Package ‘COUNT’

October 12, 2022

Type Package

Title Functions, Data and Code for Count Data

Version 1.3.4

Date 2016-10-17

Imports MASS

Depends R (>= 2.10), msme, sandwich

Author Joseph M Hilbe <`hilbe@asu.edu`>

Maintainer Andrew Robinson <`apro@unimelb.edu.au`>

Description Functions, data and code for Hilbe, J.M. 2011. Negative Binomial Regression, 2nd Edition (Cambridge University Press) and Hilbe, J.M. 2014. Modeling Count Data (Cambridge University Press).

License GPL-2

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2016-10-19 13:07:05

R topics documented:

`affairs` *affairs*

Description

Data from Fair (1978). Although Fair used a tobit model with the data, the outcome measure can be modeled as a count. In fact, Greene (2003) modeled it as Poisson, but given the amount of overdispersion in the data, employing a negative binomial model is an appropriate strategy. The data is stored in the affairs data set. Naffairs is the response variable, indicating the number of affairs reported by the participant in the past year.

Usage

```
data(affairs)
```

Format

A data frame with 601 observations on the following 18 variables.

`naffairs` number of affairs within last year

`kids` 1=have children;0= no children

`vryunhap` (1/0) very unhappily married

`unhap`	(1/0) unhappily married
`avgmarr`	(1/0) average married
`hapavg`	(1/0) happily married
`vryhap`	(1/0) very happily married
`antirel`	(1/0) anti religious
`notrel`	(1/0) not religious
`slghtrel`	(1/0) slightly religious
`smerel`	(1/0) somewhat religious
`vryrel`	(1/0) very religious
`yrsmarr1`	(1/0) >0.75 yrs
`yrsmarr2`	(1/0) >1.5 yrs
`yrsmarr3`	(1/0) >4.0 yrs
`yrsmarr4`	(1/0) >7.0 yrs
`yrsmarr5`	(1/0) >10.0 yrs
`yrsmarr6`	(1/0) >15.0 yrs

Details

rwm5yr is saved as a data frame. Count models use naffairs as response variable. 0 counts are included.

Source

Fair, R. (1978). A Theory of Extramarital Affairs, Journal of Political Economy, 86: 45-61. Greene, W.H. (2003). Econometric Analysis, Fifth Edition, New York: Macmillan.

References

Hilbe, Joseph M (2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic regression Models, Chapman & Hall/CRC

Examples

```
data(affairs)
glmaffp <- glm(naffairs ~ kids + yrsmarr2 + yrsmarr3 + yrsmarr4 + yrsmarr5,
               family = poisson, data = affairs)
summary(glmaffp)
exp(coef(glmaffp))

require(MASS)
glmaffnb <- glm.nb(naffairs ~ kids + yrsmarr2 + yrsmarr3 + yrsmarr4 + yrsmarr5,
                   data=affairs)
summary(glmaffnb)
exp(coef(glmaffnb))
```

`azcabgptca` *azcabgptca*

Description

Random subset of the 1991 Arizona Medicare data for patients hospitalized subsequent to undergoing a CABG (DRGs 106, 107) or PTCA (DRG 112) cardiovascular procedure.

Usage

```
data(azcabgptca)
```

Format

A data frame with 1959 observations on the following 6 variables.

`died` systolic blood pressure of subject

`procedure` 1=CABG; 0=PTCA

`gender` 1=male; 0=female

`age` age of subject

`los` hospital length of stay

`type` 1=emerg/urgent; 0=elective

Details

azcabgptca is saved as a data frame.

Source

Hilbe, Negative Binomial Regression, 2nd ed, Cambridge Univ Press

References

Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press

Examples

```
data(azcabgptca); attach(azcabgptca)
table(los); table(procedure, type); table(los, procedure)
summary(los)
summary(c91a <- glm(los ~ procedure+ type, family=poisson, data=azcabgptca))
modelfit(c91a)
summary(c91b <- glm(los ~ procedure+ type, family=quasipoisson, data=azcabgptca))
modelfit(c91b)
library(sandwich)
sqrt(diag(vcovHC(c91a, type="HC0")))
```

`azdrg112` *azdrg112*

Description

The data set relates to the hospital length of stay for patients having a CABG or PTCA (typel) heart procedure. The data comes from the 1995 Arizona Medicare data for DRG (Diagnostic Related Group) 112. Other predictors include gender(1=female) and age75 (1-age 75+). Type is labeled as 1=emergency or urgent admission; 0= elective. Length of stay (los) ranges from 1 to 53 days.

Usage

```
data(azdrg112)
```

Format

A data frame with 1,798 observations on the following 4 variables.

`los` hospital length of stay: 1-53 days

`gender` 1=male; 0=female

`type1` 1=emergency/urgent admission; 0=elective admission

`age75` 1=age>75; 0=age<=75

Details

azdrg112 is saved as a data frame. Count models typically use los as response variable. 0 counts are not included

Source

DRG 112 data from the 1995 Arizona Medicare (MedPar) State files

References

Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press

Examples

```
data(azdrg112)
glmazp <- glm(los ~ type1 + gender + age75, family=poisson, data=azdrg112)
summary(glmazp)
exp(coef(glmazp))
library(MASS)
glmaznb <- glm.nb(los ~ type1 + gender + age75, data=azdrg112)
summary(glmaznb)
exp(coef(glmaznb))
```

`azpro` *azpro*

Description

Data come from the 1991 Arizona cardiovascular patient files. A subset of the fields was selected to model the differential length of stay for patients entering the hospital to receive one of two standard cardiovascular procedures: CABG and PTCA. CABG is the standard acronym for Coronary Artery Bypass Graft, where the flow of blood in a diseased or blocked coronary artery or vein has been grafted to bypass the diseased sections. PTCA, or Percutaneous Transluminal Coronary Angioplasty, is a method of placing a balloon in a blocked coronary artery to open it to blood flow. It is a much less severe method of treatment for those having coronary blockage, with a corresponding reduction in risk.

Usage

```
data(azpro)
```

Format

A data frame with 3589 observations on the following 6 variables.

`los` length of hospital stay

`procedure` 1=CABG;0=PTCA

`sex` 1=Male; 0=female

`admit` 1=Urgent/Emerg; 0=elective (type of admission)

`age75` 1= Age>75; 0=Age<=75

`hospital` encrypted facility code (string)

Details

azpro is saved as a data frame. Count models use los as response variable. 0 counts are structurally excluded

Source

1991 Arizona Medpar data, cardiovascular patient files, National Health Economics & Research Co.

References

Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC

Examples

```
data(azpro)
glmazp <- glm(los ~ procedure + sex + admit, family=poisson, data=azpro)
summary(glmazp)
exp(coef(glmazp))
#glmaznb < -glm.nb(los ~ procedure + sex + admit, data=azpro)
#summary(glmaznb)
#exp(coef(glmaznb))
```

`azprocedure` *azprocedure*

Description

Data come from the 1991 Arizona cardiovascular patient files. A subset of the fields was selected to model the differential length of stay for patients entering the hospital to receive one of two standard cardiovascular procedures: CABG and PTCA. CABG is the standard acronym for Coronary Artery Bypass Graft, where the flow of blood in a diseased or blocked coronary artery or vein has been grafted to bypass the diseased sections. PTCA, or Percutaneous Transluminal Coronary Angioplasty, is a method of placing a balloon in a blocked coronary artery to open it to blood flow. It is a much less severe method of treatment for those having coronary blockage, with a corresponding reduction in risk.

