Package 'Zelig'

April 22, 2019

License GPL (>= 3)

Title Everyone's Statistical Software

Description A framework that brings together an abundance of common statistical models found across packages into a unified interface, and provides a common architecture for estimation and interpretation, as well as bridging functions to absorb increasingly more models into the package. Zelig allows each individual package, for each statistical model, to be accessed by a common uniformly structured call and set of arguments. Moreover, Zelig automates all the surrounding building blocks of a statistical work-flow--procedures and algorithms that may be essential to one user's application but which the original package developer did not use in their own research and might not themselves support. These include bootstrapping, jackknifing, and re-weighting of data. In particular, Zelig automatically generates predicted and simulated quantities of interest (such as relative risk ratios, average treatment effects, first differences and predicted and expected values) to interpret and visualize complex models.

URL https://cran.r-project.org/package=Zelig

BugReports https://github.com/IQSS/Zelig/issues

Version 5.1.6.1

Date 2018-02-27

Depends survival

Imports AER, Amelia, coda, dplyr (>= 0.3.0.2), Formula, geepack, jsonlite, sandwich, MASS, MatchIt, maxLik, MCMCpack, methods, quantreg, survey, VGAM

Suggests ei, eiPack, knitr, networkD3, optmatch, rmarkdown, testthat, tidyverse, ZeligChoice, ZeligEI, zeligverse

Collate 'assertions.R' 'model-zelig.R' 'model-timeseries.R' 'model-ma.R' 'model-ar.R' 'model-arima.R' 'model-weibull.R' 'model-tobit.R' 'model-bayes.R' 'model-tobit-bayes.R' 'model-glm.R' 'model-binchoice.R' 'model-probit.R' 'model-probit-bayes.R' 'model-poisson.R'

'model-poisson-bayes.R' 'model-oprobit-bayes.R'
'model-normal.R' 'model-normal-bayes.R' 'model-mlogit-bayes.R'
'model-gamma.R' 'model-gee.R' 'model-logit.R'
'model-logit-bayes.R' 'model-factor-bayes.R'
'model-poisson-gee.R' 'model-normal-gee.R' 'model-gamma-gee.R'
'model-binchoice-gee.R' 'model-probit-gee.R'
'model-logit-gee.R' 'model-relogit.R' 'model-quantile.R'
'model-lognorm.R' 'model-exp.R' 'model-negbinom.R'
'model-ivreg.R' 'model-ls.R' 'utils.R' 'create-json.R'
'datasets.R' 'interface.R' 'model-survey.R'
'model-binchoice-survey.R' 'model-gamma-survey.R'
'model-logit-survey.R' 'model-normal-survey.R'
'model-poisson-survey.R' 'model-probit-survey.R' 'plots.R'
'wrappers.R'

RoxygenNote 6.0.1

NeedsCompilation no

Author Christine Choirat [aut],
Christopher Gandrud [aut, cre],
James Honaker [aut],
Kosuke Imai [aut],
Gary King [aut],
Olivia Lau [aut],
IQSS Harvard University [cph]

Maintainer Christopher Gandrud <zelig.zee@gmail.com>

Repository CRAN

Date/Publication 2019-04-22 06:55:01 UTC

R topics documented:

`approval` *U.S. Presidential Approval Data*

Description

Monthy public opinion data for 2001-2006.

Usage

```
data(approval)
```

Format

A table containing 8 variables ("month", "year", "approve", "disapprove", "unsure", "sept.oct.2001", "iraq.war", and "avg.price") and 65 observations.

Source

ICPSR

References

Stuff here

ATT — *Compute simulated (sample) average treatment effects on the treated from a Zelig model estimation*

Description

Compute simulated (sample) average treatment effects on the treated from a Zelig model estimation

Usage

```
ATT(object, treatment, treated = 1, num = NULL)
```

Arguments

object	an object of class Zelig
treatment	character string naming the variable that denotes the treatment and non-treated groups.
treated	value of treatment variable indicating treatment
num	number of simulations to run. Default is 1000.

Author(s)

Christopher Gandrud

Examples

```
library(dplyr)
data(sanction)
z.att <- zelig(num ~ target + coop + mil, model = "poisson",
                 data = sanction) %>%
             ATT(treatment = "mil") %>%
             get_qi(qi = "ATT", xvalue = "TE")
```

avg — *Compute central tendancy as approrpriate to data type*

Description

Compute central tendancy as approrpriate to data type

Usage

```
avg(val)
```

Arguments

val	a vector of values

Value

a mean (if numeric) or a median (if ordered) or mode (otherwise)

---

bivariate *Sample data for bivariate probit regression*

Description

Sample data for the bivariate probit regression.

Usage

```
data(bivariate)
```

Format

A table containing 6 variables ("y1", "y2", "x1", "x2", "x3", and "x4") and 78 observations.

Source

This is a cleaned and relabelled version of the sanction data set, available in Zelig.

References

Martin, Lisa (1992). *Coercive Cooperation: Explaining Multilateral Economic Sanctions*, Princeton: Princeton University Press.

---

ci.plot *Method for plotting qi simulations across a range within a variable, with confidence intervals*

Description

Method for plotting qi simulations across a range within a variable, with confidence intervals

Usage

```
ci.plot(obj, qi="ev", var=NULL, ..., main = NULL, sub =
 NULL, xlab = NULL, ylab = NULL, xlim = NULL, ylim =
 NULL, legcol="gray20", col=NULL, leg=1, legpos=
 NULL, ci = c(80, 95, 99.9), discont=NULL)
```

Arguments

obj	A reference class zelig5 object
qi	a character-string specifying the quantity of interest to plot
var	The variable to be used on the x-axis. Default is the variable across all the chosen values with smallest nonzero variance
...	Parameters to be passed to the 'truehist' function which is implicitly called for numeric simulations
main	a character-string specifying the main heading of the plot
sub	a character-string specifying the sub heading of the plot
xlab	a character-string specifying the label for the x-axis
ylab	a character-string specifying the label for the y-axis
xlim	Limits to the x-axis
ylim	Limits to the y-axis
legcol	"legend color", an valid color used for plotting the line colors in the legend
col	a valid vector of colors of at least length 3 to use to color the confidence intervals
leg	"legend position", an integer from 1 to 4, specifying the position of the legend. 1 to 4 correspond to "SE", "SW", "NW", and "NE" respectively. Setting to 0 or "n" turns off the legend.
legpos	"legend type", exact coordinates and sizes for legend. Overrides argment "leg.type"
ci	vector of length three of confidence interval levels to draw.
discont	optional point of discontinuity along the x-axis at which to interupt the graph

Value

the current graphical parameters. This is subject to change in future implementations of Zelig

Author(s)

James Honaker

CigarettesSW *Cigarette Consumption Panel Data*

Description

Cigarette Consumption Panel Data

Format

A data set with 96 observations and 9 variables

Source

From Christian Kleiber and Achim Zeileis (2008). Applied Econometrics with R. New York: Springer-Verlag. ISBN 978-0-387-77316-2. URL https://CRAN.R-project.org/package=AER

cluster.formula *Generate Formulae that Consider Clustering*

Description

This method is used internally by the "Zelig" Package to interpret clustering in GEE models.

Usage

```
cluster.formula(formula, cluster)
```

Arguments

formula	a formula object
cluster	a vector

Value

a formula object describing clustering

coalition *Coalition Dissolution in Parliamentary Democracies*

Description

This data set contains survival data on government coalitions in parliamentary democracies (Belgium, Canada, Denmark, Finland, France, Iceland, Ireland, Israel, Italy, Netherlands, Norway, Portugal, Spain, Sweden, and the United Kingdom) for the period 1945-1987. For parsimony, country indicator variables are omitted in the sample data.

Usage

```
data(coalition)
```

Format

A table containing 7 variables ("duration", "ciep12", "invest", "fract", "polar", "numst2", "crisis") and 314 observations. For variable descriptions, please refer to King, Alt, Burns and Laver (1990).

Source

ICPSR

References

King, Gary, James E. Alt, Nancy Elizabeth Burns and Michael Laver (1990). "A Unified Model of Cabinet Dissolution in Parliamentary Democracies," *American Journal of Political Science*, vol. 34, no. 3, pp. 846-870.

Gary King, James E. Alt, Nancy Burns, and Michael Laver. ICPSR Publication Related Archive, 1115.

`coalition2` *Coalition Dissolution in Parliamentary Democracies, Modified Version*

Description

This data set contains survival data on government coalitions in parliamentary democracies (Belgium, Canada, Denmark, Finland, France, Iceland, Ireland, Israel, Italy, Netherlands, Norway, Portugal, Spain, Sweden, and the United Kingdom) for the period 1945-1987. Country indicator variables are included in the sample data.

Usage

```
data(coalition2)
```

Format

A data frame containing 8 variables ("duration", "ciep12", "invest", "fract", "polar", "numst2", "crisis", "country") and 314 observations. For variable descriptions, please refer to King, Alt, Burns and Laver (1990).

Source

ICPSR

References

King, Gary, James E. Alt, Nancy Elizabeth Burns and Michael Laver (1990). "A Unified Model of Cabinet Dissolution in Parliamentary Democracies," *American Journal of Political Science*, vol. 34, no. 3, pp. 846-870.

Gary King, James E. Alt, Nancy Burns, and Michael Laver. ICPSR Publication Related Archive, 1115.

coef,Zelig-method *Method for extracting estimated coefficients from Zelig objects*

Description

Method for extracting estimated coefficients from Zelig objects

Usage

```
## S4 method for signature 'Zelig'
coef(object)
```

Arguments

object	An Object of Class Zelig

coefficients,Zelig-method *Method for extracting estimated coefficients from Zelig objects*

Description

Method for extracting estimated coefficients from Zelig objects

Usage

```
## S4 method for signature 'Zelig'
coefficients(object)
```

Arguments

object	An Object of Class Zelig

`combine_coef_se` *Combines estimated coefficients and associated statistics from models estimated with multiply imputed data sets or bootstrapped*

Description

Combines estimated coefficients and associated statistics from models estimated with multiply imputed data sets or bootstrapped

Usage

```
combine_coef_se(obj, out_type = "matrix", bagging = FALSE,
  messages = TRUE)
```

Arguments

`obj`	a zelig object with an estimated model
`out_type`	either `"matrix"` or `"list"` specifying whether the results should be returned as a matrix or a list.
`bagging`	logical whether or not to bag the bootstrapped coefficients
`messages`	logical whether or not to return messages for what is being returned

Value

If the model uses multiply imputed or bootstrapped data then a matrix (default) or list of combined coefficients (`coef`), standard errors (`se`), z values (`zvalue`), p-values (`p`) is returned. Rubin's Rules are used to combine output from multiply imputed data. An error is returned if no imputations were included or there wasn't bootstrapping. Please use `get_coef`, `get_se`, and `get_pvalue` methods instead in cases where there are no imputations or bootstrap.

Author(s)

Christopher Gandrud and James Honaker

Source

Partially based on `mi.meld` from Amelia.

Examples

```
set.seed(123)

## Multiple imputation example
# Create fake imputed data
n <- 100
x1 <- runif(n)
x2 <- runif(n)
```

```
y <- rnorm(n)
data.1 <- data.frame(y = y, x = x1)
data.2 <- data.frame(y = y, x = x2)

# Estimate model
mi.out <- to_zelig_mi(data.1, data.2)
z.out.mi <- zelig(y ~ x, model = "ls", data = mi.out)

# Combine and extract coefficients and standard errors
combine_coef_se(z.out.mi)

## Bootstrap example
z.out.boot <- zelig(y ~ x, model = "ls", data = data.1, bootstrap = 20)
combine_coef_se(z.out.boot)
```

`createJSON` *Utility function for constructing JSON file that encodes the hierarchy of available statistical models in Zelig*

Description

Utility function for construction a JSON file that encodes the hierarchy of available statistical models.

Usage

```
createJSON(movefile=TRUE)
```

Arguments

`movefile`	Logical of whether to (TRUE) move the JSON file into path `./inst/JSON` or (FALSE) leave in working directory.

Value

Returns TRUE on successful completion of json file

Author(s)

Christine Choirat, Vito D'Orazio, James Honaker

`df.residual,Zelig-method` *Method for extracting residual degrees-of-freedom from Zelig objects*

Description

Method for extracting residual degrees-of-freedom from Zelig objects

Usage

```
## S4 method for signature 'Zelig'
df.residual(object)
```

Arguments

`object`	An Object of Class Zelig

`eidat` *Simulation Data for Ecological Inference*

Description

This dataframe contains a simulated data set to illustrate the models for ecological inference.

Usage

```
data(eidat)
```

Format

A table containing 4 variables ("t0", "t1", "x0", "x1") and 10 observations.

fitted,Zelig-method *Method for extracting estimated fitted values from Zelig objects*

Description

Method for extracting estimated fitted values from Zelig objects

Usage

```
## S4 method for signature 'Zelig'
fitted(object, ...)
```

Arguments

object	An Object of Class Zelig
...	Additional parameters to be passed to fitted

free1 *Freedom of Speech Data*

Description

Selection of individual-level survey data for freedom of speech.

Usage

```
data(free1)
```

Details

A table with 150 observations and 12 variables.

- sex1 for men and 0 for women
- ageAge of respondent in years
- educLevels of education, coded as a numeric variable with
  - 1No formal education
  - 2Less than primary school education
  - 3Completed primary school
  - 4Completed secondary school
  - 5Completed high school
  - 6Completed college
  - 7Completed post-graduate degree
- countryCharacter strings consisting of "Oceana", "Eurasia", and "Eastasia", after Orwell's *1984*.

- ySelf assessment (see below).
- v1-v6Response to vignettes (see below).

Survey respondents were asked in almost the same language for a self-assessment and for an assessment of several hypothetical persons described by written vignettes. The self assessment (`self`, in the data set), "How free do you think [name/you] [is/are] to express [him-her/your]self without fear of government reprisal?" was first asked of the survey respondent with respect to him or herself, and then after each of vignette. The possible response categories are:

- 1Completely free
- 2Very free
- 3Moderately free
- 4Slightly free
- 5Not free at all

The vignettes, ordered from most free to least free, are:

- vign1[Kay] does not like many of the government's policies. She frequently publishes her opinion in newspapers, criticizing decisions by officials and calling for change. She sees little reason these actions could lead to government reprisal.
- vign2[Michael] disagrees with many of the government's policies. Though he knows criticism is frowned upon, he doesn't believe the government would punish someone for expressing critical views. He makes his opinion known on most issues without regard to who is listening.
- vign3[Bob] has political views at odds with the government. He has heard of people occasionally being arrested for speaking out against the government, and government leaders sometimes make political speeches condemning those who criticize. He sometimes writes letters to newspapers about politics, but he is careful not to use his real name.
- vign4[Connie] does not like the government's stance on many issues. She has a friend who was arrested for being too openly critical of governmental leaders, and so she avoids voicing her opinions in public places.
- vign5[Vito] disagrees with many of the government's policies, and is very careful about whom he says this to, reserving his real opinions for family and close friends only. He knows several men who have been taken away by government officials for saying negative things in public.
- vign6[Sonny] lives in fear of being harassed for his political views. Everyone he knows who has spoken out against the government has been arrested or taken away. He never says a word about anything the government does, not even when he is at home alone with his family.

