Package 'SetMethods'

July 2, 2014

Type Package

Title SetMethods: A Package Companion to ``Set-Theoretic Methods for the Social Sciences".

Version 1.0

Date 2013-05-17

Author Mario Quaranta

Maintainer Mario Quaranta <mquaranta@luiss.it>

Description SetMethods is a package companion to the book by C. Q. Schneider and C. Wagemann ``Set-Theoretic Methods for the Social Sciences", Cambridge University Press. It contains some additional functions not present in other packages and data to replicate the examples in the book and in the online appendix.

Depends lattice, betareg, R (>= 2.14.0)

Imports lattice, betareg

License GPL-2

NeedsCompilation no

Repository CRAN

Date/Publication 2013-05-21 14:52:42

R topics documented:

`SetMethods-package` *SetMethods: A Package Companion to "Set-Theoretic Methods for the Social Sciences"*

Description

SetMethods is a package companion to the book by C. Q. Schneider and C. Wagemann "Set-Theoretic Methods for the Social Sciences", Cambridge University Press. It contains some additional functions not present in other packages and data to replicate the examples in the book and in the Online Appendix.

Details

Package:	SetMethods
Type:	Package
Version:	1.0
Date:	2013-05-17
License:	GPL-2

The package contains some additional functions to perform indirect calibration, calculate parameters of fit and produce XY plots. Furthermore, it contains all the data used in "Set-Theoretic Methods for the Social Sciences". The package is intended to be used to replicate the analyses shown in the Online Appendix of the book.

Author(s)

Mario Quaranta

Maintainer: Mario Quaranta <mario.quar@gmail.com>

References

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012). How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

---

`CronBerg` *Berg-Schlosser and Cronqvist (2005)*

Description

The `CronBerg` data frame has 18 rows and 5 variables

Usage

```
data(CronBerg)
```

Format

A data frame with 18 observations on the following 5 variables.

`GNP` a numeric vector. Condition, Gross National Product/Capita (ca. 1930). 0 if below 500$, 1 if between 550 and 850$, 2 above 850$.

`URB` a numeric vector. Condition, urbanization (population in towns with 20000 and more inhabitants); 0 if below 50 per cent; 1 if above.

`LIT` a numeric vector. Condition, literacy: 0 if below 75 per cent; 1 if above.

`INDUS` a numeric vector. Condition, Industrial Labour Force (incl. mining); 0 if below 30 per cent of active population; 1 if above.

`DEMOC` a numeric vector. Condition, stability of a democracy: 0 if not stable; 1 if stable.

Details

The data are used by Berg-Schlosser and Cronqvist (2005) to demostrate mvQCA. The original data are from Lipset (1963). Data are multi-value.

References

Berg-Schlosser, D. and Cronqvist, L. (2005) "Macro-Quantitative vs. Macro-Qualitative Methods in the Social Sciences - An Example from Empirical Democratic Theory", Historical Social Research 30, pp. 154-175.

Lipset, Seymour M. (1963) Political Man. The Social Bases of Politics. Doubleday: New York.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
data(CronBerg)
```

DTmv *Fake data for mvQCA*

Description

mvQCA data frame has 25 rows and 4 variables.

Usage

```
data(DTmv)
```

Format

A data frame with 25 observations on the following 4 variables.

`Y` a numeric vector. Outcome with 2 categories (crisp).

`A` a numeric vector. Condition with 2 categories (crisp).

`B` a numeric vector. Condition with 3 categories (multi-value).

`C` a numeric vector. Condition with 3 categories (multi-value).

Details

The data frame has only exercise purpuses to let the user learn how to perform mvQCA in R.

References

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Examples

```
data(DTmv)
```

emme *Emmenegger (2011)*

Description

The emme data frame has 19 rows and 8 variables

Usage