Usage

```
data(azprocedure)
```

Format

A data frame with 3589 observations on the following 6 variables.

`los`	length of hospital stay
`procedure`	1=CABG;0=PTCA
`sex`	1=Male; 0=female
`admit`	1=Urgent/Emerg; 0=elective (type of admission)
`age75`	1= Age>75; 0=Age<=75
`hospital`	encrypted facility code (string)

Details

azprocedure is saved as a data frame. Count models use los as response variable. 0 counts are structurally excluded

Source

1991 Arizona Medpar data, cardiovascular patient files, National Health Economics & Research Co.

References

Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC

Examples

```
library(MASS)
library(msme)

data(azprocedure)

glmazp <- glm(los ~ procedure + sex + admit, family=poisson, data=azprocedure)
summary(glmazp)
exp(coef(glmazp))

nb2 <- nbinomial(los ~ procedure + sex + admit, data=azprocedure)
summary(nb2)
exp(coef(nb2))

glmaznb <- glm.nb(los ~ procedure + sex + admit, data=azprocedure)
summary(glmaznb)
exp(coef(glmaznb))
```

`badhealth` *badhealth*

Description

From German health survey data for the year 1998 only.

Usage

```
data(badhealth)
```

Format

A data frame with 1,127 observations on the following 3 variables.

`numvisit` number of visits to doctor during 1998

`badh` 1=patient claims to be in bad health; 0=not in bad health

`age` age of patient: 20-60

Details

badhealth is saved as a data frame. Count models use numvisit as the response variable, 0 counts are included.

Source

German Health Survey, amended in Hilbe and Greene (2008).

References

Hilbe, Joseph M (2011), Negative Binomial Regression, Cambridge University Press Hilbe, J. and W. Greene (2008). Count Response Regression Models, in ed. C.R. Rao, J.P Miller, and D.C. Rao, Epidemiology and Medical Statistics, Elsevier Handbook of Statistics Series. London, UK: Elsevier.

Examples

```
data(badhealth)
glmbadp <- glm(numvisit ~ badh + age, family=poisson, data=badhealth)
summary(glmbadp)
exp(coef(glmbadp))
library(MASS)
glmbadnb <- glm.nb(numvisit ~ badh + age, data=badhealth)
summary(glmbadnb)
exp(coef(glmbadnb))
```

`fasttrakg` *fasttrakg*

Description

Data are from the Canadian National Cardiovascular Disease registry called, FASTRAK. years covered at 1996-1998. They have been grouped by covariate patterns from individual observations.

Usage

```
data(fasttrakg)
```

Format

A data frame with 15 observations on the following 9 variables.

`die`	number died from MI
`cases`	number of cases with same covariate pattern
`anterior`	1=anterior site MI; 0=inferior site MI
`hcabg`	1=history of CABG; 0=no history of CABG
`killip`	Killip level of cardiac event severity (1-4)age751= Age>75; 0=Age<=75
`kk1`	(1/0) angina; not MI
`kk2`	(1/0) moderate severity cardiac event
`kk3`	(1/0) Severe cardiac event
`kk4`	(1/0) Severe cardiac event; death

Details

fasttrakg is saved as a data frame. Count models use died as response numerator and cases as the demoninator

Source

1996-1998 FASTRAK data, Hoffman-LaRoche Canada, National Health Economics & Research Co.

References

Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press

Examples

```
library(MASS)
data(fasttrakg)
glmfp <- glm(die ~ anterior + factor(killip) + offset(log(cases)), family=poisson, data=fasttrakg)
summary(glmfp)
exp(coef(glmfp))
```

`fishing` *fishing*

Description

The fishing data is adapted from Zuur, Hilbe and Ieno (2013) to determine whether the data appears to be generated from more than one generating mechanism. The data are originally adapted from Bailey et al. (2008) who were interested in how certain deep-sea fish populations were impacted when commercial fishing began in locations with deeper water than in previous years. Given that there are 147 sites that were researched, the model is of (1) the total number of fish counted per site (totabund); (2) on the mean water depth per site (meandepth); (3) adjusted by the area of the site (sweptarea); (4) the log of which is the model offset.

Usage

```
data(fishing)
```

Format

A data frame with 147 observations on the following variables.

`totabund`	total fish counted per site
`meandepth`	mean water depth per site

`sweptarea` adjusted area of site

`density` folage density index

`site` catch site

`year` 1977-2002

`period` 0=1977-1989; 1=2000+

Details

fishing is saved as a data frame. Count models use totabund as response variable. Counts start at 2

Source

Zuur, Hilbe, Ieno (2013), A Beginner's Guide to GLM and GLMM using R,

References

Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press Zuur, Hilbe, Ieno (2013), A Beginner's Guide to GLM and GLMM using R, Highlands. Bailey M. et al (2008), "Longterm changes in deep-water fish populations in the North East Atlantic", Proc Roy Soc B 275:1965-1969.

Examples

```
## Not run:
library(MASS)
library(flexmix)
data(fishing)
attach(fishing)
fmm_pg <- flexmix(totabund~meandepth + offset(log(sweptarea)), data=rwm1984, k=2,
         model=list(FLXMRglm(totabund~., family="NB1"),
                     FLXMRglm(tpdocvis~., family="NB1")))
parameters(fmm_pg, component=1, model=1)
parameters(fmm_pg, component=2, model=1)
summary(fmm_pg)

## End(Not run)
```

`lbw` *lbw*

Description

The data come to us from Hosmer and Lemeshow (2000). Called the low birth weight (lbw) data, the response is a binary variable, low, which indicates whether the birth weight of a baby is under 2500g (low=1), or over (low=0).

Usage

```
data(lbw)
```

Format

A data frame with 189 observations on the following 10 variables.

`low` 1=low birthweight baby; 0=norml weight

`smoke` 1=history of mother smoking; 0=mother nonsmoker

`race` categorical 1-3: 1=white; 2-=black; 3=other

`age` age of mother: 14-45

`lwt` weight (lbs) at last menstrual period: 80-250 lbs

`ptl` number of false of premature labors: 0-3

`ht` 1=history of hypertension; 0 =no hypertension

`ui` 1=uterine irritability; 0 no irritability

`ftv` number of physician visits in 1st trimester: 0-6

`bwt` birth weight in grams: 709 - 4990 gr

Details

lbw is saved as a data frame. Count models can use ftv as a response variable, or convert it to grouped format

Source

Hosmer, D and S. Lemeshow (2000), Applied Logistic Regression, Wiley

References

Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC

Examples

```
data(lbw)
glmbwp <- glm(ftv ~ low + smoke + factor(race), family=poisson, data=lbw)
summary(glmbwp)
exp(coef(glmbwp))
library(MASS)
glmbwnb <- glm.nb(ftv ~ low + smoke + factor(race), data=lbw)
summary(glmbwnb)
exp(coef(glmbwnb))
```

`lbwgrp` *lbwgrp*

Description

grouped format of the lbw data. The observation level data come to us form Hosmer and Lemeshow (2000). Grouping is such that lowbw is the numerator, and cases the denominator of a binomial model, or cases may be an offset to the count variable, lowbw. Birthweights under 2500g classifies a low birthweight baby.