References

WHO's World Health Survey by Lydia Bendib, Somnath Chatterji, Alena Petrakova, Ritu Sadana, Joshua A. Salomon, Margie Schneider, Bedirhan Ustun, Maria Villanueva

Jonathan Wand, Gary King and Olivia Lau. (2007) "Anchors: Software for Anchoring Vignettes". *Journal of Statistical Software*. Forthcoming. copy at http://wand.stanford.edu/research/anchors-jss.pdf

Gary King and Jonathan Wand. "Comparing Incomparable Survey Responses: New Tools for Anchoring Vignettes," Political Analysis, 15, 1 (Winter, 2007): Pp. 46-66, copy at http://gking.harvard.edu/files/abs/c-abs.shtml.

free2 *Freedom of Speech Data*

Description

Selection of individual-level survey data for freedom of speech.

Usage

```
data(free2)
```

Details

A table with 150 observations and 12 variables.

- sex1 for men and 0 for women
- ageAge of respondent in years
- educLevels of education, coded as a numeric variable with
  - 1No formal education
  - 2Less than primary school education
  - 3Completed primary school
  - 4Completed secondary school
  - 5Completed high school
  - 6Completed college
  - 7Completed post-graduate degree
- countryCharacter strings consisting of "Oceana", "Eurasia", and "Eastasia", after Orwell's *1984*.
- ySelf assessment (see below).
- v1-v6Response to vignettes (see below).

Survey respondents were asked in almost the same language for a self-assessment and for an assessment of several hypothetical persons described by written vignettes. The self assessment (`self`, in the data set), "How free do you think [name/you] [is/are] to express [him-her/your]self without fear of government reprisal?" was first asked of the survey respondent with respect to him or herself, and then after each of vignette. The possible response categories are:

- 1Completely free
- 2Very free
- 3Moderately free
- 4Slightly free
- 5Not free at all

The vignettes, ordered from most free to least free, are:

- vign1[Kay] does not like many of the government's policies. She frequently publishes her opinion in newspapers, criticizing decisions by officials and calling for change. She sees little reason these actions could lead to government reprisal.
- vign2[Michael] disagrees with many of the government's policies. Though he knows criticism is frowned upon, he doesn't believe the government would punish someone for expressing critical views. He makes his opinion known on most issues without regard to who is listening.
- vign3[Bob] has political views at odds with the government. He has heard of people occasionally being arrested for speaking out against the government, and government leaders sometimes make political speeches condemning those who criticize. He sometimes writes letters to newspapers about politics, but he is careful not to use his real name.
- vign4[Connie] does not like the government's stance on many issues. She has a friend who was arrested for being too openly critical of governmental leaders, and so she avoids voicing her opinions in public places.
- vign5[Vito] disagrees with many of the government's policies, and is very careful about whom he says this to, reserving his real opinions for family and close friends only. He knows several men who have been taken away by government officials for saying negative things in public.
- vign6[Sonny] lives in fear of being harassed for his political views. Everyone he knows who has spoken out against the government has been arrested or taken away. He never says a word about anything the government does, not even when he is at home alone with his family.

References

WHO's World Health Survey by Lydia Bendib, Somnath Chatterji, Alena Petrakova, Ritu Sadana, Joshua A. Salomon, Margie Schneider, Bedirhan Ustun, Maria Villanueva

Jonathan Wand, Gary King and Olivia Lau. (2007) "Anchors: Software for Anchoring Vignettes". *Journal of Statistical Software*. Forthcoming. copy at http://wand.stanford.edu/research/anchors-jss.pdf

Gary King and Jonathan Wand. "Comparing Incomparable Survey Responses: New Tools for Anchoring Vignettes," Political Analysis, 15, 1 (Winter, 2007): Pp. 46-66, copy at http://gking.harvard.edu/files/abs/c-abs.shtml.

`friendship` *Simulated Example of Schoolchildren Friendship Network*

Description

This data set contains six sociomatrices of simulated data on friendship ties among schoolchildren.

Usage

```
data(friendship)
```

Format

Each variable in the dataset is a 15 by 15 matrix representing some form of social network tie held by the fictitious children. The matrices are labeled "friends", "advice", "prestige", "authority", "perpower" and "per".

The sociomatrices were combined into the friendship dataset using the format.network.data function from the netglm package by Skyler Cranmer as shown in the example.

Source

fictitious

Examples

```
## Not run:
friendship <- format.network.data(friends, advice, prestige, authority, perpower, per)

## End(Not run)
```

---

from_zelig_model *Extract the original fitted model object from a* zelig *estimation*

Description

Extract the original fitted model object from a zelig estimation

Usage

```
from_zelig_model(obj)
```

Arguments

obj	a zelig object with an estimated model

Details

Extracts the original fitted model object from a zelig estimation. This can be useful for passing output to non-Zelig post-estimation functions and packages such as texreg and stargazer for creating well-formatted presentation document tables.

Author(s)

Christopher Gandrud

Examples

```
z5 <- zls$new()
z5$zelig(Fertility ~ Education, data = swiss)
from_zelig_model(z5)
```

get_pvalue *Extract p-values from a Zelig estimated model*

Description

Extract p-values from a Zelig estimated model

Usage

```
get_pvalue(object)
```

Arguments

object	an object of class Zelig

Author(s)

Christopher Gandrud

get_qi *Extract quantities of interest from a Zelig simulation*

Description

Extract quantities of interest from a Zelig simulation

Usage

```
get_qi(object, qi = "ev", xvalue = "x", subset = NULL)
```

Arguments

object	an object of class Zelig
qi	character string with the name of quantity of interest desired: "ev" for expected values, "pv" for predicted values or "fd" for first differences.
xvalue	chracter string stating which of the set of values of x should be used for getting the quantity of interest.
subset	subset for multiply imputed data (only relevant if multiply imputed data is supplied in the original call.)

Author(s)

Christopher Gandrud

get_se *Extract standard errors from a Zelig estimated model*

Description

Extract standard errors from a Zelig estimated model

Usage

```
get_se(object)
```

Arguments

object	an object of class Zelig

Author(s)

Christopher Gandrud

grunfeld *Simulation Data for model Seemingly Unrelated Regression (sur) that corresponds to method SUR of systemfit*

Description

Dataframe contains 20 annual observations from 1935 to 1954 of 7 variables for two firms General Electric and Westinghouse. Columns are Year; Ige and Iw = Gross investment for GE and W,respectively; Fge and Fw=Market value of Firm as of begin of the year; Cge and Cw= Capital stock measure as of begin of the year.

Usage

```
data(grunfeld)
```

Format

A table containing 7 variables ("Year", "Ige", "Fge", "Cge","Iw", "Fw","Cw") and 20 observations.

hoff *Social Security Expenditure Data*

Description

This data set contains annual social security expenditure (as percent of budget lagged by two years), the relative frequency of mentions social justice received in the party's platform in each year, and whether the president is Republican or Democrat.

Usage

```
data(hoff)
```

Format

A table containing 5 variables ("year", "L2SocSec", "Just503D", "Just503R", "RGovDumy") and 36 observations.

Source

ICPSR (replication dataset s1109)

References

Gary King and Michael Laver. "On Party Platforms, Mandates, and Government Spending," *American Political Science Review*, Vol. 87, No. 3 (September, 1993): pp. 744-750.

homerun *Sample Data on Home Runs Hit By Mark McGwire and Sammy Sosa in 1998.*

Description

Game-by-game information for the 1998 season for Mark McGwire and Sammy Sosa. Data are a subset of the dataset provided in Simonoff (1998).

Usage

```
data(homerun)
```

Format

A data frame containing 5 variables ("gameno", "month", "homeruns", "playerstatus", "player") and 326 observations.

`gameno` an integer variable denoting the game number

`month` a factor variable taking with levels "March" through "September" denoting the month of the game

`homeruns` an integer vector denoting the number of homeruns hit in that game for that player

`playerstatus` an integer vector equal to "0" if the player played in the game, and "1" if they did not.

`player` an integer vector equal to "0" (McGwire) or "1" (Sosa)

Source

https://ww2.amstat.org/publications/jse/v6n3/datasets.simonoff.html

References

Simonoff, Jeffrey S. 1998. "Move Over, Roger Maris: Breaking Baseball's Most Famous Record." *Journal of Statistics Education* 6(3). Data used are a subset of the data in the article.

---

immigration *Individual Preferences Over Immigration Policy*

Description

These five datasets are part of a larger set of 10 multiply imputed data sets describing individual preferences toward immigration policy. Imputation was performed via Amelia.

Format

Each multiply-inputed data set consists of a table with 7 variables ("ipip", "wage1992", "prtyid", "ideol", "gender") and 2,485 observations. For variable descriptions, please refer to Scheve and Slaugher, 2001.

Source

National Election Survey

References

Scheve, Kenneth and Matthew Slaughter (2001). "Labor Market Competition and Individual Preferences Over Immigration Policy," *The Review of Economics and Statistics*, vol. 83, no. 1, pp. 133-145.

is_length_not_1 *Check if an object has a length greater than 1*

Description

Check if an object has a length greater than 1

Usage

```
is_length_not_1(x, msg = "Length is 1.", fail = TRUE)
```

Arguments

x	an object
msg	character string with the error message to return if `fail = TRUE`.
fail	logical whether to return an error if length is not greater than 1.

is_simsrange *Check if simulations for a range of fitted values are present in sim.out*

Description

Check if simulations for a range of fitted values are present in sim.out

Usage

```
is_simsrange(x, fail = TRUE)
```

Arguments

x	a sim.out method
fail	logical whether to return an error if simulation range is not present.

is_simsrange1 *Check if simulations for a range1 of fitted values are present in sim.out*

Description

Check if simulations for a range1 of fitted values are present in sim.out

Usage

```
is_simsrange1(x, fail = TRUE)
```

Arguments

x	a sim.out method
fail	logical whether to return an error if simulation range is not present.

is_simsx *Check if simulations for individual values are present in sim.out*

Description

Check if simulations for individual values are present in sim.out

Usage

```
is_simsx(x, fail = TRUE)
```

Arguments

x	a sim.out method
fail	logical whether to return an error if simulation range is not present.

is_simsx1 *Check if simulations for individual values for x1 are present in sim.out*

Description

Check if simulations for individual values for x1 are present in sim.out

Usage

```
is_simsx1(x, fail = TRUE)
```

Arguments

x	a sim.out method
fail	logical whether to return an error if simulation range is not present.

is_sims_present *Check if any simulations are present in sim.out*

Description

Check if any simulations are present in sim.out

Usage

```
is_sims_present(x, fail = TRUE)
```

Arguments

x	a sim.out method
fail	logical whether to return an error if no simulations are present.

is_timeseries *Check if a zelig object contains a time series model*

Description

Check if a zelig object contains a time series model

Usage

```
is_timeseries(x, msg = "Not a timeseries object.", fail = FALSE)
```

Arguments

x	a zelig object
msg	character string with the error message to return if `fail = TRUE`.
fail	logical whether to return an error if x is not a timeseries.

is_uninitializedField *Check if uninitializedField*

Description

Check if uninitializedField

Usage

```
is_uninitializedField(x, msg = "Zelig model has not been estimated.",
  fail = TRUE)
```

Arguments

x	a zelig.out method
msg	character string with the error message to return if `fail = TRUE`.
fail	logical whether to return an error if x uninitialzed.

is_varying *Check if the values in a vector vary*

Description

Check if the values in a vector vary

Usage

```
is_varying(x, msg = "Vector does not vary.", fail = TRUE)
```

Arguments

x	a vector
msg	character string with the error message to return if `fail = TRUE`.
fail	logical whether to return an error if x does not vary.

is_zelig *Check if is a zelig object*

Description

Check if is a zelig object

Usage

```
is_zelig(x, fail = TRUE)
```

Arguments

x	an object
fail	logical whether to return an error if x is not a Zelig object.

is_zeligei *Check if an object was created with ZeligEI*

Description

Check if an object was created with ZeligEI

Usage

```
is_zeligei(x, msg = "Function is not relevant for ZeligEI objects.",
  fail = TRUE)
```

Arguments

x	a zelig object
msg	character string with the error message to return if `fail = TRUE`.
fail	logical whether to return an error if x is not a timeseries.

klein — *Simulation Data for model Two-Stage Least Square (twosls) that corresponds to method 2SLS of systemfit*

Description

Dataframe contains annual observations of US economy from 1920 to 1940. The columns are, Year, C=Consumption, P=Corporate profits, P1=Previous year corporate profit,Wtot=Total wage, Wp=Private wage bill, Wg=Government wage bill,I=Investment, K1=Previous year capital stock,X=GNP,G=Government spending, T=Taxes, X1=Previous year GNP, Tm=Year-1931.

Usage

```
data(klein)
```

Format

A table containing 14 variables ("year", "C", "P", "P1","Wtot", "Wp", "Wg", "I", "K1","X", "G", "T", "X1", "Tm") and 21 observations.

Source

http://pages.stern.nyu.edu/~wgreene/Text/econometricanalysis.htm

kmenta — *Simulation Data for model Three-Stage Least Square (threesls) that corresponds to method 3SLS of systemfit*

Description

Dataframe contains 20 annual observations of a supply/demand model with 5 variables. Columns are q=Food consumption per capita, p=Ratio of food price to general consumer prices, d=Disposable income in contstant dollars, f=Ratio of preceding year's prices received by farmers to general consumer prices, a=Time index.

Usage

```
data(kmenta)
```

Format

A table containing 5 variables ("q", "p", "d", "f","a") and 20 observations.

macro *Macroeconomic Data*

Description

Selected macroeconomic indicators for Austria, Belgium, Canada, Denmark, Finland, France, Italy, Japan, the Netherlands, Norway, Sweden, the United Kingdom, the United States, and West Germany for the period 1966-1990.

Usage

```
data(macro)
```

Format

A table containing 6 variables ("country", "year", "gdp", "unem", "capmob", and "trade") and 350 observations.

Source

ICPSR

References

King, Gary, Michael Tomz and Jason Wittenberg. ICPSR Publication Related Archive, 1225.

King, Gary, Michael Tomz and Jason Wittenberg (2000). "Making the Most of Statistical Analyses: Improving Interpretation and Presentation," *American Journal of Political Science*, vol. 44, pp. 341-355.

MatchIt.url *Table of links for Zelig*

Description

Table of links for `help.zelig` for the companion MatchIt package.

Median *Compute the Statistical Median of a Vector*

Description

Compute the Statistical Median of a Vector

Usage

```
Median(x, na.rm = NULL)
```

Arguments

x	a vector of numeric or ordered values
na.rm	ignored

Value

the median of the vector

Author(s)

Matt Owen

mexico *Voting Data from the 1988 Mexican Presidental Election*

Description

This dataset contains voting data for the 1988 Mexican presidential election.

Usage

```
data(mexico)
```

Format

A table containing 33 variables and 1,359 observations.

Source

ICPSR

References

King, Gary, Michael Tomz and Jason Wittenberg (2000). "Making the Most of Statistical Analyses: Improving Interpretation and Presentation," *American Journal of Political Science*, vol. 44, pp. 341-355.

King, Tomz and Wittenberg. ICPSR Publication Related Archive, 1255.

`mi` *Enables backwards compatability for preparing non-amelia imputed data sets for* `zelig`.

