Package ‘SIMPLE.REGRESSION’

May 7, 2026

Type Package

Title OLS, Moderated, Logistic, and Count Regressions Made Simple

Version 0.3.1

Date 2026-01-19

Author Brian P. O'Connor [aut, cre]

Maintainer Brian P. O'Connor <brian.oconnor@ubc.ca>

Description Provides SPSS- and SAS-like output for least squares multiple regression, logistic regression, and count variable regressions. Detailed output is also provided for OLS moderated regression, interaction plots, and Johnson-Neyman regions of significance. The output includes standardized coefficients, partial and semi-partial correlations, collinearity diagnostics, plots of residuals, and detailed information about simple slopes for interactions. The output for some functions includes Bayes Factors and, if requested, regression coefficients from Bayesian Markov Chain Monte Carlo analyses. There are numerous options for model plots. The REGIONS_OF_SIGNIFICANCE function also provides Johnson-Neyman regions of significance and plots of interactions for both lm and lme models. There is also a function for partial and semipartial correlations and a function for conducting Cohen's set correlation analyses.

Imports graphics, stats, utils, nlme, MASS, BayesFactor, pscl, rstanarm

Depends R (>= 2.10)

LazyLoad yes

LazyData yes

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2026-01-21 11:50:09 UTC

Contents

`SIMPLE.REGRESSION-package`

SIMPLE.REGRESSION

Description

Provides SPSS- and SAS-like output for least squares multiple regression, logistic regression, and count variable regressions. Detailed output is also provided for OLS moderated regression, interaction plots, and Johnson-Neyman regions of significance. The output includes standardized coefficients, partial and semi-partial correlations, collinearity diagnostics, plots of residuals, and detailed information about simple slopes for interactions. The output for some functions includes Bayes Factors and, if requested, regression coefficients from Bayesian Markov Chain Monte Carlo (MCMC) analyses. There are numerous options for model plots.

The REGIONS_OF_SIGNIFICANCE function provides Johnson-Neyman regions of significance and plots of interactions for both lm and lme models (lme models are from the nlme package).

There is also a function for partial and semipartial correlations and a function for conducting Cohen's set correlation analyses.

References

Bauer, D. J., & Curran, P. J. (2005). Probing interactions in fixed and multilevel regression: Inferential and graphical techniques. *Multivariate Behavioral Research, 40(3),* 373-400.

Cohen, J. (1982). Set correlation as a general multivariate data-analytic method. *Multivariate Behavioral Research, 17(3)*, 301-341.

Cohen, J., Cohen, P., West, S. G., & Aiken, L. S. (2003). *Applied multiple regression/correlation analysis for the behavioral sciences* (3rd ed.). Lawrence Erlbaum Associates.

Darlington, R. B., & Hayes, A. F. (2017). *Regression analysis and linear models: Concepts, applications, and implementation.* Guilford Press.

Dunn, P. K., & Smyth, G. K. (2018). *Generalized linear models with examples in R.* Springer.

Hayes, A. F. (2018a). *Introduction to mediation, moderation, and conditional process analysis: A regression-based approach* (2nd ed.). Guilford Press.

Huitema, B. (2011). *The analysis of covariance and alternatives: Statistical methods for experiments, quasi-experiments, and single-case studies.* John Wiley & Sons.

Johnson, P. O., & Fey, L. C. (1950). The Johnson-Neyman technique, its theory, and application. *Psychometrika, 15,* 349-367.

Lorah, J. A. & Wong, Y. J. (2018). Contemporary applications of moderation analysis in counseling psychology. *Counseling Psychology, 65(5),* 629-640.

Orme, J. G., & Combs-Orme, T. (2009). *Multiple regression with discrete dependent variables.* Oxford University Press.

Pedhazur, E. J. (1997). *Multiple regression in behavioral research: Explanation and prediction.* (3rd ed.). Wadsworth Thomson Learning.

`COUNT_REGRESSION` *Count data regression*

Description

Provides SPSS- and SAS-like output for count data regression, including Poisson, quasi-Poisson, negative binomial, zero-inflated poisson, zero-inflated negative binomial, hurdle poisson, and hurdle negative binomial models. The output includes model summaries, classification tables, omnibus tests of the model coefficients, overdispersion tests, model effect sizes, the model coefficients, correlation matrix for the model coefficients, collinearity statistics, and casewise regression diagnostics.

Usage

```
COUNT_REGRESSION(data, DV, forced = NULL, hierarchical = NULL, formula=NULL,
                 model_type = 'poisson',
                 offset = NULL,
                 CI_level = 95,
                 MCMC_options = list(MCMC = FALSE, Nsamples = 10000,
                                     thin = 1, burnin = 1000,
                                     HDI_plot_est_type = 'raw'),
                 plot_type = 'residuals',
                 GoF_model_types = TRUE,
                 verbose = TRUE )
```

Arguments

`data` A dataframe where the rows are cases and the columns are the variables.

`DV` The name of the dependent variable.
Example: DV = 'outcomeVar'.

`forced` (optional) A vector of the names of the predictor variables for a forced/simultaneous entry regression. The variables can be numeric or factors.
Example: forced = c('VarA', 'VarB', 'VarC')

`hierarchical` (optional) A list with the names of the predictor variables for each step of a hierarchical regression. The variables can be numeric or factors.
Example: hierarchical = list(step1=c('VarA', 'VarB'), step2=c('VarC', 'VarD'))

`formula` (optional) Text for an R formula. Useful for testing for interactions.
Example: formula = "Aggressive_Behavior ~ Maternal_Harshness * Resiliency"")

`model_type` (optional) The name of the error distribution to be used in the model. The options are:

- "poisson" (the default),
- "quasipoisson",
- "negbin", for negative binomial,
- "zinfl_poisson", for zero-inflated poisson,
- "zinfl_negbin", for zero-inflated negative binomial, or
- "hurdle_poisson", for hurdle poisson, or
- "hurdle_negbin", for hurdle negative binomial.

Example: model_type = 'quasipoisson'

`offset` (optional) The name of the offset variable, if there is one. This variable should be in the desired metric (e.g., log). No transformation of an offset variable is performed internally.
Example: offset = 'Varname'

`CI_level` (optional) The confidence interval for the output, in whole numbers. The default is 95.

`MCMC_options` (optional) A list specifying the following options for Bayesian MCMC analyses: (1) "MCMC", Should MCMC analyses be conducted? The options are TRUE or FALSE; (2) "Nsamples", for the number of iterations or samples from the

posterior distribution; (3) "thin", for the chain-thinning interval; (4) "burnin", for the burnin period, i.e., the number of initial samples that should be dropped from the chains; and (5) "HDI_plot_est_type", for the kind of regression estimates that will appear in any requested HDI plots. The options are "standardized" or "raw".
Example: MCMC_options = list(MCMC = TRUE, Nsamples = 10000, thin = 1, burnin = 1000, HDI_plot_est_type = 'standardized')

`plot_type` (optional) The kind of plots, if any. The options are:

- 'residuals' (the default),
- 'diagnostics', for regression diagnostics,
- 'Bayes_HDI' (for MCMC posterior distributions), and
- 'none', for no plots.

Example: plot_type = 'diagnostics'

`GoF_model_types` (optional) Should fit coefficients be computed for multiple model types (Poisdon, quasi-Poisson, negative binomial, zero-inflated Poisson, zero-inflated negative binomial, and hurdle)? The default is TRUE.

`verbose` (optional) Should detailed results be displayed in console?
The options are: TRUE (default) or FALSE. If TRUE, plots of residuals are also produced.

Details

This function uses the glm function from the stats package, the negative.binomial function from the MASS package, and the zeroinfl and hurdle functions from the pscl package (Zeileis, Kleiber, & Jackman, 2008). It supplements the output from these packages with additional statistics and in formats that resemble SPSS and SAS output. The predictor variables can be numeric or factors.

The function assigns contrasts (dummy codes) to factor variables that do not already have contrasts. The baseline group for the dummy codes is determined by the alphabetic/numeric order of the factor levels. If the terms "control" or "Control" or "baseline" or "Baseline" appear in the names of a factor level, then that factor level is used as the dummy codes baseline.

The following descriptions of zero-inflated and hurdle models were provided by Atkins and Baldwin (2013), by Friendly and Meyer (2016), and at https://stats.oarc.ucla.edu/r/dae/zinb/:

Zero-inflated and hurdle models are used when there is an overabundance of zero counts (excessive, or over-dispersed zero counts). Both have two submodels, one related to the zeroes and a second related to the counts. The key difference between hurdle and zero-inflated models is how they handle zeroes: Hurdle models cleanly divide the models, with all zeroes accounted for in the logistic regression, whereas zero-inflated models treat the observed zeroes as a mixture from two latent classes that produce zeroes.

Zero-inflated models assume that the observed counts arise from a mixture of two latent classes of observations: some structural zeros for whom the DV will always be zero, and a second class for whom the observed count may be zero or above zero. The excess zeros are assumed to have been generated by a separate process from the count values and it is assumed that the excess zeros can be modeled independently.

For example, imagine that wildlife biologists want to model how many fish are being caught by visitors to a park. Some visitors do not fish (structural zeros), but there is no data on whether a person fished or not. Some visitors who did fish did not catch any fish so there are excess zeros in the data because of the people that did not fish. The variables that predict whether or not visitors fished may or may not be the same variables that predict how many fish visitors caught. Separate models for the zeroes and for the counts can be examined. Zero-inflated models assume that zero values are due to two dierent processes, e.g., that a visitor has gone fishing vs. not gone fishing. If not gone fishing, the only outcome possible is zero. If gone fishing, it is then a count process. The two parts of the a zero-inflated model are a binary (logistic) model and a count model (Poisson or negative binomial). The expected counts are expressed as a combination of the two processes.

For the zero (logistic) portion of zero-inflated models, the predicted outcomes are the zero values (excess zeros) for the DV. A positive coefficient (B) for a predictor thus means that as values on a predictor increase, the probability of observing a zero value for the DV increases.

