Package ‘MMRcaseselection’

June 3, 2020

Type Package

Title Case Classification and Selection Based on Regression Results

Version 0.1.0

Depends R (>= 2.10)

Description Researchers doing a mixed-methods analysis (nested analysis as developed by Lieberman (2005) <doi:10.1017/S0003055405051762>) can use the package for the classification of cases and case selection using results of a linear regression. One can designate cases as typical, deviant, extreme and pathway case and use different case selection strategies for the choice of a case belonging to one of these types.

License GPL-3

Encoding UTF-8

LazyData true

Imports stats, ggplot2

RoxygenNote 7.1.0

URL https://github.com/ingorohlfing/MMRcaseselection

BugReports https://github.com/ingorohlfing/MMRcaseselection

Suggests knitr, rmarkdown

VignetteBuilder knitr

Language en-US

NeedsCompilation no

Author Ingo Rohlfing [aut, cre] (<https://orcid.org/0000-0001-8715-4771>)

Maintainer Ingo Rohlfing <i.rohlfing@uni-koeln.de>

Repository CRAN

Date/Publication 2020-06-03 11:40:02 UTC
**R topics documented:**

- `extreme_on_x` ................................. 2
- `extreme_on_y` .................................. 3
- `most_deviant` ................................. 4
- `most_overpredicted` ......................... 5
- `most_typical` ................................. 5
- `most_underpredicted` ....................... 6
- `pathway` .......................................... 7
- `pathway_xvr` .................................... 8
- `predint` .......................................... 8
- `predint_plot` .................................... 9
- `residstd` ........................................ 10
- `residstd_plot` .................................. 10

**Index**

<table>
<thead>
<tr>
<th>extreme_on_x</th>
<th>Extremeness of cases on an independent variable</th>
</tr>
</thead>
</table>

**Description**

Extremeness of a case is calculated by the difference between a case’s value on the independent variable and the variable’s mean value.

**Usage**

`extreme_on_x(lmobject = NULL, ind_var = NULL)`

**Arguments**

- `lmobject` Object generated with `lm`
- `ind_var` Independent variable for which extremeness values should be calculated. Has to be entered as a character.

**Details**

Calculating the absolute value of the difference between the cases’ values and the variable’s mean value is proposed by Seawright, Jason (2016): The Case for Selecting Cases That Are Deviant or Extreme on the Independent Variable. *Sociological Methods & Research* 45 (3): 493-525. ([https://doi.org/10.1177/0049124116643556](https://doi.org/10.1177/0049124116643556))
extreme_on_y

Value
A dataframe with
- all variables in the linear model,
- absolute extremeness (absolute value of difference between variable score and mean value of variable),
- extremeness (difference between variable score and mean value of variable), which can be useful when the direction of extremeness is relevant.
The rows are ordered in decreasing order of the absolute extreme values.

Examples
```r
df <- lm(mpg ~ disp + wt, data = mtcars)
extreme_on_x(df, "wt")
```

---

extreme_on_y  Extremeness of cases on the dependent variable

Description
Extremeness of a case is calculated by the difference between a case’s value on the dependent variable and the variable’s mean value.

Usage
```r
extreme_on_y(lmobject)
```

Arguments
- `lmobject`: Object generated with `lm`

Details
Calculating the absolute value of the difference between the cases’ values and the variable’s mean value is proposed by Seawright, Jason (2016): The Case for Selecting Cases That Are Deviant or Extreme on the Independent Variable. Sociological Methods & Research 45 (3): 493-525. (https://doi.org/10.1177/0049124116643556)

Value
A dataframe with
- all variables in the linear model,
- absolute extremeness (absolute value of difference between variable score and mean value of variable),
- extremeness (difference between variable score and mean value of variable), which can be useful when the direction of extremeness is relevant.
The rows are ordered in decreasing order of the absolute extreme values.
Examples

```r
df <- lm(mpg ~ disp + wt, data = mtcars)
extreme_on_y(df)
```

---

**most_deviant**  
*Identification of the most deviant case*

**Description**

Identification of the most deviant case (= worst predicted case), based on regression estimates.

**Usage**

```r
most_deviant(lmobject)
```

**Arguments**

- `lmobject`  
  Object generated with `lm`

**Details**


**Value**

The most deviant case with the largest absolute residual of all cases.

**Examples**

```r
df <- lm(mpg ~ disp + wt, data = mtcars)
most_deviant(df)
```
**most_overpredicted**

*Identification of the most overpredicted case*

**Description**

The case with the largest negative difference between the observed value and the predicted value on the outcome. Depending on the research question, there might be a specific interest in the case for which the model performs worst and yields a larger predicted value.

**Usage**

```r
most_overpredicted(lmobject)
```

**Arguments**

- `lmobject` Object generated with `lm`

**Value**

The most overpredicted case with the largest negative residual (the most negative residual).

**Examples**

```r
df <- lm(mpg ~ disp + wt, data = mtcars)
most_overpredicted(df)
```

---

**most_typical**

*Identification of the most typical case*

**Description**

The most typical case (= best predicted case) based on regression estimates.

**Usage**

```r
most_typical(lmobject)
```

**Arguments**

- `lmobject` Object generated with `lm`

**Details**

**most_underpredicted**

**Value**

The most typical case having the smallest absolute residual of all cases.

**Examples**

```r
df <- lm(mpg ~ disp + wt, data = mtcars)
most_typical(df)
```

---

**most_underpredicted**  
**Identification of the most underpredicted case**

**Description**

The case with the largest positive difference between the observed value and the predicted value on the outcome. Depending on the research question, there might be a specific interest in the case for which the model performs worst and yields a smaller predicted value.