Description

See `to_zelig_mi`

Usage

```
mi(...)
```

Arguments

`...`	a set of `data.frame`'s

Value

an `mi` object composed of a list of data frames.

`mid` *Militarized Interstate Disputes*

Description

A small sample from the militarized interstate disputes (MID) database.

Usage

```
data(mid)
```

Format

A table containing 6 variables ("conflict", "major", "contig", "power", "maxdem", "mindem", and "years") and 3,126 observations. For full variable descriptions, please see King and Zeng, 2001.

Source

Militarized Interstate Disputes database

References

King, Gary, and Lanche Zeng (2001). "Explaining Rare Events in International Relations," *International Organization*, vol. 55, no. 3, pp. 693-715.

Jones, Daniel M., Stuart A. Bremer and David Singer (1996). "Militarized Interstate Disputes, 1816-1992: Rationale, Coding Rules, and Empirical Patterns," *Conflict Management and Peace Science*, vol. 15, no. 2, pp. 163-213.

`Mode` *Compute the Statistical Mode of a Vector*

Description

Compute the Statistical Mode of a Vector

Usage

```
Mode(x)
```

Arguments

`x`	a vector of numeric, factor, or ordered values

Value

the statistical mode of the vector. If more than one mode exists, the last one in the factor order is arbitrarily chosen (by design)

Author(s)

Christopher Gandrud and Matt Owen

`model_lookup_df` *Instructions for how to convert non-Zelig fitted model objects to Zelig. Used in to_zelig*

Description

Instructions for how to convert non-Zelig fitted model objects to Zelig. Used in to_zelig

Usage

```
model_lookup_df
```

Format

An object of class `data.frame` with 9 rows and 4 columns.

names,Zelig-method *Names method for Zelig objects*

Description

Names method for Zelig objects

Usage

```
## S4 method for signature 'Zelig'
names(x)
```

Arguments

x	An Object of Class Zelig

newpainters *The Discretized Painter's Data of de Piles*

Description

The original painters data contain the subjective assessment, on a 0 to 20 integer scale, of 54 classical painters. The newpainters data discretizes the subjective assessment by quartiles with thresholds 25%, 50%, 75%. The painters were assessed on four characteristics: composition, drawing, colour and expression. The data is due to the Eighteenth century art critic, de Piles.

Usage

```
data(newpainters)
```

Format

A table containing 5 variables ("Composition", "Drawing", "Colour", "Expression", and "School") and 54 observations.

Source

A. J. Weekes (1986).“A Genstat Primer”. Edward Arnold.

M. Davenport and G. Studdert-Kennedy (1972). “The statistical analysis of aesthetic judgement: an exploration.” *Applied Statistics*, vol. 21, pp. 324–333.

I. T. Jolliffe (1986) “Principal Component Analysis.” Springer.

References

Venables, W. N. and Ripley, B. D. (2002) “Modern Applied Statistics with S,” Fourth edition. Springer.

or_summary — *Find odds ratios for coefficients and standard errors for glm.summary class objects*

Description

Find odds ratios for coefficients and standard errors for glm.summary class objects

Usage

```
or_summary(obj, label_mod_coef = "(OR)", label_mod_se = "(OR)")
```

Arguments

obj	a glm.summary class object
label_mod_coef	character string for how to modify the coefficient label.
label_mod_se	character string for how to modify the standard error label.

PErisk — *Political Economic Risk Data from 62 Countries in 1987*

Description

Political Economic Risk Data from 62 Countries in 1987.

Usage

```
data(PErisk)
```

Format

A data frame with 62 observations on the following 6 variables. All data points are from 1987. See Quinn (2004) for more details.

country: a factor with levels 'Argentina' 'Australia' 'Austria' 'Bangladesh' 'Belgium' 'Bolivia' 'Botswana' 'Brazil' 'Burma' 'Cameroon' 'Canada' 'Chile' 'Colombia' 'Congo-Kinshasa' 'Costa Rica' 'Cote d'Ivoire' 'Denmark' 'Dominican Republic' 'Ecuador' 'Finland' 'Gambia, The' 'Ghana' 'Greece' 'Hungary' 'India' 'Indonesia' 'Iran' 'Ireland' 'Israel' 'Italy' 'Japan' 'Kenya' 'Korea, South' 'Malawi' 'Malaysia' 'Mexico' 'Morocco' 'New Zealand' 'Nigeria' 'Norway' 'Papua New Guinea' 'Paraguay' 'Philippines' 'Poland' 'Portugal' 'Sierra Leone' 'Singapore' 'South Africa' 'Spain' 'Sri Lanka' 'Sweden' 'Switzerland' 'Syria' 'Thailand' 'Togo' 'Tunisia' 'Turkey' 'United Kingdom' 'Uruguay' 'Venezuela' 'Zambia' 'Zimbabwe'

courts: an ordered factor with levels '0' < '1'.'courts' is an indicator of whether the country in question is judged to have an independent judiciary. From Henisz (2002).

barb2: a numeric vector giving the natural log of the black market premium in each country. The black market premium is coded as the black market exchange rate (local currency per dollar) divided by the official exchange rate minus 1. From Marshall, Gurr, and Harff (2002).

prsexp2: an ordered factor with levels '0' < '1' < '2' < '3' < '4' < '5', giving the lack of expropriation risk. From Marshall, Gurr, and Harff (2002).

prscorr2: an ordered factor with levels '0' < '1' < '2' < '3' < '4' < '5', measuring the lack of corruption. From Marshall, Gurr, and Harff (2002).

gdpw2: a numeric vector giving the natural log of real GDP per worker in 1985 international prices. From Alvarez et al. (1999).

Source

Mike Alvarez, Jose Antonio Cheibub, Fernando Limongi, and Adam Przeworski. 1999. "ACLP Political and Economic Database." <URL: http://www.ssc.upenn.edu/~cheibub/data/>.

Witold J. Henisz. 2002. "The Political Constraint Index (POLCON) Dataset." \ <URL: http://www-management.wharton.upenn.edu/henisz/POLCON/ContactInfo. html>.

Monty G. Marshall, Ted Robert Gurr, and Barbara Harff. 2002. "State Failure Task Force Problem Set." <URL: http://www.cidcm.umd.edu/inscr/stfail/index.htm>.

References

Kevin M. Quinn. 2004. "Bayesian Factor Analysis for Mixed Ordinal and Continuous Response." *Political Analyis*. Vol. 12, pp.338–353.

`plot,Zelig,ANY-method` *Plot method for Zelig objects*

Description

Plot method for Zelig objects

Usage

```
## S4 method for signature 'Zelig,ANY'
plot(x, y, ...)
```

Arguments

`x`	An Object of Class Zelig
`y`	unused
`...`	Additional parameters to be passed to plot

predict,Zelig-method *Method for getting predicted values from Zelig objects*

Description

Method for getting predicted values from Zelig objects

Usage

```
## S4 method for signature 'Zelig'
predict(object, ...)
```

Arguments

object	An Object of Class Zelig
...	Additional parameters to be passed to predict

qi.plot *Default Plot Design For Zelig Model QI's*

Description

Default Plot Design For Zelig Model QI's

Usage

```
qi.plot(obj, ...)
```

Arguments

obj	A reference class zelig5 object
...	Parameters to be passed to the 'truehist' function which is implicitly called for numeric simulations

Author(s)

James Honaker with panel layouts from Matt Owen

qi_slimmer *Find the median and a central interval of simulated quantity of interest distributions*

Description

Find the median and a central interval of simulated quantity of interest distributions

Usage

```
qi_slimmer(df, qi_type = "ev", ci = 0.95)
```

Arguments

df	a tidy-formatted data frame of simulated quantities of interest created by zelig_qi_to_df.
qi_type	character string either ev or pv for returning the central intervals for the expected value or predicted value, respectively.
ci	numeric. The central interval to return, expressed on the (0, 100] or the equivalent (0, 1] interval.

Details

A tidy-formatted data frame with the following columns:

- The values fitted with setx
- qi_ci_min: the minimum value of the central interval specified with ci
- qi_ci_median: the median of the simulated quantity of interest distribution
- qi_ci_max: the maximum value of the central interval specified with ci

Author(s)

Christopher Gandrud

See Also

zelig_qi_to_df

Examples

```
library(dplyr)
qi.central.interval <- zelig(Petal.Width ~ Petal.Length + Species,
             data = iris, model = "ls") %>%
             setx(Petal.Length = 2:4, Species = "setosa") %>%
             sim() %>%
             zelig_qi_to_df() %>%
             qi_slimmer()
```

residuals,Zelig-method

Method for extracting residuals from Zelig objects

Description

Method for extracting residuals from Zelig objects

Usage

```
## S4 method for signature 'Zelig'
residuals(object)
```

Arguments

object	An Object of Class Zelig

rocplot

Receiver Operator Characteristic Plots

Description

The 'rocplot' command generates a receiver operator characteristic plot to compare the in-sample (default) or out-of-sample fit for two logit or probit regressions.

Usage

```
rocplot(z1, z2,
cutoff = seq(from=0, to=1, length=100), lty1="solid",
lty2="dashed", lwd1=par("lwd"), lwd2=par("lwd"),
col1=par("col"), col2=par("col"),
main="ROC Curve",
xlab = "Proportion of 1's Correctly Predicted",
ylab="Proportion of 0's Correctly Predicted",
plot = TRUE,
...
)
```

Arguments

z1	first model
z2	second model
cutoff	A vector of cut-off values between 0 and 1, at which to evaluate the proportion of 0s and 1s correctly predicted by the first and second model. By default, this is 100 increments between 0 and 1 inclusive

lty1	the line type of the first model (defaults to 'line')
lty2	the line type of the second model (defaults to 'dashed')
lwd1	the line width of the first model (defaults to 1)
lwd2	the line width of the second model (defaults to 1)
col1	the color of the first model (defaults to 'black')
col2	the color of the second model (defaults to 'black')
main	a title for the plot (defaults to "ROC Curve")
xlab	a label for the X-axis
ylab	a lavel for the Y-axis
plot	whether to generate a plot to the selected device
...	additional parameters to be passed to the plot

Value

if plot is TRUE, rocplot simply generates a plot. Otherwise, a list with the following is produced:

roc1	a matrix containing a vector of x-coordinates and y-coordinates corresponding to the number of ones and zeros correctly predicted for the first model.
roc2	a matrix containing a vector of x-coordinates and y-coordinates corresponding to the number of ones and zeros correctly predicted for the second model.
area1	the area under the first ROC curve, calculated using Reimann sums.
area2	the area under the second ROC curve, calculated using Reimann sums.

---

sanction *Multilateral Economic Sanctions*

Description

Data on bilateral sanctions behavior for selected years during the general period 1939-1983. This data contains errors that have since been corrected. Please contact Lisa Martin before using this data for publication.

Usage

```
data(sanction)
```

Format

A table containing 8 variables ("mil", "coop", "target", "import", "export", "cost", "num", and "ncost") and 78 observations. For full variable description, see Martin, 1992.

Source

Martin, 1992

References

Martin, Lisa (1992). *Coercive Cooperation: Explaining Multilateral Economic Sanctions*, Princeton: Princeton University Press.

`seatshare` *Left Party Seat Share in 11 OECD Countries*

Description

This data set contains time-series data of the seat shares in the lower legislative house of left leaning parties over time, as well as the level of unemployment. Data follows the style used in Hibbs (1977).

Usage

```
data(seatshare)
```

Format

A table containing N variables ("country","year","unemp","leftseat") and 384 observations split across 11 countries.

Source

OECD data and Mackie and Rose (1991), extended to further years.

References

Douglas A. Hibbs. (1977). *Political Parties and Macroeconomic Policy*. American Political Science Review 71(4):1467-1487.

Thomas T. Mackie and Richard Rose. (1991). *The International Almanac of Electoral History* Macmillan: London.

`setx` *Setting Explanatory Variable Values*

Description

The `setx` function uses the variables identified in the `formula` generated by `zelig` and sets the values of the explanatory variables to the selected values. Use `setx` after `zelig` and before `sim` to simulate quantities of interest.

Usage

```
setx(obj, fn = NULL, data = NULL, cond = FALSE, ...)
```

Arguments

obj	output object from zelig
fn	a list of functions to apply to the data frame
data	a new data frame used to set the values of explanatory variables. If data = NULL (the default), the data frame called in zelig is used
cond	a logical value indicating whether unconditional (default) or conditional (choose cond = TRUE) prediction should be performed. If you choose cond = TRUE, setx will coerce fn = NULL and ignore the additional arguments in .... If cond = TRUE and data = NULL, setx will prompt you for a data frame.
...	user-defined values of specific variables for overwriting the default values set by the function fn. For example, adding var1 = mean(data$var1) or x1 = 12 explicitly sets the value of x1 to 12. In addition, you may specify one explanatory variable as a range of values, creating one observation for every unique value in the range of values

Details

This documentation describes the setx Zelig 4 compatibility wrapper function.

Value

The output is returned in a field to the Zelig object. For unconditional prediction, x.out is a model matrix based on the specified values for the explanatory variables. For multiple analyses (i.e., when choosing the by option in zelig, setx returns the selected values calculated over the entire data frame. If you wish to calculate values over just one subset of the data frame, the 5th subset for example, you may use: x.out <- setx(z.out[[5]])

Author(s)

Matt Owen, Olivia Lau and Kosuke Imai

See Also

The full Zelig manual may be accessed online at http://docs.zeligproject.org/articles/

Examples

```
# Unconditional prediction:
data(turnout)
z.out <- zelig(vote ~ race + educate, model = 'logit', data = turnout)
x.out <- setx(z.out)
s.out <- sim(z.out, x = x.out)
```

setx1 *Setting Explanatory Variable Values for First Differences*

Description

This documentation describes the setx1 Zelig 4 compatibility wrapper function. The wrapper is primarily useful for setting fitted values for creating first differences in piped workflows.

Usage

```
setx1(obj, fn = NULL, data = NULL, cond = FALSE, ...)
```

Arguments

obj	output object from zelig
fn	a list of functions to apply to the data frame
data	a new data frame used to set the values of explanatory variables. If data = NULL (the default), the data frame called in zelig is used
cond	a logical value indicating whether unconditional (default) or conditional (choose cond = TRUE) prediction should be performed. If you choose cond = TRUE, setx1 will coerce fn = NULL and ignore the additional arguments in .... If cond = TRUE and data = NULL, setx1 will prompt you for a data frame.
...	user-defined values of specific variables for overwriting the default values set by the function fn. For example, adding var1 = mean(data$var1) or x1 = 12 explicitly sets the value of x1 to 12. In addition, you may specify one explanatory variable as a range of values, creating one observation for every unique value in the range of values

Value

The output is returned in a field to the Zelig object. For unconditional prediction, x.out is a model matrix based on the specified values for the explanatory variables. For multiple analyses (i.e., when choosing the by option in zelig, setx1 returns the selected values calculated over the entire data frame. If you wish to calculate values over just one subset of the data frame, the 5th subset for example, you may use: x.out <- setx(z.out[[5]])

Author(s)

Christopher Gandrud, Matt Owen, Olivia Lau, Kosuke Imai

See Also

The full Zelig manual may be accessed online at http://docs.zeligproject.org/articles/

Examples

```
library(dplyr) # contains pipe operator %>%
data(turnout)

# plot first differences
zelig(Fertility ~ Education, data = swiss, model = 'ls') %>%
      setx(z4, Education = 10) %>%
      setx1(z4, Education = 30) %>%
      sim() %>%
      plot()
```

---

sim *Generic Method for Computing and Organizing Simulated Quantities of Interest*

Description

Simulate quantities of interest from the estimated model output from `zelig()` given specified values of explanatory variables established in `setx()`. For classical *maximum likelihood* models, `sim()` uses asymptotic normal approximation to the log-likelihood. For *Bayesian models*, Zelig simulates quantities of interest from the posterior density, whenever possible. For *robust Bayesian models*, simulations are drawn from the identified class of Bayesian posteriors. Alternatively, you may generate quantities of interest using bootstrapped parameters.

