The Probit & Logit Models

Econometrics II

Ricardo Mora

Department of Economics
Universidad Carlos III de Madrid
Máster Universitario en Desarrollo y Crecimiento Económico

Notes

Outline

Notes

Choosing Among a few Alternatives

- Set up: An agent chooses among several alternatives:
  - labor economics: participation, union membership, ...
  - demographics: marriage, divorce, # of children,...
  - industrial organization: plant building, new product,...
  - regional economics: means of transport,....
- We are going to model a choice of two alternatives (not difficult to generalize...)

The value of each alternative depends on many factors

- $U_0 = \beta_0 x_0 + \varepsilon_0$
- $U_1 = \beta_1 x_1 + \varepsilon_1$

- $\varepsilon_0, \varepsilon_1$ are effects on utility on factors UNOBSERVED TO ECONOMETRICIAN

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Choosing Among a few Alternatives

- Set up: An agent chooses among several alternatives:
  - labor economics: participation, union membership, ...
  - demographics: marriage, divorce, # of children,...
  - industrial organization: plant building, new product,...
  - regional economics: means of transport,....
- We are going to model a choice of two alternatives (not difficult to generalize...)

The value of each alternative depends on many factors

- $U_0 = \beta_0 x_0 + \varepsilon_0$
- $U_1 = \beta_1 x_1 + \varepsilon_1$

- $\varepsilon_0, \varepsilon_1$ are effects on utility on factors UNOBSERVED TO ECONOMETRICIAN

Ricardo Mora The Probit Model

Notes

Choosing Among a few Alternatives

- Set up: An agent chooses among several alternatives:
  - labor economics: participation, union membership, ...
  - demographics: marriage, divorce, # of children,...
  - industrial organization: plant building, new product,...
  - regional economics: means of transport,....
- We are going to model a choice of two alternatives (not difficult to generalize...)

The value of each alternative depends on many factors

- $U_0 = \beta_0 x_0 + \varepsilon_0$
- $U_1 = \beta_1 x_1 + \varepsilon_1$

- $\varepsilon_0, \varepsilon_1$ are effects on utility on factors UNOBSERVED TO ECONOMETRICIAN

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Choosing Among a few Alternatives

- Set up: An agent chooses among several alternatives:
  - labor economics: participation, union membership, ...
  - demographics: marriage, divorce, # of children,...
  - industrial organization: plant building, new product,...
  - regional economics: means of transport,....
- We are going to model a choice of two alternatives (not difficult to generalize...)

The value of each alternative depends on many factors

- $U_0 = \beta_0 x_0 + \varepsilon_0$
- $U_1 = \beta_1 x_1 + \varepsilon_1$

- $\varepsilon_0, \varepsilon_1$ are effects on utility on factors UNOBSERVED TO ECONOMETRICIAN

Ricardo Mora The Probit Model

Notes

Choice Under the RUM

$$\text{If } \beta_1 x_1 - \beta_0 x_0 \geq \varepsilon_0 - \varepsilon_1 \text{ then } choice = 1$$

$$\text{If } \beta_1 x_1 - \beta_0 x_0 < \varepsilon_0 - \varepsilon_1 \text{ then } choice = 0$$

- agent chooses 1 if observed advantages of 1 outweight the unobserved net advantage of 0

- note that $\varepsilon = \varepsilon_0 - \varepsilon_1$ is defined by the data collection process, not by the decision process

Fundamental Assumption: $\varepsilon = \varepsilon_0 - \varepsilon_1 \sim F$

$$Pr\left(choice = 1\right) = Pr_F\left(\varepsilon \leq \beta_1 x_1 - \beta_0 x_0\right)$$

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Choice Under the RUM

$$\text{If } \beta_1 x_1 - \beta_0 x_0 \geq \varepsilon_0 - \varepsilon_1 \text{ then } choice = 1$$

$$\text{If } \beta_1 x_1 - \beta_0 x_0 < \varepsilon_0 - \varepsilon_1 \text{ then } choice = 0$$

- agent chooses 1 if observed advantages of 1 outweight the unobserved net advantage of 0

- note that $\varepsilon = \varepsilon_0 - \varepsilon_1$ is defined by the data collection process, not by the decision process

Fundamental Assumption: $\varepsilon = \varepsilon_0 - \varepsilon_1 \sim F$

$$Pr\left(choice = 1\right) = Pr_F\left(\varepsilon \leq \beta_1 x_1 - \beta_0 x_0\right)$$

Ricardo Mora The Probit Model

Notes

Choice Under the RUM

If $\beta_1 x_1 - \beta_0 x_0 \geq \varepsilon_0 - \varepsilon_1$ then $choice = 1$

If $\beta_1 x_1 - \beta_0 x_0 < \varepsilon_0 - \varepsilon_1$ then $choice = 0$

- agent chooses 1 if observed advantages of 1 outweight the unobserved net advantage of 0

- note that $\varepsilon = \varepsilon_0 - \varepsilon_1$ is defined by the data collection process, not by the decision process

Fundamental Assumption: $\varepsilon = \varepsilon_0 - \varepsilon_1 \sim F$

$$Pr\left(choice = 1\right) = Pr_F\left(\varepsilon \leq \beta_1 x_1 - \beta_0 x_0\right)$$

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Choice Under the RUM

If $\beta_1 x_1 - \beta_0 x_0 \geq \varepsilon_0 - \varepsilon_1$ then $choice = 1$

If $\beta_1 x_1 - \beta_0 x_0 < \varepsilon_0 - \varepsilon_1$ then $choice = 0$

- agent chooses 1 if observed advantages of 1 outweight the unobserved net advantage of 0

- note that $\varepsilon = \varepsilon_0 - \varepsilon_1$ is defined by the data collection process, not by the decision process

Fundamental Assumption: $\varepsilon = \varepsilon_0 - \varepsilon_1 \sim F$

$$Pr\left(choice = 1\right) = Pr_F\left(\varepsilon \leq \beta_1 x_1 - \beta_0 x_0\right)$$

Ricardo Mora The Probit Model

Notes

The Probit & Logit Models

Probit Assumption: $\varepsilon_1, \varepsilon_0 \sim N(0, \Sigma)$ so that $\varepsilon \sim N(0,1)$

- $Pr(choice = 1) = \Phi(\beta x)$ where Φ is the cdf of the standard normal
- this is called the Probit Model
- the vector of parameters β can be consistently estimated by ML

Logit Assumption: $\varepsilon_0 - \varepsilon_1 = \varepsilon \sim$Logistic

- $Pr(choice = 1) = \frac{exp(\beta x)}{1+exp(\beta x)}$
- Easy computation!

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

The Probit & Logit Models

Probit Assumption: $\varepsilon_1, \varepsilon_0 \sim N(0, \Sigma)$ so that $\varepsilon \sim N(0,1)$

- $Pr(choice = 1) = \Phi(\beta x)$ where Φ is the cdf of the standard normal
- this is called the Probit Model
- the vector of parameters β can be consistently estimated by ML

Logit Assumption: $\varepsilon_0 - \varepsilon_1 = \varepsilon \sim$Logistic

- $Pr(choice = 1) = \frac{exp(\beta x)}{1+exp(\beta x)}$
- Easy computation!