**Usage**

```r
most_underpredicted(lmobject)
```

**Arguments**

- `lmobject`  
  Object generated with `lm`

**Value**

The most underpredicted case with the largest positive residual (the most positive residual).

**Examples**

```r
df <- lm(mpg ~ disp + wt, data = mtcars)
most_underpredicted(df)
```
Description

Calculation of pathway values, defined as the difference between residuals of full model and reduced model lacking the pathway variable. The larger the difference, the more a case qualifies as a pathway case suitable for the analysis of mechanisms.

Usage

```r
pathway(full_model, reduced_model)
```

Arguments

- `full_model` Full model including covariate of interest (= pathway variable)
- `reduced_model` Reduced model excluding covariate of interest

Details

The difference between the absolute residuals of the full and reduced model follows the approach developed by Weller and Barnes (2014): *Finding Pathways: Mixed-Method Research for Studying Causal Mechanisms*. Cambridge: Cambridge University Press. [https://doi.org/10.1017/CBO9781139644501](https://doi.org/10.1017/CBO9781139644501).


Value

A dataframe with
- all full model variables,
- full model residuals (`full_resid`),
- reduced model residuals (`reduced_resid`),
- pathway values following Weller/Barnes (`pathway_wb`),
- pathway values following Gerring (`pathway_gvalue`),
- variable showing whether Gerring’s criterion for a pathway case is met (`pathway_gstatus`)

Examples

```r
df_full <- lm(mpg ~ disp + wt, data = mtcars)
df_reduced <- lm(mpg ~ wt, data = mtcars)
pathway(df_full, df_reduced)
```
pathway_xvr  
*Plot of residuals against pathway variable*

**Description**

Plot of residuals against pathway variable

**Usage**

`pathway_xvr(full_model, reduced_model, pathway_type)`

**Arguments**

- `full_model`: Full model including covariate of interest (= pathway variable)
- `reduced_model`: Reduced model excluding covariate of interest
- `pathway_type`: Type of pathway values. `pathway_wb` are pathway values proposed by Weller and Barnes. `pathway_gvalue` are values as calculated by Gerring.

**Value**

A plot of the chosen type of pathway values against the pathway variable created with `ggplot2`.

**Examples**

```r
df_full <- lm(mpg ~ disp + wt, data = mtcars)
df_reduced <- lm(mpg ~ wt, data = mtcars)
pathway_xvr(df_full, df_reduced, pathway_type = "pathway_wb")
```

---

predint  
*Classification of cases as typical and deviant using a prediction interval.*

**Description**

Case are designated as typical (= well predicted) and deviant (= badly predicted) based on the prediction interval. The x% prediction interval represents the range that we expect to include x% of outcome values in repeated samples. For example, a 95% prediction interval ranging from 0-5 conveys that 95% of future outcome values will be in the range of 0-5. If the observed outcome is inside the prediction interval, the case is classified (or designated) as typical and as deviant otherwise.

**Usage**

`predint(lmobject, piwidth = 0.95)`
**Arguments**

- `lmObject` Object generated with `lm`
- `piwidth` Width of the prediction interval (default is 0.95).

**Details**


**Value**

A dataframe with the observed outcome, fitted outcome, upper and lower bound of the % prediction interval and classification of cases as typical or deviant.

**Examples**

```r
df <- lm(mpg ~ disp + wt, data = mtcars)
predint(df, piwidth = 0.9)
```

---

**predint_plot**

*Plot of typical and deviant cases with prediction intervals*

**Description**


**Usage**

```r
predint_plot(pred_df)
```

**Arguments**

- `pred_df` A dataframe created with `predint`.

**Value**

A plot of the observed outcome against the fitted outcome with prediction intervals and case classifications. Created with `ggplot2`.

**Examples**

```r
df <- lm(mpg ~ disp + wt, data = mtcars)
predint_status <- predint(df, piwidth = 0.9)
predint_plot(predint_status)
```
Classification of cases as typical and deviant using the standard deviation of the residuals.

Description

The share of the standard deviation of the residuals is used to designate cases as typical or deviant.

Usage

`residstd(lmobject, stdshare = 1)`

Arguments

- `lmobject`: Object generated with `lm`
- `stdshare`: Share of standard deviation of residuals distinguishing between typical and deviant cases (default is 1).

Details


Value

A dataframe with the observed outcome, fitted outcome, residual standard deviation and classification of cases as typical or deviant.

Examples

```r
df <- lm(mpg ~ disp + wt, data = mtcars)
residstd(df, stdshare = 1)
```

Plot of typical and deviant cases based on residuals’ standard deviation

Description

Plot of typical and deviant cases based on residuals’ standard deviation

Usage

`residstd_plot(resid_df)`
**Arguments**

- **resid_df**
  A dataframe created with `residstd`.

**Value**

A plot of the observed outcome against the fitted outcome with interval and case classifications. Created with `ggplot2`.

**Examples**

```r
df <- lm(mpg ~ disp + wt, data = mtcars)
residstd_status <- residstd(df, stdshare = 1)
residstd_plot(residstd_status)
```
Index

extreme_on_x, 2
extreme_on_y, 3

ggplot2, 8, 9, 11

lm, 2-6, 9, 10

most_deviant, 4
most_overpredicted, 5
most_typical, 5
most_underpredicted, 6

pathway, 7
pathway_xvr, 8
predint, 8, 9
predint_plot, 9

residstd, 10, 11
residstd_plot, 10