Usage

```
sim(obj, x, x1, y = NULL, num = 1000, bootstrap = F, bootfn = NULL,
  cond.data = NULL, ...)
```

Arguments

`obj`	output object from `zelig`
`x`	values of explanatory variables used for simulation, generated by `setx`. Not if ommitted, then `sim` will look for values in the reference class object
`x1`	optional values of explanatory variables (generated by a second call of `setx`) particular computations of quantities of interest
`y`	a parameter reserved for the computation of particular quantities of interest (average treatment effects). Few models currently support this parameter
`num`	an integer specifying the number of simulations to compute
`bootstrap`	currently unsupported
`bootfn`	currently unsupported
`cond.data`	currently unsupported
`...`	arguments reserved future versions of Zelig

Details

This documentation describes the `sim` Zelig 4 compatibility wrapper function.

Value

The output stored in `s.out` varies by model. Use the `names` function to view the output stored in `s.out`. Common elements include:

`x` the `setx` values for the explanatory variables, used to calculate the quantities of interest (expected values, predicted values, etc.).

`x1` the optional `setx` object used to simulate first differences, and other model-specific quantities of interest, such as risk-ratios.

`call` the options selected for `sim`, used to replicate quantities of interest.

`zelig.call` the original function and options for `zelig`, used to replicate analyses.

`num` the number of simulations requested.

`par` the parameters (coefficients, and additional model-specific parameters). You may wish to use the same set of simulated parameters to calculate quantities of interest rather than simulating another set.

`qi\$ev` simulations of the expected values given the model and `x`.

`qi\$pr` simulations of the predicted values given by the fitted values.

`qi\$fd` simulations of the first differences (or risk difference for binary models) for the given `x` and `x1`. The difference is calculated by subtracting the expected values given `x` from the expected values given `x1`. (If do not specify `x1`, you will not get first differences or risk ratios.)

`qi\$rr` simulations of the risk ratios for binary and multinomial models. See specific models for details.

`qi\$ate.ev` simulations of the average expected treatment effect for the treatment group, using conditional prediction. Let t_i be a binary explanatory variable defining the treatment ($t_i = 1$) and control ($t_i = 0$) groups. Then the average expected treatment effect for the treatment group is

$$\frac{1}{n}\sum_{i=1}^{n}[\,Y_i(t_i = 1) - E[Y_i(t_i = 0)] \mid t_i = 1\,],$$

where $Y_i(t_i = 1)$ is the value of the dependent variable for observation i in the treatment group. Variation in the simulations are due to uncertainty in simulating $E[Y_i(t_i = 0)]$, the counterfactual expected value of Y_i for observations in the treatment group, under the assumption that everything stays the same except that the treatment indicator is switched to $t_i = 0$.

`qi\$ate.pr` simulations of the average predicted treatment effect for the treatment group, using conditional prediction. Let t_i be a binary explanatory variable defining the treatment ($t_i = 1$) and control ($t_i = 0$) groups. Then the average predicted treatment effect for the treatment group is

$$\frac{1}{n}\sum_{i=1}^{n}[\,Y_i(t_i = 1) - Y_i(\widehat{t_i = 0}) \mid t_i = 1\,],$$

where $Y_i(t_i = 1)$ is the value of the dependent variable for observation i in the treatment group. Variation in the simulations are due to uncertainty in simulating $Y_i(\widehat{t_i = 0})$, the counterfactual predicted value of Y_i for observations in the treatment group, under the assumption that everything stays the same except that the treatment indicator is switched to $t_i = 0$.

Author(s)

Christopher Gandrud, Matt Owen, Olivia Lau and Kosuke Imai

`simulations.plot` *Plot Quantities of Interest in a Zelig-fashion*

Description

Various graph generation for different common types of simulated results from Zelig

Usage

```
simulations.plot(y, y1=NULL, xlab="", ylab="", main="", col=NULL, line.col=NULL,
axisnames=TRUE)
```

Arguments

`y`	A matrix or vector of simulated results generated by Zelig, to be graphed.
`y1`	For comparison of two sets of simulated results at different choices of covariates, this should be an object of the same type and dimension as y. If no comparison is to be made, this should be NULL.
`xlab`	Label for the x-axis.
`ylab`	Label for the y-axis.
`main`	Main plot title.
`col`	A vector of colors. Colors will be used in turn as the graph is built for main plot objects. For nominal/categorical data, this colors renders as the bar color, while for numeric data it renders as the background color.
`line.col`	A vector of colors. Colors will be used in turn as the graph is built for line color shading of plot objects.
`axisnames`	a character-vector, specifying the names of the axes

Value

nothing

Author(s)

James Honaker

sna.ex *Simulated Example of Social Network Data*

Description

This data set contains five sociomatrices of simulated data social network data.

Usage

```
data(sna.ex)
```

Format

Each variable in the dataset is a 25 by 25 matrix of simulated social network data. The matrices are labeled "Var1", "Var2", "Var3", "Var4", and "Var5".

Source

fictitious

summary,Zelig-method *Summary method for Zelig objects*

Description

Summary method for Zelig objects

Usage

```
## S4 method for signature 'Zelig'
summary(object, ...)
```

Arguments

object	An Object of Class Zelig
...	Additional parameters to be passed to summary

summary.Arima *Summary of an object of class Arima*

Description

Summary of an object of class Arima

Usage

```
## S3 method for class 'Arima'
summary(object, ...)
```

Arguments

object	An object of class Arima
...	Additional parameters

Value

The original object

SupremeCourt *U.S. Supreme Court Vote Matrix*

Description

This dataframe contains a matrix votes cast by U.S. Supreme Court justices in all cases in the 2000 term.

Usage

```
data(SupremeCourt)
```

Format

The dataframe has contains data for justices Rehnquist, Stevens, O'Connor, Scalia, Kennedy, Souter, Thomas, Ginsburg, and Breyer for the 2000 term of the U.S. Supreme Court. It contains data from 43 non-unanimous cases. The votes are coded liberal (1) and conservative (0) using the protocol of Spaeth (2003). The unit of analysis is the case citation (ANALU=0). We are concerned with formally decided cases issued with written opinions, after full oral argument and cases decided by an equally divided vote (DECTYPE=1,5,6,7).

Source

Harold J. Spaeth (2005). "Original United States Supreme Court Database: 1953-2004 Terms." <URL:http://www.as.uky.edu/polisci/ulmerproject/sctdata.htm>.

swiss — *Swiss Fertility and Socioeconomic Indicators (1888) Data*

Description

Standardized fertility measure and socio-economic indicators for each of 47 French-speaking provinces of Switzerland at about 1888.

Usage

```
data(swiss)
```

Format

A data frame with 47 observations on 6 variables, each of which is in percent, i.e., in [0,100].

[,1] Fertility Ig, "common standardized fertility measure" [,2] Agriculture [,3] Examination nation [,4] Education [,5] Catholic [,6] Infant.Mortality live births who live less than 1 year.

All variables but 'Fert' give proportions of the population.

Source

Project "16P5", pages 549-551 in

Mosteller, F. and Tukey, J. W. (1977) "Data Analysis and Regression: A Second Course in Statistics". Addison-Wesley, Reading Mass.

indicating their source as "Data used by permission of Franice van de Walle. Office of Population Research, Princeton University, 1976. Unpublished data assembled under NICHD contract number No 1-HD-O-2077."

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) "The New S Language". Wadsworth & Brooks/Cole.

table.levels — *Create a table, but ensure that the correct columns exist. In particular, this allows for entires with zero as a value, which is not the default for standard tables*

Description

Create a table, but ensure that the correct columns exist. In particular, this allows for entires with zero as a value, which is not the default for standard tables

Usage

```
table.levels(x, levels, ...)
```

Arguments

x	a vector
levels	a vector of levels
...	parameters for table

Value

a table

Author(s)

Matt Owen

`tobin` *Tobin's Tobit Data*

Description

Economists fit a parametric censored data model called the 'tobit'. These data are from Tobin's original paper.

Usage

```
data(tobin)
```

Format

A data frame with 20 observations on the following 3 variables.

durable: Durable goods purchase

age: Age in years

quant: Liquidity ratio (x 1000)

Source

J. Tobin, Estimation of relationships for limited dependent variables, Econometrica, v26, 24-36, 1958.

to_zelig *Coerce a non-Zelig fitted model object to a Zelig class object*

Description

Coerce a non-Zelig fitted model object to a Zelig class object

Usage

```
to_zelig(obj)
```

Arguments

obj	a fitted model object fitted using lm and many using glm. Note: more intended in future Zelig releases.

Author(s)

Christopher Gandrud and Ista Zhan

Examples

```
library(dplyr)
lm.out <- lm(Fertility ~ Education, data = swiss)

z.out <- to_zelig(lm.out)

# to_zelig called from within setx
setx(z.out) %>% sim() %>% plot()
```

to_zelig_mi *Bundle Multiply Imputed Data Sets into an Object for Zelig*

Description

This object prepares multiply imputed data sets so they can be used by zelig.

Usage

```
to_zelig_mi(...)
```

Arguments

...	a set of data.frame's or a single list of data.frame's

Value

an `mi` object composed of a list of data frames.

Note

This function creates a list of `data.frame` objects, which resembles the storage of imputed data sets in the `amelia` object.

Author(s)

Matt Owen, James Honaker, and Christopher Gandrud

Examples

```
# create datasets
n <- 100
x1 <- runif(n)
x2 <- runif(n)
y <- rnorm(n)
data.1 <- data.frame(y = y, x = x1)
data.2 <- data.frame(y = y, x = x2)

# merge datasets into one object as if imputed datasets

mi.out <- to_zelig_mi(data.1, data.2)

# pass object in place of data argument
z.out <- zelig(y ~ x, model = "ls", data = mi.out)
```

---

turnout *Turnout Data Set from the National Election Survey*

Description

This data set contains individual-level turnout data. It pools several American National Election Surveys conducted during the 1992 presidential election year. Only the first 2,000 observations (from a total of 15,837 observations) are included in the sample data.

Usage

```
data(turnout)
```

Format

A table containing 5 variables ("race", "age", "educate", "income", and "vote") and 2,000 observations.

Source

National Election Survey

References

King, Gary, Michael Tomz, Jason Wittenberg (2000). "Making the Most of Statistical Analyses: Improving Interpretation and Presentation," *American Journal of Political Science*, vol. 44, pp.341–355.

---

vcov,Zelig-method *Variance-covariance method for Zelig objects*

Description

Variance-covariance method for Zelig objects

Usage

```
## S4 method for signature 'Zelig'
vcov(object)
```

Arguments

object	An Object of Class Zelig

---

vcov_gee *Find vcov for GEE models*

Description

Find vcov for GEE models

Usage

```
vcov_gee(obj)
```

Arguments

obj	a geeglm class object.

vcov_rq *Find vcov for quantile regression models*

Description

Find vcov for quantile regression models

Usage

```
vcov_rq(obj)
```

Arguments

obj	a rq class object.

voteincome *Sample Turnout and Demographic Data from the 2000 Current Population Survey*

Description

This data set contains turnout and demographic data from a sample of respondents to the 2000 Current Population Survey (CPS). The states represented are South Carolina and Arkansas. The data represent only a sample and results from this example should not be used in publication.

Usage

```
data(voteincome)
```

Format

A data frame containing 7 variables ("state", "year", "vote", "income", "education", "age", "female") and 1500 observations.

`state` a factor variable with levels equal to "AR" (Arkansas) and "SC" (South Carolina)

`year` an integer vector

`vote` an integer vector taking on values "1" (Voted) and "0" (Did Not Vote)

`income` an integer vector ranging from "4" (Less than \$5000) to "17" (Greater than \$75000) denoting family income. See the CPS codebook for more information on variable coding

`education` an integer vector ranging from "1" (Less than High School Education) to "4" (More than a College Education). See the CPS codebook for more information on variable coding

`age` an integer vector ranging from "18" to "85"

`female` an integer vector taking on values "1" (Female) and "0" (Male)

Source

Census Bureau Current Population Survey

References

http://www.census.gov/cps

`Weimar` *1932 Weimar election data*

Description

This data set contains election results for 10 kreise (equivalent to precincts) from the 1932 Weimar (German) election.

Usage

```
data(Weimar)
```

Format

A table containing 11 variables and 10 observations. The variables are

Nazi Number of votes for the Nazi party

Government Number of votes for the Government

Communists Number of votes for the Communist party

FarRight Number of votes for far right parties

Other Number of votes for other parties, and non-voters

shareunemployed Proportion unemployed

shareblue Proportion working class

sharewhite Proportion white-collar workers

sharedomestic Proportion domestic servants

shareprotestants Proportion Protestant

Source

ICPSR

`zelig` *Estimating a Statistical Model*

Description

The zelig function estimates a variety of statistical models. Use `zelig` output with `setx` and `sim` to compute quantities of interest, such as predicted probabilities, expected values, and first differences, along with the associated measures of uncertainty (standard errors and confidence intervals).

Usage

```
zelig(formula, model, data, ..., by = NULL, cite = TRUE)
```

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~\, x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. For example, to run the same model on all fifty states, you could use: `z.out <- zelig(y ~ x1 + x2, data = mydata, model = 'ls',by = 's` You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

This documentation describes the `zelig` Zelig 4 compatibility wrapper function.

Additional parameters avaialable to many models include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.

- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

Author(s)

Matt Owen, Kosuke Imai, Olivia Lau, and Gary King

See Also

http://docs.zeligproject.org/articles/

Zelig-ar-class *Time-Series Model with Autoregressive Disturbance*

Description

Warning: summary does not work with timeseries models after simulation.

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.

by — a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. For example, to run the same model on all fifty states, you could use: `z.out <- zelig(y ~ x1 + x2, data = mydata, model = 'ls',by = 's` You may also use by to run models using MatchIt subclasses.

cite — If is set to 'TRUE' (default), the model citation will be printed to the console.

ts — The name of the variable containing the time indicator. This should be passed in as a string. If this variable is not provided, Zelig will assume that the data is already ordered by time.

cs — Name of a variable that denotes the cross-sectional element of the data, for example, country name in a dataset with time-series across different countries. As a variable name, this should be in quotes. If this is not provided, Zelig will assume that all observations come from the same unit over time, and should be pooled, but if provided, individual models will be run in each cross-section. If cs is given as an argument, ts must also be provided. Additionally, by must be NULL.

order — A vector of length 3 passed in as `c(p,d,q)` where p represents the order of the autoregressive model, d represents the number of differences taken in the model, and q represents the order of the moving average model.

