Probit Regression: nagdmc_probit_reg

Purpose

nagdmc_probit_reg computes a probit regression model with p parameters and is a simplified interface to **nagdmc_binomial_reg** using a probit link function.

Declaration

```
#include <nagdmc.h>
void nagdmc_probit_reg(long rec1, long nvar, long nrec, long dblk, double data[],
                       long nxvar, long xvar[], long yvar, double ycut, long bdvar,
                       double *dev, long *df, double b[], double se[], double cov[],
                       double model[], int *info);
```

Parameters

1: **rec1** – long *Input*

On entry: the index in the data of the first data record used in the analysis.

Constraint: **rec1** ≥ 0.

2: **nvar** – long *Input*

On entry: the number of variables in the data.

Constraint: **nvar** > 1.

3: **nrec** – long *Input*

On entry: the number of consecutive records, beginning at **rec1**, used in the analysis.

Constraint: **nrec** > 1.

4: **dblk** – long *Input*

On entry: the total number of records in the data block.

Constraint: **dblk** $\geq$ **rec1** $+$ **nrec**.

5: **data**[**dblk** $*$ **nvar**] – double *Input*

On entry: the data values for the jth variable (for $j = 0, 1, \ldots,$ **nvar**-1) are stored in **data**[$i*$**nvar**$+j$], for $i = 0, 1, \ldots,$ **dblk** $- 1$.

6: **nxvar** – long *Input*

On entry: the number of independent variables. If **nxvar** $= 0$ then all variables in the data, excluding **yvar** and, if ≥ 0, **bdvar**, are treated as independent variables.

Constraint: $0 \leq$ **nxvar** $<$ **nvar**.

7: **xvar**[**nxvar**] – long *Input*

On entry: the indices indicating the position in **data** in which values of the independent variables are stored. If **nxvar** $= 0$ then **xvar** must be 0, and the indices of independent variables are given by $j = 0, 1, \ldots,$ **nvar** $- 1$; $j \neq$ **yvar** and $j \neq$ **bdvar**.

Constraints: if **nxvar** > 0, $0 \leq$ **xvar**[i] $<$ **nvar**, for $i = 0, 1, \ldots,$ **nxvar** $- 1$; otherwise **xvar** must be 0.

8: **yvar** – long *Input*

On entry: the index in **data** in which values of the dependent variable are stored.

Constraints: $0 \leq$ **yvar** $<$ **nvar**; if **nxvar** > 0, **yvar** $\neq$ **xvar**[i], for $i = 0, 1, \ldots,$ **nxvar** $- 1$.

9: **ycut** – long *Input*

On entry: if **ycut** $\neq 0$, the y-variable is transformed so that values $<$ **ycut** are set to zero and values $\geq$ **ycut** are set to one.

10: **bdvar** – `long` *Input*

On entry: an index indicating the position in **data** in which the binomial denominator is stored. If **bdvar** $= -1$ a default value of one is used for all observations.

Constraint: $-1 \leq$ **bdvar** $<$ **nvar**.

11: **dev** – `double` *Output*

On exit: the deviance from the fitted model.

12: **df** – `long *` *Output*

On exit: the degrees of freedom for the deviance.

13: **b**[p] – `double` *Output*

On exit: the parameter estimates. **b**[0] is the mean parameter. **b**[i] is the coefficient of the ith variable included in the model, for $i = 1, 2, \ldots, p-1$. If **nxvar** > 0 then the order the independent variables are added to the model is defined by **xvar**, otherwise the order is defined by indices in the data.

14: **se**[p] – `double` *Output*

On exit: the standard errors of the parameters in **b**.

15: **cov**[$p * (p+1)/2$] – `double` *Output*

On exit: the first $p * (p+1)/2$ elements of **cov** contain the upper triangular part of the variance-covariance matrix of the p parameters in **b**. They are stored packed by column, i.e., the covariance between the parameter estimate given in **b**[i] and the parameter estimate given in **b**[j], $j \geq i$, is stored in **cov**[$j(j+1)/2 + i$], for $i = 0, 1, \ldots, p-1$ and $j = i, i+1, \ldots, p-1$.

16: **model**[$(3 * p * (p+1))/2 +$ **nvar** $+ 14$] – `double` *Output*

On exit: if not 0, information on the fitted model for use in the functions described in 'See Also'.

17: **info** – `int *` *Output*

On exit: **info** gives information on the success of the function call:

-4: a model value has reached a boundary.

0: the function successfully completed its task.

i; $i = 1, 2, \ldots, 10$: the specification of the ith formal parameter was incorrect.

42: invalid value for response variable.

43: invalid value for binomial denominator.

45: model has not converged.

57: there are no degrees of freedom for the error estimates.

58: the fit is exact, no error estimates.

59: more variables than observations.

98: there is an underlying computational problem (this is an unlikely error exit).

99: the function failed to allocate enough memory.

Notation

nrec	the number of observations, n.
nxvar	the number of independent variables, $p-1$.
xvar	the independent variables, X, excluding the mean.
yvar	the dependent variable, y.
bdvar	if **bdvar** ≥ 0, **bdvar** is the index in the data that defines the binomial denominator, t.
b	the parameter estimates, $\hat{\beta}$.

Description

See the description for **nagdmc_binomial_reg**.

References and Further Reading

Cook R D and Weisberg S (1982) *Residuals and Influence in Regression* Chapman and Hall.

Cox D R (1983) *Analysis of Binary Data* Chapman and Hall

McCullagh P and Nelder J A (1983) *Generalized Linear Models* Chapman and Hall.

See Also

nagdmc_binomial_reg	generalized linear model with binomial errors.
nagdmc_extr_reg	computes fitted values, residuals and leverages for a regression.
nagdmc_probit_reg	simplified version of **nagdmc_binomial_reg** using a logit link and a restricted set of parameters.
nagdmc_predict_reg	computes predictions given a fitted regression model.
probit_reg_ex.c	the example calling program.