Usage

```
data(lbwgrp)
```

Format

A data frame with 6 observations on the following 7 variables.

`lowbw`	Number of low weight babies per covariate pattern: 12-60
`cases`	Number of observations with same covariate pattern: 30-165
`smoke`	1=history of mother smoking; 0=mother nonsmoker
`race1`	(1/0): Caucasian
`race2`	(1/0): Black
`race3`	(1/0): Other
`low`	low birth weight (not valid variable in grouped format)

Details

lbwgrp is saved as a data frame. Count models: count response=lowbt; offset=log(cases); Binary: binomial numerator= lowbt; binomial denominator=cases

Source

Hosmer, D and S. Lemeshow (2000), Applied Logistic Regression, Wiley

References

Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC

Examples

```
data(lbwgrp)
glmgp <- glm(lowbw ~ smoke + race2 + race3 + offset(log(cases)), family=poisson, data=lbwgrp)
summary(glmgp)
exp(coef(glmgp))
library(MASS)
glmgnb <- glm.nb(lowbw ~  smoke + race2 + race3, data=lbwgrp)
summary(glmgnb)
exp(coef(glmgnb))
```

`logit_syn` *Logistic regression : generic synthetic binary/binomial logistic data and model*

Description

logit_syn is a generic function for developing synthetic logistic regression data and a model given user defined specifications.

Usage

```
logit_syn(nobs=50000, d=1,  xv  = c(1, 0.5, -1.5))
```

Arguments

`nobs`	number of observations in model, Default is 50000
`d`	binomial denominator, Default is 1, a binary logistic model. May use a variable containing different denominator values.
`xv`	predictor coefficient values. First argument is intercept. Use as xv = c(intercept , x1_coef, x2_coef, ...)

Details

Create a synthetic logistic regression model using the appropriate arguments. Binomial denominator must be declared. For a binary logistic model, d=1. A variable may be used as the denominator when values differ. See examples.

Value

`by`	binomial logistic numerator; number of successes
`sim.data`	synthetic data set

Author(s)

Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology Andrew Robinson, Universty of Melbourne, Australia.

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.
Hilbe, J.M. (2009), Logistic Regression Models, Chapman & Hall/CRCD

See Also

probit_syn

Examples

```
# Binary logistic regression (denominator=1)
sim.data <-logit_syn(nobs = 500, d = 1, xv = c(1, .5, -1.5))
mylogit <- glm(cbind(by,dby) ~ ., family=binomial(link="logit"), data = sim.data)
summary(mylogit)
confint(mylogit)

# Binary logistic regression with odds ratios (denominator=1); 3 predictors
sim.data <-logit_syn(nobs = 500, d = 1, xv = c(1, .75, -1.5, 1.15))
mylogit <- glm(cbind(by,dby) ~ ., family=binomial(link="logit"), data = sim.data)
exp(coef(mylogit))
exp(confint(mylogit))

# Binomial or grouped logistic regression with defined denominator, den
den <- rep(1:5, each=100, times=1)*100
sim.data <- logit_syn(nobs = 500, d = den, xv = c(1, .5, -1.5))
gby <- glm(cbind(by,dby) ~ ., family=binomial(link="logit"), data = sim.data)
summary(gby)

## Not run:
# default
sim.data <- logit_syn(nobs=500, d=1,  xv = c(2, -.55, 1.15))
dlogit <- glm(cbind(by,dby) ~ . , family=binomial(link="logit"), data = sim.data)
summary(dlogit)

## End(Not run)
```

`loomis` *loomis*

Description

Data are taken from Loomis (2003). The study relates to a survey taken on reported frequency of visits to national parks during the year. The survey was taken at park sites, thus incurring possible effects of endogenous stratification.

Usage

```
data(loomis)
```

Format

A data frame with 410 observations on the following 11 variables.

`anvisits` number of annual visits to park

`gender` 1=male;0=female

`income` income in US dollars per year, categorical: 4 levels

`income1` <=$25000

`income2` >$25000 - $55000

`income3` >$55000 - $95000

`income4` >$95000

`travel` travel time, categorical: 3 levels

`travel1` <.25 hrs

`travel2` >=.25 - <4 hrs

`travel3` >=4 hrs

Details

loomis is saved as a data frame. Count models typically use anvisits as response variable. 0 counts are included

Source

from Loomis (2003)

References

Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Loomis, J. B. (2003). Travel cost demand model based river recreation benefit estimates with on-site and household surveys: Comparative results and a correction procedure, Water Resources Research, 39(4): 1105

Examples

```
data(loomis)
glmlmp <- glm(anvisits ~ gender + factor(income) + factor(travel), family=poisson, data=loomis)
summary(glmlmp)
exp(coef(glmlmp))
library(MASS)
glmlmnb <- glm.nb(anvisits ~ gender + factor(income) + factor(travel), data=loomis)
summary(glmlmnb)
exp(coef(glmlmnb))
```

`mdvis` *mdvis*

Description

Data from a subset of the German Socio-Economic Panel (SOEP). The subset was created by Rabe-Hesketh and Skrondal (2005). Only working women are included in these data. Beginning in 1997, German health reform in part entailed a 200 co-payment as well as limits in provider reimbursement. Patients were surveyed for the one year panel (1996) prior to and the one year panel (1998) after reform to assess whether the number of physician visits by patients declined - which was the goal of reform legislation. The response, or variable to be explained by the model, is numvisit, which indicates the number of patient visits to a physician's office during a three month period.

Usage

```
data(mdvis)
```

Format

A data frame with 2,227 observations on the following 13 variables.

`numvisit`	visits to MD office 3mo prior
`reform`	1=interview yr post-reform: 1998;0=pre-reform:1996
`badh`	1=bad health; 0 = not bad health
`age`	Age(yrs 20-60)
`educ`	education(1:7-10;2=10.5-12;3=HSgrad+)
`educ1`	educ1= 7-10 years
`educ2`	educ2= 10.5-12 years
`educ3`	educ3= post secondary or high school
`agegrp`	age: 1=20-39; 2=40-49; 3=50-60
`age1`	age 20-39
`age2`	age 40-49
`age3`	age 50-60
`loginc`	log(household income in DM)

Details

mdvis is saved as a data frame. Count models typically use docvis as response variable. 0 counts are included

Source

German Socio-Economic Panel (SOEP), 1995 pre-reform; 1998 post reform. Created by Rabe-Hesketh and Skrondal (2005).

References

Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC Rabe-Hesketh, S. and A. Skrondal (2005). Multilevel and Longitudinal Modeling Using Stata, College Station: Stata Press.