Ricardo Mora The Probit Model

Notes

The Probit & Logit Models

Probit Assumption: $\varepsilon_1, \varepsilon_0 \sim N(0, \Sigma)$ so that $\varepsilon \sim N(0,1)$

- $Pr(choice = 1) = \Phi(\beta x)$ where Φ is the cdf of the standard normal
- this is called the Probit Model
- the vector of parameters β can be consistently estimated by ML

Logit Assumption: $\varepsilon_0 - \varepsilon_1 = \varepsilon \sim$Logistic

- $Pr(choice = 1) = \frac{exp(\beta x)}{1+exp(\beta x)}$
- Easy computation!

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

The Probit & Logit Models

Probit Assumption: $\varepsilon_1, \varepsilon_0 \sim N(0, \Sigma)$ so that $\varepsilon \sim N(0,1)$

- $Pr(choice = 1) = \Phi(\beta x)$ where Φ is the cdf of the standard normal
- this is called the Probit Model
- the vector of parameters β can be consistently estimated by ML

Logit Assumption: $\varepsilon_0 - \varepsilon_1 = \varepsilon \sim$Logistic

- $Pr(choice = 1) = \frac{exp(\beta x)}{1+exp(\beta x)}$
- Easy computation!

Ricardo Mora The Probit Model

Notes

The Probit & Logit Models

Probit Assumption: $\varepsilon_1, \varepsilon_0 \sim N(0, \Sigma)$ so that $\varepsilon \sim N(0,1)$

- $Pr(choice = 1) = \Phi(\beta x)$ where Φ is the cdf of the standard normal
- this is called the Probit Model
- the vector of parameters β can be consistently estimated by ML

Logit Assumption: $\varepsilon_0 - \varepsilon_1 = \varepsilon \sim$Logistic

- $Pr(choice = 1) = \frac{exp(\beta x)}{1 + exp(\beta x)}$
- Easy computation!

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

The Probit & Logit Models

Probit Assumption: $\varepsilon_1, \varepsilon_0 \sim N(0, \Sigma)$ so that $\varepsilon \sim N(0,1)$

- $Pr(choice = 1) = \Phi(\beta x)$ where Φ is the cdf of the standard normal
- this is called the Probit Model
- the vector of parameters β can be consistently estimated by ML

Logit Assumption: $\varepsilon_0 - \varepsilon_1 = \varepsilon \sim$Logistic

- $Pr(choice = 1) = \frac{exp(\beta x)}{1 + exp(\beta x)}$
- Easy computation!

Ricardo Mora The Probit Model

Notes

Example: Marriage Decision

- Consider a sample of women who have a relation
- The econometrician only observes
  - $marry \begin{cases} = 1 \text{ if married} \\ = 0 \text{ otherwise} \end{cases}$,
  - x_m: factors affecting utility of being married
  - x_s: factors affecting utility of being single
  - note that something that affects the utility of being married will also affect the utility of being single, but not in the same way (for example, pregnant status)

Notes

Controls in the Marriage Decision

- $U_m = \beta_m^0 + \beta_m^{age} age + \beta_m^{preg} pregnant + \varepsilon_m$
- $U_s = \beta_s^0 + \beta_s^{age} age + \beta_s^{preg} pregnant + \varepsilon_s$

Probit Assumption: $\varepsilon_s - \varepsilon_m | x \sim N(0,1)$

- $Pr(marry = 1) = \Phi(\beta_0 + \beta_{age} age + \beta_{preg} pregnant)$
- $\beta_0 = \beta_m^0 - \beta_s^0$
- $\beta_{age} = \beta_m^{age} - \beta_s^{age}$
- $\beta_{preg} = \beta_m^{preg} - \beta_s^{preg}$
- $var(\varepsilon_s - \varepsilon_m) = \sigma_s^2 + \sigma_m^2 - 2\sigma_{s,m} = 1$

Notes

Controls in the Marriage Decision

- $U_m = \beta_m^0 + \beta_m^{age}\, age + \beta_m^{preg}\, pregnant + \varepsilon_m$
- $U_s = \beta_s^0 + \beta_s^{age}\, age + \beta_s^{preg}\, pregnant + \varepsilon_s$

Probit Assumption: $\varepsilon_s - \varepsilon_m | x \sim N(0,1)$

- $Pr(marry = 1) = \Phi(\beta_0 + \beta_{age}\, age + \beta_{preg}\, pregnant)$
- $\beta_0 = \beta_m^0 - \beta_s^0$
- $\beta_{age} = \beta_m^{age} - \beta_s^{age}$
- $\beta_{preg} = \beta_m^{preg} - \beta_s^{preg}$
- $var(\varepsilon_s - \varepsilon_m) = \sigma_s^2 + \sigma_m^2 - 2\sigma_{s,m} = 1$

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Controls in the Marriage Decision

- $U_m = \beta_m^0 + \beta_m^{age}\, age + \beta_m^{preg}\, pregnant + \varepsilon_m$
- $U_s = \beta_s^0 + \beta_s^{age}\, age + \beta_s^{preg}\, pregnant + \varepsilon_s$

Probit Assumption: $\varepsilon_s - \varepsilon_m | x \sim N(0,1)$

- $Pr(marry = 1) = \Phi(\beta_0 + \beta_{age}\, age + \beta_{preg}\, pregnant)$
- $\beta_0 = \beta_m^0 - \beta_s^0$
- $\beta_{age} = \beta_m^{age} - \beta_s^{age}$
- $\beta_{preg} = \beta_m^{preg} - \beta_s^{preg}$
- $var(\varepsilon_s - \varepsilon_m) = \sigma_s^2 + \sigma_m^2 - 2\sigma_{s,m} = 1$

Ricardo Mora The Probit Model

Notes

Controls in the Marriage Decision

- $U_m = \beta_m^0 + \beta_m^{age} age + \beta_m^{preg} pregnant + \varepsilon_m$
- $U_s = \beta_s^0 + \beta_s^{age} age + \beta_s^{preg} pregnant + \varepsilon_s$

Probit Assumption: $\varepsilon_s - \varepsilon_m | x \sim N(0,1)$

- $Pr(marry = 1) = \Phi(\beta_0 + \beta_{age} age + \beta_{preg} pregnant)$
- $\beta_0 = \beta_m^0 - \beta_s^0$
- $\beta_{age} = \beta_m^{age} - \beta_s^{age}$
- $\beta_{preg} = \beta_m^{preg} - \beta_s^{preg}$
- $var(\varepsilon_s - \varepsilon_m) = \sigma_s^2 + \sigma_m^2 - 2\sigma_{s,m} = 1$

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Controls in the Marriage Decision

- $U_m = \beta_m^0 + \beta_m^{age} age + \beta_m^{preg} pregnant + \varepsilon_m$
- $U_s = \beta_s^0 + \beta_s^{age} age + \beta_s^{preg} pregnant + \varepsilon_s$