```
data(emme)
```

Format

A data frame with 19 observations on the following 8 variables.

`country` a factor with levels `Australia` `Austria` `Belgium` `Canada` `Denmark` `Finland` `France` `Germany` `Ireland` `Italy` `Netherlands` `NewZealand` `Norway` `Portugal` `Spain` `Sweden` `Switzerland` `UK` `USA`

`s` a numeric vector. Condition, state-society relationships.

`c` a numeric vector. Condition, non-market coordination.

`l` a numeric vector. Condition, strength of the labour movement.

`r` a numeric vector. Condition, religious denomination.

`p` a numeric vector. Condition, strenght of religious parties.

`v` a numeric vector. Condition, institutional veto points.

`jsr` a numeric vector. Outcome, job-secturity regulations.

Details

Data are used by Emmenegger (2011) to analyze job-security regulations in Western democracies. The data are fuzzy-sets.

References

Emmenegger, P. (2011) "Job-security regulations in Western democracies", European Journal of Political Research 50, pp. 336-364.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
data(emme)
```

FakeCS *Fake crisp-set data*

Description

The FakeCS data frame has 30 rows and 5 variables

Usage

```
data(FakeCS)
```

Format

A data frame with 30 observations on the following 5 variables.

`y` a numeric vector. Outcome with 2 categories (crisp-set).

`j` a numeric vector. Condition with 2 categories (crisp-set).

`z` a numeric vector. Condition with 2 categories (crisp-set).

`w` a numeric vector. Condition with 2 categories (crisp-set).

`k` a numeric vector. Condition with 2 categories (crisp-set).

Details

The data frame has only exercise purpuses to let the user learn how to perform crisp-set QCA in R.

References

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Examples

```
data(FakeCS)
```

FRSC *Freitag and Schlicht (2009)*

Description

The FRSC data frame has 16 rows and 8 variables

Usage

```
data(FRSC)
```

Format

A data frame with 16 observations on the following 8 variables.

`integrated_comp_schools` a numeric vector. Condition, percentage of Pupils Enrolled in Integrated Comprehensive Schools,

`coop_comp_schools` a numeric vector. Condition, percentage of Pupils Enrolled in Cooperative Comprehensive Schools.

`full_day_schools` a numeric vector. Condition, percentage of Pupils Enrolled in All-Day Schools

`child_care` a numeric vector. Condition, ratio of Number of Child Care Facilities to Total Population between 0 and 6 Years (percent).

`pre_schools` a numeric vector. Condition, ratio of pupils Enrolled in Pre-School to Total 6-Year-Old Population (per cent)

`early_tracking` a numeric vector. Condition, onset of Tracking, Legal Regulation.

`outcome` a numeric vector. Outcome, high Degree of Social Inequality Cases in Education.

`indep_hauptschule` a numeric vector. Condition, autonomy of the Hauptschule.

Details

Data are used by Freitag and Schlicht (2009) to analyze social inequality in education. The data are raw scores.

References

Freitag, M, and Schlicht, R. (2009) "Educational Federalism in Germany: Foundations of Social Inequalities in Education", Governance 22(1), pp. 47-72.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
data(FRSC)
```

`indirectCalibration` *Function performing the indirect calibration*

Description

indirectCalibration is a function for the indirect calibration procedure as described by Ragin (2008). It uses a binomial or a beta regression for tranforming raw scores into calibrated scores. In our opinion, using a fractional polynomial may not be appropriate to this case. In fact, we do not deal with proportions. This function requires the package `betareg`.

Usage

```
indirectCalibration(x, x_cal, binom = TRUE)
```

Arguments

`x`	vector of raw scores.
`x_cal`	vector of theoretically calibrated scores.
`binom`	logical. If indirect calibration has to be performed using binomial regression or beta regression. The default is `TRUE`, which means that binomial regression is used.

Value

It returns a vector of indirectly calibrated values.

Author(s)

Mario Quaranta

References

Ragin, C. C. (2008) Redesigning Social Inquiry: Fuzzy Sets and Beyond, The Chicago University Press: Chicago and London.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
# Generate fake data
set.seed(4)
x <- runif(20, 0, 1)
```

```
# Find quantiles
quant <- quantile(x, c(.2, .4, .5, .6, .8))

# Theoretical calibration
x_cal <- NA
x_cal[x <= quant[1]] <- 0
x_cal[x > quant[1] & x <= quant[2]] <- .2
x_cal[x > quant[2] & x <= quant[3]] <- .4
x_cal[x > quant[3] & x <= quant[4]] <- .6
x_cal[x > quant[4] & x <= quant[5]] <- .8
x_cal[x > quant[5]] <- 1
x_cal

# Indirect calibration (binomial)
a <- indirectCalibration(x, x_cal, binom = TRUE)

# Indirect calibration (beta regression)
b <- indirectCalibration(x, x_cal, binom = FALSE)

# Correlation
cor(a, b)

# Plot
plot(x, a); points(x, b, col = "red")
```