Examples

```
data(mdvis)
glmmdp <- glm(numvisit ~ reform + factor(educ) + factor(agegrp), family=poisson, data=mdvis)
summary(glmmdp)
exp(coef(glmmdp))
library(MASS)
glmmdnb <- glm.nb(numvisit ~ reform + factor(educ) + factor(agegrp), data=mdvis)
summary(glmmdnb)
exp(coef(glmmdnb))
```

`medpar` *medpar*

Description

The US national Medicare inpatient hospital database is referred to as the Medpar data, which is prepared yearly from hospital filing records. Medpar files for each state are also prepared. The full Medpar data consists of 115 variables. The national Medpar has some 14 million records, with one record for each hospilitiztion. The data in the medpar file comes from 1991 Medicare files for the state of Arizona. The data are limited to only one diagnostic group (DRG 112). Patient data have been randomly selected from the original data.

Usage

```
data(medpar)
```

Format

A data frame with 1495 observations on the following 10 variables.

`los`	length of hospital stay
`hmo`	Patient belongs to a Health Maintenance Organization, binary
`white`	Patient identifies themselves as Caucasian, binary
`died`	Patient died, binary
`age80`	Patient age 80 and over, binary
`type`	Type of admission, categorical
`type1`	Elective admission, binary
`type2`	Urgent admission,binary
`type3`	Elective admission, binary
`provnum`	Provider ID

Details

medpar is saved as a data frame. Count models use los as response variable. 0 counts are structurally excluded

Source

1991 National Medpar data, National Health Economics & Research Co.

References

Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC first used in Hardin, JW and JM Hilbe (2001, 2007), Generalized Linear Models and Extensions, Stata Press

Examples

```
library(MASS)
library(msme)
data(medpar)
glmp <- glm(los ~ hmo + white + factor(type), family=poisson, data=medpar)
summary(glmp)
exp(coef(glmp))
nb2 <- nbinomial(los ~ hmo + white + factor(type), data=medpar)
summary(nb2)
exp(coef(nb2))
glmnb <- glm.nb(los ~ hmo + white + factor(type), data=medpar)
summary(glmnb)
exp(coef(glmnb))
```

`ml.nb1` *NB1: maximum likelihood linear negative binomial regression*

Description

ml.nb1 is a maximum likelihood function for estimating linear negative binomial (NB1) data. Output consists of a table of parameter estimates, standard errors, z-value, and confidence intervals.

Usage

```
ml.nb1(formula, data, offset=0, start=NULL, verbose=FALSE)
```

Arguments

`formula`	an object of class '"formula"': a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
`data`	a mandatory data frame containing the variables in the model.
`offset`	this can be used to specify an _a priori_ known component to be included in the linear predictor during fitting. The offset should be provided on the log scale.
`start`	an optional vector of starting values for the parameters.
`verbose`	a logical flag to indicate whether the fit information should be printed.

Details

ml.nb1 is used like glm.nb, but without saving ancillary statistics.

Value

The function returns a dataframe with the following components:

`Estimate`	ML estimate of the parameter
`SE`	Asymptotic estimate of the standard error of the estimate of the parameter
`Z`	The Z statistic of the asymptotic hypothesis test that the population value for the parameter is 0.
`LCL`	Lower 95% confidence interval for the parameter estimate.
`UCL`	Upper 95% confidence interval for the parameter estimate.

Author(s)

Andrew Robinson, Universty of Melbourne, Australia, and Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

`glm.nb`, `ml.nbc`, `ml.nb2`

Examples

```
# Table 10.8, Hilbe. J.M. (2011), Negative Binomial Regression,
#   2nd ed. Cambridge University Press (adapted)
data(medpar)
medpar$type <- factor(medpar$type)
med.nb1 <- ml.nb1(los ~ hmo + white + type, data = medpar)
med.nb1
```

`ml.nb2` *NB2: maximum likelihood linear negative binomial regression*

Description

ml.nb2 is a maximum likelihood function for estimating linear negative binomial (NB2) data. Output consists of a table of parameter estimates, standard errors, z-value, and confidence intervals.

Usage

```
ml.nb2(formula, data, offset=0, start=NULL, verbose=FALSE)
```

Arguments

`formula`	an object of class '"formula"': a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
`data`	a mandatory data frame containing the variables in the model.
`offset`	this can be used to specify an _a priori_ known component to be included in the linear predictor during fitting. The offset should be provided on the log scale.
`start`	an optional vector of starting values for the parameters.
`verbose`	a logical flag to indicate whether the fit information should be printed.

Details

ml.nb2 is used like glm.nb, but without saving ancillary statistics.

Value

The function returns a dataframe with the following components:

`Estimate`	ML estimate of the parameter
`SE`	Asymptotic estimate of the standard error of the estimate of the parameter
`Z`	The Z statistic of the asymptotic hypothesis test that the population value for the parameter is 0.
`LCL`	Lower 95% confidence interval for the parameter estimate.
`UCL`	Upper 95% confidence interval for the parameter estimate.

Author(s)

Andrew Robinson, Universty of Melbourne, Australia, and Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

`glm.nb`, `ml.nbc`, `ml.nb1`

Examples

```
# Table 8.7, Hilbe. J.M. (2011), Negative Binomial Regression,
#   2nd ed. Cambridge University Press (adapted)
data(medpar)
medpar$type <- factor(medpar$type)
med.nb2 <- ml.nb2(los ~ hmo + white + type, data = medpar)
med.nb2
```

`ml.nbc` *NBC: maximum likelihood linear negative binomial regression*

Description

ml.nbc is a maximum likelihood function for estimating canonical linear negative binomial (NB-C) data.

Usage

```
ml.nbc(formula, data, start=NULL, verbose=FALSE)
```

Arguments

`formula`	an object of class '"formula"': a symbolic description of the model to be fitted. The details of model specification are given under 'Details'.
`data`	a mandatory data frame containing the variables in the model.
`start`	an optional vector of starting values for the parameters.
`verbose`	a logical flag to indicate whether the fit information should be printed.

Details

ml.nbc is used like glm.nb, but without saving ancillary statistics.

Value

The function returns a dataframe with the following components:

`Estimate`	ML estimate of the parameter
`SE`	Asymptotic estimate of the standard error of the estimate of the parameter
`Z`	The Z statistic of the asymptotic hypothesis test that the population value for the parameter is 0.
`LCL`	Lower 95% confidence interval for the parameter estimate.
`UCL`	Upper 95% confidence interval for the parameter estimate.

Author(s)

Andrew Robinson, Universty of Melbourne, Australia, and Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

`glm.nb`, `ml.nb1`, `ml.nb2`

Examples

```
# Table 10.12, Hilbe. J.M. (2011), Negative Binomial Regression,
#   2nd ed. Cambridge University Press (adapted)

## Not run:
data(medpar)
nobs <- 50000
x2 <- runif(nobs)
x1 <- runif(nobs)
xb <- 1.25*x1 + .1*x2 - 1.5
mu <- 1/(exp(-xb)-1)
p <- 1/(1+mu)
r <- 1
gcy <- rnbinom(nobs, size=r, prob = p)
test <- data.frame(gcy, x1, x2)
nbc <- ml.nbc(gcy ~ x1 + x2, data=test)
nbc

## End(Not run)
```

`ml.pois` *NB2: maximum likelihood Poisson regression*

Description

ml.pois is a maximum likelihood function for estimating Poisson data. Output consists of a table of parameter estimates, standard errors, z-value, and confidence intervals. An offset may be declared as an option.