Probit Assumption: $\varepsilon_s - \varepsilon_m | x \sim N(0,1)$

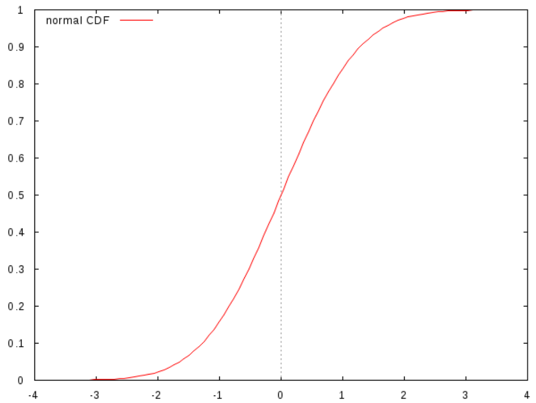
- $Pr(marry = 1) = \Phi(\beta_0 + \beta_{age} age + \beta_{preg} pregnant)$
- $\beta_0 = \beta_m^0 - \beta_s^0$
- $\beta_{age} = \beta_m^{age} - \beta_s^{age}$
- $\beta_{preg} = \beta_m^{preg} - \beta_s^{preg}$
- $var(\varepsilon_s - \varepsilon_m) = \sigma_s^2 + \sigma_m^2 - 2\sigma_{s,m} = 1$

Ricardo Mora The Probit Model

Notes

A Graphical Interpretation of the Probit Model

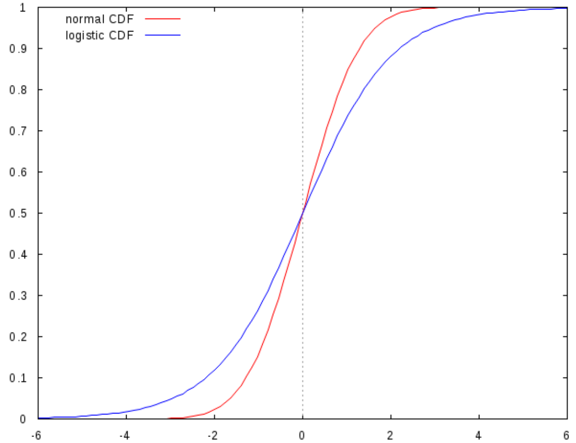
The probability to participate is a nonlinear function of the index function $\beta_0 + \beta_{age} age + \beta_{preg} pregnant$



Notes

Logit vs. Probit

Tails are thicker in the logit



Notes

Interpretation of the Slopes and Marginal Effects

when the control x_j appears in both utilities...

- only the net effect on the index function, $\beta_j = \beta^j_m - \beta^j_s$, is identified

normality (nonlinearity) assumption

- "net slope" β_j captures the marginal effect on index function βx of an increase of one unit of control x_j
- the marginal effect on the probability of marriage is more complex
  - if x_j is continuous, $\frac{\partial Pr(marry=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if x_j is discrete, $\Delta Pr(marry = 1) = \Phi(\beta x_1) - \Phi(\beta x_0)$ where x_1 is the vector of controls with the final value for x_j and x_0 is the vector of controls with the initial value for x_j

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Interpretation of the Slopes and Marginal Effects

when the control x_j appears in both utilities...

- only the net effect on the index function, $\beta_j = \beta^j_m - \beta^j_s$, is identified

normality (nonlinearity) assumption

- "net slope" β_j captures the marginal effect on index function βx of an increase of one unit of control x_j
- the marginal effect on the probability of marriage is more complex
  - if x_j is continuous, $\frac{\partial Pr(marry=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if x_j is discrete, $\Delta Pr(marry = 1) = \Phi(\beta x_1) - \Phi(\beta x_0)$ where x_1 is the vector of controls with the final value for x_j and x_0 is the vector of controls with the initial value for x_j

Ricardo Mora The Probit Model

Notes

Interpretation of the Slopes and Marginal Effects

when the control x_j appears in both utilities...

- only the net effect on the index function, $\beta_j = \beta_m^j - \beta_s^j$, is identified

normality (nonlinearity) assumption

- "net slope" β_j captures the marginal effect on index function βx of an increase of one unit of control x_j
- the marginal effect on the probability of marriage is more complex
  - if x_j is continuous, $\frac{\partial Pr(marry=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if x_j is discrete, $\Delta Pr(marry=1) = \Phi(\beta x_1) - \Phi(\beta x_0)$ where x_1 is the vector of controls with the final value for x_j and x_0 is the vector of controls with the initial value for x_j

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Interpretation of the Slopes and Marginal Effects

when the control x_j appears in both utilities...

- only the net effect on the index function, $\beta_j = \beta_m^j - \beta_s^j$, is identified

normality (nonlinearity) assumption

- "net slope" β_j captures the marginal effect on index function βx of an increase of one unit of control x_j
- the marginal effect on the probability of marriage is more complex
  - if x_j is continuous, $\frac{\partial Pr(marry=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if x_j is discrete, $\Delta Pr(marry=1) = \Phi(\beta x_1) - \Phi(\beta x_0)$ where x_1 is the vector of controls with the final value for x_j and x_0 is the vector of controls with the initial value for x_j

Ricardo Mora The Probit Model

Notes

Interpretation of the Slopes and Marginal Effects

when the control x_j appears in both utilities...

- only the net effect on the index function, $\beta_j = \beta_m^j - \beta_s^j$, is identified

normality (nonlinearity) assumption

- "net slope" β_j captures the marginal effect on index function βx of an increase of one unit of control x_j
- the marginal effect on the probability of marriage is more complex
  - if x_j is continuous, $\frac{\partial Pr(marry=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if x_j is discrete, $\Delta Pr(marry=1) = \Phi(\beta x_1) - \Phi(\beta x_0)$ where x_1 is the vector of controls with the final value for x_j and x_0 is the vector of controls with the initial value for x_j

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Interpretation of the Slopes and Marginal Effects

when the control x_j appears in both utilities...

- only the net effect on the index function, $\beta_j = \beta_m^j - \beta_s^j$, is identified

normality (nonlinearity) assumption

- "net slope" β_j captures the marginal effect on index function βx of an increase of one unit of control x_j
- the marginal effect on the probability of marriage is more complex
  - if x_j is continuous, $\frac{\partial Pr(marry=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if x_j is discrete, $\Delta Pr(marry=1) = \Phi(\beta x_1) - \Phi(\beta x_0)$ where x_1 is the vector of controls with the final value for x_j and x_0 is the vector of controls with the initial value for x_j

Ricardo Mora The Probit Model

Notes

Interpretation of the Slopes and Marginal Effects

when the control x_j appears in both utilities...

- only the net effect on the index function, $\beta_j = \beta_m^j - \beta_s^j$, is identified

normality (nonlinearity) assumption

- "net slope" β_j captures the marginal effect on index function βx of an increase of one unit of control x_j
- the marginal effect on the probability of marriage is more complex
  - if x_j is continuous, $\frac{\partial Pr(marry=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if x_j is discrete, $\Delta Pr(marry = 1) = \Phi(\beta x_1) - \Phi(\beta x_0)$ where x_1 is the vector of controls with the final value for x_j and x_0 is the vector of controls with the initial value for x_j

Notes

The Density in the Probit Model

Assumption: iid random sample

- let the true value be β_0
- then, under the Probit model

$$Pr(married|x) = \begin{cases} \Phi(\beta_0 x) \text{ if } marry = 1 \\ 1 - \Phi(\beta_0 x) \text{ if } marry = 0 \end{cases}$$

Notes

The Density in the Probit Model

Assumption: iid random sample

- let the true value be β_0
- then, under the Probit model

$$Pr(married\,|x) = \begin{cases} \Phi(\beta_0 x) \text{ if } marry = 1 \\ 1 - \Phi(\beta_0 x) \text{ if } marry = 0 \end{cases}$$