Details

Currently only the Reference class syntax for time series. This model does not accept Bootstraps or weights.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_ar.html

Examples

```
data(seatshare)
subset <- seatshare[seatshare$country == "UNITED KINGDOM",]
ts.out <- zelig(formula = unemp ~ leftseat, model = "ar", ts = "year", data = subset)
summary(ts.out)
```

`Zelig-arima-class` *Autoregressive and Moving-Average Models with Integration for Time-Series Data*

Description

Warning: `summary` does not work with timeseries models after simulation.

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. For example, to run the same model on all fifty states, you could use: `z.out <- zelig(y ~ x1 + x2, data = mydata, model = 'ls',by = 's` You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.
`ts`	The name of the variable containing the time indicator. This should be passed in as a string. If this variable is not provided, Zelig will assume that the data is already ordered by time.
`cs`	Name of a variable that denotes the cross-sectional element of the data, for example, country name in a dataset with time-series across different countries. As a variable name, this should be in quotes. If this is not provided, Zelig will assume that all observations come from the same unit over time, and should be pooled, but if provided, individual models will be run in each cross-section. If `cs` is given as an argument, `ts` must also be provided. Additionally, `by` must be `NULL`.
`order`	A vector of length 3 passed in as `c(p,d,q)` where p represents the order of the autoregressive model, d represents the number of differences taken in the model, and q represents the order of the moving average model.

Details

Currently only the Reference class syntax for time series. This model does not accept Bootstraps or weights.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_arima.html

Examples

```
data(seatshare)
subset <- seatshare[seatshare$country == "UNITED KINGDOM",]
ts.out <- zarima$new()
ts.out$zelig(unemp ~ leftseat, order = c(1, 0, 1), data = subset)

# Set fitted values and simulate quantities of interest
ts.out$setx(leftseat = 0.75)
ts.out$setx1(leftseat = 0.25)
ts.out$sim()
```

`Zelig-bayes-class` *Bayes Model object for inheritance across models in Zelig*

Description

Bayes Model object for inheritance across models in Zelig

Methods

`get_coef(nonlist = FALSE)` Get estimated model coefficients

`zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE)` The zelig function estimates a variety of statistical models

---

`Zelig-binchoice-class` *Binary Choice object for inheritance across models in Zelig*

Description

Binary Choice object for inheritance across models in Zelig

`Zelig-binchoice-gee-class` *Object for Binary Choice outcomes in Generalized Estimating Equations for inheritance across models in Zelig*

Description

Object for Binary Choice outcomes in Generalized Estimating Equations for inheritance across models in Zelig

`Zelig-binchoice-survey-class` *Object for Binary Choice outcomes with Survey Weights for inheritance across models in Zelig*

Description

Object for Binary Choice outcomes with Survey Weights for inheritance across models in Zelig

`Zelig-class` *Zelig reference class*

Description

Zelig website: http://zeligproject.org/

Fields

`fn` R function to call to wrap

`formula` Zelig formula

`weights` [forthcoming]

`name` name of the Zelig model

`data` data frame or matrix

`by` split the data by factors

`mi` work with imputed dataset

`idx` model index

`zelig.call` Zelig function call

`model.call` wrapped function call

`zelig.out` estimated zelig model(s)

`setx.out` set values

`setx.labels` pretty-print qi

`bsetx` is x set?

`bsetx1` is x1 set?

`bsetrange` is range set?

`bsetrange1` is range1 set?

`range` range

`range1` range1

`test.statistics` list of test statistics

`sim.out` simulated qi's

`simparam` simulated parameters

`num` number of simulations

`authors` Zelig model authors

`zeligauthors` Zelig authors

`modelauthors` wrapped model authors

`packageauthors` wrapped package authors

`refs` citation information

`year` model is released

`description` model description

`url` model URL

`url.docs` model documentation URL

`category` model category

`vignette.url` vignette URL

`json` JSON export

`ljson` JSON export

`outcome` JSON export

`wrapper` JSON export

`explanatory` JSON export

`mcunit.test` unit testing

`with.feedback` Feedback

`robust.se` return robust standard errors

Methods

`ATT(treatment, treated = 1, quietly = TRUE, num = NULL)` Generic Method for Computing Simulated (Sample) Average Treatment Effects on the Treated

`cite()` Provide citation information about Zelig and Zelig model, and about wrapped package and wrapped model

`feedback()` Send feedback to the Zelig team

`from_zelig_model()` Extract the original fitted model object from a zelig call. Note only works for models using directly wrapped functions.

`get_coef(nonlist = FALSE)` Get estimated model coefficients

`get_df_residual()` Get residual degrees-of-freedom

`get_fitted(...)` Get estimated fitted values

`get_model_data()` Get data used to estimate the model

`get_names()` Return Zelig object field names

`get_predict(...)` Get predicted values

`get_pvalue()` Get estimated model p-values

`get_qi(qi = "ev", xvalue = "x", subset = NULL)` Get quantities of interest

`get_residuals(...)` Get estimated model residuals

`get_se()` Get estimated model standard errors

`get_vcov()` Get estimated model variance-covariance matrix

`graph(...)` Plot the quantities of interest

`help()` Open the model vignette from http://zeligproject.org/

`packagename()` Automatically retrieve wrapped package name

`references(style = "sphinx")` Construct a reference list specific to a Zelig model.

`set(..., fn = list(numeric = mean, ordered = Median))` Setting Explanatory Variable Values

`sim(num = NULL)` Generic Method for Computing and Organizing Simulated Quantities of Interest

`simATT(simparam, data, depvar, treatment, treated)` Simulate an Average Treatment on the Treated

`summarise(...)` Display a Zelig object

`summarize(...)` Display a Zelig object

`toJSON()` Convert Zelig object to JSON format

`zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE)` The zelig function estimates a variety of statistical models

---

`Zelig-exp-class` *Exponential Regression for Duration Dependent Variables*

Description

Exponential Regression for Duration Dependent Variables

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. For example, to run the same model on all fifty states, you could use: z.out <- zelig(y ~ x1 + x2, data = mydata, model = 'ls',by = 's You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
robust	defaults to FALSE. If TRUE, zelig() computes robust standard errors based on sandwich estimators and the options selected in cluster.
if	robust = TRUE, you may select a variable to define groups of correlated observations. Let x3 be a variable that consists of either discrete numeric values, character strings, or factors that define strata. Then z.out <- zelig(y ~ x1 + x2, robust = TRUE, cluster = "x3",model = "exp", data = mydata) means that the observations can be correlated within the strata defined by the variable x3, and that robust standard errors should be calculated according to those clusters. If robust = TRUE but cluster is not specified, zelig() assumes that each observation falls into its own cluster.

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

Methods

`zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE)` The zelig function estimates a variety of statistical models

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_exp.html

Examples

```
library(Zelig)
data(coalition)
library(survival)
z.out <- zelig(Surv(duration, ciep12) ~ fract + numst2, model = "exp",
               data = coalition)
summary(z.out)
```

`Zelig-factor-bayes-class` *Bayesian Factor Analysis*

Description

Bayesian Factor Analysis

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `~ Y1 + Y2 + Y3`, where Y1, Y2, and Y3 are variables of interest in factor analysis (manifest variables), assumed to be normally distributed. The model requires a minimum of three manifest variables contained in the same dataset. The `+` symbol means "inclusion" not "addition."
`factors`	number of the factors to be fitted (defaults to 2).
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).

...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

In addition, zelig() accepts the following additional arguments for model specification:

- lambda.constraints: list containing the equality or inequality constraints on the factor loadings. Choose from one of the following forms:
- varname = list(): by default, no constraints are imposed.
- varname = list(d, c): constrains the dth loading for the variable named varname to be equal to c.
- varname = list(d, +): constrains the dth loading for the variable named varname to be positive;
- varname = list(d, -): constrains the dth loading for the variable named varname to be negative.
- std.var: defaults to FALSE (manifest variables are rescaled to zero mean, but retain observed variance). If TRUE, the manifest variables are rescaled to be mean zero and unit variance.

In addition, zelig() accepts the following additional inputs for bayes.factor:

- burnin: number of the initial MCMC iterations to be discarded (defaults to 1,000).
- mcmc: number of the MCMC iterations after burnin (defaults to 20,000).
- thin: thinning interval for the Markov chain. Only every thin-th draw from the Markov chain is kept. The value of mcmc must be divisible by this value. The default value is 1.
- verbose: defaults to FALSE. If TRUE, the progress of the sampler (every 10
- seed: seed for the random number generator. The default is NA which corresponds to a random seed 12345.
- Lambda.start: starting values of the factor loading matrix Λ, either a scalar (all unconstrained loadings are set to that value), or a matrix with compatible dimensions. The default is NA, where the start value are set to be 0 for unconstrained factor loadings, and 0.5 or - 0.5 for constrained factor loadings (depending on the nature of the constraints).
- Psi.start: starting values for the uniquenesses, either a scalar (the starting values for all diagonal elements of Ψ are set to be this value), or a vector with length equal to the number of manifest variables. In the latter case, the starting values of the diagonal elements of Ψ take the values of Psi.start. The default value is NA where the starting values of the all the uniquenesses are set to be 0.5.
- store.lambda: defaults to TRUE, which stores the posterior draws of the factor loadings.
- store.scores: defaults to FALSE. If TRUE, stores the posterior draws of the factor scores. (Storing factor scores may take large amount of memory for a large number of draws or observations.)

The model also accepts the following additional arguments to specify prior parameters:

- `l0`: mean of the Normal prior for the factor loadings, either a scalar or a matrix with the same dimensions as Λ. If a scalar value, that value will be the prior mean for all the factor loadings. Defaults to 0.
- `L0`: precision parameter of the Normal prior for the factor loadings, either a scalar or a matrix with the same dimensions as Λ. If `L0` takes a scalar value, then the precision matrix will be a diagonal matrix with the diagonal elements set to that value. The default value is 0, which leads to an improper prior.
- `a0`: the shape parameter of the Inverse Gamma prior for the uniquenesses is `a0`/2. It can take a scalar value or a vector. The default value is 0.001.
- `b0`: the scale parameter of the Inverse Gamma prior for the uniquenesses is `b0`/2. It can take a scalar value or a vector. The default value is 0.001.

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

Methods

`zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE)` The zelig function estimates a variety of statistical models

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_factorbayes.html

Examples

```
## Not run:
data(swiss)
names(swiss) <- c("Fert", "Agr", "Exam", "Educ", "Cath", "InfMort")
z.out <- zelig(~ Agr + Exam + Educ + Cath + InfMort,
model = "factor.bayes", data = swiss,
factors = 2, verbose = FALSE,
a0 = 1, b0 = 0.15, burnin = 500, mcmc = 5000)

z.out$geweke.diag()
z.out <- zelig(~ Agr + Exam + Educ + Cath + InfMort,
model = "factor.bayes", data = swiss, factors = 2,
```

```
lambda.constraints =
   list(Exam = list(1,"+"),
        Exam = list(2,"-"),
        Educ = c(2, 0),
        InfMort = c(1, 0)),
verbose = FALSE, a0 = 1, b0 = 0.15,
burnin = 500, mcmc = 5000)

z.out$geweke.diag()
z.out$heidel.diag()
z.out$raftery.diag()
summary(z.out)

## End(Not run)
```

`Zelig-gamma-class` *Gamma Regression for Continuous, Positive Dependent Variables*

Description

Gamma Regression for Continuous, Positive Dependent Variables

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `bootstrap`: logical or numeric. If `FALSE` don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_gamma.html

Examples

```
library(Zelig)
data(coalition)
z.out <- zelig(duration ~ fract + numst2, model = "gamma", data = coalition)
summary(z.out)
```

`Zelig-gamma-gee-class` *Generalized Estimating Equation for Gamma Regression*

Description

Generalized Estimating Equation for Gamma Regression

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.

model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
corstr:character	string specifying the correlation structure: "independence", "exchangeable", "ar1", "unstructured" and "userdefined"
See	geeglm in package geepack for other function arguments.
id:	where id is a variable which identifies the clusters. The data should be sorted by id and should be ordered within each cluster when appropriate
corstr:	character string specifying the correlation structure: "independence", "exchangeable", "ar1", "unstructured" and "userdefined"
geeglm:	See geeglm in package geepack for other function arguments

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_gammagee.html

Examples

```
library(Zelig)
data(coalition)
coalition$cluster <- c(rep(c(1:62), 5),rep(c(63), 4))
sorted.coalition <- coalition[order(coalition$cluster),]
z.out <- zelig(duration ~ fract + numst2, model = "gamma.gee",id = "cluster",
               data = sorted.coalition,corstr = "exchangeable")
summary(z.out)
```

Zelig-gamma-survey-class

Gamma Regression with Survey Weights

Description

Gamma Regression with Survey Weights

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.

- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_gammasurvey.html

Examples

```
library(Zelig)
data(api, package="survey")
z.out1 <- zelig(api00 ~ meals + yr.rnd, model = "gamma.survey",
weights = ~pw, data = apistrat)
summary(z.out1)
```

Zelig-gee-class *Generalized Estimating Equations Model object for inheritance across models in Zelig*

Description

Generalized Estimating Equations Model object for inheritance across models in Zelig

Methods

zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE) The zelig function estimates a variety of statistical models

Zelig-glm-class *Generalized Linear Model object for inheritance across models in Zelig*

Description

Generalized Linear Model object for inheritance across models in Zelig

Methods

`zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE)` The zelig function estimates a variety of statistical models

Zelig-ivreg-class *Instrumental-Variable Regression*

Description

Instrumental-Variable Regression

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means `inclusion'' notaddition.” You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.
`formula`	specification(s) of the regression relationship

instruments	the instruments. Either instruments is missing and formula has three parts as in y ~ x1 + x2 | z1 + z2 + z3 (recommended) or formula is y ~ x1 + x2 and instruments is a one-sided formula ~ z1 + z2 + z3. Using instruments is not recommended with zelig.
model, x, y	logicals. If TRUE the corresponding components of the fit (the model frame, the model matrices , the response) are returned.
...	further arguments passed to methods. See also zelig.

Details

Additional parameters avaialable to many models include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Regressors and instruments for ivreg are most easily specified in a formula with two parts on the right-hand side, e.g., y ~ x1 + x2 | z1 + z2 + z3, where x1 and x2 are the regressors and z1, z2, and z3 are the instruments. Note that exogenous regressors have to be included as instruments for themselves. For example, if there is one exogenous regressor ex and one endogenous regressor en with instrument in, the appropriate formula would be y ~ ex + en | ex + in. Equivalently, this can be specified as y ~ ex + en | . - en + in, i.e., by providing an update formula with a . in the second part of the formula. The latter is typically more convenient, if there is a large number of exogenous regressors.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

Methods

zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE) The zelig function estimates a variety of statistical models

Source

ivreg is from Christian Kleiber and Achim Zeileis (2008). Applied Econometrics with R. New York: Springer-Verlag. ISBN 978-0-387-77316-2. URL https://CRAN.R-project.org/package=AER

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_ivreg.html Fit instrumental-variable regression by two-stage least squares. This is equivalent to direct instrumental-variables estimation when the number of instruments is equal to the number of predictors.

zelig, Greene, W. H. (1993) *Econometric Analysis*, 2nd ed., Macmillan.