Hurdle models also deal with an excess of zero DV values, but without assuming that zero values arise from a mixture of two latent classes of observations. Imagine that it is (somehow) known that every visitor to a park did in fact fish. There could be an excess of zeroes because many of the visitors did not know how to fish. A separate logistic regression submodel is used to distinguish zero counts from the larger counts. The submodel for the positive counts is a truncated Poisson or negative-binomial model, excluding the zero counts. In other words, there is one process and submodel accounting for the zero counts and a separate process accounting for the positive counts, once the zero hurdle has been crossed. In zero-inflation models, the first process generates only extra zeros beyond those of the regular Poisson distribution. For hurdle models, the first process generates all of the zeros. In hurdle models, the zero values are considered fully observed, rather than latent.

For the zero (logistic) portion of hurdle models, the predicted outcomes are for going from zero to greater than zero values for the DV. A positive coefficient (B) for a predictor thus means that as values on a predictor increase, the probability of crossing the hurdle (obtaining a value higher than zero) for the DV increases.

Predicted values, for selected levels of the predictor variables, can be produced and plotted using the PLOT_MODEL funtion in this package.

The Bayesian MCMC analyses can be time-consuming for larger datasets. The MCMC analyses are conducted using functions, and their default settings, from the rstanarm package (Goodrich, Gabry, Ali, & Brilleman, 2024). Family = 'quasipoisson' analyses are currently not possible for the MCMC analyses. model_type = 'poisson' is therefore used instead.

The Bayesian MCMC analyses are currently not available for zero-inflated poisson and zero-inflated negative binomial models.

The MCMC results can be verified using the model checking functions in the rstanarm package (e.g., Muth, Oravecz, & Gabry, 2018).

Good sources for interpreting count data regression residuals and diagnostics plots:

- rpubs.com/benhorvath
- library.virginia.edu

Value

An object of class "COUNT_REGRESSION". The object is a list containing the following possible components:

`model` All of the glm function output for the regression model.

`modelsum`	All of the summary.glm function output for the regression model.
`modeldata`	All of the predictor and outcome raw data that were used in the model, along with regression diagnostic statistics for each case.
`collin_diags`	Collinearity diagnostic coefficients for models without interaction terms.
`chain_dat`	The MCMC chains.
`Bayes_HDIs`	The Bayesian HDIs.

Author(s)

Brian P. O'Connor

References

Atkins, D. C., Baldwin, S. A., Zheng, C., Gallop, R. J., & Neighbors, C. (2013). A tutorial on count regression and zero-altered count models for longitudinal substance use data. *Psychology of Addictive Behaviors, 27(1),* 166177. https://doi.org/10.1037/a0029508

Atkins, D. C., & Gallop, R. J. (2007). Rethinking how family researchers model infrequent outcomes: A tutorial on count regression and zero-inflated models. *Journal of Family Psychology, 21(4),* 726-735.

Beaujean, A. A., & Grant, M. B. (2019). Tutorial on using regression models with count outcomes using R. *Practical Assessment, Research, and Evaluation: Vol. 21, Article 2.*

Coxe, S., West, S.G., & Aiken, L.S. (2009). The analysis of count data: A gentle introduction to Poisson regression and its alternatives. *Journal of Personality Assessment, 91,* 121-136.

Dunn, P. K., & Smyth, G. K. (2018). *Generalized linear models with examples in R.* Springer.

Friendly, M., & Meyer, D. (2016). *Discrete Data Analysis with R: Visualization and Modeling Techniques for Categorical and Count Data.* Chapman and Hall/CRC.

Hardin, J. W., & Hilbe, J. M. (2007). *Generalized linear models and extensions.* Stata Press.

Muth, C., Oravecz, Z., & Gabry, J. (2018). User-friendly Bayesian regression modeling: A tutorial with rstanarm and shinystan. *The Quantitative Methods for Psychology, 14(2),* 99119. https://doi.org/10.20982/tqmp.14.2.p099

Orme, J. G., & Combs-Orme, T. (2009). *Multiple regression with discrete dependent variables.* Oxford University Press.

Rindskopf, D. (2023). Generalized linear models. In H. Cooper, M. N. Coutanche, L. M. McMullen, A. T. Panter, D. Rindskopf, & K. J. Sher (Eds.), *APA handbook of research methods in psychology: Data analysis and research publication,* (2nd ed., pp. 201-218). American Psychological Association.

Zeileis, A., Kleiber, C., & Jackman, S. (2008). Regression Models for Count Data in R. *Journal of Statistical Software, 27(8).* https://www.jstatsoft.org/v27/i08/.

Examples

```
# for Kremelburg, 2011, p.262 & p. 282: poisson regression
COUNT_REGRESSION(data=data_Kremelburg_2011, DV='OVRJOYED',
                 forced=c('AGE','EDUC','REALRINC','female'))


# for Kremelburg, 2011, p. 266-267 & p. 284:
# negative binomial regression & with Bayesian MCMC analyses & HDI plots
COUNT_REGRESSION(data=data_Kremelburg_2011, DV='HURTATWK',
                 forced=c('AGE','EDUC','REALRINC','female'),
                 model_type = 'negbin',
                 MCMC_options = list(MCMC = TRUE, Nsamples = 10000,
                                     thin = 1, burnin = 1000,
                                     HDI_plot_est_type = 'raw'),
                 plot_type = 'Bayes_HDI')

# for Orme, 2009, p. 160: with an offset variable
COUNT_REGRESSION(data=data_Orme_2009_5, DV='NumberAdopted', forced=c('Married'),
                 offset='lnYearsFostered')

# zero-inflated poisson regression
COUNT_REGRESSION(data=data_Kremelburg_2011, DV='HURTATWK',
                 forced=c('AGE','EDUC','REALRINC','female'),
                 model_type = 'zinfl_poisson',
                 plot_type = 'diagnostics')

# hurdle negative binomial regression
COUNT_REGRESSION(data=data_Kremelburg_2011, DV='HURTATWK',
                 forced=c('AGE','EDUC','REALRINC','female'),
                 model_type = 'hurdle_negbin',
                 plot_type = 'diagnostics')
```

`data_Bauer_Curran_2005` *data_Bauer_Curran_2005*

Description

Multilevel moderated regression data from Bauer and Curran (2005).

Usage

```
data(data_Bauer_Curran_2005)
```

Source

Bauer, D. J., & Curran, P. J. (2005). Probing interactions in fixed and multilevel regression: Inferential and graphical techniques. *Multivariate Behavioral Research, 40(3),* 373-400.

Examples

```
head(data_Bauer_Curran_2005)

# for Bauer & Curran, 2005, p. 395
HSBmod <-nlme::lme(MathAch ~ Sector + CSES + CSES:Sector,
                   data = data_Bauer_Curran_2005,
                   random = ~1 + CSES|School, method = "ML")
summary(HSBmod)

REGIONS_OF_SIGNIFICANCE(model=HSBmod,
                        plot_title='Johnson-Neyman Regions of Significance',
                        Xaxis_label='Child SES',
                        Yaxis_label='Slopes of School Sector on Math achievement')
```

---

data_Bodner_2016 *data_Bodner_2016*

Description

Moderated regression data used by Bodner (2016) to illustrate the tumble graphs method of plotting interactions. The data were also used by Bauer and Curran (2005).

Usage

```
data(data_Bodner_2016)
```

Source

Bodner, T. E. (2016). Tumble Graphs: Avoiding misleading end point extrapolation when graphing interactions from a moderated multiple regression analysis. *Journal of Educational and Behavioral Statistics, 41(6),* 593-604.

Bauer, D. J., & Curran, P. J. (2005). Probing interactions in fixed and multilevel regression: Inferential and graphical techniques. *Multivariate Behavioral Research, 40(3),* 373-400.

Examples

```
head(data_Bodner_2016)

# replicates p 599 of Bodner (2016)
MODERATED_REGRESSION(data=data_Bodner_2016, DV='math90',
                     IV='Anti90', IV_range='tumble',
                     MOD='Hyper90', MOD_levels='quantiles',
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     COVARS=c('age90month','female','grade90','minority'),
                     center = FALSE,
                     plot_type = 'interaction')
```

`data_Chapman_Little_2016`
data_Chapman_Little_2016

Description

Moderated regression data from Chapman and Little (2016).

Usage

```
data(data_Chapman_Little_2016)
```

Source

Chapman, D. A., & Little, B. (2016). Climate change and disasters: How framing affects justifications for giving or withholding aid to disaster victims. *Social Psychological and Personality Science, 7,* 13-20.

Examples

```
head(data_Chapman_Little_2016)

# the data used by Hayes (2018, Introduction to Mediation, Moderation, and
# Conditional Process Analysis: A Regression-Based Approach), replicating p. 239
MODERATED_REGRESSION(data=data_Chapman_Little_2016, DV='justify',
                     IV='frame', IV_range='tumble',
                     MOD='skeptic', MOD_levels='AikenWest',
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     center = FALSE,
                     plot_type = 'regions')
```

`data_Cohen_Aiken_West_2003_7`
data_Cohen_Aiken_West_2003_7

Description

Moderated regression data for a continuous predictor and a continuous moderator from Cohen, Cohen, West, & Aiken (2003, Chapter 7).

Usage

```
data(data_Cohen_Aiken_West_2003_7)
```

Source

Cohen, J., Cohen, P., West, S. G., & Aiken, L. S. (2003). *Applied multiple regression/correlation analysis for the behavioral sciences* (3rd ed.). Lawrence Erlbaum Associates.

Examples

```
head(data_Cohen_Aiken_West_2003_7)

# replicates p 276 of Chapter 7 of Cohen, Cohen, West, & Aiken (2003)
MODERATED_REGRESSION(data=data_Cohen_Aiken_West_2003_7, DV='yendu',
                     IV='xage', IV_range='tumble',
                     MOD='zexer', MOD_levels='AikenWest',
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     center = TRUE,
                     plot_type = 'regions')
```

---

data_Cohen_Aiken_West_2003_9 *data_Cohen_Aiken_West_2003_9*

Description

Moderated regression data for a continuous predictor and a categorical moderator from Cohen, Cohen, West, & Aiken (2003, Chapter 9).

Usage

```
data(data_Cohen_Aiken_West_2003_9)
```

Source

Cohen, J., Cohen, P., West, S. G., & Aiken, L. S. (2003). *Applied multiple regression/correlation analysis for the behavioral sciences* (3rd ed.). Lawrence Erlbaum Associates.

