Package ‘mistwosources’

November 4, 2016

Type  Package
Title  Probit models with two misclassified responses
Version 1.0
Date  2016-10-07
Author  Scott J. Cook, Betsabe Blas, Raymond J. Carroll and Samiran Sinha
Maintainer  Betsabe Blas <betsabe.bg@gmail.com>
Description  Obtains the maximum likelihood estimates of the regression parameters of the probit model with misclassified responses from two sources.
License  GPL-2
NeedsCompilation  no

R topics documented:

mistwosources-package ........................................ 1
misclass_a1xz_a2xz ............................................. 2
misclass_a1_a2 .................................................. 4
misclass_phi1 ................................................... 6

Index

mistwosources-package  Probit models with two misclassified responses

Description

Obtains the maximum likelihood estimates of the regression parameters of the probit model with misclassified responses from two sources.

Details

This package contains three functions that are all used to fit the probit model when the response variable is subject to misclassification, and the corresponding methodology is proposed in the manuscript entitled, "Two wrongs make a right: addressing underreporting in binary data from multiple sources". The function "misclass_a1_a2" is used when the misclassification probabilities from the two sources are constant, the function "misclass_a1xz_a2xz" is used when the misclassification probabilities from the two sources depend on covariates. The third function "misclass_phi1" is used to fit a probit model when two misclassified responses are combined into one, and the corresponding misclassification probability depends on covariates.
Author(s)
Scott J. Cook, Betsabe Blas, Raymond J. Carroll and Samiran Sinha
Maintainer: Betsabe Blas <betsabe.bg@gmail.com>

References

misclass_a1xz_a2xz  Estimating probit model while misclassified mechanisms depend on covariates

Description
The function produces the parameter estimate of a probit model for the response variable given a set of covariates. However, instead of the true response, the observed data contains two variables that are potentially misclassified version of the true response (for details, see Cook et al). In particular, here we assume that the misclassification mechanism depends on covariates.

Usage
misclass_a1xz_a2xz(y1 = y1, y2 = y2, x, z1, z2, print.summary = TRUE)

Arguments
y1  an n × 1 vector of the response from source 1
y2  an n × 1 vector of the response from source 2
x   an n × p matrix of exogeneous variable
z1  an n × a matrix of exogeneous variables for source 1
z2  an n × b matrix of exogeneous variables for source 2
print.summary  If print.summary=T, prints a summary of the final optim function estimates.
               Default: print.summary=T

Details
Let $y_1$ and $y_2$ be the two reported responses that are misclassified version of the underlying true response $y_T$. Both $y_1$ and $y_2$ are binary, and assume that $y_1$ depends on covariates $x$ and $z_1$, $y_2$ depends on covariates $x$ and $z_2$. Let $z = (z_1, z_2)$. The probability model for $y_T$ is $Pr(y_T = 1|x) = \Phi(x^T \beta)$, here we are interested in estimating the regression coefficient $\beta$. The misclassification probabilities are $\alpha_1(x, z_1) = Pr(y_1 = 0|y_T = 1, x, z_1) = \Phi((x^T, z_1^T) \eta_1)$ and $\alpha_2(x, z_2) = Pr(y_2 = 0|y_T = 1, x, z_2) = \Phi((x^T, z_2^T) \eta_2)$, where $\alpha_1$ and $\alpha_2$ depend on covariates $x$, $z_1$, and $z_2$. We assume $Pr(y_1 = 0|y_T = 0, x, z_1) = Pr(y_2 = 0|y_T = 0, x, z_2) = 1$ for all $x$ and $z$. Here, we can write the probabilities $Pr(y_1 = r, y_2 = s|x, z)$, $r, s = 0, 1$, in terms of $\alpha_1(x, z_1)$ and $\alpha_2(x, z_2)$, and $Pr(y_T = 1|x)$, then we can write the log-likelihood function for the probit model with misclassification. The log-likelihood function is maximized using the optim function. The outputs are described below.
Value

- **estimate**: coefficient estimates
- **value**: the maximized value of the log-likelihood
- **convergence**: integer codes from the `optim` function. An integer code 0 indicates successful convergence
- **message**: a character string giving any additional information returned by the `optim` function, or NULL
- **hessian**: a symmetric matrix giving an estimate of the Hessian at the solution found, and the square root of diagonal elements are the standard error of the parameter estimates.

