Package ‘XLS’

October 12, 2022

Imports mpoly

Title A Modeling Approach that Optimizes Future Errors in Least Squares

Version 0.1.0

Maintainer Samet Sokel <a_s@eskisehir.edu.tr>

Description
Given the date column as an ascending entry, future errors are included in the sum of squares of error that should be minimized based on the number of steps and weights you determine. Thus, it is prevented that the variables affect each other's coefficients unrealistically.

License GPL (>= 3)

Encoding UTF-8

RoxygenNote 7.1.2

Suggests rmarkdown, knitr

BugReports https://github.com/sametsoekel/eXtreme-Least-Squares/issues

NeedsCompilation no

Author Samet Sokel [aut, cre] (<https://orcid.org/0000-0002-4429-5125>),
Yavuz Acar [aut, rev]

Repository CRAN

Date/Publication 2022-03-10 22:10:12 UTC

R topics documented:

xls.fit 2
xls.objfun 2
xls.prep 3

Index **4**

xls.fit — *Fitting an eXtreme Least Squares Model*

Description

Almost the same interface as `stats::lm`. Just includes two parameters more, error_weights and error_ahead_level

Usage

```
xls.fit(formula, data, error_weights = NULL, error_ahead_level = 4)
```

Arguments

`formula`	An object of class "formula": a symbolic description of the model to be fitted.
`data`	A "data.frame" (with no missing values) object containing the variables in the model.
`error_weights`	A numeric vector including error weights by order. If NULL, it is created automatically by error_ahead_level amount, decreasing at equal intervals.
`error_ahead_level`	An integer which represents how many steps further the parameters will be optimized for each data point.

Value

A `lm` object whose coefficients are optimized by the mentioned method.

Examples

```
df <- datasets::airquality

ordered_df <- df[with(df,order(Month,Day)),]

model <- xls.fit(Ozone ~ Solar.R + Wind + Temp,ordered_df,
error_weights = c(0.4,0.3,0.2,0.1),error_ahead_level = 4)
```

xls.objfun — *Preparing eXtreme Least Squares Nonlinear Objective Function*

Description

Automatically used in `xls.fit()` No need to use if the objective function is not specifically desired to be achieved.

Usage

```
xls.objfun(data, error_column_name, error_weights, error_ahead_level)
```

Arguments

- `data` — A data.frame object which is returned by `xls.prep`. Tip: `xls.prep`'s `.$data` sub object returns the data.frame
- `error_column_name` — Symbolic error column's name. By default, it is named "error_symbolic" by `xls.prep()`
- `error_weights` — A numeric vector including error weights by order.
- `error_ahead_level` — An integer which represents how many steps further the parameters will be optimized for each data point.

Value

A `function` object.

`xls.prep` — *Preparing eXtreme Least Squares Data*

Description

Automatically used in `xls.fit()` No need to use if the raw data is not specifically desired to be achieved.

Usage

```
xls.prep(formula, data, dependent_var)
```

Arguments

- `formula` — An object of class "formula": a symbolic description of the model to be fitted.
- `data` — A data.frame object.
- `dependent_var` — A character which is the same as left hand side variable in specified formula.

Value

A `list` object which contains a `data.frame` object to be modeled and character vector of independent variables.

Index

xls.fit, 2
xls.objfun, 2
xls.prep, 3