Examples

```
head(data_Cohen_Aiken_West_2003_9)

# replicates p 376 of Chapter 9 of Cohen, Cohen, West, & Aiken (2003)
MODERATED_REGRESSION(data=data_Cohen_Aiken_West_2003_9, DV='SALARY',
                     IV='PUB', IV_range='tumble',
                     MOD='DEPART_f', MOD_type = 'factor', MOD_levels='AikenWest',
                     MOD_reflevel = 'psychology',
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     center = TRUE,
                     plot_type = 'interaction')
```

data_DeLeo_2013 *data_DeLeo_2013*

Description

A dataset with multiple continuous variables that simulate the data from De Leo and Wulfert (2013). The dataset is used in the examples for the present PARTIAL_COR and SET_CORRELATION functions.

Usage

```
data(data_DeLeo_2013)
```

Source

De Leo, J. A., & Wulfert, E. (2013). Problematic internet use and other risky behaviors in college students: An application of problem-behavior theory. *Psychology of Addictive Behaviors, 27(1),* 133-141.

Examples

```
head(data_DeLeo_2013)

# bipartial
SET_CORRELATION(data=data_DeLeo_2013,
                IVs = c('Grade_Point_Average','Family_Morals','Social_Support',
                        'Intolerance_of_Deviance','Impulsivity',
                        'Social_Interaction_Anxiety'),
                DVs = c('Problematic_Internet_Use','Tobacco_Use','Alcohol_Use',
                        'Illicit_Drug_Use'),
                IV_covars = c('Age','Parents_Income'),
                DV_covars = c('Gambling_Behavior','Unprotected_Sex'),
                display_cormats=TRUE)
```

data_Green_Salkind_2014 *data_Green_Salkind_2014*

Description

Mutiple regression data from Green and Salkind (2018).

Usage

```
data(data_Green_Salkind_2014)
```

Source

Green, S. B., & Salkind, N. J. (2014). Lesson 34: Multiple linear regression (pp. 257-269). In, *Using SPSS for Windows and Macintosh: Analyzing and understanding data.* New York, NY: Pearson.

Examples

```
head(data_Green_Salkind_2014)

# forced (simultaneous) entry; replicating the output on p. 263
OLS_REGRESSION(data=data_Green_Salkind_2014, DV='injury',
               forced=c('quads','gluts','abdoms','arms','grip'))

# hierarchical entry; replicating the output on p. 265-266
OLS_REGRESSION(data=data_Green_Salkind_2014, DV='injury',
               hierarchical = list( step1=c('quads','gluts','abdoms'),
                                    step2=c('arms','grip')) )
```

`data_Halvorson_2022_log` *data_Halvorson_2022_log*

Description

Logistic regression data from Halvorson et al. (2022, p. 291).

Usage

```
data(data_Halvorson_2022_log)
```

Source

Halvorson, M. A., McCabe, C. J., Kim, D. S., Cao, X., & King, K. M. (2022). Making sense of some odd ratios: A tutorial and improvements to present practices in reporting and visualizing quantities of interest for binary and count outcome models. *Psychology of Addictive Behaviors, 36(3),* 284-295.

Examples

```
head(data_Halvorson_2022_log)

# replicating Figure 3, p 292
log_Halvorson <-
  LOGISTIC_REGRESSION(data=data_Halvorson_2022_log, DV='Y', forced=c('x1','x2'),
                      plot_type = 'diagnostics')

# high & low values for x2
x2_high <- mean(data_Halvorson_2022_log$x1) + sd(data_Halvorson_2022_log$x1)
```

```
x2_low  <- mean(data_Halvorson_2022_log$x1) - sd(data_Halvorson_2022_log$x1)

PLOT_MODEL(modobject = log_Halvorson,
           IV_focal_1 = 'x1',
           IV_focal_2 = 'x2',  IV_focal_2_values = c(x2_low, x2_high),
           bootstrap=TRUE, N_sims=1000, CI_level=95,
           ylim = c(0, 1),
           xlab = 'x1',
           ylab = 'Expected Probability',
           title = 'Probability of Y by x1 and x2 for Simulated Data Example')
```

---

data_Halvorson_2022_pois *data_Halvorson_2022_pois*

Description

Poisson regression data from Halvorson et al. (2022, p. 293).

Usage

```
data(data_Halvorson_2022_pois)
```

Source

Halvorson, M. A., McCabe, C. J., Kim, D. S., Cao, X., & King, K. M. (2022). Making sense of some odd ratios: A tutorial and improvements to present practices in reporting and visualizing quantities of interest for binary and count outcome models. *Psychology of Addictive Behaviors, 36(3),* 284-295.

Examples

```
head(data_Halvorson_2022_pois)

# replicating Table 3, p 293
pois_Halvorson <-
  COUNT_REGRESSION(data=data_Halvorson_2022_pois, DV='Neg_OH_conseqs',
                   forced=c('Gender','Positive_urgency','Planning',
                            'Sensation_seeking'),
                   plot_type = 'diagnostics')

# replicating Figure 4, p 294
PLOT_MODEL(modobject = pois_Halvorson,
           IV_focal_1 = 'Positive_urgency',
           IV_focal_2 = 'Gender',
           bootstrap=FALSE, N_sims=1000, CI_level=95,
           ylim = c(0, 20),
```

```
            xlab = 'Positive Urgency',
            ylab = 'Expected Count of Alcohol Consequences',
            title = 'Expected Count of Alcohol Consequences
                     by Positive Urgency and Gender')
```

---

data_Huitema_2011 *data_Huitema_2011*

Description

Moderated regression data for a continuous predictor and a dichotomous moderator from Huitema (2011, p. 253).

Usage

```
data(data_Huitema_2011)
```

Source

Huitema, B. (2011). *The analysis of covariance and alternatives: Statistical methods for experiments, quasi-experiments, and single-case studies.* Hoboken, NJ: Wiley.

Examples

```
head(data_Huitema_2011)

# replicating results on p. 254 for the Johnson-Neyman technique for a categorical moderator
MODERATED_REGRESSION(data=data_Huitema_2011, DV='Y',
                     IV='X', IV_range='tumble',
                     MOD='D', MOD_type = 'factor',
                     center = FALSE,
                     plot_type = 'interaction',
                     JN_type = 'Huitema')
```

---

data_Kremelburg_2011 *data_Kremelburg_2011*

Description

Logistic and Poisson regression data from Kremelburg (2011).

Usage

```
data(data_Kremelburg_2011)
```

Source

Kremelburg, D. (2011). Chapter 6: Logistic, ordered, multinomial, negative binomial, and Poisson regression. *Practical statistics: A quick and easy guide to IBM SPSS Statistics, STATA, and other statistical software.* Sage.

Examples

```
head(data_Kremelburg_2011)

# for Kremelburg, 2011, p. 244
LOGISTIC_REGRESSION(data = data_Kremelburg_2011, DV='OCCTRAIN',
                    forced = c('AGE','female','EDUC','REALRINC'))

# for Kremelburg, 2011, p.262 & p. 282: poisson regression
COUNT_REGRESSION(data=data_Kremelburg_2011, DV='OVRJOYED',
                 forced=c('AGE','EDUC','REALRINC','female'))

# for Kremelburg, 2011, p. 266-267 & p. 284: negative binomial regression
COUNT_REGRESSION(data=data_Kremelburg_2011, DV='HURTATWK',
                 forced=c('AGE','EDUC','REALRINC','female'),
                 model_type = 'negbin')
```

`data_Lorah_Wong_2018` *data_Lorah_Wong_2018*

Description

Moderated regression data from Lorah and Wong (2018).

Usage

```
data(data_Lorah_Wong_2018)
```

Source

Lorah, J. A. & Wong, Y. J. (2018). Contemporary applications of moderation analysis in counseling psychology. *Journal of Counseling Psychology, 65(5),* 629-640.

Examples

```
head(data_Lorah_Wong_2018)

model_Lorah <-
MODERATED_REGRESSION(data=data_Lorah_Wong_2018, DV='suicidal',
                     IV='burden', IV_range='tumble',
                     MOD='belong_thwarted', MOD_levels='quantiles',
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     COVARS='depression', center = TRUE,
```

```
                    plot_type = 'regions')

REGIONS_OF_SIGNIFICANCE(model=model_Lorah,
                        plot_title='Johnson-Neyman Regions of Significance',
                        Xaxis_label='Thwarted Belongingness',
                        Yaxis_label='Slopes of Burdensomeness on Suicical Ideation',
                        legend_label=NULL)
```

---

data_Meyers_2013 *data_Meyers_2013*

Description

Logistic regression data from Myers et al. (2013).

Usage

```
data(data_Meyers_2013)
```

Source

Meyers, L. S., Gamst, G. C., & Guarino, A. J. (2013). Chapter 30: Binary logistic regression. *Performing data analysis using IBM SPSS.* Hoboken, NJ: Wiley.

Examples

```
head(data_Meyers_2013)

# for p. 263
LOGISTIC_REGRESSION(data= data_Meyers_2013, DV='graduated', forced= c('sex','family_encouragement'))
```

---

data_OConnor_Dvorak_2001
data_OConnor_Dvorak_2001

Description

Moderated regression data from O'Connor and Dvorak (2001)

Details

A data frame with scores for 131 male adolescents on resiliency, maternal harshness, and aggressive behavior. The data are from O'Connor and Dvorak (2001, p. 17) and are provided as trial moderated regression data for the MODERATED_REGRESSION and REGIONS_OF_SIGNIFICANCE functions.

References

O'Connor, B. P., & Dvorak, T. (2001). Conditional associations between parental behavior and adolescent problems: A search for personality-environment interactions. *Journal of Research in Personality, 35,* 1-26.

Examples

```
head(data_OConnor_Dvorak_2001)

# for O'Connor & Dvorak, 2001, p. 17; with numeric values for IV_range & MOD_levels='AikenWest'
mharsh_agg <-
  MODERATED_REGRESSION(data=data_OConnor_Dvorak_2001, DV='Aggressive_Behavior',
                       IV='Maternal_Harshness', IV_range=c(1,7.7),
                       MOD='Resiliency', MOD_levels='AikenWest',
                       quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                       center = FALSE,
                       plot_type = 'interaction',
                       DV_range = c(1,6),
                       Xaxis_label='Maternal Harshness',
                       Yaxis_label='Adolescent Aggressive Behavior',
                       legend_label='Resiliency')

REGIONS_OF_SIGNIFICANCE(model=mharsh_agg,
           plot_title='Slopes of Maternal Harshness on Aggression by Resiliency',
           Xaxis_label='Resiliency',
           Yaxis_label='Slopes of Maternal Harshness on Aggressive Behavior ')
```

---

data_Orme_2009_2 *data_Orme_2009_2*

Description

Logistic regression data from Orme and Combs-Orme (2009), Chapter 2.