Notes

The Density in the Probit Model

Assumption: iid random sample

- let the true value be β_0
- then, under the Probit model

$$Pr(married\,|x) = \begin{cases} \Phi(\beta_0 x) \text{ if } marry = 1 \\ 1 - \Phi(\beta_0 x) \text{ if } marry = 0 \end{cases}$$

Notes

The Likelihood of an Observation

- the likelihood replaces in the density the true vector β_0 with any vector β
- then, the likelihood for individual i takes the form

$$L_i(\beta) = \begin{cases} \Phi(\beta x_i) \text{ if } marry_i = 1 \\ 1 - \Phi(\beta x_i) \text{ if } marry_i = 0 \end{cases}$$

- o, more conveniently,

$$L_i(\beta) = [\Phi(\beta x_i)]^{marry_i} [1 - \Phi(\beta x_i)]^{(1-marry_i)}$$

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

The Likelihood of an Observation

- the likelihood replaces in the density the true vector β_0 with any vector β
- then, the likelihood for individual i takes the form

$$L_i(\beta) = \begin{cases} \Phi(\beta x_i) \text{ if } marry_i = 1 \\ 1 - \Phi(\beta x_i) \text{ if } marry_i = 0 \end{cases}$$

- o, more conveniently,

$$L_i(\beta) = [\Phi(\beta x_i)]^{marry_i} [1 - \Phi(\beta x_i)]^{(1-marry_i)}$$

Ricardo Mora The Probit Model

Notes

The Likelihood of an Observation

- the likelihood replaces in the density the true vector β_0 with any vector β
- then, the likelihood for individual i takes the form

$$L_i(\beta) = \begin{cases} \Phi(\beta x_i) \text{ if } marry_i = 1 \\ 1 - \Phi(\beta x_i) \text{ if } marry_i = 0 \end{cases}$$

- o, more conveniently,

$$L_i(\beta) = [\Phi(\beta x_i)]^{marry_i} [1 - \Phi(\beta x_i)]^{(1 - marry_i)}$$

Notes

The Loglikelihood

- first, we take the logs

$$l_i(\beta) = marry_i \log(\Phi(\beta x_i)) + (1 - marry_i) \log(1 - \Phi(\beta x_i))$$

- then we compute the likelihood for the entire *iid* sample

$$l(\beta) = \sum_i l_i(\beta)$$

- hence

$$l(\beta) = \sum_i \{marry_i \log(\Phi(\beta x_i)) + (1 - marry_i) \log(1 - \Phi(\beta x_i))\}$$

Notes

The Loglikelihood

- first, we take the logs

$$l_i(\beta) = marry_i \log(\Phi(\beta x_i)) + (1 - marry_i)\log(1 - \Phi(\beta x_i))$$

- then we compute the likelihood for the entire *iid* sample

$$l(\beta) = \sum_i l_i(\beta)$$

- hence

$$l(\beta) = \sum_i \{marry_i \log(\Phi(\beta x_i)) + (1 - marry_i)\log(1 - \Phi(\beta x_i))\}$$

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

The Loglikelihood

- first, we take the logs

$$l_i(\beta) = marry_i \log(\Phi(\beta x_i)) + (1 - marry_i)\log(1 - \Phi(\beta x_i))$$

- then we compute the likelihood for the entire *iid* sample

$$l(\beta) = \sum_i l_i(\beta)$$

- hence

$$l(\beta) = \sum_i \{marry_i \log(\Phi(\beta x_i)) + (1 - marry_i)\log(1 - \Phi(\beta x_i))\}$$

Ricardo Mora The Probit Model

Notes

ML Estimation of Probit Model

Definition

- the MLE is the vector $\hat{\beta}^{ML}$ such that

$$\hat{\beta}^{ML} = \arg\max \sum_i \{marry_i \log\left(\Phi(\beta x_i)\right) + (1 - marry_i)\log\left(1 - \Phi(\beta x_i)\right)\}$$

- because of the nonlinear nature of the maximization problem, there are not explicit formulas for the probit ML estimates
- instead, numerical optimization is used, and, usually, only a few iterations are needed
- in Stata, several algorithms can be used

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

ML Estimation of Probit Model

Definition

- the MLE is the vector $\hat{\beta}^{ML}$ such that

$$\hat{\beta}^{ML} = \arg\max \sum_i \{marry_i \log\left(\Phi(\beta x_i)\right) + (1 - marry_i)\log\left(1 - \Phi(\beta x_i)\right)\}$$

- because of the nonlinear nature of the maximization problem, there are not explicit formulas for the probit ML estimates
- instead, numerical optimization is used, and, usually, only a few iterations are needed
- in Stata, several algorithms can be used

Ricardo Mora The Probit Model

Notes

ML Estimation of Probit Model

Definition

- the MLE is the vector $\hat{\beta}^{ML}$ such that

$$\hat{\beta}^{ML} = \arg\max \sum_i \{marry_i \log(\Phi(\beta x_i)) + (1 - marry_i) \log(1 - \Phi(\beta x_i))\}$$

- because of the nonlinear nature of the maximization problem, there are not explicit formulas for the probit ML estimates
- instead, numerical optimization is used, and, usually, only a few iterations are needed
- in Stata, several algorithms can be used

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

ML Estimation of Probit Model

Definition

- the MLE is the vector $\hat{\beta}^{ML}$ such that

$$\hat{\beta}^{ML} = \arg\max \sum_i \{marry_i \log(\Phi(\beta x_i)) + (1 - marry_i) \log(1 - \Phi(\beta x_i))\}$$

- because of the nonlinear nature of the maximization problem, there are not explicit formulas for the probit ML estimates
- instead, numerical optimization is used, and, usually, only a few iterations are needed
- in Stata, several algorithms can be used

Ricardo Mora The Probit Model

Notes

The Perfect Prediction Problem

- suppose that vector $\widetilde{\beta}$ perfectly predicts $marry_i$ in the sense that for a given scalar k, $\widetilde{\beta}x > k$ iff $marry = 1$
- then the same thing is true for any multiple of $\widetilde{\beta}$: the sample identification condition is violated
- this may be due to several reasons
  - one control may be a perfect classifier: drop it
  - the model may be trivially misspecified (like predicting marriage among married individuals)
  - the sample may simply be not large enough

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

The Perfect Prediction Problem

- suppose that vector $\widetilde{\beta}$ perfectly predicts $marry_i$ in the sense that for a given scalar k, $\widetilde{\beta}x > k$ iff $marry = 1$
- then the same thing is true for any multiple of $\widetilde{\beta}$: the sample identification condition is violated
- this may be due to several reasons
  - one control may be a perfect classifier: drop it
  - the model may be trivially misspecified (like predicting marriage among married individuals)
  - the sample may simply be not large enough

Ricardo Mora The Probit Model

Notes

The Perfect Prediction Problem

- suppose that vector $\widetilde{\beta}$ perfectly predicts $marry_i$ in the sense that for a given scalar k, $\widetilde{\beta}x > k$ iff $marry = 1$
- then the same thing is true for any multiple of $\widetilde{\beta}$: the sample identification condition is violated
- this may be due to several reasons
  - one control may be a perfect classifier: drop it
  - the model may be trivially misspecified (like predicting marriage among married individuals)
  - the sample may simply be not large enough