Usage

```
ml.pois(formula, data, offset=0, start=NULL, verbose=FALSE)
```

Arguments

`formula`	an object of class ’"formula"’: a symbolic description of the model to be fitted.
`data`	a mandatory data frame containing the variables in the model.
`offset`	this can be used to specify an _a priori_ known component to be included in the linear predictor during fitting. The offset should be provided on the log scale.
`start`	an optional vector of starting values for the parameters.
`verbose`	a logical flag to indicate whether the fit information should be printed.

Details

ml.pois is used like glm, but does not provide ancillary statistics.

Value

The function returns a dataframe with the following components:

`Estimate`	ML estimate of the parameters
`SE`	Asymptotic estimate of the standard error of the estimate of the parameter
`Z`	The Z statistic of the asymptotic hypothesis test that the population value for the parameter is 0.
`LCL`	Lower 95% confidence interval for the parameter estimates.
`UCL`	Upper 95% confidence interval for the parameter estimates.

Author(s)

Andrew Robinson, Universty of Melbourne, Australia, and Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

`glm.nb`, `ml.nbc`, `ml.nb1`

Examples

```
# Table 8.7, Hilbe. J.M. (2011), Negative Binomial Regression,
#   2nd ed. Cambridge University Press (adapted)
data(medpar)
medpar$type <- factor(medpar$type)
med.pois <- ml.pois(los ~ hmo + white + type, data = medpar)
med.pois

data(rwm5yr)
lyear <- log(rwm5yr$year)
rwm.poi <- ml.pois(docvis ~ outwork + age + female, offset=lyear, data =
```

```
rwm5yr)
rwm.poi
exp(rwm.poi$Estimate)
exp(rwm.poi$LCL)
exp(rwm.poi$UCL)
```

`modelfit` *Fit Statistics for generalized linear models*

Description

modelfit is used following a glm() or glm.nb() model to produce a list of model fit statistics.

Usage

```
modelfit(x)
```

Arguments

`x`	the only argument is the name of the fitted glm or glm.nb function model

Details

modelfit is to be used as a post-estimation function, following the use of glm() or glm.nb().

Value

`obs`	number of model observatiions
`aic`	AIC statistic
`xvars`	number of model predictors
`rdof`	residial degrees of freedom
`aic_n`	AIC, 'aic'/'obs'
`ll`	log-likelihood
`bic_r`	BIC - Raftery parameterization
`bic_l`	BIC - log-likelihood Standard definition (Stata)
`bic_qh`	Hannan-Quinn IC statistic (Limdep)

Note

modelfit.r must be loaded into memory in order to be effectve. Users may past modelfit.r into script editor to run, as well as load it.

Author(s)

Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of technology

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

Hilbe, J.M. (2009), Logistic Regression Models, Chapman Hall/CRC

See Also

`glm`, `glm.nb`

Examples

```
## Hilbe (2011), Table 9.17
library(MASS)
data(lbwgrp)
nb9_3 <- glm.nb(lowbw ~ smoke + race2 + race3 + offset(log(cases)), data=lbwgrp)
summary(nb9_3)
exp(coef(nb9_3))
modelfit(nb9_3)
```

`myTable` *Frequency table*

Description

mytable is used to produce a table of frequencies, proportion and cumulative proportions for a count variable

Usage

```
myTable(x)
```

Arguments

`x`	the only argument is the name of the count variable

Details

myTable is used as either a diagnostic to view the distribution of a count variable, or as a frequency distribution display in its own right. myTable is given in Table 9.40 in Hilbe (2011).

Value

`x`	count value
`Freq`	Frequency of count
`Prop`	Proportion
`CumProp`	Cumulative proportion

Author(s)

Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press. Hilbe, J.M. (2009), Logistic Regression Models, Chapman Hall/CRC

See Also

`modelfit`

Examples

```
data(medpar)
myTable(medpar$los)
```

`nb1_syn` *Negative binomial (NB1): generic synthetic linear negative binomial data and model*

Description

nb1_syn is a generic function for developing synthetic NB1 data and a model given user defined specifications.

Usage

```
nb1_syn(nobs=50000, delta=1, xv = c(1, 0.75, -1.25))
```

Arguments

`nobs`	number of observations in model, Default is 50000
`delta`	NB1 heterogeneity or ancillary parameter
`xv`	predictor coefficient values. First argument is intercept. Use as xv = c(intercept , x1_coef, x2_coef, ...)

Details

Create a synthetic linear negative binomial (NB1) regression model using the appropriate arguments. Model data with predictors indicated as a group with a period (.). See examples.

Data can be modeled using the ml.nb1.r function in the COUNT package, or by using the gamlss function in the gamlss package, using the "family=NBII" option.

Value

`nb1y`	Negative binomial (NB1) response; number of counts
`sim.data`	synthetic data set

Author(s)

Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology Andrew Robinson, Universty of Melbourne, Australia.

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

`nb2_syn`, `nbc_syn`

Examples

```
sim.data <- nb1_syn(nobs = 5000, delta = .5, xv = c(.5, 1.25, -1.5))
mynb1 <- ml.nb1(nb1y ~ . , data = sim.data)
mynb1

## Not run:
# use gamlss to model NB1 data
library(gamlss)
sim.data <- nb1_syn(nobs = 5000, delta = .5, xv = c(.5, 1.25, -1.5))
mynb1 <- gamlss( nb1y ~ . , family=NBII, data = sim.data)
mynb1

## End(Not run)

## Not run:
# default
sim.data <- nb1_syn()
dnb1 <- ml.nb1(nb1y ~ . , data = sim.data)
dnb1

## End(Not run)
```

`nb2.obs.pred` *Table of negative binomial counts: observed vs predicted proportions and difference*

Description

nb2.obs.pred is used to produce a table of a negative binomial model count response with mean observed vs mean predicted proportions, and their difference.

Usage

```
nb2.obs.pred(len, model)
```

Arguments

`len`	highest count for the table
`model`	name of the negative binomial model created

Details

nb2.obs.pred is used to determine where disparities exist in the mean observed and predicted proportions in the range of model counts. nb2.obs.pred is used in Table 9.28 and other places in Hilbe (2011). nb2.obs.pred follows glm.nb(), where both y=TRUE and model=TRUE options must be used.

Value

`Count`	count value
`obsPropFreq`	Observed proportion of counts
`avgp`	Predicted proportion of counts
`Diff`	Difference in observed vs predicted

Author(s)

Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology Andrew Robinson, University of Melbourne, Australia

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

`myTable`

Examples

```
library(MASS)

data(medpar)
mdpar <- glm.nb(los ~ hmo+white+type2+type3, data=medpar, y=TRUE, model=TRUE)
nb2.obs.pred(len=25, model=mdpar)
```

`nb2_syn` *Negative binomial (NB2): generic synthetic negative binomial data and model*

Description

nb2_syn is a generic function for developing synthetic NB2 data and a model given user defined specifications.