Usage

```
data(data_Orme_2009_2)
```

Source

Orme, J. G., & Combs-Orme, T. (2009). *Multiple Regression With Discrete Dependent Variables.* Oxford University Press.

Examples

```
head(data_Orme_2009_2)

# for Orme, 2009, p. 60
LOGISTIC_REGRESSION(data = data_Orme_2009_2, DV='ContinueFostering',
                    forced= c('zResources', 'Married'))
```

---

data_Orme_2009_5 *data_Orme_2009_5*

Description

Data for count regression from Orme and Combs-Orme (2009), Chapter 5.

Usage

```
data(data_Orme_2009_5)
```

Source

Orme, J. G., & Combs-Orme, T. (2009). *Multiple Regression With Discrete Dependent Variables.* Oxford University Press.

Examples

```
head(data_Orme_2009_5)

# for Orme, 2009, p. 175: negative binomial regression with an offset variable
COUNT_REGRESSION(data=data_Orme_2009_5, DV='NumberAdopted',
                 forced=c('Married','zParentRole'),
                 model_type = 'negbin',
                 offset='lnYearsFostered')
```

---

data_Pedhazur_1997 *data_Pedhazur_1997*

Description

Moderated regression data for a continuous predictor and a dichotomous moderator from Pedhazur (1997, p. 588).

Usage

```
data(data_Pedhazur_1997)
```

Source

Pedhazur, E. J. (1997). *Multiple regression in behavioral research: Explanation and prediction.* (3rd ed.). Fort Worth, Texas: Wadsworth Thomson Learning.

Examples

```
head(data_Pedhazur_1997)

# replicating results on p. 594 for the Johnson-Neyman technique for a categorical moderator

MODERATED_REGRESSION(data=data_Pedhazur_1997, DV='Y',
                     IV='X', IV_range='tumble',
                     MOD='Directive', MOD_type = 'factor', MOD_levels='quantiles',
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     center = FALSE,
                     plot_type = 'interaction',
                     JN_type = 'Pedhazur')
```

---

data_Pituch_Stevens_2016 *data_Pituch_Stevens_2016*

Description

Logistic regression data from Pituch and Stevens (2016), Chapter 11.

Usage

```
data(data_Pituch_Stevens_2016)
```

Source

Pituch, K. A., & Stevens, J. P. (2016). *Applied multivariate statistics for the social sciences : Analyses with SAS and IBMs SPSS,* (6th ed.). Routledge.

Examples

```
# for p. 463
LOGISTIC_REGRESSION(data = data_Pituch_Stevens_2016, DV='Health',
                    forced= c('Treatment','Motivation'))
```

LOGISTIC_REGRESSION *Logistic regression*

Description

Logistic regression analyses with SPSS- and SAS-like output. The output includes model summaries, classification tables, omnibus tests of model coefficients, the model coefficients, likelihood ratio tests for the predictors, overdispersion tests, model effect sizes, the correlation matrix for the model coefficients, collinearity statistics, and casewise regression diagnostics.

Usage

```
LOGISTIC_REGRESSION(data, DV, forced = NULL, hierarchical = NULL, formula=NULL,
                    ref_category = NULL,
                    family = 'binomial',
                    CI_level = 95,
                    MCMC_options = list(MCMC = FALSE, Nsamples = 10000,
                                        thin = 1, burnin = 1000,
                                        HDI_plot_est_type = 'standardized'),
                    plot_type = 'residuals',
                    verbose = TRUE)
```

Arguments

`data` A dataframe where the rows are cases and the columns are the variables.

`DV` The name of the dependent variable.
Example: DV = 'outcomeVar'.

`forced` (optional) A vector of the names of the predictor variables for a forced/simultaneous entry regression. The variables can be numeric or factors.
Example: forced = c('VarA', 'VarB', 'VarC')

`hierarchical` (optional) A list with the names of the predictor variables for each step of a hierarchical regression. The variables can be numeric or factors.
Example: hierarchical = list(step1=c('VarA', 'VarB'), step2=c('VarC', 'VarD'))

`formula` (optional) Text for an R formula. Useful for testing for interactions.
Example: formula = "Aggressive_Behavior ~ Maternal_Harshness * Resiliency"")

`ref_category` (optional) The reference category for DV.
Example: ref_category = 'alive'

`family` (optional) The name of the error distribution to be used in the model. The options are:

- "binomial" (the default), or
- "quasibinomial", which should be used when there is overdispersion.

Example: family = 'quasibinomial'

`CI_level` (optional) The confidence interval for the output, in whole numbers. The default is 95.

`MCMC_options` (optional) A list specifying the following options for Bayesian MCMC analyses: (1) "MCMC", Should MCMC analyses be conducted? The options are TRUE or FALSE; (2) "Nsamples", for the number of iterations or samples from the posterior distribution; (3) "thin", for the chain-thinning interval; (4) "burnin", for the burnin period, i.e., the number of initial samples that should be dropped from the chains; and (5) "HDI_plot_est_type", for the kind of regression estimates that will appear in any requested HDI plots. The options are "standardized" or "raw".
Example: MCMC_options = list(MCMC = TRUE, Nsamples = 10000, thin = 1, burnin = 1000, HDI_plot_est_type = 'standardized')

`plot_type` (optional) The kind of plots, if any. The options are:

- 'residuals' (the default),
- 'diagnostics', for regression diagnostics,
- 'Bayes_HDI' (for MCMC posterior distributions), and
- 'none', for no plots.

Example: plot_type = 'diagnostics'

`verbose` (optional) Should detailed results be displayed in console?
The options are: TRUE (default) or FALSE. If TRUE, plots of residuals are also produced.

Details

This function uses the glm function from the stats package and supplements the output with additional statistics and in formats that resembles SPSS and SAS output. The predictor variables can be numeric or factors.

The function assigns contrasts (dummy codes) to factor variables that do not already have contrasts. The baseline group for the dummy codes is determined by the alphabetic/numeric order of the factor levels. If the terms "control" or "Control" or "baseline" or "Baseline" appear in the names of a factor level, then that factor level is used as the dummy codes baseline.

Predicted values for this model, for selected levels of the predictor variables, can be produced and plotted using the PLOT_MODEL funtion in this package.

The Bayesian MCMC analyses can be time-consuming for larger datasets. The MCMC analyses are conducted using functions, and their default settings, from the rstanarm package (Goodrich, Gabry, Ali, & Brilleman, 2024). The MCMC results can be verified using the model checking functions in the rstanarm package (e.g., Muth, Oravecz, & Gabry, 201).

Good sources for interpreting logistic regression residuals and diagnostics plots:

- rpubs.com/benhorvath
- library.virginia.edu

Value

An object of class "LOGISTIC_REGRESSION". The object is a list containing the following possible components:

`model` All of the glm function output for the regression model.

`modelsum`	All of the summary.glm function output for the regression model.
`modeldata`	All of the predictor and outcome raw data that were used in the model, along with regression diagnostic statistics for each case.
`collin_diags`	Collinearity diagnostic coefficients for models without interaction terms.
`chain_dat`	The MCMC chains.
`Bayes_HDIs`	The Bayesian HDIs.

Author(s)

Brian P. O'Connor

References

Dunn, P. K., & Smyth, G. K. (2018). *Generalized linear models with examples in R.* Springer.

Field, A., Miles, J., & Field, Z. (2012). *Discovering statistics using R.* Los Angeles, CA: Sage.

Goodrich, B., Gabry, J., Ali, I., & Brilleman, S. (2024). *rstanarm: Bayesian applied regression modeling via Stan.* R package version 2.32.1, https://mc-stan.org/rstanarm/.

Hair, J. F., Black, W. C., Babin, B. J., & Anderson, R. E. (2014). *Multivariate data analysis,* (8th ed.). Lawrence Erlbaum Associates.

Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. (2013) *Applied logistic regression.* (3rd ed.). John Wiley & Sons.

Muth, C., Oravecz, Z., & Gabry, J. (2018). User-friendly Bayesian regression modeling: A tutorial with rstanarm and shinystan. *The Quantitative Methods for Psychology, 14(2),* 99119.
https://doi.org/10.20982/tqmp.14.2.p099

Orme, J. G., & Combs-Orme, T. (2009). *Multiple regression with discrete dependent variables.* Oxford University Press.

Pituch, K. A., & Stevens, J. P. (2016). *Applied multivariate statistics for the social sciences: Analyses with SAS and IBM's SPSS,* (6th ed.). Routledge.

Rindskopf, D. (2023). Generalized linear models. In H. Cooper, M. N. Coutanche, L. M. McMullen, A. T. Panter, D. Rindskopf, & K. J. Sher (Eds.), *APA handbook of research methods in psychology: Data analysis and research publication,* (2nd ed., pp. 201-218). American Psychological Association.

Examples

```
# Meyers, 2013, p. 263: forced (simultaneous) entry
LOGISTIC_REGRESSION(data = data_Meyers_2013, DV='graduated',
                    forced=c('sex','family_encouragement'),
                    plot_type = 'diagnostics')
```

```
# for Kremelburg, 2011, p. 244: hierarchical entry, with Bayesian MCMC analyses & HDI plots
LOGISTIC_REGRESSION(data = data_Kremelburg_2011, DV='OCCTRAIN',
                    hierarchical=list( step1=c('AGE', 'female'),
                                       step2=c('EDUC','REALRINC')),
                    MCMC_options = list(MCMC = TRUE, Nsamples = 10000,
                                        thin = 1, burnin = 1000,
                                        HDI_plot_est_type = 'raw'),
                    plot_type = 'Bayes_HDI')
```

MODERATED_REGRESSION *Moderated multiple regression*

Description

Conducts moderated regression analyses for two-way interactions with extensive options for interaction plots, including Johnson-Neyman regions of significance. The output includes the Anova Table (Type III tests), standardized coefficients, partial and semi-partial correlations, collinearity statistics, casewise regression diagnostics, plots of residuals and regression diagnostics, and detailed information about simple slopes. The output includes Bayes Factors and, if requested, regression coefficients from Bayesian Markov Chain Monte Carlo (MCMC) analyses.