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

The Perfect Prediction Problem

- suppose that vector $\widetilde{\beta}$ perfectly predicts $marry_i$ in the sense that for a given scalar k, $\widetilde{\beta}x > k$ iff $marry = 1$
- then the same thing is true for any multiple of $\widetilde{\beta}$: the sample identification condition is violated
- this may be due to several reasons
  - one control may be a perfect classifier: drop it
  - the model may be trivially misspecified (like predicting marriage among married individuals)
  - the sample may simply be not large enough

Ricardo Mora The Probit Model

Notes

The Perfect Prediction Problem

- suppose that vector $\widetilde{\beta}$ perfectly predicts $marry_i$ in the sense that for a given scalar k, $\widetilde{\beta}x > k$ iff $marry = 1$
- then the same thing is true for any multiple of $\widetilde{\beta}$: the sample identification condition is violated
- this may be due to several reasons
  - one control may be a perfect classifier: drop it
  - the model may be trivially misspecified (like predicting marriage among married individuals)
  - the sample may simply be not large enough

Notes

The Perfect Prediction Problem

- suppose that vector $\widetilde{\beta}$ perfectly predicts $marry_i$ in the sense that for a given scalar k, $\widetilde{\beta}x > k$ iff $marry = 1$
- then the same thing is true for any multiple of $\widetilde{\beta}$: the sample identification condition is violated
- this may be due to several reasons
  - one control may be a perfect classifier: drop it
  - the model may be trivially misspecified (like predicting marriage among married individuals)
  - the sample may simply be not large enough

Notes

Asymptotic Properties and Testing

under general conditions, MLE is consistent, asymptotically normal, and asymptotically efficient

- we can construct (asymptotic) t tests and confidence intervals (just as with OLS, 2SLS, and IV)
- exclusion restrictions
  - the Lagrange multiplier or score test only requires estimating model under the null
  - the Wald test requires estimation of only the unrestricted model
  - the likelihood ratio (LR) test requires estimation of both models

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Asymptotic Properties and Testing

under general conditions, MLE is consistent, asymptotically normal, and asymptotically efficient

- we can construct (asymptotic) t tests and confidence intervals (just as with OLS, 2SLS, and IV)
- exclusion restrictions
  - the Lagrange multiplier or score test only requires estimating model under the null
  - the Wald test requires estimation of only the unrestricted model
  - the likelihood ratio (LR) test requires estimation of both models

Notes

Asymptotic Properties and Testing

under general conditions, MLE is consistent, asymptotically normal, and asymptotically efficient

- we can construct (asymptotic) t tests and confidence intervals (just as with OLS, 2SLS, and IV)
- exclusion restrictions
  - the Lagrange multiplier or score test only requires estimating model under the null
  - the Wald test requires estimation of only the unrestricted model
  - the likelihood ratio (LR) test requires estimation of both models

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Asymptotic Properties and Testing

under general conditions, MLE is consistent, asymptotically normal, and asymptotically efficient

- we can construct (asymptotic) t tests and confidence intervals (just as with OLS, 2SLS, and IV)
- exclusion restrictions
  - the Lagrange multiplier or score test only requires estimating model under the null
  - the Wald test requires estimation of only the unrestricted model
  - the likelihood ratio (LR) test requires estimation of both models

Notes

Asymptotic Properties and Testing

under general conditions, MLE is consistent, asymptotically normal, and asymptotically efficient

- we can construct (asymptotic) t tests and confidence intervals (just as with OLS, 2SLS, and IV)
- exclusion restrictions
  - the Lagrange multiplier or score test only requires estimating model under the null
  - the Wald test requires estimation of only the unrestricted model
  - the likelihood ratio (LR) test requires estimation of both models

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Asymptotic Properties and Testing

under general conditions, MLE is consistent, asymptotically normal, and asymptotically efficient

- we can construct (asymptotic) t tests and confidence intervals (just as with OLS, 2SLS, and IV)
- exclusion restrictions
  - the Lagrange multiplier or score test only requires estimating model under the null
  - the Wald test requires estimation of only the unrestricted model
  - the likelihood ratio (LR) test requires estimation of both models

Ricardo Mora The Probit Model

Notes

The Likelihood Ratio Test

The LR test

- it is based on the difference in loglikelihood functions
- as with the F tests in linear regression, restricting models leads to no-larger loglikelihoods

$$LR = 2(l_{ur} - l_r) \xrightarrow{a} \chi_q$$

where q is the number of restrictions

Notes

Probit & Logit Estimation in Stata

- `probit`: computes Maximum Likelihood probit estimation
- `logit`: computes Maximum Likelihood logit estimation
- margins (`mfx`): marginal means, predictive margins, marginal effects, and average marginal effects
- `test`: Wald tests of simple and composite linear hypothesis
- `lincom`: point estimates, standard errors, testing, and inference for linear combinations of coefficients
- `predict`: predictions, residuals, influence statistics, and other diagnostic measures
- `e(ll)`: returns the log-likelihood for the last estimated model

We are going to use `probit`, `test`, `lincom`, `predict`, `e(ll)`, and `logit`

Notes

Probit & Logit Estimation in Stata

- `probit`: computes Maximum Likelihood probit estimation
- `logit`: computes Maximum Likelihood logit estimation
- margins (`mfx`): marginal means, predictive margins, marginal effects, and average marginal effects
- `test`: Wald tests of simple and composite linear hypothesis
- `lincom`: point estimates, standard errors, testing, and inference for linear combinations of coefficients
- `predict`: predictions, residuals, influence statistics, and other diagnostic measures
- `e(ll)`: returns the log-likelihood for the last estimated model

We are going to use `probit`, `test`, `lincom`, `predict`, `e(ll)`, and `logit`

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Probit & Logit Estimation in Stata

- `probit`: computes Maximum Likelihood probit estimation
- `logit`: computes Maximum Likelihood logit estimation
- margins (`mfx`): marginal means, predictive margins, marginal effects, and average marginal effects
- `test`: Wald tests of simple and composite linear hypothesis
- `lincom`: point estimates, standard errors, testing, and inference for linear combinations of coefficients
- `predict`: predictions, residuals, influence statistics, and other diagnostic measures
- `e(ll)`: returns the log-likelihood for the last estimated model

We are going to use `probit`, `test`, `lincom`, `predict`, `e(ll)`, and `logit`

Ricardo Mora The Probit Model

Notes

Probit & Logit Estimation in Stata

- probit: computes Maximum Likelihood probit estimation
- logit: computes Maximum Likelihood logit estimation
- margins (mfx): marginal means, predictive margins, marginal effects, and average marginal effects
- test: Wald tests of simple and composite linear hypothesis
- lincom: point estimates, standard errors, testing, and inference for linear combinations of coefficients
- predict: predictions, residuals, influence statistics, and other diagnostic measures
- e(ll): returns the log-likelihood for the last estimated model

We are going to use probit, test, lincom, predict, e(ll), and logit

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Probit & Logit Estimation in Stata

- probit: computes Maximum Likelihood probit estimation
- logit: computes Maximum Likelihood logit estimation
- margins (mfx): marginal means, predictive margins, marginal effects, and average marginal effects
- test: Wald tests of simple and composite linear hypothesis
- lincom: point estimates, standard errors, testing, and inference for linear combinations of coefficients
- predict: predictions, residuals, influence statistics, and other diagnostic measures
- e(ll): returns the log-likelihood for the last estimated model