Usage

```
nb2_syn(nobs = 50000, off = 0, alpha = 1, xv = c(1, 0.75, -1.5))
```

Arguments

`nobs`	number of observations in model, Default is 50000
`alpha`	NB2 heterogeneity or ancillary parameter
`off`	optional: log of offset variable
`xv`	predictor coefficient values. First argument is intercept. Use as xv = c(intercept , x1_coef, x2_coef, ...)

Details

Create a synthetic negative binomial (NB2) regression model using the appropriate arguments. Model data with predictors indicated as a group with a period (.). Offset optional. If no offset is desired, drop "off= loff" from nb2_syn function call and "+ loff" from glm.nb function call. See examples.

Data can be estimated using the glm.nb() function, or the ml.nb2() function in the COUNT package, or by using the gamlss function in the gamlss package, with "family=NBI" option.

Value

`nby`	Negative binomial response; number of counts
`sim.data`	synthetic data set

Author(s)

Andrew Robinson, Universty of Melbourne, Australia, and Joseph M. Hilbe, Arizona State University, Jet Propulsion Laboratory, California Institute of Technology

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

`poisson_syn`, `nb1_syn`, `nbc_syn`

Examples

```
library(MASS)

sim.data <- nb2_syn(nobs = 500, alpha = .5, xv = c(2, .75, -1.25))
mynb2 <- glm.nb(nby ~ . , data = sim.data)
summary(mynb2)
confint(mynb2)

# with offset
oset <- rep(1:5, each=100, times=1)*100
loff <- log(oset)
sim.data <- nb2_syn(nobs = 500, off = loff, alpha = .5, xv = c(1.2, -.75, .25, -1.3))
mypof <- glm.nb(nby ~ . + loff, data = sim.data)
summary(mypof)
confint(mypof)

# without offset, exponentiated coefficients, CI's
sim.data <- nb2_syn(nobs = 500, alpha = .75, xv = c(1, .5, -1.4))
mynbf <- glm.nb(nby ~ . , data = sim.data)
exp(coef(mynbf))
exp(confint(mynbf))

## Not run:
# default, without offset
sim.data <- nb2_syn()
dnb2 <- glm.nb(nby ~ . , data = sim.data)
summary(dnb2)

## End(Not run)

# use ml.nb2.r function
sim.data <- nb2_syn(nobs = 500, alpha = .5, xv = c(2, .75, -1.25))
mynb2x <- ml.nb2(nby ~ . , data = sim.data)
mynb2x

## Not run:
# use gamlss function for modeling data after sim.data created
library(gamlss)
sim.data <- nb2_syn(nobs = 500, alpha = .5, xv = c(2, .75, -1.25))
gamnb <- gamlss(nby ~ ., family=NBI, data = sim.data)
gamnb

## End(Not run)
```

`nbc_syn` *Negative binomial (NB-C): generic synthetic canonical negative binomial data and model*

Description

nbc_syn is a generic function for developing synthetic NB-C data and a model given user defined specifications.

Usage

```
nbc_syn(nobs=50000, alpha=1.15, xv = c(-1.5, -1.25, -.1))
```

Arguments

`nobs`	number of observations in model, Default is 50000
`alpha`	NB-C heterogeneity or ancillary parameter
`xv`	predictor coefficient values. First argument is intercept. Use as xv = c(intercept , x1_coef, x2_coef, ...)

Details

Create a synthetic canonial negative binomial (NB-C) regression model using the appropriate arguments. Model data with predictors indicated as a group with a period (.). Data can be modeled using the ml.nbc.r function in the COUNT package. See examples.

Value

`nbcy`	Canonical negative binomial (NB-C) response; number of counts
`sim.data`	synthetic data set

Author(s)

Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology Andrew Robinson, Universty of Melbourne, Australia.

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

`nb2_syn`, `nb1_syn`

Examples

```
## Not run:
sim.data <- nbc_syn(nobs = 50000, alpha = 1.15, xv = c(-1.5, -1.25, -.1))
mynbc <- ml.nbc(nbcy ~ . , data = sim.data)
mynbc

# default
sim.data <- nbc_syn()
```

```
dnbc <- ml.nbc(nbcy ~ . , data = sim.data)
dnbc

## End(Not run)
```

`nuts` *nuts*

Description

Squirrel data set (nuts) from Zuur, Hilbe, and Ieno (2013). As originally reported by Flaherty et al (2012), researchers recorded information about squirrel behavior and forest attributes across various plots in Scotland's Abernathy Forest. The study focused on the following variables. response cones number of cones stripped by red squirrels per plot predictor sntrees standardized number of trees per plot sheight standardized mean tree height per plot scover standardized percentage of canopy cover per plot The stripped cone count was only taken when the mean diameter of trees was under 0.6m (dbh).

Usage

```
data(nuts)
```

Format

A data frame with 52 observations on the following 8 variables.

`cones`	number cones stripped by squirrels
`ntrees`	number of trees per plot
`dbh`	number DBH per plot
`height`	mean tree height per plot
`cover`	canopy closure (as a percentage)
`sntrees`	standardized number of trees per plot
`sheight`	standardized mean tree height per plot
`scover`	standardized canopy closure (as a percentage)

Details

nuts is saved as a data frame. Count models use ntrees as response variable. Counts start at 3

Source

Zuur, Hilbe, Ieno (2013), A Beginner's Guide to GLM and GLMM using R, Highlands

References

Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press Zuur, Hilbe, Ieno (2013), A Beginner's Guide to GLM and GLMM using R, Highlands. Flaherty, S et al (2012), "The impact of forest stand structure on red squirrels habitat use", Forestry 85:437-444.

Examples

```
data(nuts)
nut <- subset(nuts, dbh < 0.6)
# sntrees <- scale(nuts$ntrees)
# sheigtht <- scale(nuts$height)
# scover <- scale(nuts$cover)
summary(PO <- glm(cones ~ sntrees + sheight + scover, family=quasipoisson, data=nut))
```

`poi.obs.pred` *Table of Poisson counts: observed vs predicted proportions and difference*

Description

poi.obs.pred is used to produce a table of a Poisson model count response with mean observed vs mean predicted proportions, and their difference.

Usage

```
poi.obs.pred(len, model)
```

Arguments

`len`	highest count for the table
`model`	name of the Poisson model created

Details

poi.obs.pred is used to determine where disparities exist in the mean observed and predicted proportions in the range of model counts. poi.obs.pred is used in Table 6.15 and other places in Hilbe (2011). poi.obs.pred follows glm(), where both y=TRUE and model=TRUE options must be used.

Value

`Count`	count value
`obsPropFreq`	Observed proportion of counts
`avgp`	Predicted proportion of counts
`Diff`	Difference in observed vs predicted

Author(s)

Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology Andrew Robinson, University of Melbourne, Australia

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

`myTable`

Examples

```
data(medpar)
mdpar <- glm(los ~ hmo+white+type2+type3, family=poisson, data=medpar, y=TRUE, model=TRUE)
poi.obs.pred(len=25, model=mdpar)
```

`poisson_syn` *Poisson : generic synthetic Poisson data and model*

Description

poisson_syn is a generic function for developing synthetic Poisson data and a model given user defined specifications.