KA *Koenig-Archibugi (2004)*

Description

The KA data frame has 13 rows and 5 variables

Usage

```
data(KA)
```

Format

A data frame with 13 observations on the following 5 variables.

`supranat` a numeric vector. Government support for supranational CFSP.

`identmass` a numeric vector. European identity of the general public.

`conform` a numeric vector. Policy conformity.

`region` a numeric vector. Regional governance.

`capab` a numeric vector. Material capabilities.

Details

Data are used by Koenig-Archibugi (2004) to analyze government preferences for istitutional change in EU foreign and security policy. Data are fuzzy-sets.

References

Koenig-Archibugi, M. (2004) "Explaining Government Preferences for Istitutional Change in EU Foreign and Security Policy", International Organization 58, pp.137-174.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
data(KA)
```

LipsetCS *Lipset (1959), crisp-set*

Description

The `LipsetCS` data frame has 18 rows and 6 variables

Usage

```
data(LipsetCS)
```

Format

A data frame with 18 observations on the following 6 variables.

`DEVELOPED` a numeric vector. Condition, economically developed country.

`URBAN` a numeric vector. Condition, urbanized countries.

`LITERATE` a numeric vector. Condition, countries with high literacy rate.

`INDUSTRIAL` a numeric vector. Condition, Industrialized countries.

`GOVSTAB` a numeric vector. Condition, politically stable countries.

`SURVIVED` a numeric vector. Outcome, survival of democracy during the inter-war period.

Details

Data used by Ragin (2009) to illustrates the variants of QCA. Originally by Lipset (1959). Data are crisp-sets.

References

Lipset, S. M. (1959) "Some Social Requisites of Democracy: Economic Development and Political Legitimacy", American Political Science Review 53, pp. 69-105.

Ragin, C. C. (2009) "Qualitative Comparative Analysis. Using Fuzzy Sets (fsQCA)." In Rihoux, B., and Ragin, C. C. (eds.) Configurational Comparative Methods. Qualitative Comparative Analysis (QCA) and Related Techniques. Thousand Oaks, CA and London: Sage, pp. 87-121.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann.

Examples

```
data(LipsetCS)
```

LipsetFS *Lipset (1959), fuzzy-set*

Description

The LipsetFS data frame has 18 rows and 6 variables

Usage

```
data(LipsetFS)
```

Format

A data frame with 18 observations on the following 6 variables.

`Survived` a numeric vector. Outcome, survival of democracy during the inter-war period.

`DEVELOPED` a numeric vector. Condition, economically developed countries.

`URBAN` a numeric vector. Condition, urbanized countries.

`LITERATE` a numeric vector. Condition, countries with high literacy rate.

`INDUSTRIAL` a numeric vector. Condition, industrialized countries.

`STABLE` a numeric vector. Condition, politically stable countries.

Details

Data used by Ragin (2009) to illustrates the variants of QCA. Originally by Lipset (1959). Data are fuzzy-sets.

References

Lipset, S. M. (1959) "Some Social Requisites of Democracy: Economic Development and Political Legitimacy", American Political Science Review 53, pp. 69-105.

Ragin, C. C. (2009) "Qualitative Comparative Analysis. Using Fuzzy Sets (fsQCA)." In Rihoux, B., and Ragin, C. C. (eds.) Configurational Comparative Methods. Qualitative Comparative Analysis (QCA) and Related Techniques. Thousand Oaks, CA and London: Sage, pp. 87-121.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann.

Examples

```
data(LipsetFS)
```

LipsetRaw *Lipset (1959), raw data*

Description

The LipsetRaw data frame has 18 rows and 6 variables

Usage