Examples

```
library(Zelig)
library(dplyr) # for the pipe operator %>%
# load and transform data
data("CigarettesSW")
CigarettesSW$rprice <- with(CigarettesSW, price/cpi)
CigarettesSW$rincome <- with(CigarettesSW, income/population/cpi)
CigarettesSW$tdiff <- with(CigarettesSW, (taxs - tax)/cpi)
# log second stage independent variables, as logging internally for ivreg is
# not currently supported
CigarettesSW$log_rprice <- log(CigarettesSW$rprice)
CigarettesSW$log_rincome <- log(CigarettesSW$rincome)
z.out1 <- zelig(log(packs) ~ log_rprice + log_rincome |
log_rincome + tdiff + I(tax/cpi),data = CigarettesSW, subset = year == "1995",model = "ivreg")
summary(z.out1)
library(Zelig)
library(AER) # for sandwich vcov
library(dplyr) # for the pipe operator %>%

# load and transform data
data("CigarettesSW")
CigarettesSW$rprice <- with(CigarettesSW, price/cpi)
CigarettesSW$rincome <- with(CigarettesSW, income/population/cpi)
CigarettesSW$tdiff <- with(CigarettesSW, (taxs - tax)/cpi)

# log second stage independent variables, as logging internally for ivreg is
# not currently supported
CigarettesSW$log_rprice <- log(CigarettesSW$rprice)
CigarettesSW$log_rincome <- log(CigarettesSW$rincome)

# estimate model
z.out1 <- zelig(log(packs) ~ log_rprice + log_rincome |
                    log_rincome + tdiff + I(tax/cpi),
                    data = CigarettesSW,
                    model = "ivreg")
summary(z.out1)
```

`Zelig-logit-bayes-class`

Bayesian Logit Regression

Description

Bayesian Logit Regression

Arguments

formula	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
...	additional arguments passed to `zelig`, relevant for the model to be estimated.
by	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `burnin`: number of the initial MCMC iterations to be discarded (defaults to 1,000).
- `mcmc`: number of the MCMC iterations after burnin (defaults to 10,000).
- `thin`: thinning interval for the Markov chain. Only every thin-th draw from the Markov chain is kept. The value of mcmc must be divisible by this value. The default value is 1.
- `verbose`: defaults to FALSE. If TRUE, the progress of the sampler (every 10%) is printed to the screen.
- `seed`: seed for the random number generator. The default is `NA` which corresponds to a random seed of 12345.
- `beta.start`: starting values for the Markov chain, either a scalar or vector with length equal to the number of estimated coefficients. The default is `NA`, such that the maximum likelihood estimates are used as the starting values.

Use the following parameters to specify the model's priors:

- `b0`: prior mean for the coefficients, either a numeric vector or a scalar. If a scalar value, that value will be the prior mean for all the coefficients. The default is 0.
- `B0`: prior precision parameter for the coefficients, either a square matrix (with the dimensions equal to the number of the coefficients) or a scalar. If a scalar value, that value times an identity matrix will be the prior precision parameter. The default is 0, which leads to an improper prior.

Use the following arguments to specify optional output for the model:

- `bayes.resid`: defaults to FALSE. If TRUE, the latent Bayesian residuals for all observations are returned. Alternatively, users can specify a vector of observations for which the latent residuals should be returned.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_logitbayes.html

Examples

```
data(turnout)
z.out <- zelig(vote ~ race + educate, model = "logit.bayes",data = turnout, verbose = FALSE)
```

`Zelig-logit-class` *Logistic Regression for Dichotomous Dependent Variables*

Description

Logistic Regression for Dichotomous Dependent Variables

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.

model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
below	(defaults to 0) The point at which the dependent variable is censored from below. If any values in the dependent variable are observed to be less than the censoring point, it is assumed that that particular observation is censored from below at the observed value. (See for a Bayesian implementation that supports both left and right censoring.)
robust	defaults to FALSE. If TRUE, zelig() computes robust standard errors based on sandwich estimators (see and) and the options selected in cluster.
if	robust = TRUE, you may select a variable to define groups of correlated observations. Let x3 be a variable that consists of either discrete numeric values, character strings, or factors that define strata. Then z.out <- zelig(y ~ x1 + x2, robust = TRUE, cluster = "x3", model = "tobit", data = mydata) means that the observations can be correlated within the strata defined by the variable x3, and that robust standard errors should be calculated according to those clusters. If robust = TRUE but cluster is not specified, zelig() assumes that each observation falls into its own cluster.

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

Methods

`show(signif.stars = FALSE, subset = NULL, bagging = FALSE)` Display a Zelig object

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_logit.html

Examples

```
library(Zelig)
data(turnout)
z.out1 <- zelig(vote ~ age + race, model = "logit", data = turnout,
                cite = FALSE)
summary(z.out1)
summary(z.out1, odds_ratios = TRUE)
x.out1 <- setx(z.out1, age = 36, race = "white")
s.out1 <- sim(z.out1, x = x.out1)
summary(s.out1)
plot(s.out1)
```

`Zelig-logit-gee-class` *Generalized Estimating Equation for Logit Regression*

Description

Generalized Estimating Equation for Logit Regression

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.

by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
id:	where id is a variable which identifies the clusters. The data should be sorted by id and should be ordered within each cluster when appropriate
corstr:	character string specifying the correlation structure: "independence", "exchangeable", "ar1", "unstructured" and "userdefined"
geeglm:	See geeglm in package geepack for other function arguments

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_logitgee.html

Examples

```
data(turnout)
turnout$cluster <- rep(c(1:200), 10)
sorted.turnout <- turnout[order(turnout$cluster),]

z.out1 <- zelig(vote ~ race + educate, model = "logit.gee",
id = "cluster", data = sorted.turnout)

summary(z.out1)
x.out1 <- setx(z.out1)
s.out1 <- sim(z.out1, x = x.out1)
summary(s.out1)
```

```
plot(s.out1)
```

`Zelig-logit-survey-class`
Logit Regression with Survey Weights

Description

Logit Regression with Survey Weights

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: `http://docs.zeligproject.org/articles/`.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.
`below`	(defaults to 0) The point at which the dependent variable is censored from below. If any values in the dependent variable are observed to be less than the censoring point, it is assumed that that particular observation is censored from below at the observed value. (See for a Bayesian implementation that supports both left and right censoring.)
`robust`	defaults to FALSE. If TRUE, zelig() computes robust standard errors based on sandwich estimators (see and) and the options selected in cluster.
`if`	robust = TRUE, you may select a variable to define groups of correlated observations. Let x3 be a variable that consists of either discrete numeric values, character strings, or factors that define strata. Then z.out <- zelig(y ~ x1 + x2, robust = TRUE, cluster = "x3", model = "tobit", data = mydata) means that the observations can be correlated within the strata defined by the variable x3, and that robust standard errors should be calculated according to those clusters. If robust = TRUE but cluster is not specified, zelig() assumes that each observation falls into its own cluster.

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_logitsurvey.html

Examples

```
data(api, package = "survey")
apistrat$yr.rnd.numeric <- as.numeric(apistrat$yr.rnd == "Yes")
z.out1 <- zelig(yr.rnd.numeric ~ meals + mobility, model = "logit.survey",
               weights = apistrat$pw, data = apistrat)

summary(z.out1)
x.low <- setx(z.out1, meals= quantile(apistrat$meals, 0.2))
x.high <- setx(z.out1, meals= quantile(apistrat$meals, 0.8))
s.out1 <- sim(z.out1, x = x.low, x1 = x.high)
summary(s.out1)
plot(s.out1)
```

Zelig-lognorm-class *Log-Normal Regression for Duration Dependent Variables*

Description

Log-Normal Regression for Duration Dependent Variables

Arguments

formula	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
...	additional arguments passed to `zelig`, relevant for the model to be estimated.
by	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
robust	defaults to FALSE. If TRUE, zelig() computes robust standard errors based on sandwich estimators (see and) based on the options in cluster.
cluster	if robust = TRUE, you may select a variable to define groups of correlated observations. Let x3 be a variable that consists of either discrete numeric values, character strings, or factors that define strata. Then means that the observations can be correlated within the strata defined by the variable x3, and that robust standard errors should be calculated according to those clusters. If robust = TRUE but cluster is not specified, zelig() assumes that each observation falls into its own cluster.

Details

Additional parameters avaialable to many models include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If `FALSE` don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or

individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

Methods

`zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE)` The zelig function estimates a variety of statistical models

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_lognorm.html

Examples

```
library(Zelig)
data(coalition)
z.out <- zelig(Surv(duration, ciep12) ~ fract + numst2, model ="lognorm",  data = coalition)
summary(z.out)
```

Zelig-ls-class *Least Squares Regression for Continuous Dependent Variables*

Description

Least Squares Regression for Continuous Dependent Variables

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.

cite If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

Methods

zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE) The zelig function estimates a variety of statistical models

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_ls.html

Examples

```
library(Zelig)
data(macro)
z.out1 <- zelig(unem ~ gdp + capmob + trade, model = "ls", data = macro,
cite = FALSE)
summary(z.out1)
```

Zelig-ma-class *Time-Series Model with Moving Average*

Description

Warning: summary does not work with timeseries models after simulation.

Arguments

formula	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
...	additional arguments passed to `zelig`, relevant for the model to be estimated.
by	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
ts	The name of the variable containing the time indicator. This should be passed in as a string. If this variable is not provided, Zelig will assume that the data is already ordered by time.
cs	Name of a variable that denotes the cross-sectional element of the data, for example, country name in a dataset with time-series across different countries. As a variable name, this should be in quotes. If this is not provided, Zelig will assume that all observations come from the same unit over time, and should be pooled, but if provided, individual models will be run in each cross-section. If `cs` is given as an argument, `ts` must also be provided. Additionally, `by` must be `NULL`.
order	A vector of length 3 passed in as `c(p,d,q)` where p represents the order of the autoregressive model, d represents the number of differences taken in the model, and q represents the order of the moving average model.

Details

Currently only the Reference class syntax for time series. This model does not accept Bootstraps or weights.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_ma.html

Examples

```
data(seatshare)
subset <- seatshare[seatshare$country == "UNITED KINGDOM",]
ts.out <- zelig(formula = unemp ~ leftseat, model = "ma", ts = "year", data = subset)
summary(ts.out)
```

Zelig-mlogit-bayes-class

Bayesian Multinomial Logistic Regression

Description

Bayesian Multinomial Logistic Regression

Arguments

formula	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
...	additional arguments passed to `zelig`, relevant for the model to be estimated.
by	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

zelig() accepts the following arguments for mlogit.bayes:

- `baseline`: either a character string or numeric value (equal to one of the observed values in the dependent variable) specifying a baseline category. The default value is NA which sets the baseline to the first alphabetical or numerical unique value of the dependent variable.

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `burnin`: number of the initial MCMC iterations to be discarded (defaults to 1,000).
- `mcmc`: number of the MCMC iterations after burnin (defaults to 10,000).
- `mcmc.method`: either "MH" or "slice", specifying whether to use Metropolis Algorithm or slice sampler. The default value is MH.
- `thin`: thinning interval for the Markov chain. Only every thin-th draw from the Markov chain is kept. The value of mcmc must be divisible by this value. The default value is 1.
- `tune`: tuning parameter for the Metropolis-Hasting step, either a scalar or a numeric vector (for kk coefficients, enter a kk vector). The tuning parameter should be set such that the acceptance rate is satisfactory (between 0.2 and 0.5). The default value is 1.1.
- `verbose`: defaults to FALSE. If TRUE, the progress of the sampler (every 10%) is printed to the screen.
- `seed`: seed for the random number generator. The default is `NA` which corresponds to a random seed of 12345.
- `beta.start`: starting values for the Markov chain, either a scalar or vector with length equal to the number of estimated coefficients. The default is `NA`, such that the maximum likelihood estimates are used as the starting values.

Use the following parameters to specify the model's priors:

- `b0`: prior mean for the coefficients, either a numeric vector or a scalar. If a scalar value, that value will be the prior mean for all the coefficients. The default is 0.
- `B0`: prior precision parameter for the coefficients, either a square matrix (with the dimensions equal to the number of the coefficients) or a scalar. If a scalar value, that value times an identity matrix will be the prior precision parameter. The default is 0, which leads to an improper prior.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_mlogitbayes.html

Examples

```
data(mexico)
z.out <- zelig(vote88 ~ pristr + othcok + othsocok,model = "mlogit.bayes",
data = mexico,verbose = FALSE)
```

`Zelig-negbin-class` *Negative Binomial Regression for Event Count Dependent Variables*

Description

Negative Binomial Regression for Event Count Dependent Variables

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `bootstrap`: logical or numeric. If `FALSE` don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

Methods

zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE) The zelig function estimates a variety of statistical models

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_negbin.html

Examples

```
library(Zelig)
data(sanction)
z.out <- zelig(num ~ target + coop, model = "negbin", data = sanction)
summary(z.out)
```

Zelig-normal-bayes-class

Bayesian Normal Linear Regression

Description

Bayesian Normal Linear Regression

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).

`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to many models include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: `http://docs.zeligproject.org/articles/weights.html`.
- `burnin`: number of the initial MCMC iterations to be discarded (defaults to 1,000).
- `mcmc`: number of the MCMC iterations after burnin (defaults to 10,000).
- `thin`: thinning interval for the Markov chain. Only every thin-th draw from the Markov chain is kept. The value of mcmc must be divisible by this value. The default value is 1.
- `verbose`: defaults to FALSE. If TRUE, the progress of the sampler (every 10%) is printed to the screen.
- `seed`: seed for the random number generator. The default is `NA` which corresponds to a random seed of 12345.
- `beta.start`: starting values for the Markov chain, either a scalar or vector with length equal to the number of estimated coefficients. The default is `NA`, such that the maximum likelihood estimates are used as the starting values.

