Econometrics II
Tutorial Problems No. 1

Lennart Hoogerheide & Agnieszka Borowska

15.02.2017

1 Summary

- **Binary Response Model:** A model for a binary (or dummy, i.e. with two possible outcomes 0 and 1) dependent variable.
- **Response Probability:** In a binary response model, the probability that the dependent variable takes on the value one, conditional on explanatory variables.
- **Linear Probability Model:** The multiple linear regression model with a binary dependent variable, where the response probability is linear in the parameters.

  [bad idea! the probability can be estimated outside the $[0, 1]$ interval]
- **Logit Model:** A model for binary response where the response probability is the logit function evaluated at a linear function of the explanatory variables.

$$G(z) = \frac{1}{1 - \exp(-z)} = \frac{\exp(z)}{1 + \exp(z)}.$$

- **Probit Model:** A model for binary responses where the response probability is the standard normal cumulative distribution function (CDF) evaluated at a linear function of the explanatory variables.

$$G(z) = \Phi(z) = \int_{-\infty}^{z} \phi(v)dv = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{v^2}{2}\right) dv.$$

- **Latent Variable Model:** A model where the observed dependent variable is assumed to be a function of an underlying latent, or unobserved, variable.

  [interpretation of binary logit/probit model]
- **Partial Effect at the Average (PEA):** In models with nonconstant partial effects, the partial effect evaluated at the average values of the explanatory variables.

  [Substitute averages $\bar{x}_1, \ldots, \bar{x}_k$, where k is the number of regressors.]
- **Average Partial Effect (APE):** For nonconstant partial effects, the partial effect averaged across the specified population.

  [$\frac{1}{n}\sum_{i=1}^{n} \frac{\partial \mathbb{P}(y_i=1|x_i)}{\partial x_j} = \frac{1}{n}\sum_{i=1}^{n} g(\hat{\beta}_0 + \hat{\beta}_1 x_{i1} + \cdots + \hat{\beta}_k x_{ik}) \cdot \beta_j$]
- **Akaike Information Criterion (AIC):** A general measure for relative quality of models estimated with maximum likelihood, computed as

$$AIC = -2\frac{\ln L}{n} + 2\frac{k}{n},$$

  where $\ln L$ is the maximum value of likelihood, n is the number of observations and k is the number of parameters.
- **Schwarz Criterion (SC):** A general measure for relative quality of models estimated with maximum likelihood, computed as

$$SC = -2\frac{\ln L}{n} + \ln(n)\frac{k}{n},$$

  where $\ln L$ is the maximum value of likelihood, n is the number of observations and k is the number of parameters.

- **Pseudo R-Squared (McFadden R-Squared):** A goodness-of-fit measure that is particularly used for logit/probit models.
$$\text{McFadden R-Squared} = 1 - \frac{\ln L}{\ln L_0},$$
where $\ln L$ is the maximum value of loglikelihood and $\ln L_0$ is the maximum value of likelihood in model with only a constant term.

  [not only a relative quality measure, unlike AIC or SC]

- **Percent Correctly Predicted (Hit Rate):** In a binary response model, the percentage of times the prediction of zero or one coincides with the actual outcome. Percentage of observations with $\tilde{y}_i = y_i$, where
$$\tilde{y}_i = \begin{cases} 1, & \text{if } \hat{\mathbb{P}}(y_i = 1|x_i) = G(x_i'\hat{\beta}) > c, \\ 0, & \text{if } \hat{\mathbb{P}}(y_i = 1|x_i) = G(x_i'\hat{\beta}) \leq c, \end{cases}$$
where c is typically chosen as 0.5.

2 Extra Topics

2.1 The Perfect Classifier Problem

Recall: the loglikelihood

$$\begin{aligned} \ln L(\beta) &= \ln p(y_1, \ldots, y_n | x_1, \ldots, x_n) \\ &= \sum_{i=1}^{n} \Big\{ \underbrace{y_i \ln[G(x_i'\beta)]}_{(*)} + \underbrace{(1 - y_i) \ln[1 - G(x_i'\beta)]}_{(**)} \Big\}. \end{aligned} \tag{1}$$

We have
$$0 < G(x_i'\beta) < 1,$$
hence
$$-\infty < \ln[G(x_i'\beta)] < 0.$$
Notice that
$$\begin{aligned} y_i = 1 &\Rightarrow (*) < 0 \;\&\; (**) = 0, \\ y_i = 0 &\Rightarrow (*) = 0 \;\&\; (**) < 0. \end{aligned}$$

Perfect fit:
$$\begin{aligned} y_i = 1 &\iff G(x_i'\beta) = 1, \\ y_i = 0 &\iff G(x_i'\beta) = 0. \end{aligned}$$
This could happen only when
$$y_i = 1 \iff x_i'\beta = \infty, \tag{2}$$
$$y_i = 0 \iff x_i'\beta = -\infty. \tag{3}$$
We say that the loglikelihood (1) is *bounded above by* 0, and it achieves this bound if (2) and (3) hold.

Now, suppose that there is some linear combination of the independent variables, say $x_i'\beta^\bullet$, such that
$$y_i = 1 \iff x_i'\beta^\bullet > 0, \tag{4}$$
$$y_i = 0 \iff x_i'\beta^\bullet < 0. \tag{5}$$
In other words, there is some range of the regressor(s) for which y_i is always 1 or 0. Then, we say that $x_i'\beta^\bullet$ describes a **separating hyperplane** (see Figure 1) and there is **complete separation** of the data. $x_i'\beta^\bullet$ is said to be a **perfect classifier**, since it allows us to predict y_i with perfect accuracy for every observation.