Author(s)

Scott J. Cook, Betsabe Blas, Raymond J. Carroll and Samiran Sinha

Maintainer: Betsabe Blas <betsabe.bg@gmail.com>

References


Examples

```r
## Case 1: vector covariates
## We will generate dataset by setting x, z1 and z2 from N(0,1)

n <- 1000  ## sample size

## The independent variables are given by

x <- rnorm(n)
z1 <- rnorm(n)z2 <- rnorm(n)

## Generate the true binary response y_true, with covariate x

beta1_true <- 1
beta0_true <- 1

lm <- beta0_true + beta1_true*x
pr.probit <- pnorm(lm)
y_true <- rbinom(n, 1, pr.probit)

## Generate the misclassified variable y1 from source 1.

delta2 <- 1
beta2 <- 1

lm_a1 <- -0.7 + delta2*x + beta2*z1
pr_a1.probit <- pnorm(lm_a1)
alpha1.probit <- rbinom(n, 1, pr_a1.probit)
y1 <- y_true * (1 - alpha1.probit)

## Generate the misclassified variable y1 from source 2.

delta3 <- 1
beta3 <- 1

lm_a2 <- -1.4 + delta3*x + beta3*z2
pr_a2.probit <- pnorm(lm_a2)
alpha2.probit <- rbinom(n, 1, pr_a2.probit)
y2 <- y_true * (1 - alpha2.probit)

## End of data generation
```
## Now, we will fit the function misclass_a1xz_a2xz

misclass_a1xz_a2xz(y1,y2,x=x,z1=z1,z2=z2)

## Case 2: matrix covariates

## We will generate dataset by setting x, z1 and z2 from N(0,1)

n <- 1000  ## sample size

## The independent variables are given by

x1 <- rnorm(n)
x2 <- rnorm(n)
z1.1 <- rnorm(n)
z1.2 <- rnorm(n)
z2.1 <- rnorm(n)
z2.2 <- rnorm(n)

## Generate the true binary response y_true, with covariates x1 and x2

beta1_true <- 1
beta2_true <- 1
beta0_true <- -1

lm <- beta0_true + beta1_true*x1 + beta2_true*x2
pr.probit <- pnorm(lm)
y_true <- rbinom(n, 1, pr.probit)

## Generate the misclassified variable y1 from source 1.

delta21 <- 1
delta22 <- 1
misclass_phi1

beta2.1 <- 1
beta2.2 <- 1

lm_a1 <- -0.7 + delta21*x1 + delta22*x2 + beta2.1*z1.1 + beta2.2*z1.2
pr_a1.probit <- pnorm(lm_a1)
alpha1.probit <- rbinom(n, 1, pr_a1.probit)
y1 <- y_true*(1-alpha1.probit)

## Generate the misclassified variable y1 from source 2.

delta31 <- 1
delta32 <- 1
misclass_phi2

beta3.1 <- 1
beta3.2 <- 1

lm_a2 <- -1.4 + delta31*x1 + delta32*x2 + beta3.1*z2.1 + beta3.2*z2.2
pr_a2.probit <- pnorm(lm_a2)
alpha2.probit <- rbinom(n, 1, pr_a2.probit)
y2 <- y_true*(1-alpha2.probit)

## End of data generation

x <- cbind(x1, x2)
z1 <- cbind(z1.1, z1.2)
z2 <- cbind(z2.1, z2.2)

## Now, we will fit the function misclass_a1xz_a2xz

misclass_a1xz_a2xz(y1, y2, x=x, z1=z1, z2=z2)

---

**Description**

The function produces parameter estimates of a probit model for the response variable given a set of covariates. However, instead of the true response, the observed data contains two variables that
are potentially misclassified version of the true response (for details, see Cook et al). In particular, here we assume that the misclassification mechanism is independent of covariates.

Usage

misclass_a1_a2(y1 = y1, y2 = y2, x = x, a1 = 0.001, a2 = 0.001, bmat = NULL, print.summary = TRUE)

Arguments

y1 an n × 1 vector of the response from source 1
y2 an n × 1 vector of the response from source 2
x an n × p matrix of the independent variables
a1 starting value for α₁, the default is a1=0.001
a2 starting value for α₂, the default is a2=0.001
bmat starting values for the coefficient of x, β, the default is bmat=NULL, uses standard probit values
print.summary if print.summary=T, prints a summary of the optim function estimates. Default: Default: print.summary=T