Use the following parameters to specify the model's priors:

- `b0`: prior mean for the coefficients, either a numeric vector or a scalar. If a scalar value, that value will be the prior mean for all the coefficients. The default is 0.
- `B0`: prior precision parameter for the coefficients, either a square matrix (with the dimensions equal to the number of the coefficients) or a scalar. If a scalar value, that value times an identity matrix will be the prior precision parameter. The default is 0, which leads to an improper prior.
- `c0`: c0/2 is the shape parameter for the Inverse Gamma prior on the variance of the disturbance terms.
- `d0`: d0/2 is the scale parameter for the Inverse Gamma prior on the variance of the disturbance terms.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See `http://docs.zeligproject.org/articles/getters.html` for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_normalbayes.html

Examples

```
data(macro)
z.out <- zelig(unem ~ gdp + capmob + trade, model = "normal.bayes", data = macro, verbose = FALSE)


data(macro)
z.out <- zelig(unem ~ gdp + capmob + trade, model = "normal.bayes",
data = macro, verbose = FALSE)

z.out$geweke.diag()
z.out$heidel.diag()
z.out$raftery.diag()
summary(z.out)

x.out <- setx(z.out)
s.out1 <- sim(z.out, x = x.out)
summary(s.out1)

x.high <- setx(z.out, trade = quantile(macro$trade, prob = 0.8))
x.low <- setx(z.out, trade = quantile(macro$trade, prob = 0.2))

s.out2 <- sim(z.out, x = x.high, x1 = x.low)
summary(s.out2)
```

Zelig-normal-class *Normal Regression for Continuous Dependent Variables*

Description

Normal Regression for Continuous Dependent Variables

Arguments

formula	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.

data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
below	(defaults to 0) The point at which the dependent variable is censored from below. If any values in the dependent variable are observed to be less than the censoring point, it is assumed that that particular observation is censored from below at the observed value. (See for a Bayesian implementation that supports both left and right censoring.)
robust	defaults to FALSE. If TRUE, zelig() computes robust standard errors based on sandwich estimators (see and) and the options selected in cluster.
if	robust = TRUE, you may select a variable to define groups of correlated observations. Let x3 be a variable that consists of either discrete numeric values, character strings, or factors that define strata. Then z.out <- zelig(y ~ x1 + x2, robust = TRUE, cluster = "x3", model = "tobit", data = mydata) means that the observations can be correlated within the strata defined by the variable x3, and that robust standard errors should be calculated according to those clusters. If robust = TRUE but cluster is not specified, zelig() assumes that each observation falls into its own cluster.
formula	a model fitting formula

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_normal.html

Examples

```
data(macro)
z.out1 <- zelig(unem ~ gdp + capmob + trade, model = "normal",
data = macro)
summary(z.out1)
x.high <- setx(z.out1, trade = quantile(macro$trade, 0.8))
x.low <- setx(z.out1, trade = quantile(macro$trade, 0.2))
s.out1 <- sim(z.out1, x = x.high, x1 = x.low)
summary(s.out1)
plot(s.out1)
```

Zelig-normal-gee-class

Generalized Estimating Equation for Normal Regression

Description

Generalized Estimating Equation for Normal Regression

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

`robust`	defaults to TRUE. If TRUE, consistent standard errors are estimated using a "sandwich" estimator.
`corstr`	defaults to "independence". It can take on the following arguments:
`Independence`	(corstr = independence): cor(yit,yit')=0, for all t,t' with t not equal to t'. It assumes that there is no correlation within the clusters and the model becomes equivalent to standard normal regression. The "working" correlation matrix is the identity matrix.
`Fixed`	corstr = fixed): If selected, the user must define the "working" correlation matrix with the R argument rather than estimating it from the model.
`id:`	where id is a variable which identifies the clusters. The data should be sorted by id and should be ordered within each cluster when appropriate
`corstr:`	character string specifying the correlation structure: "independence", "exchangeable", "ar1", "unstructured" and "userdefined"
`geeglm:`	See geeglm in package geepack for other function arguments
`Mv:`	defaults to 1. It specifies the number of periods of correlation and only needs to be specified when `corstr` is stat_M_dep, non_stat_M_dep, or AR-M.
`R:`	defaults to NULL. It specifies a user-defined correlation matrix rather than estimating it from the data. The argument is used only when corstr is "fixed". The input is a TxT matrix of correlations, where T is the size of the largest cluster.

Details

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_normalgee.html

Examples

```
library(Zelig)
data(macro)
z.out <- zelig(unem ~ gdp + capmob + trade, model ="normal.gee", id = "country",
        data = macro, corstr = "AR-M")
summary(z.out)
```

`Zelig-normal-survey-class` *Normal Regression for Continuous Dependent Variables with Survey Weights*

Description

Normal Regression for Continuous Dependent Variables with Survey Weights

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~\, x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `bootstrap`: logical or numeric. If `FALSE` don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_normalsurvey.html

Examples

```
library(Zelig)
data(api, package = "survey")
z.out1 <- zelig(api00 ~ meals + yr.rnd, model = "normal.survey",eights = ~pw, data = apistrat)
summary(z.out1)
```

`Zelig-oprobit-bayes-class`

Bayesian Ordered Probit Regression

Description

Bayesian Ordered Probit Regression

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to many models include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `burnin`: number of the initial MCMC iterations to be discarded (defaults to 1,000).
- `mcmc`: number of the MCMC iterations after burnin (defaults to 10,000).
- `thin`: thinning interval for the Markov chain. Only every thin-th draw from the Markov chain is kept. The value of mcmc must be divisible by this value. The default value is 1.
- `verbose`: defaults to FALSE. If TRUE, the progress of the sampler (every 10%) is printed to the screen.
- `seed`: seed for the random number generator. The default is `NA` which corresponds to a random seed of 12345.
- `beta.start`: starting values for the Markov chain, either a scalar or vector with length equal to the number of estimated coefficients. The default is `NA`, such that the maximum likelihood estimates are used as the starting values.

Use the following parameters to specify the model's priors:

- `b0`: prior mean for the coefficients, either a numeric vector or a scalar. If a scalar value, that value will be the prior mean for all the coefficients. The default is 0.
- `B0`: prior precision parameter for the coefficients, either a square matrix (with the dimensions equal to the number of the coefficients) or a scalar. If a scalar value, that value times an identity matrix will be the prior precision parameter. The default is 0, which leads to an improper prior.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

Vignette: http://docs.zeligproject.org/articles/zelig_oprobitbayes.html

---

Zelig-poisson-bayes-class

Bayesian Poisson Regression

Description

Bayesian Poisson Regression

Arguments

formula	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
...	additional arguments passed to `zelig`, relevant for the model to be estimated.
by	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `burnin`: number of the initial MCMC iterations to be discarded (defaults to 1,000).
- `mcmc`: number of the MCMC iterations after burnin (defaults to 10,000).
- `tune`: Metropolis tuning parameter, either a positive scalar or a vector of length kk, where kk is the number of coefficients. The tuning parameter should be set such that the acceptance rate of the Metropolis algorithm is satisfactory (typically between 0.20 and 0.5). The default value is 1.1.
- `thin`: thinning interval for the Markov chain. Only every thin-th draw from the Markov chain is kept. The value of mcmc must be divisible by this value. The default value is 1.
- `verbose`: defaults to FALSE. If TRUE, the progress of the sampler (every 10%) is printed to the screen.
- `seed`: seed for the random number generator. The default is `NA` which corresponds to a random seed of 12345.
- `beta.start`: starting values for the Markov chain, either a scalar or vector with length equal to the number of estimated coefficients. The default is `NA`, such that the maximum likelihood estimates are used as the starting values.

Use the following parameters to specify the model's priors:

- b0: prior mean for the coefficients, either a numeric vector or a scalar. If a scalar value, that value will be the prior mean for all the coefficients. The default is 0.
- B0: prior precision parameter for the coefficients, either a square matrix (with the dimensions equal to the number of the coefficients) or a scalar. If a scalar value, that value times an identity matrix will be the prior precision parameter. The default is 0, which leads to an improper prior.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_poissonbayes.html

Examples

```
data(sanction)
z.out <- zelig(num ~ target + coop, model = "poisson.bayes",data = sanction, verbose = FALSE)
```

Zelig-poisson-class *Poisson Regression for Event Count Dependent Variables*

Description

Poisson Regression for Event Count Dependent Variables

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.

by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
id:	where id is a variable which identifies the clusters. The data should be sorted by id and should be ordered within each cluster when appropriate
corstr:	character string specifying the correlation structure: "independence", "exchangeable", "ar1", "unstructured" and "userdefined"
geeglm:	See geeglm in package geepack for other function arguments

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_poisson.html

Examples

```
library(Zelig)
data(sanction)
z.out <- zelig(num ~ target + coop, model = "poisson", data = sanction)
summary(z.out)
```

Zelig-poisson-gee-class

Generalized Estimating Equation for Poisson Regression

Description

Generalized Estimating Equation for Poisson Regression

Arguments

formula	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
...	additional arguments passed to `zelig`, relevant for the model to be estimated.
by	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
id:	where id is a variable which identifies the clusters. The data should be sorted by id and should be ordered within each cluster when appropriate
corstr:	character string specifying the correlation structure: "independence", "exchangeable", "ar1", "unstructured" and "userdefined"
geeglm:	See geeglm in package geepack for other function arguments

Details

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `bootstrap`: logical or numeric. If `FALSE` don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_poissongee.html

Examples

```
library(Zelig)
data(sanction)
sanction$cluster <- c(rep(c(1:15), 5), rep(c(16), 3))
sorted.sanction <- sanction[order(sanction$cluster),]
z.out <- zelig(num ~ target + coop, model = "poisson.gee",id = "cluster", data = sorted.sanction)
summary(z.out)
```

`Zelig-poisson-survey-class`

Poisson Regression with Survey Weights

Description

Poisson Regression with Survey Weights

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.

by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_poissonsurvey.html

Examples

```
library(Zelig)
data(api, package="survey")
z.out1 <- zelig(enroll ~ api99 + yr.rnd , model = "poisson.survey", data = apistrat)
summary(z.out1)
```

Zelig-probit-bayes-class

Bayesian Probit Regression

Description

Bayesian Probit Regression

Arguments

formula	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
model	the name of a statistical model to estimate. For a list of supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
...	additional arguments passed to `zelig`, relevant for the model to be estimated.
by	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. For example, to run the same model on all fifty states, you could use: `z.out <- zelig(y ~ x1 + x2, data = mydata, model = 'ls',by = 's` You may also use `by` to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `burnin`: number of the initial MCMC iterations to be discarded (defaults to 1,000).
- `mcmc`: number of the MCMC iterations after burnin (defaults to 10,000).
- `thin`: thinning interval for the Markov chain. Only every thin-th draw from the Markov chain is kept. The value of mcmc must be divisible by this value. The default value is 1.
- `verbose`: defaults to FALSE. If TRUE, the progress of the sampler (every 10%) is printed to the screen.
- `seed`: seed for the random number generator. The default is `NA` which corresponds to a random seed of 12345.
- `beta.start`: starting values for the Markov chain, either a scalar or vector with length equal to the number of estimated coefficients. The default is `NA`, such that the maximum likelihood estimates are used as the starting values.

Use the following parameters to specify the model's priors:

- `b0`: prior mean for the coefficients, either a numeric vector or a scalar. If a scalar value, that value will be the prior mean for all the coefficients. The default is 0.

- B0: prior precision parameter for the coefficients, either a square matrix (with the dimensions equal to the number of the coefficients) or a scalar. If a scalar value, that value times an identity matrix will be the prior precision parameter. The default is 0, which leads to an improper prior.

Use the following arguments to specify optional output for the model:

- bayes.resid: defaults to FALSE. If TRUE, the latent Bayesian residuals for all observations are returned. Alternatively, users can specify a vector of observations for which the latent residuals should be returned.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_probitbayes.html

Examples

```
data(turnout)
z.out <- zelig(vote ~ race + educate, model = "probit.bayes",data = turnout, verbose = FALSE)
summary(z.out)
```

Zelig-probit-class *Probit Regression for Dichotomous Dependent Variables*

Description

Probit Regression for Dichotomous Dependent Variables

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.

`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

Additional parameters avaialable to this model include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: `http://docs.zeligproject.org/articles/weights.html`.
- `bootstrap`: logical or numeric. If `FALSE` don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: `http://docs.zeligproject.org/articles/bootstraps.html`.

See Also

Vignette: `http://docs.zeligproject.org/articles/zelig_probit.html`

Examples

```
data(turnout)
z.out <- zelig(vote ~ race + educate, model = "probit", data = turnout)
summary(z.out)
x.out <- setx(z.out)
s.out <- sim(z.out, x = x.out)
summary(s.out)
plot(s.out)
```

Zelig-probit-gee-class

Generalized Estimating Equation for Probit Regression

Description

Generalized Estimating Equation for Probit Regression

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
corstr:character	string specifying the correlation structure: "independence", "exchangeable", "ar1", "unstructured" and "userdefined"
See	geeglm in package geepack for other function arguments.
id:	where id is a variable which identifies the clusters. The data should be sorted by id and should be ordered within each cluster when appropriate
corstr:	character string specifying the correlation structure: "independence", "exchangeable", "ar1", "unstructured" and "userdefined"
geeglm:	See geeglm in package geepack for other function arguments

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_probitgee.html

Examples

```
data(turnout)
turnout$cluster <- rep(c(1:200), 10)
sorted.turnout <- turnout[order(turnout$cluster),]
z.out1 <- zelig(vote ~ race + educate, model = "probit.gee",
id = "cluster", data = sorted.turnout)
summary(z.out1)
```

Zelig-probit-survey-class

Probit Regression with Survey Weights

Description

@param formula a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.

Arguments

`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.

cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
below:	point at which the dependent variable is censored from below. If the dependent variable is only censored from above, set below = -Inf. The default value is 0.
above:	point at which the dependent variable is censored from above. If the dependent variable is only censored from below, set above = Inf. The default value is Inf.

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- burnin: number of the initial MCMC iterations to be discarded (defaults to 1,000).
- mcmc: number of the MCMC iterations after burnin (defaults to 10,000).
- thin: thinning interval for the Markov chain. Only every thin-th draw from the Markov chain is kept. The value of mcmc must be divisible by this value. The default value is 1.
- verbose: defaults to FALSE. If TRUE, the progress of the sampler (every 10%) is printed to the screen.
- seed: seed for the random number generator. The default is NA which corresponds to a random seed of 12345.
- beta.start: starting values for the Markov chain, either a scalar or vector with length equal to the number of estimated coefficients. The default is NA, such that the maximum likelihood estimates are used as the starting values.

Use the following parameters to specify the model's priors:

- b0: prior mean for the coefficients, either a numeric vector or a scalar. If a scalar value, that value will be the prior mean for all the coefficients. The default is 0.
- B0: prior precision parameter for the coefficients, either a square matrix (with the dimensions equal to the number of the coefficients) or a scalar. If a scalar value, that value times an identity matrix will be the prior precision parameter. The default is 0, which leads to an improper prior.
- c0: c0/2 is the shape parameter for the Inverse Gamma prior on the variance of the disturbance terms.
- d0: d0/2 is the scale parameter for the Inverse Gamma prior on the variance of the disturbance terms.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_probitsurvey.html

Examples

```
data(api, package="survey")
z.out1 <- zelig(enroll ~ api99 + yr.rnd ,
model = "poisson.survey", data = apistrat)
summary(z.out1)
x.low <- setx(z.out1, api99= quantile(apistrat$api99, 0.2))
x.high <- setx(z.out1, api99= quantile(apistrat$api99, 0.8))
s.out1 <- sim(z.out1, x=x.low, x1=x.high)
summary(s.out1)
plot(s.out1)
```

Zelig-quantile-class *Quantile Regression for Continuous Dependent Variables*

Description

Quantile Regression for Continuous Dependent Variables

Arguments

formula	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
...	additional arguments passed to `zelig`, relevant for the model to be estimated.
by	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

In addition to the standard inputs, zelig takes the following additional options for quantile regression:

- tau: defaults to 0.5. Specifies the conditional quantile(s) that will be estimated. 0.5 corresponds to estimating the conditional median, 0.25 and 0.75 correspond to the conditional quartiles, etc. tau vectors with length greater than 1 are not currently supported. If tau is set outside of the interval [0,1], zelig returns the solution for all possible conditional quantiles given the data, but does not support inference on this fit (setx and sim will fail).
- se: a string value that defaults to "nid". Specifies the method by which the covariance matrix of coefficients is estimated during the sim stage of analysis. se can take the following values, which are passed to the summary.rq function from the quantreg package. These descriptions are copied from the summary.rq documentation.
  - "iid" which presumes that the errors are iid and computes an estimate of the asymptotic covariance matrix as in KB(1978).
  - "nid" which presumes local (in tau) linearity (in x) of the the conditional quantile functions and computes a Huber sandwich estimate using a local estimate of the sparsity.
  - "ker" which uses a kernel estimate of the sandwich as proposed by Powell(1990).
- ...: additional options passed to rq when fitting the model. See documentation for rq in the quantreg package for more information.