```
data(LipsetRaw)
```

Format

A data frame with 18 observations on the following 6 variables.

SURVIVED a numeric vector. Outcome, survival of democracy during the inter-war period.

DEVELOPED a numeric vector. Condition, level of economic development.

URBAN a numeric vector. Condition, level of urbanization.

LITERATE a numeric vector. Condition, level of literacy.

INDUSTRIAL a numeric vector. Condition, level of industrialization.

UNSTABLE a numeric vector. Condition, politically stable countries.

Details

Data used by Ragin (2009) to illustrates the variants of QCA. Originally by Lipset (1959). Data are raw-scores.

References

Lipset, S. M. (1959) "Some Social Requisites of Democracy: Economic Development and Political Legitimacy", American Political Science Review 53, pp. 69-105.

Ragin, C. C. (2009) "Qualitative Comparative Analysis. Using Fuzzy Sets (fsQCA)." In Rihoux, B., and Ragin, C. C. (eds.) Configurational Comparative Methods. Qualitative Comparative Analysis (QCA) and Related Techniques. Thousand Oaks, CA and London: Sage, pp. 87-121.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann.

Examples

```
data(LipsetRaw)
```

`Pennings` *Pennings (2003)*

Description

The `Pennings` data frame has 45 rows and 5 variables

Usage

```
data(Pennings)
```

Format

A data frame with 45 observations on the following 5 variables.

`K` a numeric vector. Outcome, constitutional control.

`C` a numeric vector. Condition, consensus democracy.

`P` a numeric vector. Condition, presidentialism.

`N` a numeric vector. Condition, new democracy.

`R` a numeric vector. Condition, rigid constritution.

Details

Data used by Pennings (2009) to explain constitutional control. Data are fuzzy-sets.

References

Pennings, P. (2009) "Beyond Dichotomous Explanations: Explaining Constitutional control of the Executive with Fuzzy-sets", European Journal of Political Research 42, pp. 541-567.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann.

Examples

```
data(Pennings)
```

QCAfit *Function calculating the parameters of fit*

Description

QCAfit is a function calculating parameters of fit useful in QCA and fsQCA that are consistency, coverage, PRI and PRODUCT. It works with both single and multiple conditions.

Usage

```
QCAfit(x, y, cond.lab = NULL, necessity = FALSE, negation = FALSE)
```

Arguments

x	is a vector containing the values of a condition or a matrix with more than one conditions.
y	is a vector containing the values of the output.
cond.lab	is a vector containing the label(s) of the condition(s).
necessity	logical. It indicates if the output should be for sufficient or necessary condition(s). By default, TRUE, the function returns a table of parameters of fit for sufficient condition(s) (Consistency, Coverage, PRI and Product). When it set to FALSE the function returns a table of parameters of fit for necessary condition(s) (Consistency, Coverage, Relevance of Necessity).
negation	logical. It indicates if the parameters of fit should be computed for the positive or the negative outcome. By default it is set to be FALSE, so it returns parameters of fit for the positive outcome.

Value

It returns a matrix containing the parameters of fit for each condition.

Author(s)

Mario Quaranta

References

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
# Generate fake data
set.seed(1234)

a <- runif(100, 0, 1)
b <- runif(100, 0, 1)
c <- runif(100, 0, 1)
y <- runif(100, 0, 1)

# Only one condition, for sufficiency
QCAfit(a, y, cond.lab = "A")

# With three conditions, for sufficiency
QCAfit(cbind(a, b, c), y, cond.lab = c("A", "B", "C"))

# Only one condition, for necessity
QCAfit(a, y, cond.lab = "A", necessity = TRUE)

# With three conditions, for necessity
QCAfit(cbind(a, b, c), y, cond.lab = c("A", "B", "C"), necessity = TRUE)

# With three conditions and negated output, for sufficiency
QCAfit(cbind(a, b, c), y, cond.lab = c("A", "B", "C"), negation = TRUE)
```

Samford *Samford (2010)*

Description

The Samford data frame has 61 rows and 4 variables

Usage

```
data(Samford)
```

Format

A data frame with 61 observations on the following 4 variables.

`Y` a numeric vector. Outcone, trade liberalization.

`G` a numeric vector. Condition, lack of weak growth.