We are going to use probit, test, lincom, predict, e(ll), and logit

Ricardo Mora The Probit Model

Notes

Probit & Logit Estimation in Stata

- `probit`: computes Maximum Likelihood probit estimation
- `logit`: computes Maximum Likelihood logit estimation
- margins (`mfx`): marginal means, predictive margins, marginal effects, and average marginal effects
- `test`: Wald tests of simple and composite linear hypothesis
- `lincom`: point estimates, standard errors, testing, and inference for linear combinations of coefficients
- `predict`: predictions, residuals, influence statistics, and other diagnostic measures
- `e(ll)`: returns the log-likelihood for the last estimated model

We are going to use `probit`, `test`, `lincom`, `predict`, `e(ll)`, and `logit`

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Probit & Logit Estimation in Stata

- `probit`: computes Maximum Likelihood probit estimation
- `logit`: computes Maximum Likelihood logit estimation
- margins (`mfx`): marginal means, predictive margins, marginal effects, and average marginal effects
- `test`: Wald tests of simple and composite linear hypothesis
- `lincom`: point estimates, standard errors, testing, and inference for linear combinations of coefficients
- `predict`: predictions, residuals, influence statistics, and other diagnostic measures
- `e(ll)`: returns the log-likelihood for the last estimated model

We are going to use `probit`, `test`, `lincom`, `predict`, `e(ll)`, and `logit`

Ricardo Mora The Probit Model

Notes

Probit & Logit Estimation in Stata

- `probit`: computes Maximum Likelihood probit estimation
- `logit`: computes Maximum Likelihood logit estimation
- margins (`mfx`): marginal means, predictive margins, marginal effects, and average marginal effects
- `test`: Wald tests of simple and composite linear hypothesis
- `lincom`: point estimates, standard errors, testing, and inference for linear combinations of coefficients
- `predict`: predictions, residuals, influence statistics, and other diagnostic measures
- `e(ll)`: returns the log-likelihood for the last estimated model

We are going to use `probit`, `test`, `lincom`, `predict`, `e(ll)`, and `logit`

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Probit & Logit Estimation in Stata

- `probit`: computes Maximum Likelihood probit estimation
- `logit`: computes Maximum Likelihood logit estimation
- margins (`mfx`): marginal means, predictive margins, marginal effects, and average marginal effects
- `test`: Wald tests of simple and composite linear hypothesis
- `lincom`: point estimates, standard errors, testing, and inference for linear combinations of coefficients
- `predict`: predictions, residuals, influence statistics, and other diagnostic measures
- `e(ll)`: returns the log-likelihood for the last estimated model

We are going to use `probit`, `test`, `lincom`, `predict`, `e(ll)`, and `logit`

Ricardo Mora The Probit Model

Notes

probit *depvar* *indvars* [if] [in] [weight],[options]

- *depvar*: negative is 0, all other nonmissing values is 1
- output shows χ^2_q statistic test for null that all slopes are zero
- some interesting options:
  - noconstant: suppress constant term
  - constraints(*constraints*) apply specified linear constraints

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

probit *depvar* *indvars* [if] [in] [weight],[options]

- *depvar*: negative is 0, all other nonmissing values is 1
- output shows χ^2_q statistic test for null that all slopes are zero
- some interesting options:
  - noconstant: suppress constant term
  - constraints(*constraints*) apply specified linear constraints

Ricardo Mora The Probit Model

Notes

`probit` *depvar* *indvars* [if] [in] [weight],[options]

- *depvar*: negative is 0, all other nonmissing values is 1
- output shows χ^2_q statistic test for null that all slopes are zero
- some interesting options:
  - `noconstant`: suppress constant term
  - `constraints(`*constraints*`)` apply specified linear constraints

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

`probit` *depvar* *indvars* [if] [in] [weight],[options]

- *depvar*: negative is 0, all other nonmissing values is 1
- output shows χ^2_q statistic test for null that all slopes are zero
- some interesting options:
  - `noconstant`: suppress constant term
  - `constraints(`*constraints*`)` apply specified linear constraints

Notes

A Simulated Example: Participation Decision

The Probit Model

- $U_m = 0.3 + 0.05 * educ + 0.5 * kids + \varepsilon_m$
- $U_h = 0.8 - 0.02 * educ + 2 * kids + \varepsilon_h$
- $\varepsilon_h, \varepsilon_m \sim N(0, \Sigma)$ such that $\varepsilon \sim N(0,1)$

- education brings utility if you work, disutility if you don't
- having a kid brings more utility if you don't work
- $\beta x = -0.5 + 0.07 * educ - 1.5 * kids$

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

A Simulated Example: Participation Decision

The Probit Model

- $U_m = 0.3 + 0.05 * educ + 0.5 * kids + \varepsilon_m$
- $U_h = 0.8 - 0.02 * educ + 2 * kids + \varepsilon_h$
- $\varepsilon_h, \varepsilon_m \sim N(0, \Sigma)$ such that $\varepsilon \sim N(0,1)$

- education brings utility if you work, disutility if you don't
- having a kid brings more utility if you don't work
- $\beta x = -0.5 + 0.07 * educ - 1.5 * kids$

Ricardo Mora The Probit Model

Notes

A Simulated Example: Participation Decision

The Probit Model

- $U_m = 0.3 + 0.05 * educ + 0.5 * kids + \varepsilon_m$
- $U_h = 0.8 - 0.02 * educ + 2 * kids + \varepsilon_h$
- $\varepsilon_h, \varepsilon_m \sim N(0, \Sigma)$ such that $\varepsilon \sim N(0,1)$

- education brings utility if you work, disutility if you don't
- having a kid brings more utility if you don't work
- $\beta x = -0.5 + 0.07 * educ - 1.5 * kids$

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

A Simulated Example: Participation Decision

The Probit Model

- $U_m = 0.3 + 0.05 * educ + 0.5 * kids + \varepsilon_m$
- $U_h = 0.8 - 0.02 * educ + 2 * kids + \varepsilon_h$
- $\varepsilon_h, \varepsilon_m \sim N(0, \Sigma)$ such that $\varepsilon \sim N(0,1)$

- education brings utility if you work, disutility if you don't
- having a kid brings more utility if you don't work
- $\beta x = -0.5 + 0.07 * educ - 1.5 * kids$

Ricardo Mora The Probit Model

Notes

probit Output

probit *work educ kids*

```
Probit regression                                 Number of obs   =       5000
                                                  LR chi2(2)      =    1449.25
                                                  Prob > chi2     =     0.0000
Log likelihood = -2548.1786                       Pseudo R2       =     0.2214

------------------------------------------------------------------------------
        work |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        educ |   .0741842   .0056425    13.15   0.000     .0631251    .0852432
        kids |   -1.43585   .0409208   -35.09   0.000    -1.516054   -1.355647
       _cons |  -.5886567    .079498    -7.40   0.000    -.7444698   -.4328435
------------------------------------------------------------------------------
```


Notes

Predicting the Probabilities

Computing $\hat{\Pr}(y_i = 1 | x_i)$

predict *p_hat* ,p

- for each observation, if $\hat{\Pr}(y_i = 1 | x_i) > 0.5$ then $\hat{y}_i = 1$
- the percent correctly predicted is the % for which $\hat{y}_i$ matches y_i
- it is possible to get high percentages correctly predicted in useless models
  - suppose that $\Pr(y_i = 0) = 0.9$
  - always predicting $\hat{y}_i = 0$ will lead to 90% correctly predicted!

