Package ‘BiDimRegression’

May 16, 2018

Version 2.0.0
Date 2018-05-09
Title Calculates the Bidimensional Regression Between Two 2D Configurations
Imports Formula, methods
Depends R (>= 1.8.0)
Description Calculates the bidimensional regression between two 2D configurations following the approach by Tobler (1965).
License GPL-3
URL https://CRAN.R-project.org/package=BiDimRegression
NeedsCompilation no
Repository CRAN
RoxygenNote 6.0.1
LazyData true
Suggests testthat, knitr, rmarkdown, dplyr, ggplot2
VignetteBuilder knitr
Author Claus-Christian Carbon [aut], Alexander (Sasha) Pastukhov [cre, ctb]
Maintainer Alexander (Sasha) Pastukhov <pastukhov.alexander@gmail.com>
Date/Publication 2018-05-16 11:32:59 UTC

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>anova.lm2</td>
<td>2</td>
</tr>
<tr>
<td>BiDimRegression</td>
<td>3</td>
</tr>
<tr>
<td>CarbonExample1Data</td>
<td>4</td>
</tr>
<tr>
<td>CarbonExample2Data</td>
<td>5</td>
</tr>
<tr>
<td>CarbonExample3Data</td>
<td>6</td>
</tr>
<tr>
<td>EyegazeData</td>
<td>6</td>
</tr>
<tr>
<td>FriedmanKohlerData1</td>
<td>7</td>
</tr>
</tbody>
</table>
Description

Anova for lm2 objects, returns a table with pairwise comparisons between models or, if only one model was supplied, with the null model.

Usage

```r
## S3 method for class 'lm2'
anova(object, ...)
```

Arguments

- `object` an object of class "lm2"
- `...` further objects of class "lm2"

Value

an anova data frame

See Also

`lm2`

Examples

```r
lm2euc <- lm2(depV1+depV2~indepV1+indepV2, NakayaData, transformation = 'Euclidean')
lm2aff <- lm2(depV1+depV2~indepV1+indepV2, NakayaData, transformation = 'Affine')
anova(lm2euc, lm2aff)
```
**BiDimRegression**

*Calculates the bidimensional regression between two 2D configurations*

**Description**

Calculates the bidimensional regression between two 2D configurations using both Euclidean and Affine transformations following the approach by Tobler (1965). This function assumes strict data format and returns all coefficients and statistics in a single structure. Same functionality is now re-implemented in a R-friendly style, see `lm2` function.

**Usage**

`BiDimRegression(coord)`

**Arguments**

- `coord` table that must contain two columns for dependent variables (named `depV1` and `depV2`) and two columns for independent variables (named `indepV1` and `indepV2`).

**Value**

an S3 class `BiDimRegression` containing all essential measures of the bidimensional regression:

- `euclidean.r`, `affine.r` - the regression coefficient, defined analogously to Pearson’s r.
- `euclidean.rsqr`, `affine.rsqr` - the squared regression coefficient.
- `euclidean.diABSqr`, `affine.diABSqr` - the squared distortion index for dependent variables; following Waterman and Gordon's (1984) extension of the bidimensional regression, it provides a measure of comparison of distortions, but the range of values is 0 to 1 following Friedman and Kohler (2003).
- `euclidean.dMaxABSqr`, `affine.dMaxABSqr` - the maximal squared distortion index for dependent variables.
- `euclidean.diYSqr`, `affine.diYSqr` - the distortion index for independent variables.
- `euclidean.dMaxYSqr`, `affine.dMaxYSqr` - the maximal squared distortion index for independent variables.
- `euclidean.scaleFactorX`, `affine.scaleFactorX` - the scaling factor of the first dimension (1.0 means no scaling; values below 1.0 indicate a contraction, values above 1.0 indicate an expansion).
- `euclidean.scaleFactorY`, `affine.scaleFactorY` - the scaling factor of the second dimension.
- `euclidean.angleDEG`, `affine.angleDEG` - the rotation angle in degrees.
- `euclidean.shear`, `affine.shear` - shearing of the transformed configuration, always zero for the Euclidean transformation.
• `euclidean.ttestDF`, `affine.ttestDF` - degrees of freedom (DF) for the t-tests regarding the model parameters (alphas and betas).
• `euclidean.alpha1.*`, `euclidean.alpha2.*`, `affine.alpha1.*`, `affine.alpha2.*` - intercept vectors, information includes .coeff for coefficient, .SE for standard error, tValue for t-statistics, and pValue for significance.
• `euclidean.fvalue`, `affine.fvalue` - F-statistics, following the advice of Nakaya (1997).
• `euclidean.df1`, `affine.df1` - degrees of freedom of the nominator used for the F-statistics propagated by Nakaya (1997); df1 = p-2, with p is the number of elements needed to calculate the referring model: p=4 for the Euclidean and p=6 for the affine geometry Nakaya, 1997, Table 1.
• `euclidean.df2`, `affine.df2` - degrees of freedom of the denominator used for the F-statistics propagated by Nakaya (1997); df2 = 2n-p, with p is the number of elements needed to calculate the referring model (see df1) and n is the number of coordinate pairs.
• `euclidean.pvalue`, `affine.pvalue` - the significance level based on the preceding F-statistics.
• `euclidean.dAICso`, `affine.dAICso` - the AIC difference between the regarding bidimensional regression model and the bidimensional null model (S0) according to Nakaya (1997), formula 56.
• `eucVSaff.*` - statistical comparison between Euclidean and Affine models, include .fValue for F-statistics, .df1 and .df2 for the degrees of freedom, .pValue for the significance level, and .dAIC for AIC difference between two models.

**See Also**

`lm2`

**Examples**

```r
resultingMeasures <- BiDimRegression(NakayaData)
pdf(resultingMeasures)
```

---

**Description**

Example 1 from the domain of aesthetics to show how the method can be utilized for assessing the similarity of two portrayed persons, actually the Mona Lisa in the world famous Louvre version and the only recently re-discovered Prado version.

Usage

data(CarbonExample1Data)

Format

A data frame with 36 observations on the following 4 variables.

   depV1    a numeric vector
   depV2    a numeric vector
   indepV1  a numeric vector
   indepV2  a numeric vector

Examples

data(CarbonExample1Data)
## maybe str(CarbonExample1Data); plot(CarbonExample1Data) ...

---

CarbonExample2Data


Description

Example 2 originates from the area of geography and inspects the accuracy of different maps of the city of Paris which were created over the last 350 years as compared to a recent map

Usage

data(CarbonExample2Data)

Format

A data frame with 13 observations on the following 4 variables.

   depV1    a numeric vector
   depV2    a numeric vector
   indepV1  a numeric vector
   indepV2  a numeric vector

Examples

data(CarbonExample2Data)
## maybe str(CarbonExample2Data); plot(CarbonExample2Data) ...
CarbonExample3Data  


Description

Example 3 focuses on demonstrating how good a cognitive map recalculated from averaged cognitive distance data fits with a related real map.

Usage

data(CarbonExample3Data)

Format

A data frame with 10 observations on the following 4 variables.

- depV1  a numeric vector
- depV2  a numeric vector
- indepV1 a numeric vector
- indepV2 a numeric vector

Examples

data(CarbonExample3Data)
## maybe str(CarbonExample3Data) ; plot(CarbonExample3Data) ...

EyegazeData  

Eye gaze calibration data

Description

A dataset containing a monocular eye gaze recording with calibration sequence. Courtesy of Bamberger Baby Institut: BamBI.

Usage

EyegazeData
**FriedmanKohlerData1**

**Format**

A data frame with 365 rows and 6 variables:

- **time** sample timestamp, in milliseconds
- **x, y** recorded gaze, in internal eye tracker units
- **target_x, target_y** location of the calibration target on the screen, in pixels
- **target** index of the target within the sequence

**Description**


**Usage**

data(FriedmanKohlerData1)

**Format**

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

- **depV1** a numeric vector
- **depV2** a numeric vector
- **indepV1** a numeric vector
- **indepV2** a numeric vector

**Examples**

data(FriedmanKohlerData1)
## maybe str(FriedmanKohlerData1) ; plot(FriedmanKohlerData1) ...