Usage

```
MODERATED_REGRESSION(data, DV, IV, MOD,
                     IV_type = 'numeric', IV_range = 'tumble',
                     MOD_type='numeric', MOD_levels='quantiles',
                     MOD_range=NULL, MOD_reflevel=NULL,
                     quantiles_IV = c(.1, .9), quantiles_MOD = c(.25, .5, .75),
                     COVARS = NULL,
                     center = TRUE,
                     CI_level = 95,
                     MCMC_options = list(MCMC = FALSE, Nsamples = 10000,
                                         thin = 1, burnin = 1000,
                                         HDI_plot_est_type = 'raw'),
                     plot_type = 'residuals', plot_title=NULL, DV_range = NULL,
                     Xaxis_label = NULL, Yaxis_label=NULL, legend_label=NULL,
                     JN_type = 'Huitema',
                     verbose = TRUE )
```

Arguments

`data` A dataframe where the rows are cases and the columns are the variables.

`DV` The name of the dependent variable.
Example: DV = 'outcomeVar'

`IV` The name of the independent variable.
Example: IV = 'varA'

`MOD` The name of the moderator variable
Example: MOD = 'varB'

`IV_type` (optional) The type of independent variable. The options are 'numeric' (the default) or 'factor'.
Example: IV_type = 'factor'

`IV_range` (optional) The independent variable range for a moderated regression plot. The options are:

- 'tumble' (the default), for tumble graphs following Bodner (2016)
- 'quantiles', in which case the 10th and 90th quantiles of the IV will be used (alternative values can be specified using the quantiles_IV argument);
- 'AikenWest', in which case the IV mean - one SD, and the IV mean + one SD, will be used;
- a vector of two user-provided values (e.g., c(1, 10)); and
- NULL, in which case the minimum and maximum IV values will be used.

Example: IV_range = 'AikenWest'

`MOD_type` (optional) The type of moderator variable. The options are 'numeric' (the default) or 'factor'.
Example: MOD_type = 'factor'

`MOD_levels` (optional) The levels of the moderator variable to be used if MOD is continuous. The options are:

- 'quantiles', in which case the .25, .5, and .75 quantiles of the MOD variable will be used (alternative values can be specified using the quantiles_MOD argument);
- 'AikenWest', in which case the mean of MOD, the mean of MOD - one SD, and the mean of MOD + one SD, will be used; and
- a vector of two user-provided values.

Example: MOD_levels = c(1, 10)

`MOD_range` (optional) The range of the MOD values to be used in the Johnson-Neyman regions of significance analyses. The options are: NULL (the default), in which case the minimum and maximum MOD values will be used; and a vector of two user-provided values.
Example: MOD_range = c(1, 10)

`MOD_reflevel` (optional) The level of MOD, if it is a factor, to be used as the baseline in the factor dummy codes. If a baseline value is not provided, and if the terms "control" or "Control" or "baseline" or "Baseline" appear in the names of a factor level, then that factor level will be used as the baseline; otherwise the baseline will be the earliest of the alphabetically- ordered factor levels.

`quantiles_IV` (optional) The quantiles of the independent variable to be used as the IV range for a moderated regression plot.
Example: quantiles_IV = c(.10, .90)

`quantiles_MOD` (optional) The quantiles the moderator variable to be used as the MOD simple slope values in the moderated regression analyses.
Example: quantiles_MOD = c(.25, .5, .75)

`COVARS` (optional) The name(s) of possible covariates.
Example: COVARS = c('CovarA', 'CovarB', 'CovarC')

`center` (optional) Logical, indicating whether the IV and MOD variables should be centered (default = TRUE).
Example: center = FALSE

`CI_level` (optional) The confidence interval for the output, in whole numbers. CI_level is also used in the Johnson-Neyman regions of significance computations. The default is 95.

`MCMC_options` (optional) A list specifying the following options for Bayesian MCMC analyses: (1) "MCMC", Should MCMC analyses be conducted? The options are TRUE or FALSE; (2) "Nsamples", for the number of iterations or samples from the posterior distribution; (3) "thin", for the chain-thinning interval; (4) "burnin", for the burnin period, i.e., the number of initial samples that should be dropped from the chains; and (5) "HDI_plot_est_type", for the kind of regression estimates that will appear in any requested HDI plots. The options are "standardized" or "raw".
Example: MCMC_options = list(MCMC = TRUE, Nsamples = 10000, thin = 1, burnin = 1000, HDI_plot_est_type = 'raw')

`plot_type` (optional) The kind of plot, if any. The options are:

- 'residuals' (the default)
- 'diagnostics' (for regression diagnostics)
- 'interaction' (for a traditional moderated regression interaction plot)
- 'regions' (for a moderated regression Johnson-Neyman regions of significance plot),
- 'Bayes_HDI' (for MCMC posterior distributions), and
- 'none' (for no plots).

Example: plot_type = 'diagnostics'

`plot_title` (optional) The plot title.
Example: plot_title = 'Interaction Plot'

`DV_range` (optional) The range of Y-axis values for the plot.
Example: DV_range = c(1,10)

`Xaxis_label` (optional) A label for the X axis to be used in the requested plot.
Example: Xaxis_label = 'IV name'

`Yaxis_label` (optional) A label for the Y axis to be used in the requested plot.
Example: Yaxis_label = 'DV name'

`legend_label` (optional) A legend label for the plot.
Example: legend_label = 'MOD name'

`JN_type` (optional) The formula to be used in computing the critical F value for the Johnson-Neyman regions of significance analyses. The options are 'Huitema' (the default), or 'Pedhazur'.
Example: JN_type = 'Pedhazur'

`verbose` Should detailed results be displayed in console? The options are: TRUE (default) or FALSE. If TRUE, plots of residuals are also produced.

Details

The MCMC analyses are conducted using functions, and their default settings, from the rstanarm package (Goodrich, Gabry, Ali, & Brilleman, 2024). MCMC analyses can be time-consuming for larger datasets.

The Bayes Factor analyses are conducted using functions, and their default settings, from the BayesFactor package (Morey & Rouder, 2024).

The Bayes factor values for the predictor variables are based solely on the predictor t & df values, using the ttest.tstat function from the BayesFactor package.

Value

An object of class "MODERATED_REGRESSION". The object is a list containing the following possible components:

`modelsum`	All of the summary.lm function output for the regression model without interaction terms.
`anova_table`	Anova Table (Type III tests).
`mainRcoefs`	Predictor coefficients for the model without interaction terms.
`modeldata`	All of the predictor and outcome raw data that were used in the model, along with regression diagnostic statistics for each case.
`collin_diags`	Collinearity diagnostic coefficients for models without interaction terms.
`modelXNsum`	Regression model statistics with interaction terms.
`RsqchXn`	Rsquared change for the interaction.
`fsquaredXN`	fsquared change for the interaction.
`xnRcoefs`	Predictor coefficients for the model with interaction terms.
`simslop`	The simple slopes.
`simslopZ`	The standardized simple slopes.
`plotdon`	The plot data for a moderated regression.
`JN.data`	The Johnson-Neyman results for a moderated regression.
`ros`	The Johnson-Neyman regions of significance for a moderated regression.
`chain_dat`	The MCMC chains.
`Bayes_HDIs`	The Bayesian HDIs.

Author(s)

Brian P. O'Connor

References

Bodner, T. E. (2016). Tumble graphs: Avoiding misleading end point extrapolation when graphing interactions from a moderated multiple regression analysis. *Journal of Educational and Behavioral Statistics, 41,* 593-604.

Cohen, J., Cohen, P., West, S. G., & Aiken, L. S. (2003). *Applied multiple regression/correlation* *analysis for the behavioral sciences* (3rd ed.). Lawrence Erlbaum Associates.

Darlington, R. B., & Hayes, A. F. (2017). *Regression analysis and linear models: Concepts, applications, and implementation.* Guilford Press.

Goodrich, B., Gabry, J., Ali, I., & Brilleman, S. (2024). *rstanarm: Bayesian applied regression modeling via Stan.* R package version 2.32.1, https://mc-stan.org/rstanarm/.

Hayes, A. F. (2018a). *Introduction to mediation, moderation, and conditional process analysis: A regression-based approach* (2nd ed.). Guilford Press.

Hayes, A. F., & Montoya, A. K. (2016). A tutorial on testing, visualizing, and probing an interaction involving a multicategorical variable in linear regression analysis. *Communication Methods and Measures, 11,* 1-30.

Lee M. D., & Wagenmakers, E. J. (2014) *Bayesian cognitive modeling: A practical course.* Cambridge University Press.

Morey, R. & Rouder, J. (2024). *BayesFactor: Computation of Bayes Factors for Common Designs.* R package version 0.9.12-4.7, https://github.com/richarddmorey/bayesfactor.

O'Connor, B. P. (1998). All-in-one programs for exploring interactions in moderated multiple regression. *Educational and Psychological Measurement, 58,* 833-837.

Pedhazur, E. J. (1997). *Multiple regression in behavioral research: Explanation and prediction.* (3rd ed.). Wadsworth Thomson Learning.