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

Methods

zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE) The zelig function estimates a variety of statistical models

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_quantile.html

Examples

```
library(Zelig)
data(stackloss)
z.out1 <- zelig(stack.loss ~ Air.Flow + Water.Temp + Acid.Conc.,
model = "rq", data = stackloss,tau = 0.5)
summary(z.out1)
```

`Zelig-relogit-class` *Rare Events Logistic Regression for Dichotomous Dependent Variables*

Description

Rare Events Logistic Regression for Dichotomous Dependent Variables

Arguments

`formula`	a symbolic representation of the model to be estimated, in the form `y ~ x1 + x2`, where `y` is the dependent variable and `x1` and `x2` are the explanatory variables, and `y`, `x1`, and `x2` are contained in the same dataset. (You may include more than two explanatory variables, of course.) The `+` symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form `x1*x2` without computing them in prior steps; `I(x1*x2)` to include only the interaction term and exclude the main effects; and quadratic terms in the form `I(x1^2)`.
`model`	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
`data`	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by `Amelia` or `to_zelig_mi`).
`...`	additional arguments passed to `zelig`, relevant for the model to be estimated.
`by`	a factor variable contained in `data`. If supplied, `zelig` will subset the data frame based on the levels in the `by` variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use `by` to run models using MatchIt subclasses.
`cite`	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

The relogit procedure supports four optional arguments in addition to the standard arguments for zelig(). You may additionally use:

- `tau`: a vector containing either one or two values for `tau`, the true population fraction of ones. Use, for example, tau = c(0.05, 0.1) to specify that the lower bound on tau is 0.05 and the upper bound is 0.1. If left unspecified, only finite-sample bias correction is performed, not case-control correction.

- `case.control`: if tau is specified, choose a method to correct for case-control sampling design: "prior" (default) or "weighting".
- `bias.correct`: a logical value of `TRUE` (default) or `FALSE` indicating whether the intercept should be corrected for finite sample (rare events) bias.

Additional parameters avaialable to many models include:

- `weights`: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- `bootstrap`: logical or numeric. If `FALSE` don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, `zelig` will return an object with elements including `coefficients`, `residuals`, and `formula` which may be summarized using `summary(z.out)` or individually extracted using, for example, `coef(z.out)`. See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using `from_zelig_model`.

Methods

`modcall_formula_transformer()` Transform model call formula.

`show(signif.stars = FALSE, subset = NULL, bagging = FALSE)` Display a Zelig object

`zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE)` The zelig function estimates a variety of statistical models

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_relogit.html

Examples

```
library(Zelig)
data(mid)
z.out1 <- zelig(conflict ~ major + contig + power + maxdem + mindem + years,
              data = mid, model = "relogit", tau = 1042/303772)
summary(z.out1)
```

Zelig-survey-class *Survey models in Zelig for weights for complex sampling designs*

Description

Survey models in Zelig for weights for complex sampling designs

Methods

`zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE)` The zelig function estimates a variety of statistical models

Zelig-timeseries-class *Time-series models in Zelig*

Description

Time-series models in Zelig

Methods

`packagename()` Automatically retrieve wrapped package name

`sim(num = NULL)` Generic Method for Computing and Organizing Simulated Quantities of Interest

`zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE)` The zelig function estimates a variety of statistical models

Zelig-tobit-bayes-class *Bayesian Tobit Regression*

Description

Bayesian Tobit Regression

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
below:	point at which the dependent variable is censored from below. If the dependent variable is only censored from above, set below = -Inf. The default value is 0.
above:	point at which the dependent variable is censored from above. If the dependent variable is only censored from below, set above = Inf. The default value is Inf.
below:	point at which the dependent variable is censored from below. If the dependent variable is only censored from above, set below = -Inf. The default value is 0.
above:	point at which the dependent variable is censored from above. If the dependent variable is only censored from below, set above = Inf. The default value is Inf.

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- burnin: number of the initial MCMC iterations to be discarded (defaults to 1,000).
- mcmc: number of the MCMC iterations after burnin (defaults to 10,000).
- thin: thinning interval for the Markov chain. Only every thin-th draw from the Markov chain is kept. The value of mcmc must be divisible by this value. The default value is 1.
- verbose: defaults to FALSE. If TRUE, the progress of the sampler (every 10%) is printed to the screen.
- seed: seed for the random number generator. The default is NA which corresponds to a random seed of 12345.

- beta.start: starting values for the Markov chain, either a scalar or vector with length equal to the number of estimated coefficients. The default is NA, such that the maximum likelihood estimates are used as the starting values.

Use the following parameters to specify the model's priors:

- b0: prior mean for the coefficients, either a numeric vector or a scalar. If a scalar value, that value will be the prior mean for all the coefficients. The default is 0.
- B0: prior precision parameter for the coefficients, either a square matrix (with the dimensions equal to the number of the coefficients) or a scalar. If a scalar value, that value times an identity matrix will be the prior precision parameter. The default is 0, which leads to an improper prior.
- c0: c0/2 is the shape parameter for the Inverse Gamma prior on the variance of the disturbance terms.
- d0: d0/2 is the scale parameter for the Inverse Gamma prior on the variance of the disturbance terms.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_tobitbayes.html

Examples

```
data(turnout)
z.out <- zelig(vote ~ race + educate, model = "tobit.bayes",data = turnout, verbose = FALSE)
```

Zelig-tobit-class *Linear Regression for a Left-Censored Dependent Variable*

Description

Linear Regression for a Left-Censored Dependent Variable

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).
model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.
below	(defaults to 0) The point at which the dependent variable is censored from below. If any values in the dependent variable are observed to be less than the censoring point, it is assumed that that particular observation is censored from below at the observed value.
above	(defaults to 0) The point at which the dependent variable is censored from above If any values in the dependent variable are observed to be more than the censoring point, it is assumed that that particular observation is censored from above at the observed value.
robust	defaults to FALSE. If TRUE, zelig() computes robust standard errors based on sandwich estimators and the options selected in cluster.
cluster	if robust = TRUE, you may select a variable to define groups of correlated observations. Let x3 be a variable that consists of either discrete numeric values, character strings, or factors that define strata. Then z.out <- zelig(y ~ x1 + x2, robust = TRUE, cluster = "x3", model = "tobit", data = mydata)means that the observations can be correlated within the strata defined by the variable x3, and that robust standard errors should be calculated according to those clusters. If robust = TRUE but cluster is not specified, zelig() assumes that each observation falls into its own cluster.

Details

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.

- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

Methods

zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE) The zelig function estimates a variety of statistical models

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_tobit.html

Examples

```
library(Zelig)
data(tobin)
z.out <- zelig(durable ~ age + quant, model = "tobit", data = tobin)
summary(z.out)
```

Zelig-weibull-class *Weibull Regression for Duration Dependent Variables*

Description

Weibull Regression for Duration Dependent Variables

Arguments

formula	a symbolic representation of the model to be estimated, in the form y ~ x1 + x2, where y is the dependent variable and x1 and x2 are the explanatory variables, and y, x1, and x2 are contained in the same dataset. (You may include more than two explanatory variables, of course.) The + symbol means "inclusion" not "addition." You may also include interaction terms and main effects in the form x1*x2 without computing them in prior steps; I(x1*x2) to include only the interaction term and exclude the main effects; and quadratic terms in the form I(x1^2).

model	the name of a statistical model to estimate. For a list of other supported models and their documentation see: http://docs.zeligproject.org/articles/.
data	the name of a data frame containing the variables referenced in the formula or a list of multiply imputed data frames each having the same variable names and row numbers (created by Amelia or to_zelig_mi).
...	additional arguments passed to zelig, relevant for the model to be estimated.
by	a factor variable contained in data. If supplied, zelig will subset the data frame based on the levels in the by variable, and estimate a model for each subset. This can save a considerable amount of effort. You may also use by to run models using MatchIt subclasses.
cite	If is set to 'TRUE' (default), the model citation will be printed to the console.

Details

In addition to the standard inputs, zelig() takes the following additional options for weibull regression:

- robust: defaults to FALSE. If TRUE, zelig() computes robust standard errors based on sandwich estimators based on the options in cluster.
- cluster: if robust = TRUE, you may select a variable to define groups of correlated observations. Let x3 be a variable that consists of either discrete numeric values, character strings, or factors that define strata. Then z.out <- zelig(y ~ x1 + x2, robust = TRUE, cluster = "x3", mod means that the observations can be correlated within the strata defined by the variable x3, and that robust standard errors should be calculated according to those clusters. If robust=TRUErobust=TRUE but cluster is not specified, zelig() assumes that each observation falls into its own cluster.

Additional parameters avaialable to this model include:

- weights: vector of weight values or a name of a variable in the dataset by which to weight the model. For more information see: http://docs.zeligproject.org/articles/weights.html.
- bootstrap: logical or numeric. If FALSE don't use bootstraps to robustly estimate uncertainty around model parameters due to sampling error. If an integer is supplied, the number of boostraps to run. For more information see: http://docs.zeligproject.org/articles/bootstraps.html.

Value

Depending on the class of model selected, zelig will return an object with elements including coefficients, residuals, and formula which may be summarized using summary(z.out) or individually extracted using, for example, coef(z.out). See http://docs.zeligproject.org/articles/getters.html for a list of functions to extract model components. You can also extract whole fitted model objects using from_zelig_model.

Methods

zelig(formula, data, model = NULL, ..., weights = NULL, by, bootstrap = FALSE) The zelig function estimates a variety of statistical models

See Also

Vignette: http://docs.zeligproject.org/articles/zelig_weibull.html

Examples

```
data(coalition)
z.out <- zelig(Surv(duration, ciep12) ~ fract + numst2,model = "weibull", data = coalition)
```

Zelig.url *Table of links for Zelig*

Description

Table of links for `help.zelig` for the core Zelig package.

---

zelig_qi_to_df *Extract simulated quantities of interest from a zelig object*

Description

Extract simulated quantities of interest from a zelig object

Usage

```
zelig_qi_to_df(obj)
```

Arguments

`obj`	a zelig object with simulated quantities of interest

Details

A simulated quantities of interest in a tidy data formatted `data.frame`. This can be useful for creating custom plots.

Each row contains a simulated value and each column contains:

- `setx_value` whether the simulations are from the base `x` `setx` or the contrasting `x1` for finding first differences.
- The fitted values specified in `setx` including a `by` column if `by` was used in the `zelig` call.
- `expected_value`
- `predicted_value`

For multinomial reponse models, a separate column is given for the expected probability of each outcome in the form `expected_*`. Additionally, there a is column of the predicted outcomes (`predicted_value`).

Author(s)

Christopher Gandrud

Source

For a discussion of tidy data see https://www.jstatsoft.org/article/view/v059i10.

See Also

qi_slimmer

Examples

```
#### QIs without first difference or range, from covariates fitted at
## central tendencies
z.1 <- zelig(Petal.Width ~ Petal.Length + Species, data = iris,
             model = "ls")
z.1 <- setx(z.1)
z.1 <- sim(z.1)
head(zelig_qi_to_df(z.1))

#### QIs for first differences
z.2 <- zelig(Petal.Width ~ Petal.Length + Species, data = iris,
             model = "ls")
z.2a <- setx(z.2, Petal.Length = 2)
z.2b <- setx(z.2, Petal.Length = 4.4)
z.2 <- sim(z.2, x = z.2a, x1 = z.2a)
head(zelig_qi_to_df(z.2))

#### QIs for first differences, estimated by Species
z.3 <- zelig(Petal.Width ~ Petal.Length, by = "Species", data = iris,
             model = "ls")
z.3a <- setx(z.3, Petal.Length = 2)
z.3b <- setx(z.3, Petal.Length = 4.4)
z.3 <- sim(z.3, x = z.3a, x1 = z.3a)
head(zelig_qi_to_df(z.3))

#### QIs for a range of fitted values
z.4 <- zelig(Petal.Width ~ Petal.Length + Species, data = iris,
             model = "ls")
z.4 <- setx(z.4, Petal.Length = 2:4)
z.4 <- sim(z.4)
head(zelig_qi_to_df(z.4))

#### QIs for a range of fitted values, estimated by Species
z.5 <- zelig(Petal.Width ~ Petal.Length, by = "Species", data = iris,
            model = "ls")
z.5 <- setx(z.5, Petal.Length = 2:4)
z.5 <- sim(z.5)
head(zelig_qi_to_df(z.5))

#### QIs for two ranges of fitted values
```

```
z.6 <- zelig(Petal.Width ~ Petal.Length + Species, data = iris,
             model = "ls")
z.6a <- setx(z.6, Petal.Length = 2:4, Species = "setosa")
z.6b <- setx(z.6, Petal.Length = 2:4, Species = "virginica")
z.6 <- sim(z.6, x = z.6a, x1 = z.6b)

head(zelig_qi_to_df(z.6))
```

`zelig_setx_to_df` *Extracted fitted values from a Zelig object with* `setx` *values*

Description

Extracted fitted values from a Zelig object with `setx` values

Usage

```
zelig_setx_to_df(obj)
```

Arguments

`obj`	a zelig object with simulated quantities of interest

Details

Fitted (`setx`) values in a tidy data formatted `data.frame`. This was designed to enable the WhatIf package's `whatif` function to extract "counterfactuals".

Author(s)

Christopher Gandrud

Examples

```
#### QIs without first difference or range, from covariates fitted at
## central tendencies
z.1 <- zelig(Petal.Width ~ Petal.Length + Species, data = iris,
             model = "ls")
z.1 <- setx(z.1)
zelig_setx_to_df(z.1)

#### QIs for first differences
z.2 <- zelig(Petal.Width ~ Petal.Length + Species, data = iris,
             model = "ls")
z.2 <- setx(z.2, Petal.Length = 2)
z.2 <- setx1(z.2, Petal.Length = 4.4)
zelig_setx_to_df(z.2)

#### QIs for first differences, estimated by Species
```

```
z.3 <- zelig(Petal.Width ~ Petal.Length, by = "Species", data = iris,
             model = "ls")
z.3 <- setx(z.3, Petal.Length = 2)
z.3 <- setx1(z.3, Petal.Length = 4.4)
zelig_setx_to_df(z.3)

#### QIs for a range of fitted values
z.4 <- zelig(Petal.Width ~ Petal.Length + Species, data = iris,
             model = "ls")
z.4 <- setx(z.4, Petal.Length = 2:4)
zelig_setx_to_df(z.4)

#### QIs for a range of fitted values, estimated by Species
z.5 <- zelig(Petal.Width ~ Petal.Length, by = "Species", data = iris,
             model = "ls")
z.5 <- setx(z.5, Petal.Length = 2:4)
zelig_setx_to_df(z.5)

#### QIs for two ranges of fitted values
z.6 <- zelig(Petal.Width ~ Petal.Length + Species, data = iris,
             model = "ls")
z.6 <- setx(z.6, Petal.Length = 2:4, Species = "setosa")
z.6 <- setx1(z.6, Petal.Length = 2:4, Species = "virginica")
zelig_setx_to_df(z.6)
```

Index