Package 'lm.beta'

January 9, 2023

Type Package

Title Add Standardized Regression Coefficients to Linear-Model-Objects

Version 1.7-1

Date 2023-01-01

Maintainer Stefan Behrendt <r@behrendt-stefan.de>

Description Adds standardized regression coefficients to objects created by 'lm'. Also extends the S3 methods 'print', 'summary' and 'coef' with additional boolean argument 'standardized' an provides 'xtable'-support.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Imports xtable

Suggests knitr

VignetteBuilder knitr

Author Stefan Behrendt [aut, cre]

Date/Publication 2023-01-09 13:00:02 UTC

R topics documented:

`lm.beta-package` *Add Standardized Regression Coefficients to* `lm`*-Objects*

Description

Adds standardized regression coefficients to objects created by `lm`.

Also extends the S3 methods `print`, `summary` and `coef` with additional boolean argument standardized.

Please regard:

Package `lm.beta` works in the way of common statistical softwares like SPSS by standardizing the coefficients after estimating them using the standard deviations or similar measures of the used variables. So there are unstandardized and standardized coefficients available simultaneously.

Standardizing before estimating is not (yet) available in this package, but by using the command `scale` you can do this by using basic commands. Hereby please regard that the option `center` influences the way of interpretation of the intercept.

Package `lm.beta` standardizes all coefficients disregarding the use in interpretation. In this version, all types of scales of the variables (metrical, categorical, ...), all types of contrasts, interaction effects and additional terms on both sides of the formula can be handled if `lm` can handle them. The sensitive use in interpretation has to be regarded by the user.

Details

Package:	lm.beta
Type:	Package
Version:	1.6-1
Date:	2018-06-20
License:	GPL(>=2)

Author(s)

Stefan Behrendt <r@behrendt-stefan.de>

References

Urban, D., Mayerl, J., Sackmann, R. (Hrsg.) *Regressionsanalyse : Theorie, Technik und Anwendung*, VS-Verlag, 4. Aufl.

Vittinghoff, E. et al (2005) *Regression methods in biostatistics: Linear, logistic, survival, and repeated measures models*, Springer, p 75

See Also

`lm.beta`, `lm`

Examples

```
## Taken from lm help
##
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2, 10, 20, labels = c("Ctl","Trt"))
weight <- c(ctl, trt)
lm.D9 <- lm(weight ~ group)

# standardize
lm.D9.beta <- lm.beta(lm.D9)
print(lm.D9.beta)
summary(lm.D9.beta)
coef(lm.D9.beta)
```

`coef.lm.beta` *Print Coefficients of Standardized Linear-Model-Object*

Description

S3-method `coef` for object `lm.beta`.

Usage

```
## S3 method for class 'lm.beta'
coef(object, standardized = TRUE, ...)
```

Arguments

`object`	An R object of class `lm.beta`
`standardized`	Logical. Should the standardized values be returned?
`...`	additional arguments. (Currently ignored.)

Details

If `standardized=FALSE`, the unstandardized regression coefficients are printed like if calling standard `coef.lm`-method, else (the standard value) the standardized regression coefficients are printed.

Value

named numeric Vector of (un)standardized regression coefficients.

Author(s)

Stefan Behrendt, <r@behrendt-stefan.de>

See Also

lm.beta for creating the lm.beta-object.

Examples

```
## Taken from lm help
##
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2, 10, 20, labels = c("Ctl","Trt"))
weight <- c(ctl, trt)
lm.D9 <- lm(weight ~ group)

# standardize
lm.D9.beta <- lm.beta(lm.D9)
coef(lm.D9.beta)
coef(lm.D9.beta,standardized=FALSE)
```

---

lm.beta *Add Standardized Regression Coefficients to Linear-Model-Objects*

Description

Adds standardized regression coefficients to objects created by lm.

Usage

```
lm.beta(object, complete.standardization = FALSE)
```

Arguments

object	An R object of type lm
complete.standardization	Logical. (See Details.)

Details

Calculates the standardized regression coefficients by common method used for example in SPSS. For translating the formula, functions model.matrix (for the right-hand side) and model.frame (for the left-hand side) are used. Additionally the case weights are regarded. So all options saved in the lm-object are supported.