Examples

```
# for Lorah & Wong, 2018, p. 630: with IV_range = 'AikenWest'
MODERATED_REGRESSION(data=data_Lorah_Wong_2018, DV='suicidal', IV='burden', MOD='belong_thwarted',
                     IV_range='AikenWest',
                     MOD_levels='quantiles',
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     center = TRUE, COVARS='depression',
                     plot_type = 'interaction', plot_title=NULL, DV_range = c(1,1.25))

# for Lorah & Wong, 2018, p. 630: with IV_range = 'tumble', &
# with Bayesian MCMC analyses & HDI plots
MODERATED_REGRESSION(data=data_Lorah_Wong_2018, DV='suicidal', IV='burden', MOD='belong_thwarted',
                     IV_range='tumble',
                     MOD_levels='quantiles',
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     center = TRUE, COVARS='depression',
                     MCMC_options = list(MCMC = FALSE, Nsamples = 10000,
                                         thin = 1, burnin = 1000,
                                         HDI_plot_est_type = 'raw'),
                      plot_type = 'Bayes_HDI', plot_title=NULL, DV_range = c(1,1.25))

# for O'Connor & Dvorak, 2001, p. 17; with numeric values for IV_range & MOD_levels='AikenWest'
```

```
MODERATED_REGRESSION(data=data_OConnor_Dvorak_2001, DV='Aggressive_Behavior',
                     IV='Maternal_Harshness', MOD='Resiliency',
                     IV_range=c(1,7.7),
                     MOD_levels='AikenWest', MOD_range=NULL,
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     center = FALSE,
                     plot_type = 'interaction',
                     DV_range = c(1,6),
                     Xaxis_label='Maternal Harshness',
                     Yaxis_label='Adolescent Aggressive Behavior',
                     legend_label='Resiliency')
```

OLS_REGRESSION *Ordinary least squares regression*

Description

Provides SPSS- and SAS-like output for ordinary least squares simultaneous entry regression and hierarchical entry regression. The output includes the Anova Table (Type III tests), standardized coefficients, partial and semi-partial correlations, collinearity statistics, casewise regression diagnostics, plots of residuals and regression diagnostics. The output includes Bayes Factors and, if requested, regression coefficients from Bayesian Markov Chain Monte Carlo (MCMC) analyses.

Usage

```
OLS_REGRESSION(data, DV, forced=NULL, hierarchical=NULL, formula=NULL,
               CI_level = 95,
               MCMC_options = list(MCMC = FALSE, Nsamples = 10000,
                                   thin = 1, burnin = 1000,
                                   HDI_plot_est_type = 'standardized'),
               plot_type = 'residuals',
               verbose=TRUE, ...)
```

Arguments

data	A dataframe where the rows are cases and the columns are the variables.
DV	The name of the dependent variable. Example: DV = 'outcomeVar'
forced	(optional) A vector of the names of the predictor variables for a forced/simultaneous entry regression. The variables can be numeric or factors. Example: forced = c('VarA', 'VarB', 'VarC')
hierarchical	(optional) A list with the names of the predictor variables for each step of a hierarchical regression. The variables can be numeric or factors. Example: hierarchical = list(step1=c('VarA', 'VarB'), step2=c('VarC', 'VarD'))
formula	(optional) Text for an R formula. Useful for testing for interactions. Example: formula = "Aggressive_Behavior ~ Maternal_Harshness * Resiliency"")

CI_level (optional) The confidence interval for the output, in whole numbers. The default is 95.

MCMC_options (optional) A list specifying the following options for Bayesian MCMC analyses: (1) "MCMC", Should MCMC analyses be conducted? The options are TRUE or FALSE; (2) "Nsamples", for the number of iterations or samples from the posterior distribution; (3) "thin", for the chain-thinning interval; (4) "burnin", for the burnin period, i.e., the number of initial samples that should be dropped from the chains; and (5) "HDI_plot_est_type", for the kind of regression estimates that will appear in any requested HDI plots. The options are "standardized" or "raw".
Example: MCMC_options = list(MCMC = TRUE, Nsamples = 10000, thin = 1, burnin = 1000, HDI_plot_est_type = 'standardized')

plot_type (optional) The kind of plots, if any. The options are:

- 'residuals' (the default)
- 'diagnostics' (for regression diagnostics),
- 'Bayes_HDI' (for MCMC posterior distributions), or
- 'none' (for no plots).

Example: plot_type = 'diagnostics'

verbose Should detailed results be displayed in console? The options are: TRUE (default) or FALSE. If TRUE, plots of residuals are also produced.

... (dots, for internal purposes only at this time.)

Details

This function uses the lm function from the stats package, supplements the output with additional statistics, and it formats the output so that it resembles SPSS and SAS regression output. The predictor variables can be numeric or factors.

The function assigns contrasts (dummy codes) to factor variables that do not already have contrasts. The baseline group for the dummy codes is determined by the alphabetic/numeric order of the factor levels. If the terms "control" or "Control" or "baseline" or "Baseline" appear in the names of a factor level, then that factor level is used as the dummy codes baseline.

The MCMC analyses are conducted using functions, and their default settings, from the rstanarm package (Goodrich, Gabry, Ali, & Brilleman, 2024). MCMC analyses can be time-consuming for larger datasets.

The Bayes Factor analyses are conducted using functions, and their default settings, from the BayesFactor package (Morey & Rouder, 2024).

The Bayes factor values for the predictor variables are based solely on the predictor t & df values, using the ttest.tstat function from the BayesFactor package.

Good sources for interpreting residuals and diagnostics plots:

- library.virginia.edu
- sthda.com

Value

An object of class "OLS_REGRESSION". The object is a list containing the following possible components:

- `model` All of the lm function output for the regression model without interaction terms.
- `modelsum` All of the summary.lm function output for the regression model without interaction terms.
- `anova_table` Anova Table (Type III tests).
- `mainRcoefs` Predictor coefficients for the model without interaction terms.
- `modeldata` All of the predictor and outcome raw data that were used in the model, along with regression diagnostic statistics for each case.
- `collin_diags` Collinearity diagnostic coefficients for models without interaction terms.
- `chain_dat` The MCMC chains.
- `Bayes_HDIs` The Bayesian HDIs.

Author(s)

Brian P. O'Connor

References

Bodner, T. E. (2016). Tumble graphs: Avoiding misleading end point extrapolation when graphing interactions from a moderated multiple regression analysis. *Journal of Educational and Behavioral Statistics, 41,* 593-604.

Cohen, J., Cohen, P., West, S. G., & Aiken, L. S. (2003). *Applied multiple regression/correlation analysis for the behavioral sciences* (3rd ed.). Lawrence Erlbaum Associates.

Darlington, R. B., & Hayes, A. F. (2017). *Regression analysis and linear models: Concepts, applications, and implementation.* Guilford Press.

Goodrich, B., Gabry, J., Ali, I., & Brilleman, S. (2024). *rstanarm: Bayesian applied regression modeling via Stan.* R package version 2.32.1, https://mc-stan.org/rstanarm/.

Hayes, A. F. (2018a). *Introduction to mediation, moderation, and conditional process analysis: A regression-based approach* (2nd ed.). Guilford Press.

Hayes, A. F., & Montoya, A. K. (2016). A tutorial on testing, visualizing, and probing an interaction involving a multicategorical variable in linear regression analysis. *Communication Methods and Measures, 11,* 1-30.

Lee M. D., & Wagenmakers, E. J. (2014) *Bayesian cognitive modeling: A practical course.* Cambridge University Press.

Morey, R. & Rouder, J. (2024). *BayesFactor: Computation of Bayes Factors for Common Designs.* R package version 0.9.12-4.7, https://github.com/richarddmorey/bayesfactor.

O'Connor, B. P. (1998). All-in-one programs for exploring interactions in moderated multiple regression. *Educational and Psychological Measurement, 58,* 833-837.

Pedhazur, E. J. (1997). *Multiple regression in behavioral research: Explanation and prediction.* (3rd ed.). Wadsworth Thomson Learning.

Examples

```
# for Green_Salkind_2014, p. 263: forced (simultaneous) entry
OLS_REGRESSION(data=data_Green_Salkind_2014, DV='injury',
               forced = c('quads','gluts','abdoms','arms','grip'))

# for Green_Salkind_2014, p. 265: hierarchical entry with Bayesian MCMC analyses & HDI plots
OLS_REGRESSION(data=data_Green_Salkind_2014, DV='injury',
               hierarchical = list(step1=c('quads','gluts','abdoms'),
                                   step2=c('arms','grip')),
               MCMC_options = list(MCMC = TRUE, Nsamples = 10000,
                                   thin = 1, burnin = 1000,
                                   HDI_plot_est_type = 'raw'),
               plot_type = 'Bayes_HDI')

# for O'Connor & Dvorak, 2001, p. 17; 2-way interaction specified via formula
OLS_REGRESSION(data=data_OConnor_Dvorak_2001,
               formula = 'Aggressive_Behavior ~ Maternal_Harshness * Resiliency')
```

`PARTIAL_COR` *Partial and semipartial correlations*

Description

Produces partial correlations between two or more variables (in set Y) while statistically controlling for one or more covariates (set C). It also produces partial correlations, semipartial correlations, and standardized regression coefficients for predicting variables (in set Y) from one or more set X variables.

Usage

```
PARTIAL_COR(data, Y, X=NULL, C=NULL, Ncases=NULL, verbose=TRUE)
```

Arguments

`data`	Either a dataframe of raw data (where the rows are cases and the columns are the variables), or a square correlation matrix with row and column names.
`Y`	The names of one or more continuous variables in data. Example: Y = c('var1', 'var2', 'var3')
`C`	The names of one or more continuous variables in data to be partialled out of the Y variable correlations. Example: C = c('var4', 'var5')

`X` The names of one or more continuous predictor variables in data.
Example: X = c('var6', 'var7', 'var8')

`Ncases` The number of cases. Required only when the input (data) is a correlation matrix.

`verbose` Should detailed results be displayed in console?
The options are: TRUE (default) or FALSE.

Details

Y must be provided along with either one, or both, of C and X.

If Y and C are provided, but not X, then the function computes:

- the correlations between the Y variables after partialling the C variables out of the Y variables.

If Y and X are provided, but not C, then the function computes:

- the standardized betas for the X variables predicting the Y variables;
- the partial correlations for the X variables predicting the Y variables. In other words, for any given X variable, the other X variables are partialled out of both the given X variable and the Y variables. And,
- the semi-partial correlations for the X variables predicting the Y variables. In other words, for any given X variable, the other X variables are partialled out of the given X variable and the Y variables remain as they are, untouched.

If Y, X, and C are provided, then the function computes:

- the correlations between the Y variables after partialling the C variables out of the Y variables;
- the betas for the X variables predicting the C-partialled Y variables;
- the partial correlations for the X variables predicting the C-partialled Y variables. In other words, for any given X variable, the other X variables are partialled out of both the given X variable and the C-partialled Y variables. And,
- the semi-partial correlations for the X variables predicting the C-partialled Y variables. In other words, for any given X variable, the other X variables are partialled out of the given X variable but not out of the C-partialled Y variables.

Value

A list containing the correlations, standardized regression coefficients (betas), partial correlations, semi-partial correlations, t-test values, and p values.