`H` a numeric vector. Condition, lack of hyper-inflation

`HorG` a numeric vector. Condition, H or G.

Details

Data are used by Samford (2010) to analyze rapid trade liberalization in Latin America. Data are fuzzy-sets.

References

Samford, S. (2010) "Averting 'Disruption and Reversal': Reassessing the Logic of Rapid Trade Reform in Latin America", Politics and Policy 38(3), pp. 373-407.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
data(Samford)
```

`Selbst` *Selbst, practicing the truth table algorithm data*

Description

The `Selbst` data frame has 130 rows and 4 variables

Usage

```
data(Selbst)
```

Format

A data frame with 130 observations on the following 4 variables.

`A` a numeric vector. Condition, crisp-set.

`B` a numeric vector. Condition, crisp-set.

`C` a numeric vector. Condition, crisp-set.

`Y` a numeric vector. Condition, crisp-set.

Details

The authors of the data are Carsten Schnieder and Claudius Wagemann. The data are used in the on-line appendix of "Set-Theoretic Methods for the Social Sciences" to practice the truth table algorithm. Data are crisp-sets.

References

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
data(Selbst)
```

`Selbst.disappear` *Selbst, disappearing necessary condition data*

Description

The `Selbst.disappear` data frame has 98 rows and 4 variables

Usage

```
data(Selbst.disappear)
```

Format

A data frame with 98 observations on the following 4 variables.

`A` a numeric vector. Condition, crisp-set.

`B` a numeric vector. Condition, crisp-set.

`C` a numeric vector. Condition, crisp-set.

`Y` a numeric vector. Outcome, crisp-set.

Details

The authors of the data are Carsten Schnieder and Claudius Wagemann. The data are used in the on-line appendix of "Set-Theoretic Methods for the Social Sciences" to show the disappearance of a necessary condition. Data are crisp-sets.

References

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
data(Selbst.disappear)
```

`VisCS` *Vis (2009), crisp set data*

Description

The `VisCS` data frame has 25 rows and 4 variables

Usage

```
data(VisCS)
```

Format

A data frame with 25 observations on the following 4 variables.

`P` a numeric vector. Condition, weak political positions, with parties in government expecting losses at the next election.

`S` a numeric vector. Condition, deteriorating economic situation.

`R` a numeric vector. Condition, government dominated by parties from the right of the political spectrum.

`U` a numeric vector. Outcome, unpopular reform.

Details

Data are used by Vis (2009) to analyze the pursuit of unpopular reforms by governments. Data are crisp-sets.

References

Vis, B. (2009) "Government and Unpopular Social Policy Reforms: Biting the Bullet or Steering Clear?", European Journal of Political Research 48, pp. 31-57.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
data(VisCS)
```

`VisFS` *Vis (2009), fuzzy set data*

Description

The `VisFS` data frame has 25 rows and 4 variables

Usage

```
data(VisFS)
```

Format

A data frame with 25 observations on the following 4 variables.

`p` a numeric vector. Condition, weak political positions, with parties in government expecting losses at the next election.

`s` a numeric vector. Condition, deteriorating economic situation.

`r` a numeric vector. Condition, government dominated by parties from the right of the political spectrum.

`u` a numeric vector. Outcome, unpopular reform.

Details

Data are used by Vis (2009) to analyze the pursuit of unpopular reforms by governments. Data are fuzzy-sets.

References

Vis, B. (2009) "Government and Unpopular Social Policy Reforms: Biting the Bullet or Steering Clear?", European Journal of Political Research 48, pp. 31-57.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
data(VisFS)
```

xy.plot *Function producing enhanced XY plots*

Description

xy.plot produces xyplots and provides coverage and consistency values. The advantage over `fsplot()` (in the QCA3 package) is its larger flexibility and that it does not need a dataset to work, it just need two vectors. Several graphic parameters can be decided by the user.

Usage