Usage

```
poisson_syn(nobs = 50000, off = 0, xv = c(1, -.5,  1))
```

Arguments

`nobs`	number of observations in model, Default is 50000
`off`	optional: log of offset variable
`xv`	predictor coefficient values. First argument is intercept. Use as xv = c(intercept , x1_coef, x2_coef, ...)

Details

Create a synthetic Poisson regression model using the appropriate arguments. Offset optional. Model data with predictors indicated as a group with a period (.). See examples.

Value

`py`	Poisson response; number of counts
`sim.data`	synthetic data set

Author(s)

Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology Andrew Robinson, Universty of Melbourne, Australia.

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press.

See Also

nb2_syn

Examples

```
# standard Poisson model with two predictors and intercept
sim.data <- poisson_syn(nobs = 500, xv = c(2, .75, -1.25))
mypo <- glm(py ~ . , family=poisson, data = sim.data)
summary(mypo)
confint(mypo)

# Poisson with offset and three predictors
oset <- rep(1:5, each=100, times=1)*100
loff <- log(oset)
sim.data <- poisson_syn(nobs = 500, off = loff, xv = c(1.2, -.75, .25, -1.3))
mypof <- glm(py ~ . + loff, family=poisson, data = sim.data)
summary(mypof)
confint(mypof)

# Poisson without offset, exponentiated coefficients, CI's
sim.data <- poisson_syn(nobs = 500, xv = c(2, .75, -1.25))
mypo <- glm(py ~ . , family=poisson, data = sim.data)
exp(coef(mypo))
exp(confint(mypo))

## Not run:
# default (without offset)
sim.data <- poisson_syn()
dmypo <- glm( py ~ . , family=poisson, data = sim.data)
summary(dmypo)

## End(Not run)
```

`probit_syn` *Probit regression : generic synthetic binary/binomial probit data and model*

Description

probit_syn is a generic function for developing synthetic probit regression data and a model given user defined specifications.

Usage

```
probit_syn(nobs=50000, d=1,  xv = c(1, 0.5, -1.5))
```

Arguments

`nobs`	number of observations in model, Default is 50000
`d`	binomial denominator, Default is 1, a binary probit model. May use a variable containing different denominator values.
`xv`	predictor coefficient values. First argument is intercept. Use as xv = c(intercept , x1_coef, x2_coef, ...)

Details

Create a synthetic probit regression model using the appropriate arguments. Binomial denominator must be declared. For a binary probit model, d=1. A variable may be used as the denominator when values differ. See examples.

Value

`py`	binomial probit numerator; number of successes
`sim.data`	synthetic data set

Author(s)

Joseph M. Hilbe, Arizona State University, and Jet Propulsion Laboratory, California Institute of Technology Andrew Robinson, Universty of Melbourne, Australia.

References

Hilbe, J.M. (2011), Negative Binomial Regression, second edition, Cambridge University Press. Hilbe, J.M. (2009), Logistic Regression Models, Chapman & Hall/CRCD

See Also

`logit_syn`

Examples

```
# Binary probit regression (denominator=1)
sim.data <-probit_syn(nobs = 5000, d = 1, xv = c(1, .5, -1.5))
myprobit <- glm(cbind(py,dpy) ~ ., family=binomial(link="probit"), data = sim.data)
summary(myprobit)
confint(myprobit)

# Binary probit regression with 3 predictors (denominator=1)
sim.data <-probit_syn(nobs = 5000, d = 1, xv = c(1, .75, -1.5, 1.15))
myprobit <- glm(cbind(py,dpy) ~ ., family=binomial(link="probit"), data = sim.data)
summary(myprobit)
confint(myprobit)

# Binomial or grouped probit regression with defined denominator, den
den <- rep(1:5, each=1000, times=1)*100
sim.data <- probit_syn(nobs = 5000, d = den, xv = c(1, .5, -1.5))
```

```
gpy <- glm(cbind(py,dpy) ~ ., family=binomial(link="probit"), data = sim.data)
summary(gpy)

## Not run:
# default
sim.data <- probit_syn()
dprobit <- glm(cbind(py,dpy) ~ . , family=binomial(link="probit"), data = sim.data)
summary(dprobit)

## End(Not run)
```

`rwm` *rwm*

Description

German health registry for the years 1984-1988. Health information for years prior to health reform.

Usage

```
data(rwm)
```

Format

A data frame with 27,326 observations on the following 4 variables.

`docvis` number of visits to doctor during year (0-121)

`age` age: 25-64

`educ` years of formal education (7-18)

`hhninc` household yearly income in DM/1000)

Details

rwm is saved as a data frame. Count models typically use docvis as response variable. 0 counts are included

Source

German Health Reform Registry, years pre-reform 1984-1988, From Hilbe and Greene (2008)

References

Hilbe, Joseph M (2011), Negative Binomial Regression, Cambridge University Press Hilbe, J.M. and W.H. Greene (2008), "Count Response Regression Models", in Rao, CR, JP Miller and DC Rao (eds), Handbook of Statistics 27: Epidemiology and Medical Statistics, Amsterdam: Elsevier. pp. 210-252.

Examples

```
data(rwm)
glmrwp <- glm(docvis ~ age + educ + hhninc, family=poisson, data=rwm)
summary(glmrwp)
exp(coef(glmrwp))
library(MASS)
glmrwnb <- glm.nb(docvis ~ age + educ + hhninc, data=rwm)
summary(glmrwnb)
exp(coef(glmrwnb))
```

`rwm1984` *rwm1984*

Description

German health registry for the year 1984.

Usage

`data(rwm1984)`

Format

A data frame with 3,874 observations on the following 17 variables.

`docvis`	number of visits to doctor during year (0-121)
`hospvis`	number of days in hospital during year (0-51)
`edlevel`	educational level (categorical: 1-4)
`age`	age: 25-64
`outwork`	out of work=1; 0=working
`female`	female=1; 0=male
`married`	married=1; 0=not married
`kids`	have children=1; no children=0
`hhninc`	household yearly income in marks (in Marks)
`educ`	years of formal education (7-18)
`self`	self-employed=1; not self employed=0
`edlevel1`	(1/0) not high school graduate
`edlevel2`	(1/0) high school graduate
`edlevel3`	(1/0) university/college
`edlevel4`	(1/0) graduate school

Details

rwm1984 is saved as a data frame. Count models typically use docvis as response variable. 0 counts are included

Source

German Health Reform Registry, year=1984, in Hilbe and Greene (2007)

References

Hilbe, Joseph, M (2014), Modeling Count Data, Cambridge University Press Hilbe, Joseph M (2011), Negative Binomial Regression, Cambridge University Press Hilbe, J. and W. Greene (2008). Count Response Regression Models, in ed. C.R. Rao, J.P Miller, and D.C. Rao, Epidemiology and Medical Statistics, Elsevier Handbook of Statistics Series. London, UK: Elsevier.