Usage
data(FriedmanKohlerData2)

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

- depv1 a numeric vector
- depv2 a numeric vector
- indepv1 a numeric vector
- indepv2 a numeric vector

Examples
data(FriedmanKohlerData2)
## maybe str(FriedmanKohlerData2); plot(FriedmanKohlerData2) ...

lm2
Fitting Bidimensional Regression Models

Description
lm2 is used to fit bidimensional linear regression models using Euclidean and Affine transformations following the approach by Tobler (1965).

Usage
lm2(formula, data, transformation)
Arguments

formula  a symbolic description of the model to be fitted in the format \( A + B \sim C + D \), where \( A \) and \( B \) are dependent and \( C \) and \( D \) are independent variables.
data  a data frame containing variables for the model.
transformation  the transformation to be used, either 'euclidean', 'affine', or 'projective'.

Value

lm2 returns an object of class "lm2". An object of class "lm" is a list containing at least the following components:

transformation  string with the transformation type (euclidean, affine, or projective)
npredictors  number of predictors used in the model: 4 for euclidean, 6 for affine, 8 for projective.
df_model, df_residual  degrees of freedom for the model and for the residuals
transformation_matrix  \( 3 \times 3 \) transformation matrix
coeff  transformation coefficients, with \( a \) denoting the intercept terms.
transformed_coeff  scale, angle, and sheer coefficients, depends on transformation.
fitted_values  data frame containing fitted values for the original data set
residuals  data frame containing residuals for the original fit
r_squared, adj_r_squared  R-squared and adjusted R-squared.
F, p_value  F-statistics and the corresponding p-value, given the df_model and df_residual degrees of freedom.
daIC  Akaike Information Criterion (AIC) difference between the regression model and the null model. A negative values indicates that the regression model is better. See Nakaya (1997).
distortion_index  Distortion index following Waterman and Gordon (1984), as adjusted by Friedman and Kohler (2003)

lm  an underlying linear model for Euclidean and affine transformations.
formula  formula, describing input and output columns
data  data used to fit the model
Call  function call information, incorporates the formula, transformation, and data.

See Also

anova.lm2 BiDimRegression
Examples

```r
lm2euc <- lm2(depV1 + depV2 ~ indepV1 + indepV2, NakayaData, 'euclidean')
lm2aff <- lm2(depV1 + depV2 ~ indepV1 + indepV2, NakayaData, 'affine')
lm2prj <- lm2(depV1 + depV2 ~ indepV1 + indepV2, NakayaData, 'projective')
anova(lm2euc, lm2aff, lm2prj)
predict(lm2euc)
summary(lm2euc)
```

Description


Usage

data(NakayaData)

Format

A data frame with 19 observations on the following 4 variables.

- `depV1` a numeric vector
- `depV2` a numeric vector
- `indepV1` a numeric vector
- `indepV2` a numeric vector

Examples

data(NakayaData)
## maybe str(NakayaData); plot(NakayaData) ...
predict.lm2

Predict method for Bidimensional Regression Model Fits

Description

Predicted values based on the bidimensional regressional model object.

Usage

## S3 method for class 'lm2'
predict(object, newdata, 
         Mandatory arguments
         ...
         Optional arguments)

Arguments

- **object**: an object of class "lm2"
- **newdata**: An optional two column data frame with independent variables. If omitted, the fitted values are used.
- **...**: optional arguments

Value

a two column data frame with predicted values for dependent variables.

See Also

lm2

Examples

lm2euc <- lm2(depV1+depV2~indepV1+indepV2, NakayaData, transformation = 'Euclidean')
predict(lm2euc, NakayaData[, 3:4])
Index

*Topic datasets
  CarbonExample1Data, 4
  CarbonExample2Data, 5
  CarbonExample3Data, 6
  EyegazeData, 6
  FriedmanKohlerData1, 7
  FriedmanKohlerData2, 8
  NakayaData, 10

anova.lm2, 2, 9

BiDimRegression, 3, 9

CarbonExample1Data, 4
CarbonExample2Data, 5
CarbonExample3Data, 6

EyegazeData, 6

FriedmanKohlerData1, 7
FriedmanKohlerData2, 8

linear model, 9
lm2, 2–4, 8, 11

NakayaData, 10

predict.lm2, 11