Author(s)

Brian P. O'Connor

References

Cohen, J., Cohen, P., West, S. G., & Aiken, L. S. (2003). *Applied multiple regression/correlation analysis for the behavioral sciences* (3rd ed.). Lawrence Erlbaum Associates.

Examples

```
# C, but no X variables
PARTIAL_COR(data = data_DeLeo_2013,
            Y = c('Problematic_Internet_Use','Tobacco_Use',
                  'Alcohol_Use','Illicit_Drug_Use'),
            C = c('Age','Parents_Income'),
            X = NULL)

# X, but no C variables
PARTIAL_COR(data = data_DeLeo_2013,
            Y = c('Problematic_Internet_Use','Tobacco_Use',
                  'Alcohol_Use','Illicit_Drug_Use'),
            C = NULL,
            X = c('Impulsivity','Social_Interaction_Anxiety',
                  'Social_Support','Intolerance_of_Deviance','Family_Morals',
                  'Grade_Point_Average','Depression','Family_Conflict'))

# both X & C variables
PARTIAL_COR(data = data_DeLeo_2013,
            Y = c('Problematic_Internet_Use','Tobacco_Use',
                  'Alcohol_Use','Illicit_Drug_Use'),
            C = c('Age','Parents_Income'),
            X = c('Impulsivity','Social_Interaction_Anxiety',
                  'Social_Support','Intolerance_of_Deviance','Family_Morals',
                  'Grade_Point_Average','Depression','Family_Conflict'))
```

PLOT_MODEL *Plots predicted values for a regression model*

Description

Plots predicted values of the outcome variable for specified levels of predictor variables for OLS_REGRESSION, MODERATED_REGRESSION, LOGISTIC_REGRESSION, and COUNT_REGRESSION models from this package.

Usage

```
PLOT_MODEL(modobject,
           IV_focal_1, IV_focal_1_values=NULL,
           IV_focal_2=NULL, IV_focal_2_values=NULL,
           IVs_nonfocal_values = NULL,
           bootstrap=FALSE, N_sims=100, CI_level=95,
           xlim=NULL, xlab=NULL,
           ylim=NULL, ylab=NULL,
           title = NULL,
           plot_save = FALSE, plot_save_type = 'png',
           cols_user = NULL,
           verbose=TRUE)
```

Arguments

modobject
: The returned output from the OLS_REGRESSION, MODERATED_REGRESSION, LOGISTIC_REGRESSION, or COUNT_REGRESSION functions in this package.

IV_focal_1
: The name of the focal, varying predictor variable.
Example: IV_focal_1 = 'age'

IV_focal_1_values
: (optional) Values for IV_focal_1, for which predictions of the outcome will be produced and plotted. IV_focal_1_values will appear on the x-axis in the plot. If IV_focal_1 is numeric and IV_focal_1_values is not provided, then a sequence based on the range of the model data values for IV_focal_1 will be used. If IV_focal_1 is a factor & IV_focal_1_values is not provided, then the factor levels from the model data values for IV_focal_1 will be used.
Example: IV_focal_1_values = seq(20, 80, 1)
Example: IV_focal_1_values = c(20, 40, 60)

IV_focal_2
: (optional) If desired, the name of a second focal predictor variable for the plot.
Example: IV_focal_2 = 'height'

IV_focal_2_values
: (optional) Values for IV_focal_2 for which predictions of the outcome will be produced and plotted. If IV_focal_2 is numeric and IV_focal_2_values is not provided, then the following three values for IV_focal_2_values, derived from the model data, will be used for plotting: the mean, one SD below the mean, and one SD above the mean. If IV_focal_2 is a factor & IV_focal_2_values is not provided, then the factor levels from the model data values for IV_focal_2 will be used.
Example: IV_focal_2_values = c(20, 40, 60)

IVs_nonfocal_values
: (optional) A list with the desired constant values for the non focal predictors, if any. If IVs_nonfocal_values is not provided, then the mean values of numeric non focal predictors and the baseline values of factors will be used as the defaults. It is also possible to specify values for only some of the IVs_nonfocal variables on this argument.
Example: IVs_nonfocal_values = list(AGE = 25, EDUC = 12)

bootstrap
: (optional) Should bootstrapping be used for the confidence intervals? The options are TRUE or FALSE (the default).

N_sims
: (optional) The number of bootstrap simulations.
Example: N_sims = 1000

CI_level
: (optional) The desired confidence interval, in whole numbers.
Example: CI_level = 95

xlim
: (optional) The x-axis limits for the plot.
Example: xlim = c(1, 9)

xlab
: (optional) A x-axis label for the plot.
Example: xlab = 'IVname'

ylim
: (optional) The y-axis limits for the plot.
Example: ylim = c(0, 80)

ylab (optional) A y-axis label for the plot.
Example: ylab = 'DVname'

title (optional) A title for the plot.
Example: title = 'OLS prediction of DV'

plot_save Should a plot be saved to disk? TRUE or FALSE (the default).

plot_save_type The output format if plot_save = TRUE. The options are 'bitmap', 'tiff', 'png' (the default), 'jpeg', and 'bmp'.

cols_user A vector of colours for the levels of IV_focal_1 or IV_focal_2.
If NULL, the default colours are selected, in order, from this vector: cols_user <- c("mediumvioletred", 'black', "blue", 'cyan2', "red", 'limegreen', "yellow", 'blueviolet'). If there are more than 7 levels of levels of IV_focal_1 or IV_focal_2, then "rainbow" is used to determine the colours.

verbose Should detailed results be displayed in console?
The options are: TRUE (default) or FALSE

Details

A plot with both IV_focal_1 and IV_focal_2 predictor variables will look like an interaction plot. But it is only a true interaction plot if the required product term(s) was entered as a predictor when the model was created.

Value

A matrix with the levels of the variables that were used for the plot along with the predicted values, confidence intervals, and se.fit values.

Author(s)

Brian P. O'Connor

Examples

```
ols_GS <-
OLS_REGRESSION(data=data_Green_Salkind_2014, DV='injury',
               hierarchical = list( step1=c('age','quads','gluts','abdoms'),
                                    step2=c('arms','grip')) )

PLOT_MODEL(modobject = ols_GS,
           IV_focal_1 = 'gluts', IV_focal_1_values=NULL,
           IV_focal_2 = 'age', IV_focal_2_values=NULL,
           IVs_nonfocal_values = NULL,
           bootstrap=TRUE, N_sims=100, CI_level=95,
           ylim=NULL, ylab=NULL, title=NULL,
           verbose=TRUE)

ols_LW <-
MODERATED_REGRESSION(data=data_Lorah_Wong_2018, DV='suicidal', IV='burden', MOD='belong_thwarted',
                     IV_range='tumble',
                     MOD_levels='quantiles',
```

```
                    quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                    COVARS='depression',
                    plot_type = 'interaction', DV_range = c(1,1.25))

PLOT_MODEL(modobject = ols_LW,
           IV_focal_1 = 'burden', IV_focal_1_values=NULL,
           IV_focal_2 = 'belong_thwarted', IV_focal_2_values=NULL,
           bootstrap=TRUE, N_sims=100, CI_level=95)

logmod_Meyers <-
  LOGISTIC_REGRESSION(data = data_Meyers_2013, DV='graduated',
                      forced = c('sex','family_encouragement'))

PLOT_MODEL(modobject = logmod_Meyers,
           IV_focal_1 = 'family_encouragement', IV_focal_1_values=NULL,
           IV_focal_2=NULL, IV_focal_2_values=NULL,
           bootstrap=FALSE, N_sims=100, CI_level=95)

pois_Krem <-
  COUNT_REGRESSION(data=data_Kremelburg_2011, DV='OVRJOYED', forced=NULL,
                   hierarchical= list(step1=c('AGE', 'female'),
                                      step2=c('EDUC','REALRINC','DEGREE')) )

PLOT_MODEL(modobject = pois_Krem,
           IV_focal_1 = 'AGE',
           IV_focal_2 = 'DEGREE',
           IVs_nonfocal_values = list(EDUC = 5, female = '2'),
           bootstrap=FALSE, N_sims=100, CI_level=95)
```

REGIONS_OF_SIGNIFICANCE *Plots of Johnson-Neyman regions of significance for interactions*

Description

Plots of Johnson-Neyman regions of significance for interactions in moderated multiple regression, for both MODERATED_REGRESSION models (which are produced by this package) and for lme models (from the nlme package).

Usage

```
REGIONS_OF_SIGNIFICANCE(model,
                           IV_range=NULL, MOD_range=NULL,
                           plot_title=NULL, Xaxis_label=NULL,
                           Yaxis_label=NULL, legend_label=NULL,
                           names_IV_MOD=NULL)
```

Arguments

`model`	The name of a MODERATED_REGRESSION model, or of an lme model from the nlme package.
`IV_range`	(optional) The range of the IV to be used in the plot. Example: IV_range = c(1, 10)
`MOD_range`	(optional) The range of the MOD values to be used in the plot. Example: MOD_range = c(2, 4, 6)
`plot_title`	(optional) The plot title. Example: plot_title = 'Regions of Significance Plot'
`Xaxis_label`	(optional) A label for the X axis to be used in the plot. Example: Xaxis_label = 'IV name'
`Yaxis_label`	(optional) A label for the Y axis to be used in the plot. Example: Yaxis_label = 'DV name'
`legend_label`	(optional) The legend label. Example: legend_label = 'Simple Slopes'
`names_IV_MOD`	(optional) and for lme/nlme models only. Use this argument to ensure that the IV and MOD variables are correctly identified for the plot. There are three scenarios in particular that may require specification of this argument: • when there are covariates in addition to IV & MOD as predictors, • if the order of the variables in model is not IV then MOD, or, • if the IV is a two-level factor (because lme alters the variable names in this case). Example: names_IV_MOD = c('IV name', 'MOD name')

Value

A list with the following possible components:

`JN.data`	The Johnson-Neyman results for a moderated regression.
`ros`	The Johnson-Neyman regions of significance for a moderated regression.

Author(s)

Brian P. O'Connor

References

Bauer, D. J., & Curran, P. J. (2005). Probing interactions in fixed and multilevel regression: Inferential and graphical techniques. *Multivariate Behavioral Research, 40(3),* 373-400.