Examples

```
library(MASS)
library(msme)
data(rwm1984)

glmrp <- glm(docvis ~ outwork + female + age + factor(edlevel), family=poisson, data=rwm1984)
summary(glmrp)
exp(coef(glmrp))

summary(nb2 <- nbinomial(docvis ~ outwork + female + age + factor(edlevel), data=rwm1984))
exp(coef(nb2))

summary(glmrnb <- glm.nb(docvis ~ outwork + female + age + factor(edlevel), data=rwm1984))
exp(coef(glmrnb))
```

`rwm5yr` *rwm5yr*

Description

German health registry for the years 1984-1988. Health information for years immediately prior to health reform.

Usage

```
data(rwm5yr)
```

Format

A data frame with 19,609 observations on the following 17 variables.

`id` patient ID (1=7028)

`docvis` number of visits to doctor during year (0-121)

`hospvis` number of days in hospital during year (0-51)

`year` year; (categorical: 1984, 1985, 1986, 1987, 1988)

`edlevel` educational level (categorical: 1-4)

`age`	age: 25-64
`outwork`	out of work=1; 0=working
`female`	female=1; 0=male
`married`	married=1; 0=not married
`kids`	have children=1; no children=0
`hhninc`	household yearly income in marks (in Marks)
`educ`	years of formal education (7-18)
`self`	self-employed=1; not self employed=0
`edlevel1`	(1/0) not high school graduate
`edlevel2`	(1/0) high school graduate
`edlevel3`	(1/0) university/college
`edlevel4`	(1/0) graduate school

Details

rwm5yr is saved as a data frame. Count models typically use docvis as response variable. 0 counts are included

Source

German Health Reform Registry, years pre-reform 1984-1988, in Hilbe and Greene (2007)

References

Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press Hilbe, Joseph M (2011), Negative Binomial Regression, Cambridge University Press Hilbe, J. and W. Greene (2008). Count Response Regression Models, in ed. C.R. Rao, J.P Miller, and D.C. Rao, Epidemiology and Medical Statistics, Elsevier Handbook of Statistics Series. London, UK: Elsevier.

Examples

```
library(MASS)
data(rwm5yr)

glmrp <- glm(docvis ~ outwork + female + age + factor(edlevel), family=poisson, data=rwm5yr)
summary(glmrp)
exp(coef(glmrp))

## Not run:
library(msme)
nb2 <- nbinomial(docvis ~ outwork + female + age + factor(edlevel), data=rwm5yr)
summary(nb2)
exp(coef(nb2))

glmrnb <- glm.nb(docvis ~ outwork + female + age + factor(edlevel), data=rwm5yr)
summary(glmrnb)
exp(coef(glmrnb))

## End(Not run)
```

`ships` *ships*

Description

Data set used in McCullagh & Nelder (1989), Hardin & Hilbe (2003), and other sources. The data contains values on the number of reported accidents for ships belonging to a company over a given time period. When a ship was constructed is also recorded.

Usage

```
data(ships)
```

Format

A data frame with 40 observations on the following 7 variables.

`accident`	number of shipping accidents
`op`	1=ship operated 1975-1979;0=1965-74
`co.65.69`	ship was in construction 1965-1969 (1/0)
`co.70.74`	ship was in construction 1970-1974 (1/0)
`co.75.79`	ship was in construction 1975-1979 (1/0)
`service`	months in service
`ship`	ship identification : 1-5

Details

ships is saved as a data frame. Count models use accident as the response variable, with log(service) as the offset. ship can be used as a panel identifier.

Source

McCullagh and Nelder, 1989.

References

Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC Hardin, JW and JM Hilbe (2001, 2007), Generalized Linear Models and Extensions, Stata Press McCullagh, P.A, and J. Nelder (1989), Generalized Linear Models, Chapman & Hall

Examples

```
data(ships)
glmshp <- glm(accident ~ op + co.70.74 + co.75.79 + offset(log(service)),
              family=poisson, data=ships)
summary(glmshp)
exp(coef(glmshp))
library(MASS)
glmshnb <- glm.nb(accident ~ op + co.70.74 + co.75.79 + offset(log(service)),
                   data=ships)
summary(glmshnb)
exp(coef(glmshnb))
## Not run:
library(gee)
shipgee <- gee(accident ~ op + co.70.74 + co.75.79 + offset(log(service)),
              data=ships, family=poisson, corstr="exchangeable", id=ship)
summary(shipgee)

## End(Not run)
```

`smoking` *smoking*

Description

A simple data set with only 6 observations.

Usage

```
data(smoking)
```

Format

A data frame with 6 observations on the following 4 variables.

`sbp` systolic blood pressure of subject

`male` 1=male; 0=female

`smoker` 1=hist of smoking; 0= no hist of smoking

`age` age of subject

Details

smoking is saved as a data frame.

Source

none

References

Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press

Examples

```
sbp    <- c(131,132,122,119,123,115)
male   <- c(1,1,1,0,0,0)
smoker <- c(1,1,0,0,1,0)
age    <- c(34,36,30,32,26,23)
summary(reg1 <- lm(sbp~ male+smoker+age))
```

`titanic` *titanic*

Description

The data is an observation-based version of the 1912 Titanic passenger survival log,

Usage

```
data(titanic)
```

Format

A data frame with 1316 observations on the following 4 variables.

`class` a factor with levels `1st class` `2nd class` `3rd class` `crew`

`age` a factor with levels `child` `adults`

`sex` a factor with levels `women` `man`

`survived` a factor with levels `no` `yes`

Details

titanic is saved as a data frame. Used to assess risk ratios

Source

Found in many other texts

References

Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC

Examples

```
data(titanic)
titanic$survival <- titanic$survived == "yes"
glmlr <- glm(survival ~ age + sex + factor(class), family=binomial, data=titanic)
summary(glmlr)
```

`titanicgrp` *titanicgrp*

Description

The data is an grouped version of the 1912 Titanic passenger survival log,

Usage

```
data(titanicgrp)
```

Format

A data frame with 12 observations on the following 5 variables.

`survive`	number of passengers who survived
`cases`	number of passengers with same pattern of covariates
`age`	1=adult; 0=child
`sex`	1=Male; 0=female
`class`	ticket class 1= 1st class; 2= second class; 3= third class

Details

titanicgrp is saved as a data frame. Used to assess risk ratios

Source

Found in many other texts

References

Hilbe, Joseph M (2014), Modeling Count Data, Cambridge University Press Hilbe, Joseph M (2007, 2011), Negative Binomial Regression, Cambridge University Press Hilbe, Joseph M (2009), Logistic Regression Models, Chapman & Hall/CRC

Examples

```
library(MASS)
library(msme)
data(titanicgrp)
glmlr <- glm(survive ~ age + sex + factor(class) + offset(log(cases)),
             family=poisson, data=titanicgrp)
summary(glmlr)
exp(coef(glmlr))

lcases <- titanicgrp$cases
nb2o <- nbinomial(survive ~ age + sex + factor(class),
                                        formula2 =~ age + sex,
                                        offset = lcases,
                                        mean.link="log",
                                        scale.link="log_s",
                                        data=titanicgrp)
summary(nb2o)
exp(coef(nb2o))
```

Index