Details

Let \( y_1 \) and \( y_2 \) be the two reported responses that are misclassified versions of the true response \( y_T \). Both \( y_1 \) and \( y_2 \) are binary. Let \( z = (z_1, z_2) \). The misclassification probabilities \( \alpha_1 = Pr(y_1 = 0|y_T = 1) \) and \( \alpha_2 = Pr(y_2 = 0|y_T = 1) \) are unknown constants. We assume \( Pr(y_1 = 0|y_T = 1) = Pr(y_2 = 0|y_T = 0) = 0 \). The probability of observing the correctly classified value of the dependent variable is \( Pr(y_T = 1|x) = \Phi(x^T \beta) \), here we are interested in estimating the regression coefficient \( \beta \). We can write the probabilities \( Pr(y_1 = r, y_2 = s|x, z) \), \( r, s = 0, 1 \), in terms of \( \alpha_1 \) and \( \alpha_2 \), and \( Pr(y_T = 1|x) \), then we can write the log-likelihood function for the probit model with misclassification. The log-likelihood function is maximized using the optim function. In order to avoid convergence issues, the function misclass_a1_a2 estimates \( \alpha_1^* \) and \( \alpha_2^* \), where \( \alpha_1 = \Phi(\alpha_1^*) \) and \( \alpha_2 = \Phi(\alpha_2^*) \). The variance-covariance matrix is calculated using the hessian option in the optim function. The vector estimate contains the estimated values of \( \beta, \alpha_1^* = \Phi^{-1}(\alpha_1) = qnorm(\alpha_1) \), and \( \alpha_2^* = \Phi^{-1}(\alpha_2) = qnorm(\alpha_2) \). Similarly, stderr and vmat report the standard error estimates and the full covariance matrix for \( (\beta, \alpha_1^*, \alpha_2^*) \). The starting value for \( \beta \) can be changed using bmat option in misclass_a1_a2.

Value

| estimate | standard errors for the regression coefficient \( \beta \) |
| stderr   | standard errors for the estimate                   |
| vmat     | full covariance matrix                            |
| convergence | 0 indicates successful completion and 1 indicates that the iteration limit maxit in the optim function had been reached |
| value    | the maximized value of the log-likelihood          |
| a1       | the estimate for \( \alpha_1 \)                    |
| a2       | the estimate for \( \alpha_2 \)                    |
| smat     | standard errors for \( a1 \) and \( a2 \)           |
misclass_phi1

Fitting a probit model while two misclassified responses are combined into one

Description

The function produces parameter estimate of a probit model for the response variable given covariates. However, instead of the true response \( y_T \), the observed data contain two variables \( y_1 \) and \( y_2 \) that are potentially misclassified version of the true response (for details, see Cook et al.), which are used to define a combined variable \( y_{sum} = I(y_1 + y_2 \geq 0) \), where \( I \) is the indicator function. In particular, here we assume that the misclassification mechanism depends on covariates.
Usage

misclass_phi1(y1 = y1, y2 = y2, x = x, z1 = z1, z2 = z2, bmat = NULL, print.summary = TRUE)

Arguments

y1 an n × 1 vector of the response from source 1
y2 an n × 1 vector of the response from source 2
x an n × p matrix of the independent variables
z1 an n × a matrix of exogeneous variables from source 1
z2 an n × b matrix of exogeneous variables from source 2
bmat starting values for the parameters β and η, the default is bmat=NULL, that uses standard probit values of \( y_{sum} \sim x \) and \( y_{sum} \sim x + z_1 + z_2 \), respectively
print.summary If print.summary=T, prints a summary of the final optim function estimates. Default: print.summary=T

Details

Let \( y_1 \) and \( y_2 \) be the two reported responses that are misclassified versions of the true response \( y_T \). Both \( y_1 \) and \( y_2 \) are binary, and assume that \( y_1 \) depends on covariates \( x \) and \( z_1 \), \( y_2 \) depends on covariates \( x \) and \( z_2 \). Let \( z = (z_1, z_2) \). The probability model for \( y_T \) is \( Pr(y_T = 1|x) = \Phi(x^T \beta) \).
Here we are interested in estimating the regression coefficient \( \beta \). The misclasification probability is \( \gamma(x, z_1, z_2) = Pr(y_{sum} = 0|y_T = 1, x, z_1, z_2) = \Phi\{(x^T, z_1^T, z_2^T)\eta\} \), \( \gamma(x, z_1, z_2) \) depends on the covariate \( x \) and \( z \). We assume \( Pr(y_{sum} = 0|y_T = 0, x, z_1, z_2) = 1 \) for all \( x \) and \( z \). Here, we can write the probability \( Pr(y_{sum} = 1|x, z) \) in terms of \( \gamma(x, z_1, z_2) \) and \( Pr(y_T = 1|x, z) \), then we can write the log-likelihood function for the probit model with misclassification. The log-likelihood function is maximized using the optim function. The outputs are described below.