In the case of models with intercept, the standardization results in the same estimates as lm(..., data = scale(data)).

In the case of models without intercept, there are two different types of standardization available. (1) Complete standardization (complete.standardization = TRUE) results in the same estimates

as `lm(..., data = scale(data))` and therefore results in the same estimates as the same model with intercept. (2) Incomplete standardization (`complete.standardization = FALSE`, the standard value) results in the same estimates as `lm(..., data = scale(data, center = FALSE))`. This estimation is implemented in IBM SPSS Statistics. For a theoretical justification see *Eisenhauer 2003*.

Please regard:

Package `lm.beta` standardizes the coefficients after estimating them using the standard deviations or similar measures of the used variables. So there are unstandardized and standardized coefficients available simultaneously.

Standardizing before estimating is not (yet) available in this package, but by using the function `scale` you can do this by using basic commands. Hereby please regard that the option `center` influences the way of interpretation of the intercept.

Package `lm.beta` standardizes all coefficients disregarding the use in interpretation. In this version, all types of scales of the variables (metrical, categorical, ...), all types of contrasts, interaction effects and additional terms on both sides of the formula can be handled if `lm` can handle them. The sensitive use in interpretation has to be regarded by the user.

Value

A list of class `lm.beta` like a `lm`-object extended by

- **standardized.coefficients** named vector of the standardized coefficients.

Note

Some S3 methods, where standardized coefficients mind, are extended, the others work unchanged.

Author(s)

Stefan Behrendt, <r@behrendt-stefan.de>

References

Eisenhauer, J.G. (2003). Regression through the Origin. In *Teching Statistics*, 25(3).

Urban, D., Mayerl, J., Sackmann, R. (Hrsg.) *Regressionsanalyse : Theorie, Technik und Anwendung*. VS-Verlag, 4th ed.

Vittinghoff, E. et al (2005) *Regression methods in biostatistics: Linear, logistic, survival, and repeated measures models*, Springer, p 75

See Also

`lm` for creating the demanded object and `print.lm.beta`, `summary.lm.beta` and `coef.lm.beta` as well as `xtable.lm.beta` for extended S3-methods.

Examples

```
## Taken from lm help
##
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2, 10, 20, labels = c("Ctl","Trt"))
weight <- c(ctl, trt)
lm.D9 <- lm(weight ~ group)

# standardize
lm.D9.beta <- lm.beta(lm.D9)
print(lm.D9.beta)
summary(lm.D9.beta)
coef(lm.D9.beta)
xtable::xtable(lm.D9.beta)
```

`print.lm.beta` *Print Standardized Linear-Model-Object*

Description

S3-method `print` for object `lm.beta`.

Usage

```
## S3 method for class 'lm.beta'
print(x, standardized = TRUE, ...)
```

Arguments

`x`	An R object of class `lm.beta`
`standardized`	Logical. Should the standardized values be printed?
`...`	Additional arguments to pass to `print.lm`

Details

If `standardized=FALSE`, the standard `print.lm`-method is called, else (the standard value) the regression coefficients are replaced by the standardized ones.

The additional arguments are in case of `standardized=FALSE` passed to `print.lm`, else they are passed to `print` for classes `call` and `vector`.

Value

Original object.

Author(s)

Stefan Behrendt, <r@behrendt-stefan.de>

See Also

lm.beta for creating the lm.beta-object.

Examples

```
## Taken from lm help
##
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2, 10, 20, labels = c("Ctl","Trt"))
weight <- c(ctl, trt)
lm.D9 <- lm(weight ~ group)

# standardize
lm.D9.beta <- lm.beta(lm.D9)
print(lm.D9.beta)
print(lm.D9.beta,standardized=FALSE)
```

---

summary.lm.beta *Summarize Standardized Linear-Model-Object*

Description

S3-method summary for object lm.beta.

Usage

```
## S3 method for class 'lm.beta'
summary(object, standardized = TRUE, ...)
```

Arguments

object	An R object of class lm.beta
standardized	Logical. Should the standardized values be integrated?
...	Additional arguments to pass to summary.lm

Details

If standardized=FALSE, the standard summary.lm-method is called, else (the standard value) the standardized regression coefficients are added into the coefficient table.

The additional arguments are passed to summary.lm.