```
xy.plot(x, y,
        ylim = c(-0.05, 1.05), xlim = c(-0.05, 1.05),
        pch = 19, col = "black", main = "XY plot",
        ylab = "Outcome", xlab = "Condition",
        mar = c(4, 4, 4, 1), mgp = c(2.2, 0.8, 0),
        cex.fit = 0.6, cex.axis = 0.7, cex.main = 1,
        necessity = FALSE, show.hv = TRUE, show.fit = TRUE,
        pos.fit = "top", case.lab = TRUE, labs = NULL,
        cex.lab = 0.8, offset.x = 0, offset.y = 0,
        pos = 4, srt = 0,
        ident = FALSE)
```

Arguments

x	vector containing the condition.
y	vector containing the outcome.
ylim	limits of y-axis. The default is `c(-0.05, 1.05)`.
xlim	limits of x-axis. The default is `c(-0.05, 1.05)`.
pch	plotting "character". The default is `19`. See ?pch.
col	color for the plotting "character". The default is `"black"`. See ?par.
main	an overall title for the plot. The default is `"XY plot"`. See ?title.
ylab	a title for the y-axis. The default is `"Outcome"`. See ?title.
xlab	a title for the x-axis. The default is `"Condition"`. See ?title.
mar	A numerical vector of the form `c(bottom, left, top, right)` which gives the number of lines of margin to be specified on the four sides of the plot. The default is `c(4, 4, 4, 1)`. See ?par.
mgp	The margin line (in `mex` units) for the axis title, axis labels and axis line. Note that `mgp[1]` affects title whereas `mgp[2:3]` affect axis. The default is `c(2.2, 0.8, 0)`. See ?par.
cex.fit	character expansion for the parameters of fit. The defaut is `0.6`. See ?pch or ?text.
cex.axis	character expansion for the x-axis and y-axis. The defaut is `0.7`. See ?pch or ?text.

cex.main	character expansion for the overall title of the plot. The defaut is 1. See ?pch or ?text.
necessity	logical. Indicates if the parameters of fit are calculated for a sufficient or necessary condition. The default is FALSE, therefore it calculates the parameters of fit for sufficiency. To get the parameters of fit for necessary conditions set necessity as TRUE.
show.hv	logical. Indicates if horizontal and vertical lines at 0.5 have to be shown. The default is TRUE.
show.fit	logical. Indicates if parameters of fit have to be shown. The default is TRUE.
pos.fit	character. Indicates the position of the parameters of fit. The positions are "top", which places the parameters of fit outsite the plotting are just below the main title, or "corner", which places the parameters of fit in the corner of the plotting are. The default is pos.fit = "top".
case.lab	logical. Indicates if cases have to be labeled. The default is TRUE.
labs	the vector of case labels. The default is NULL.
cex.lab	character expansion for case labels. The default is 0.8.
offset.x	is a numerical value that sets the offset for case labels position on the x-axis. The default is 0.
offset.y	is a numerical value that sets the offset for case labels position on the y-axis. The default is 0.
pos	a position specifier for the case labels. Values of 1, 2, 3 and 4, respectively indicate positions below, to the left of, above and to the right of the specified coordinates. The default is 4. See ?text.
srt	indicates the rotation of the case labels in degrees. The default is 0. See ?par.
ident	logical. Indicates if identify() has to be used to label the cases. When set to TRUE case.lab has to be FALSE and labels have to be provided by the user. The defaults is FALSE. See ?identify.

Value

It returns an enhanced XY plot.

Author(s)

Mario Quaranta.

References

Ragin, C. C. (2008) Redesigning Social Inquiry: Fuzzy Sets and Beyond. The Chicago University Press: Chicago and London.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge Univeristy Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Examples

```
# Generate fake data
set.seed(123)
x <- runif(40, 0, 1)
y <- runif(40, 0, 1)

# Default
xy.plot(x, y)

# With labels
xy.plot(x, y, case.lab = TRUE, labs = 1:40)

# With labels and bigger measures of fit
xy.plot(x, y, case.lab = TRUE, labs = 1:40, cex.fit = 1)

# With labels and bigger title
xy.plot(x, y, case.lab = TRUE, labs = 1:40, cex.main = 1.5)

# Generate fake data the have perfect sufficiency
set.seed(123)
x <- runif(50, 0, 1)
y <- runif(50, 0, 1)

for(i in 1:length(y)) {
    while(x[i] > y[i]) {
        y[i] <- runif(1, 0, 1)
        x[i] <- runif(1, 0, 1)
    }
}