Huitema, B. (2011). *The analysis of covariance and alternatives: Statistical methods for experiments, quasi-experiments, and single-case studies.* John Wiley & Sons.

Johnson, P. O., & Neyman, J. (1936). Tests of certain linear hypotheses and their application to some educational problems. *Statistical Research Memoirs, 1,* 57-93.

Johnson, P. O., & Fey, L. C. (1950). The Johnson-Neyman technique, its theory, and application. *Psychometrika, 15,* 349-367.

Pedhazur, E. J. (1997). *Multiple regression in behavioral research: Explanation and prediction.* (3rd ed.). Wadsworth Thomson Learning

Rast, P., Rush, J., Piccinin, A. M., & Hofer, S. M. (2014). The identification of regions of significance in the effect of multimorbidity on depressive symptoms using longitudinal data: An application of the Johnson-Neyman technique. *Gerontology, 60,* 274-281.

Examples

```
# for Cohen, Cohen, West, & Aiken, 2003, Chapter 7, p 276
CAW_7 <-
MODERATED_REGRESSION(data=data_Cohen_Aiken_West_2003_7, DV='yendu',
                     IV='xage',IV_range='tumble',
                     MOD='zexer', MOD_levels='quantiles',
                     quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                     plot_type = 'interaction')

REGIONS_OF_SIGNIFICANCE(model=CAW_7)

# for Bauer & Curran, 2005, p. 395
HSBmod <-nlme::lme(MathAch ~ Sector + CSES + CSES:Sector,
                   data = data_Bauer_Curran_2005,
                   random = ~1 + CSES|School, method = "ML")
summary(HSBmod)

REGIONS_OF_SIGNIFICANCE(model=HSBmod,
                        plot_title='Johnson-Neyman Regions of Significance',
                        Xaxis_label='Child SES',
                        Yaxis_label='Slopes of School Sector on Math achievement')


# for O'Connor & Dvorak, 2001, p. 17; with numeric values for IV_range & MOD_levels='AikenWest'
mharsh_agg <-
  MODERATED_REGRESSION(data=data_OConnor_Dvorak_2001, DV='Aggressive_Behavior',
                       IV='Maternal_Harshness', IV_range=c(1,7.7),
                       MOD='Resiliency', MOD_levels='AikenWest',
                       quantiles_IV=c(.1, .9), quantiles_MOD=c(.25, .5, .75),
                       center = FALSE,
                       plot_type = 'interaction',
                       DV_range = c(1,6),
                       Xaxis_label='Maternal Harshness',
                       Yaxis_label='Adolescent Aggressive Behavior',
                       legend_label='Resiliency')

REGIONS_OF_SIGNIFICANCE(model=mharsh_agg,
                        plot_title='Johnson-Neyman Regions of Significance',
                        Xaxis_label='Resiliency',
                      Yaxis_label='Slopes of Maternal Harshness on Aggressive Behavior')
```

SET_CORRELATION *Cohen's Set Correlation Analysis*

Description

Performs Cohen's set correlation analysis of associations between two sets of variables while statistically controlling for one or more other variables. Estimates of overall, multivariate association between the two sets of variables are provided, along with partial correlations and output from OLS regression analyses for each dependent variable.

Usage

```
SET_CORRELATION(data, IVs, DVs, IV_covars=NULL, DV_covars=NULL,
                Ncases=NULL, verbose=TRUE, display_cormats=FALSE)
```

Arguments

`data`	Either a dataframe of raw data (where the rows are cases and the columns are the variables), or a square correlation matrix with row and column names.
`IVs`	The name(s) of the independent/predictor variable(s) in data. Example: IVs = c('var1', 'var2', 'var3')
`DVs`	The name(s) of the dependent variable(s) in data. Example: DVs = c('var4', 'var5', 'var6')
`IV_covars`	The name(s) of the variable(s), if any, to be partialled out of the IVs. Example: IV_covars = c('var7', 'var8')
`DV_covars`	The name(s) of the variable(s), if any, to be partialled out of the DVs. Example: DV_covars = c('var9', 'var10')
`Ncases`	The number of cases. Required only when the input (data) is a correlation matrix.
`verbose`	Should detailed results be displayed in console? The options are: TRUE (default) or FALSE.
`display_cormats`	Should the variable correlation matrices be displayed in console? The options are: TRUE or FALSE(default).

Details

Set correlation analysis and canonical correlation analysis are both fully multivariate methods for examining associations between two sets of variables. However, in CCA the focus is on linear combinations of predictor and criterion variables, which are often difficult to interpret. In contrast, in set correlation analysis the focus is typically on the associations between two sets of variables while statistically controlling for other variables (rather than on linear combinations). The outcome variables of interest in set correlation analysis are the (possibly partialled) dependent variables themselves and not composites of variables.

A key feature of set correlation analysis is the option of examining the overlap between two sets of variables while statistically controlling for one or more other variables. The covariates that are removed from one set of variables (e.g., the DVs) may or may not be the same covariates that are removed from the other set of variables (e.g., the IVs).

In the present function, when there is a wish to statistically remove the same covariates from both sets (i.e., from both the IVs and DVs), then simply enter the same covariate names on both the IV_covars and DV_covars arguments.

The options together result in five different types of data scenarios that can be examined:

Whole, in which the associations between two sets (IVs and DVs) are assessed without any partialling out whatsoever;

Partial, in which the associations between two sets (IVs and DVs) are assessed while partialling the same covariates (one or more) out of both the IVs and DVs;

X Semipartial, in which the associations between two sets (IVs and DVs) are assessed while partialling one or more covariates out of the IV set while leaving the variables in the DV set untouched (unpartialled);

Y Semipartial, in which the associations between two sets (IVs and DVs) are assessed while partialling one or more covariates out of the DV set while leaving the variables in the IV set untouched (unpartialled); and

Bipartial, in which the associations between two sets (IVs and DVs) are assessed while partialling one or more covariates out of the DV set and while partialling one or more other (different) covariates out of the IV set.

The set correlation analyses in this function are conducted using only the correlations between the variables. When raw data are entered into the function, the variable correlation matrix is computed and becomes the sole basis of all further set correlation analyses.

Value

An object of class "SET_CORRELATION". The object is a list containing the following components:

`bigR`	The Pearson correlation matrix for the variables in the analyses.
`Ryy`	The correlations between the DVs.
`Rxx`	The correlations between the IVs.
`Rx_y`	The correlation between the DVs and IVs.
`betas`	The standardized betas.
`se_betas`	The standard errors of the standardized betas.
`t`	The t test values for the standardized betas.
`pt`	The p values for the t tests for the standardized betas.

Author(s)

Brian P. O'Connor

References

Cohen, J. (1982). Set correlation as a general multivariate data-analytic method. *Multivariate Behavioral Research, 17(3)*, 301-341.

Cohen, J. (1988). Set correlation and multivariate Methods. In J. Cohen, *Statistical power analysis for the behavioral sciences* (2nd ed., pp. 467-530). Mahwah, NJ: Erlbaum.

Cohen, J. (1993). Set correlation. In G. Keren & C. Lewis (Eds.), *A handbook for data analysis in the behavioral sciences: Statistical issues* (pp. 165-198). Mahwah, NJ: Erlbaum.

Cohen, J., Cohen, P., West, S. G., & Aiken, L. S. (2003). Multiple dependent variables: Set correlation. In, *Applied multiple regression/correlation analysis for the behavioral sciences* (3rd ed., pp. 608-628). Lawrence Erlbaum Associates.

Examples

```
# data from Cohen et al. (2003)
Cohen_2003_p621 <- '
 1.0
 .53  1.0
 .62  .61  1.0
 .19  .23  .03  1.0
-.09  .10  .10 -.02  1.0
 .08  .18  .12  .02  .05  1.0
 .02  .02  .03  .00  .06  .22  1.0
-.12 -.10 -.06 -.02  .18 -.07 -.01  1.0
 .08  .15  .12 -.02  .02  .36 -.05 -.03  1.0'

Cohen_2003_p621_noms <- c('ADHD', 'CD', 'ODD', 'Sex', 'Age', 'MONLY',
                          'MWORK', 'MAGE', 'Poverty')

Cohen_2003_p621 <- data.matrix( read.table(text=Cohen_2003_p621, fill=TRUE,
                                           col.names=Cohen_2003_p621_noms,
                                           row.names=Cohen_2003_p621_noms ))
Cohen_2003_p621[upper.tri(Cohen_2003_p621)] <-
  t(Cohen_2003_p621)[upper.tri(Cohen_2003_p621)]

# whole
SET_CORRELATION(data=Cohen_2003_p621,
                IVs = c('Sex', 'Age', 'MONLY', 'MWORK', 'MAGE', 'Poverty'),
                DVs = c('ADHD', 'CD', 'ODD'),
                IV_covars = NULL,
                DV_covars = NULL,
                Ncases = 701)

# bipartial
SET_CORRELATION(data=data_DeLeo_2013,
                IVs = c('Grade_Point_Average','Family_Morals','Social_Support',
                  'Intolerance_of_Deviance','Impulsivity','Social_Interaction_Anxiety'),
                DVs = c('Problematic_Internet_Use','Tobacco_Use',
                        'Alcohol_Use','Illicit_Drug_Use'),
```

```
                IV_covars = c('Age','Parents_Income'),
                DV_covars = c('Gambling_Behavior','Unprotected_Sex'),
                display_cormats=TRUE)

# X semipartial
SET_CORRELATION(data=data_DeLeo_2013,
                IVs = c('Grade_Point_Average','Family_Morals','Social_Support',
                  'Intolerance_of_Deviance','Impulsivity','Social_Interaction_Anxiety'),
                DVs = c('Problematic_Internet_Use','Tobacco_Use',
                        'Alcohol_Use','Illicit_Drug_Use'),
                IV_covars = c('Age','Parents_Income'),
                DV_covars = NULL)

# partial
SET_CORRELATION(data=data_DeLeo_2013,
                IVs = c('Grade_Point_Average','Family_Morals','Social_Support',
                  'Intolerance_of_Deviance','Impulsivity','Social_Interaction_Anxiety'),
                DVs = c('Problematic_Internet_Use','Tobacco_Use',
                        'Alcohol_Use','Illicit_Drug_Use'),
                IV_covars = c('Age','Parents_Income'),
                DV_covars = c('Age','Parents_Income'))
```

Index