Value

estimate coefficient estimates
stderr standard errors for estimate
vmat full covariance matrix
convergence 0 indicates successful completion and 1 indicates that the iteration limit maxit in the optim function had been reached
value the maximized value of the log-likelihood

Author(s)

Scott J. Cook, Betsabe Blas, Raymond J. Carroll and Samiran Sinha
Maintainer: Betsabe Blas <betsabe.bg@gmail.com>

References

Examples

## Case 1: vector covariates

### We will generate dataset by setting x=(x1, x2), z1 and z2 from N(0,1)

\( n < 1000 \) ## sample size

### The independent variables are given by

x1 <- rnorm(n)

z1 <- rnorm(n)

z2 <- rnorm(n)

## Generate the true binary response \( y_{true} \), with covariates x1 and x2

\( \beta_{1\_true} <- 1 \)

beta0_true <- -1

lm <- beta0_true + beta1_true*x1

pr.probit <- pnorm(lm)

y_true <- rbinom(n, 1, pr.probit)

## Generate the misclassified variable \( y_{1} \) from source 1.

\( \delta_{21} <- 1 \)

misclass_phi1

beta2 <- 1

lm_a1 <- -0.7 + delta21*x1 + beta2*z1

pr_a1.probit <- pnorm(lm_a1)

alpha1.probit <- rbinom(n, 1, pr_a1.probit)

y1 <- y_true*(1-alpha1.probit)

## Generate the misclassified variable \( y_{1} \) from source 2.

\( \delta_{31} <- 1 \)

misclass_phi1

beta3 <- 1

lm_a2 <- -1.4 + delta31*x1 + beta3*z2

pr_a2.probit <- pnorm(lm_a2)

alpha2.probit <- rbinom(n, 1, pr_a2.probit)

y2 <- y_true*(1-alpha2.probit)

## Former variable \( y_{sum}=I(y_{1}+y_{2}\geq 1) \)

y_both <- as.numeric(y1+y2>=1)

misclass_phi1(y1,y2,x1,z1,z2)

## Case 2: matrix covariates

### We will generate dataset by setting x=(x1, x2), z1=(z1.1,z1.2) and z2=(z2.1,z2.2)

\( n < 1000 \) ## sample size

### The independent variables are given by

x1 <- rnorm(n)

x2 <- rnorm(n)

z1.1 <- rnorm(n)

z1.2 <- rnorm(n)

z2.1 <- rnorm(n)

z2.2 <- rnorm(n)

## Generate the true binary response \( y_{true} \), with covariates x1 and x2

\( \beta_{1\_true} <- 1 \)

beta2_true <- 1

beta0_true <- -1

lm <- beta0_true + beta1_true*x1+beta2_true*x2

pr.probit <- pnorm(lm)

y_true <- rbinom(n, 1, pr.probit)

## Generate the misclassified variable \( y_{1} \) from source 1.

\( \delta_{21} <- 1 \)

misclass_phi1

beta2.1 <- 1

\( \delta_{22} <- 1 \)

misclass_phi1
misclass_phi1

beta2.2 <- 1
lm_a1 <- -0.7 + delta21*x1 + delta22*x2 + beta2.1*z1.1 + beta2.2*z1.2
pr_a1.probit <- pnorm(lm_a1)
alpha1.probit <- rbinom(n,1,pr_a1.probit)
y1 <- y_true*alpha1.probit
## Generate the misclassified variable y1 from source 2.
delta31 <- 1
delta32 <- 1
beta3.1 <- 1
beta3.2 <- 1
lm_a2 <- -1.4 + delta31*x1 + delta32*x2 + beta3.1*z2.1 + beta3.2*z2.2
pr_a2.probit <- pnorm(lm_a2)
alpha2.probit <- rbinom(n,1,pr_a2.probit)
y2 <- y_true*alpha2.probit
## Former variable y_sum=I(y1+y2>=1)
y_both <- as.numeric(y1+y2>=1)
x <- cbind(x1,x2)
z1 <- cbind(z1.1,z1.2)
z2 <- cbind(z2.1,z2.2)
misclass_phi1(y1,y2,x,z1,z2)
Index

*Topic misclassification
  misclass_a1_a2, 4
  misclass_a1xz_a2xz, 2
  misclass_phi1, 6

*Topic package
  mistwosources-package, 1

*Topic probit model
  misclass_a1_a2, 4
  misclass_a1xz_a2xz, 2
  misclass_phi1, 6

  misclass_a1_a2, 4
  misclass_a1xz_a2xz, 2
  misclass_phi1, 6
  mistwosources (mistwosources-package), 1
  mistwosources-package, 1