# Default
xy.plot(x, y)
```

`xy.plot.lat` *Function producing enhanced XY plots with Lattice*

Description

`xy.plot.lat` produces XY plots using the `lattice` package and provides coverage and consistency values. The advantage over `fsplot()` (in the `QCA3` package) is its larger flexibility and that it does not need a dataset to work, it just needs two vectors. Several graphic parameters can be decided by the user.

Usage

```
xy.plot.lat(x, y,
            ylim = c(-0.05, 1.05), xlim = c(-0.05, 1.05),
            main = "", pch = 19, col = "black", cex.fit = 1,
            ylab = "Outcome", xlab = "Condition",
```

```
pos.fit = "top", strip.cex = 0.8,
necessity = FALSE, show.fit = TRUE, case.lab = FALSE,
lab.pos = 4, labs = NULL,
show.hv = TRUE)
```

Arguments

x	vector containing the condition.
y	vector containing the outcome.
ylim	limits of y-axis. The default is `c(-0.05, 1.05)`.
xlim	limits of x-axis. The default is `c(-0.05, 1.05)`.
main	an overall title for the plot. The default is `""`. See ?xyplot.
pch	plotting "character". The default is `19`. See ?pch or ?xyplot.
col	color for the plotting "character". The default is `"black"`. See ?par or ?xyplot.
cex.fit	character expansion for the parameters of fit. The defaut is `0.6`. See ?pch or ?text.
ylab	a title for the y-axis. The default is `"Outcome"`. See ?title or ?xyplot.
xlab	a title for the x-axis. The default is `"Condition"`. See ?title or ?xyplot.
pos.fit	character. Indicates the position of the parameters of fit. The positions are `"top"`, which places the parameters of fit in a strip on top of the plotting area, or `"corner"`, which places the parameters of fit in the corner of the plotting are. The default is `pos.fit = "top"`.
strip.cex	character expansion for the parameters of fit when `pos.fit = "top"`. The defaut is `0.8`. See ?pch or ?text.
necessity	logical. Indicates if the parameters of fit are calculated for a sufficient or necessary condition. The default is `FALSE`, therefore it calculates the parameters of fit for sufficiency. To get the parameters of fit for necessary conditions set `necessity` as `TRUE`.
show.fit	logical. Indicates if parameters of fit have to be shown. The default is `TRUE`.
case.lab	logical. Indicates if cases have to be labeled. The default is `TRUE`.
lab.pos	a position specifier for the case labels. Values of 1, 2, 3 and 4, respectively indicate positions below, to the left of, above and to the right of the specified coordinates. The default is `4`. See ?text.
labs	the vector of case labels. The default is `NULL`.
show.hv	logical. Indicates if horizontal and vertical lines at 0.5 have to be shown. The default is `TRUE`.

Value

It returns an enhanced XY plot using the `lattice` package.

Author(s)

Mario Quaranta

References

Ragin, C. C. (2008) Redesigning Social Inquiry: Fuzzy Sets and Beyond, The Chicago University Press: Chicago and London.

Schneider, C. Q., Wagemann, C. (2012) Set-Theoretic Methods for the Social Sciences, Cambridge University Press: Cambridge.

Schneider, C. Q., Wagemann, C., Quaranta, M. (2012) How To... Use Software for Set-Theoretic Analysis. Online Appendix to "Set-Theoretic Methods for the Social Sciences". Available at www.cambridge.org/schneider-wagemann

Sarkar, D. (2008) Lattice: Multivariate Data Visualization with R, Springer: Berlin.

Examples

```
# Generate fake data to have perfect necessity
set.seed(123)
x <- runif(60, 0, 1)
y <- runif(60, 0, 1)

for(i in 1:length(y)) {
    while(x[i] < y[i]) {
        y[i] <- runif(1, 0, 1)
        x[i] <- runif(1, 0, 1)
    }
}

# Default with blue dots and pch = 1
xy.plot.lat(x, y, pch = 1, col = "blue")

# Parameters of fit in the corners with blue dots and pch = 1
xy.plot.lat(x, y, pch = 1, col = "blue", pos.fit = "corner")
```

Index