Notes

Predicting the Probabilities

Computing $\hat{\Pr}(y_i = 1|x_i)$

```
predict p_hat ,p
```

- for each observation, if $\hat{\Pr}(y_i = 1|x_i) > 0.5$ then $\hat{y}_i = 1$
- the percent correctly predicted is the % for which $\hat{y}_i$ matches y_i
- it is possible to get high percentages correctly predicted in useless models
  - suppose that $\Pr(y_i = 0) = 0.9$
  - always predicting $\hat{y}_i = 0$ will lead to 90% correctly predicted!

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Predicting the Probabilities

Computing $\hat{\Pr}(y_i = 1|x_i)$

predict p_hat ,p

- for each observation, if $\hat{\Pr}(y_i = 1|x_i) > 0.5$ then $\hat{y}_i = 1$
- the percent correctly predicted is the % for which $\hat{y}_i$ matches y_i
- it is possible to get high percentages correctly predicted in useless models
  - suppose that $\Pr(y_i = 0) = 0.9$
  - always predicting $\hat{y}_i = 0$ will lead to 90% correctly predicted!

Ricardo Mora The Probit Model

Notes

Predicting the Probabilities

Computing $\hat{\Pr}(y_i = 1 | x_i)$

```
predict p_hat ,p
```

- for each observation, if $\hat{\Pr}(y_i = 1 | x_i) > 0.5$ then $\hat{y}_i = 1$
- the percent correctly predicted is the % for which $\hat{y}_i$ matches y_i
- it is possible to get high percentages correctly predicted in useless models
  - suppose that $\Pr(y_i = 0) = 0.9$
  - always predicting $\hat{y}_i = 0$ will lead to 90% correctly predicted!

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Predicting the Probabilities

Computing $\hat{\Pr}(y_i = 1 | x_i)$

```
predict p_hat ,p
```

- for each observation, if $\hat{\Pr}(y_i = 1 | x_i) > 0.5$ then $\hat{y}_i = 1$
- the percent correctly predicted is the % for which $\hat{y}_i$ matches y_i
- it is possible to get high percentages correctly predicted in useless models
  - suppose that $\Pr(y_i = 0) = 0.9$
  - always predicting $\hat{y}_i = 0$ will lead to 90% correctly predicted!

Ricardo Mora The Probit Model

Notes

Predicting the Probabilities

Computing $\hat{\Pr}(y_i = 1 | x_i)$

predict *p_hat* ,p

- for each observation, if $\hat{\Pr}(y_i = 1 | x_i) > 0.5$ then $\hat{y}_i = 1$
- the percent correctly predicted is the % for which $\hat{y}_i$ matches y_i
- it is possible to get high percentages correctly predicted in useless models
  - suppose that $\Pr(y_i = 0) = 0.9$
  - always predicting $\hat{y}_i = 0$ will lead to 90% correctly predicted!

Notes

Understanding Coefficients & Marginal Effects

- the column "`Coeff.`" refers to the ML estimates $\hat{\beta}^{ML}$
- in contrast to the linear model, in the probit model the coefficients do not capture the marginal effect on output when a control changes
  - if control x_j is continuous, $\frac{\partial Pr(y=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if control x_j is discrete, $\Delta Pr(work = 1) = \Phi(\beta x_1) - \Phi(\beta x_0)$
- since the model is non-linear, marginal effects depend on the values of the other controls
- to get marginal effects instead of coefficients we can use command `dprobit`
- we can also use the commands `margins` or `mfx`

Notes

Understanding Coefficients & Marginal Effects

- the column "Coeff." refers to the ML estimates $\hat{\beta}^{ML}$
- in contrast to the linear model, in the probit model the coefficients do not capture the marginal effect on output when a control changes
  - if control x_j is continuous, $\frac{\partial Pr(y=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if control x_j is discrete, $\Delta Pr(work = 1) = \Phi(\beta x_1) - \Phi(\beta x_0)$
- since the model is non-linear, marginal effects depend on the values of the other controls
- to get marginal effects instead of coefficients we can use command dprobit
- we can also use the commands margins or mfx

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Understanding Coefficients & Marginal Effects

- the column "Coeff." refers to the ML estimates $\hat{\beta}^{ML}$
- in contrast to the linear model, in the probit model the coefficients do not capture the marginal effect on output when a control changes
  - if control x_j is continuous, $\frac{\partial Pr(y=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if control x_j is discrete, $\Delta Pr(work = 1) = \Phi(\beta x_1) - \Phi(\beta x_0)$
- since the model is non-linear, marginal effects depend on the values of the other controls
- to get marginal effects instead of coefficients we can use command dprobit
- we can also use the commands margins or mfx

Ricardo Mora The Probit Model

Notes

Understanding Coefficients & Marginal Effects

- the column "`Coeff.`" refers to the ML estimates $\hat{\beta}^{ML}$
- in contrast to the linear model, in the probit model the coefficients do not capture the marginal effect on output when a control changes
  - if control x_j is continuous, $\frac{\partial Pr(y=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if control x_j is discrete, $\Delta Pr(work = 1) = \Phi(\beta x_1) - \Phi(\beta x_0)$
- since the model is non-linear, marginal effects depend on the values of the other controls
- to get marginal effects instead of coefficients we can use command `dprobit`
- we can also use the commands `margins` or `mfx`

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Understanding Coefficients & Marginal Effects

- the column "`Coeff.`" refers to the ML estimates $\hat{\beta}^{ML}$
- in contrast to the linear model, in the probit model the coefficients do not capture the marginal effect on output when a control changes
  - if control x_j is continuous, $\frac{\partial Pr(y=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if control x_j is discrete, $\Delta Pr(work = 1) = \Phi(\beta x_1) - \Phi(\beta x_0)$
- since the model is non-linear, marginal effects depend on the values of the other controls
- to get marginal effects instead of coefficients we can use command `dprobit`
- we can also use the commands `margins` or `mfx`

Ricardo Mora The Probit Model

Notes

Understanding Coefficients & Marginal Effects

- the column "`Coeff.`" refers to the ML estimates $\hat{\beta}^{ML}$
- in contrast to the linear model, in the probit model the coefficients do not capture the marginal effect on output when a control changes
  - if control x_j is continuous, $\frac{\partial Pr(y=1)}{\partial x_j} = \phi(\beta x)\beta_j$
  - if control x_j is discrete, $\Delta Pr(work = 1) = \Phi(\beta x_1) - \Phi(\beta x_0)$
- since the model is non-linear, marginal effects depend on the values of the other controls
- to get marginal effects instead of coefficients we can use command `dprobit`
- we can also use the commands `margins` or `mfx`

