
#Using Date of sowing as Main-plot factor and varieties as sub-plot factor and using LSD test
#Split plot analysis with LSD test for Yield
splitplot(splitdata[4],splitdata$Replication,splitdata$Date_of_Sowing,splitdata$Varities,1)
#Split plot analysis with LSD test for both Yield and Plant Height
splitplot(splitdata[4:5],splitdata$Replication,splitdata$Date_of_Sowing,splitdata$Varities,1)
```

`sqrtretransform` *Re-transform the square root transformed data*

Description

Retransform the square root transformed data. When square root transformation is done, the mean of the treatments needs to be re-transformed for comparison.

Usage

```
sqrtretransform(transformed.mean, if.zero.present)
```

Arguments

`transformed.mean`	vector of mean which needs to be re-transformed
`if.zero.present`	0 if zero was present in the data prior to transformation of data. 1 if zero was absent in the data prior to transformation

Value

Square root re-transformed vector

Examples

```
vector<-c(19,10,30,60,50,10,5)
#Square root re-transform and zero was absent in the data prior to transformation
sqrtretransform(vector,1)
```

---

sqrttransform *Square root transformation of the numeric vector*

Description

The function carries out square root transformation of each values of vector. If one of values of a vector is 0, 0.5 is added to each observation.

Usage

```
sqrttransform(numeric.vector)
```

Arguments

numeric.vector data vector to be transformed

Value

Square root transformed data

Examples

```
vector<-c(0,25,36,6,9,25,70)
sqrttransform(vector)
```

---

stripplot *Analysis of Strip plot design*

Description

The function gives ANOVA, R-square of the model, normality testing of residuals, SEm (standard error of mean), SEd (standard error of difference), interpretation of ANOVA results and multiple comparison test for means

Usage

```
stripplot(data, block, column, row, mean.comparison.test)
```

Arguments

`data`	dependent variables
`block`	vector containing replications
`column`	vector containing column strip levels
`row`	vector containing row strip levels
`mean.comparison.test`	0 for no test, 1 for LSD test, 2 for Duncan test and 3 for HSD test

Value

ANOVA, interpretation of ANOVA, R-square, normality test result, SEm, SEd and multiple comparison test result

Examples

```
data(splitdata)
#Split data is used for sake of demonstration
#Using Date of sowing as Column factor and varieties as Row factor and using LSD test for Yield only
stripplot(splitdata[4],splitdata$Replication,splitdata$Date_of_Sowing,splitdata$Varities,1)
#Using Date of sowing as Column factor and varieties as Row factor and using LSD test for both var.
stripplot(splitdata[4:5],splitdata$Replication,splitdata$Date_of_Sowing,splitdata$Varities,1)
```

`yieldconvert` *Convert the yield data of plot into different units*

Description

The function converts the yield data of plot into qtl/ha, tonnes/ha, qtl/acre or tonnes/acre depending on the option chosen.

Usage

```
yieldconvert(yield.in.kg, length.of.plot, width.of.plot, choose.convert.to)
```

Arguments

`yield.in.kg`	yield data in kilograms
`length.of.plot`	length of plot in m
`width.of.plot`	width of the plot in m
`choose.convert.to`	0 for qtl/ha, 1 for tonnes/ha, 2 for qtl/acre and 3 for tonnes/acre

Value

converted yield

Examples

```
#Convert yield vector obtained from 10m x 5m plot into different forms
yield<-c(10,15,12,16,19,25,30,25,11)
#For converting into qtl/ha
yieldconvert(yield,10,5,0)
#For converting into tonnes/ha
yieldconvert(yield,10,5,1)
#For converting into qtl/acre
yieldconvert(yield,10,5,2)
#For converting into tonnes/acre
yieldconvert(yield,10,5,3)
```

Index