Value

Adapted summary.lm-object, in case of standardized=TRUE with additional class summary.lm.beta.

Author(s)

Stefan Behrendt, <r@behrendt-stefan.de>

See Also

lm.beta for creating the lm.beta-object.

Examples

```
## Taken from lm help
##
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2, 10, 20, labels = c("Ctl","Trt"))
weight <- c(ctl, trt)
lm.D9 <- lm(weight ~ group)

# standardize
lm.D9.beta <- lm.beta(lm.D9)
summary(lm.D9.beta)
summary(lm.D9.beta,standardized=FALSE)
```

---

xtable.lm.beta — *Coerce Coefficent Table of Standardized Linear-Model-Object to LaTeX- and HTML-Tables*

Description

S3-method xtable for object lm.beta and summary.lm.beta.

Usage

```
## S3 method for class 'lm.beta'
xtable(x, standardized = TRUE, caption = NULL, label = NULL,
                          align = NULL, digits = NULL, display = NULL,
                          auto = FALSE, ...)
## S3 method for class 'summary.lm.beta'
xtable(x, caption = NULL, label = NULL, align = NULL,
                                  digits = NULL, display = NULL, auto = FALSE,
                                  ...)
```

Arguments

x	An R object of class 'lm.beta' or 'summary.lm.beta'.
standardized	Logical. Should the standardized values be returned?
caption	Character vector of length 1 or 2 containing the table's caption or title. If length is 2, the second item is the "short caption" used when LaTeX generates a "List of Tables". Set to NULL to suppress the caption. Default value is NULL.
label	Character vector of length 1 containing the LaTeX label or HTML anchor. Set to NULL to suppress the label. Default value is NULL.
align	Character vector of length equal to the number of columns of the resulting table, indicating the alignment of the corresponding columns. Also, "|" may be used to produce vertical lines between columns in LaTeX tables, but these are effectively ignored when considering the required length of the supplied vector. If a character vector of length one is supplied, it is split as strsplit(align, "")[[1]] before processing. Since the row names are printed in the first column, the length of align is one greater than ncol(x) if x is a data.frame. Use "l", "r", and "c" to denote left, right, and center alignment, respectively. Use "p{3cm}" etc. for a LaTeX column of the specified width. For HTML output the "p" alignment is interpreted as "l", ignoring the width request. Default depends on the class of x.
digits	Numeric vector of length equal to one (in which case it will be replicated as necessary) or to the number of columns of the resulting table **or** matrix of the same size as the resulting table, indicating the number of digits to display in the corresponding columns. Since the row names are printed in the first column, the length of the vector digits or the number of columns of the matrix digits is one greater than ncol(x) if x is a data.frame. Default depends on the class of x. If values of digits are negative, the corresponding values of x are displayed in scientific format with abs(digits) digits.
display	Character vector of length equal to the number of columns of the resulting table, indicating the format for the corresponding columns. Since the row names are printed in the first column, the length of display is one greater than ncol(x) if x is a data.frame. These values are passed to the formatC function. Use "d" (for integers), "f", "e", "E", "g", "G", "fg" (for reals), or "s" (for strings). "f" gives numbers in the usual xxx.xxx format; "e" and "E" give n.ddde+nn or n.dddE+nn (scientific format); "g" and "G" put x[i] into scientific format only if it saves space to do so. "fg" uses fixed format as "f", but digits as number of *significant* digits. Note that this can lead to quite long result strings. Default depends on the class of x.
auto	Logical, indicating whether to apply automatic format when no value is passed to align, digits, or display. This 'autoformat' (based on xalign, xdigits, and xdisplay) can be useful to quickly format a typical matrix or data.frame. Default value is FALSE.
...	Additional arguments. (Currently ignored.)

Details

see xtable

Value

see xtable

Author(s)

Stefan Behrendt, <r@behrendt-stefan.de>

See Also

lm.beta for creating the lm.beta-object.

Examples

```
## Taken from lm help
##
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2, 10, 20, labels = c("Ctl","Trt"))
weight <- c(ctl, trt)
lm.D9 <- lm(weight ~ group)

# standardize
lm.D9.beta <- lm.beta(lm.D9)
xtable::xtable(lm.D9.beta)
```

Index