Notes

`dprobit` *work educ kids*

```
Probit regression, reporting marginal effects           Number of obs =   5000
                                                        LR chi2(2)    =1449.25
                                                        Prob > chi2   = 0.0000
Log likelihood = -2548.1786                             Pseudo R2     = 0.2214

------------------------------------------------------------------------------
    work |      dF/dx   Std. Err.      z    P>|z|     x-bar  [    95% C.I.   ]
---------+--------------------------------------------------------------------
    educ |   .0268932   .0020397    13.15   0.000   13.5644   .022895  .030891
   kids*|  -.5076016   .0126823   -35.09   0.000     .5956 -.532458 -.482745
---------+--------------------------------------------------------------------
  obs. P |       .362
 pred. P |   .3308434  (at x-bar)
------------------------------------------------------------------------------
(*) dF/dx is for discrete change of dummy variable from 0 to 1
    z and P>|z| correspond to the test of the underlying coefficient being 0
```


Notes

Individual Marginal Effects: Discrete Change

we want to estimate the change in probability when x changes from x_0 to x_1

Discrete change

- after estimation of the model, predict index function $\hat{\beta}^{ML}x_0$
- replace values in controls from scenario 0, x_0, to scenario 1, x_1
- predict index function $\hat{\beta}^{ML}x_1$
- generate the individual marginal effects

$$\Phi\left(\hat{\beta}^{ML}x_1\right) - \Phi\left(\hat{\beta}^{ML}x_0\right)$$

Notes

The Random Utility Model
The Probit & Logit Models
Estimation & Inference
Probit & Logit Estimation in Stata
Summary

Individual Marginal Effects: Discrete Change

we want to estimate the change in probability when x changes from x_0 to x_1

Discrete change

- after estimation of the model, predict index function $\hat{\beta}^{ML}x_0$
- replace values in controls from scenario 0, x_0, to scenario 1, x_1
- predict index function $\hat{\beta}^{ML}x_1$
- generate the individual marginal effects

$$\Phi\left(\hat{\beta}^{ML}x_1\right) - \Phi\left(\hat{\beta}^{ML}x_0\right)$$

Notes

Example: The Effect of Having A Kid

```
// marginal effects of having a kid
gen kids_old=kids
replace kids=1
predict p1,p
replace kids=0
predict p0,p
gen Mg_kid = p1 - p0
bysort educ: sum Mg_kid
```


Notes

bysort *educ*: su *Mg_kid*

```
-> educ = 8

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |       721   -.4257113           0  -.4257113  -.4257113

-> educ = 12

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |      2255   -.4901687           0  -.4901687  -.4901687

-> educ = 16

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |      1502    -.524038           0   -.524038   -.524038

-> educ = 21

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |       522   -.5134011           0  -.5134011  -.5134011
```

- the effect of having a kid changes with education
- higher education makes individuals more likely to have indexes βx closer to 0.5 (the probit slope is largest at 0.5)
- how would you make the "kid" effect smaller with higher education?

Notes

bysort *educ* : su *Mg_kid*

```
-> educ = 8

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |       721   -.4257113           0  -.4257113  -.4257113

-> educ = 12

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |      2255   -.4901687           0  -.4901687  -.4901687

-> educ = 16

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |      1502    -.524038           0   -.524038   -.524038

-> educ = 21

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |       522   -.5134011           0  -.5134011  -.5134011
```

- the effect of having a kid changes with education
- higher education makes individuals more likely to have indexes βx closer to 0.5 (the probit slope is largest at 0.5)
- how would you make the "kid" effect smaller with higher education?

Notes

bysort *educ* : su *Mg_kid*

```
-> educ = 8

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |       721   -.4257113           0  -.4257113  -.4257113

-> educ = 12

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |      2255   -.4901687           0  -.4901687  -.4901687

-> educ = 16

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |      1502    -.524038           0   -.524038   -.524038

-> educ = 21

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |       522   -.5134011           0  -.5134011  -.5134011
```

- the effect of having a kid changes with education
- higher education makes individuals more likely to have indexes βx closer to 0.5 (the probit slope is largest at 0.5)
- how would you make the "kid" effect smaller with higher education?

Notes

bysort *educ*: su *Mg_kid*

```
-> educ = 8

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |       721   -.4257113           0  -.4257113  -.4257113

-> educ = 12

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |      2255   -.4901687           0  -.4901687  -.4901687

-> educ = 16

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |      1502    -.524038           0   -.524038   -.524038

-> educ = 21

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
      Mg_kid |       522   -.5134011           0  -.5134011  -.5134011
```

- the effect of having a kid changes with education
- higher education makes individuals more likely to have indexes βx closer to 0.5 (the probit slope is largest at 0.5)
- how would you make the “kid” effect smaller with higher education?

Notes

Individual Marginal Effects: Infinitessimal Change

Calculus approximation

- after estimation, predict the index function $\hat{\beta}^{ML}x$
- generate the calculus approximation: $\phi\left(\hat{\beta}^{ML}x\right)\hat{\beta}_j^{ML}$

Notes

Example of Calculus Approximation

On individual values

```
 // marginal effects of one extra year of education (individual calculus approximation)
. predict xb_hat,xb
. gen Mg_educ_cal=normalden(xb_hat)*_b[educ]          // this is the individual's marginal effect
. su Mg_educ_cal

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
 Mg_educ_cal |      5000    .0212441    .0058988     .01063    .0295949
```

On average values

```
. // marginal effects of one extra year of education  (at the mean values)
. rename educ educ_old
. rename kids kids_old
. egen educ = mean(educ_old) if e(sample)
. egen kids = mean(kids_old) if e(sample)
. predict xb_hat_avg,xb
. gen Mg_educ_avg=normalden(xb_hat_avg)*_b[educ]      // this is the marginal effect on averages
. su Mg_educ_avg

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
 Mg_educ_avg |      5000    .0268932           0   .0268932   .0268932
```


Notes

Logit Estimation

logit *educ kids*

```
Logistic regression                               Number of obs   =       5000
                                                  LR chi2(2)      =    1446.73
                                                  Prob > chi2     =     0.0000
Log likelihood = -2549.4381                       Pseudo R2       =     0.2210

------------------------------------------------------------------------------
        work |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        educ |   .1263039   .0097563    12.95   0.000     .1071819    .1454258
        kids |  -2.386564   .0715243   -33.37   0.000    -2.526749   -2.246379
       _cons |  -1.031709   .1346306    -7.66   0.000    -1.295581   -.7678384
------------------------------------------------------------------------------
```

mfx

```
Marginal effects after logit
      y  = Pr(work) (predict)
         =  .32302366
------------------------------------------------------------------------------
variable |      dy/dx    Std. Err.     z    P>|z|  [    95% C.I.   ]      X
---------+--------------------------------------------------------------------
    educ |     .02762      .00212   13.02   0.000   .023462  .031778   13.5644
   kids*|  -.5102708      .01283  -39.78   0.000  -.535413 -.485129     .5956
------------------------------------------------------------------------------
(*) dy/dx is for discrete change of dummy variable from 0 to 1
```

Logit & Probit $\hat{\beta}$ are not comparable, but marginal effects are.

Notes

Summary

- not all parameters of the RUM can be estimated
- the Probit and Logit models identify how each control affects the probability of participation
- ML estimation requires numerical methods
- under general conditions, ML estimates are consistent, asymptotically normal, and asymptotically efficient
- significance tests and general restrictions tests are easy to carry out with the Probit model
- `Stata` allows for probit and logit estimation of the random utility model by ML

Notes

Notes