Problem? Yes, for ML estimation! Then, it is possible to make the value of $\ln L$ arbitrarily close to 0 (the upper bound) by choosing β arbitrarily large (in an absolute sense)[1]. Hence, no finite ML estimator exists.

[1] Formally: by setting $\beta = \gamma\beta^\bullet$ and letting $\gamma \to \infty$.

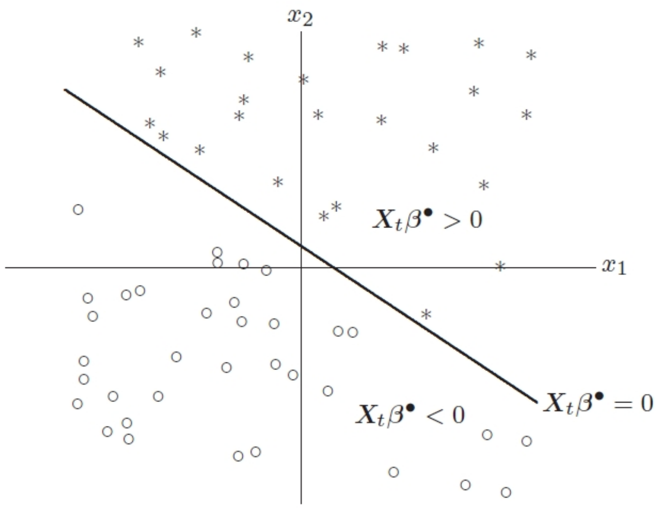


Figure 1: Figure 11.2 from Davidson and MacKinnon (1999), "Econometric Theory and Methods": A perfect classifier yields a separating hyperplane.

This is exactly what any nonlinear maximization algorithm will attempt to do if there exists a vector $\beta^\bullet$ for which conditions (4) and (5) are satisfied. Because of the numerical limitations, the algorithm will eventually terminate (with some numerical error) at a value of $\ln L$ slightly less than 0.

This is likely to occur in practice when the sample is very small, when almost all of the y_i are equal to 0 or almost all of them are equal to 1, or when the model fits extremely well.

The next topic is designed to give you a feel for the circumstances in which ML estimation is likely to fail because there is a perfect classifier.

2.2 Simulation from the latent variable model[2]

Consider the latent variable model

$$y_i^* = \beta_0 + \beta_1 x_i + e_i,$$
$$e_i \sim \mathcal{N}(0,1),$$
$$y_i = \begin{cases} 1, & \text{if } y_i^* > 0, \\ 0, & \text{if } y_i^* \leq 0 \end{cases}$$

Suppose that $x_i \sim \mathcal{N}(0,1)$*. We will generate* $5,000$ *samples of* 20 *observations on* (x_i, y_i) *pairs in the following way:*

- $1,000$ *assuming that* $\beta_0 = 0$ *and* $\beta_1 = 1$*;*
- $1,000$ *assuming that* $\beta_0 = 1$ *and* $\beta_1 = 1$*;*
- $1,000$ *assuming that* $\beta_0 = -1$ *and* $\beta_1 = 1$*;*
- $1,000$ *assuming that* $\beta_0 = 0$ *and* $\beta_1 = 2$*;*
- $1,000$ *assuming that* $\beta_0 = 0$ *and* $\beta_1 = 3$.

For each of the $5,000$ *samples, we will attempt to estimate a probit model. We are interested in the following question: In each of the five cases, what proportion of the time does the estimation fail because of perfect classifiers? We also want to explain why there will be more failures in some cases than in others. Next, we will repeat this exercise for five sets of* $1,000$ *samples of size* 40*, with the same parameter values. This will allow us to draw a conclusion about the effect of sample size on the perfect classifier problem.*

Figure 2 presents an EViews code for the first case ($N = 20$ with β_0 and β_1).

[2] Based on Exercise 11.5 from Davidson and MacKinnon (1999), "Econometric Theory and Methods".

Program: LATENTVARIABLE_DM17_5 - (h:\desktop\econometric2\latentvariable_dm1...

Run | Print | Save | SaveAs | Cut | Copy | Paste | InsertTxt | Find | Replace | Wrap+/- | LineNum+/- | Encrypt

```
wfcreate(wf=latentvariable_dm17_5_0_1) u 20
'Control Variables
!N = 20
!M =1000
setmaxerrs 6*!M '6 because if the estimation fails no coefs, stderrs and loglik are created, and
assigning of these creates next errors

'Parameters
!beta0 = 0
!beta1 = 1

matrix(!N,!M) xs
matrix(!N,!M) us
matrix(!N,!M) ys
matrix(!N,!M) y_stars

matrix(2,!M) eq_coeff
matrix(2,!M) eq_stderrs
matrix(1,!M) eq_loglik

for !i=1 to !M
    series u = nrnd
    matplace(us,u,1,!i)
    series x = nrnd
    matplace(xs,x,1,!i)
    series y_star = !beta0 + !beta1*x + u
    matplace(y_stars,y_star,1,!i)
    series y = @recode(y_star>0, 1, 0)
    matplace(ys,y,1,!i)

    equation eq.binary(d="n") y c x
    eq_coeff(1,!i) = eq.@coefs(1)
    eq_coeff(2,!i) = eq.@coefs(2)
    eq_stderrs(1,!i) = eq.@stderrs(1)
    eq_stderrs(2,!i) = eq.@stderrs(2)

    eq_loglik = eq.@logl
next

scalar err_no1 = @errorcount/6

wfsave "H:\Desktop\Econometric2\DM17_5_N20_betas_0_1"
```

Figure 2: EViews code for simulation of $M = 1,000$ scenarios of binary response model with $N = 20$ independent variables, with $\beta_0 = 0$ and $\beta_1 = 1$, and for probit estimation. But don't worry, you will not be asked for EViews commands at the exam!

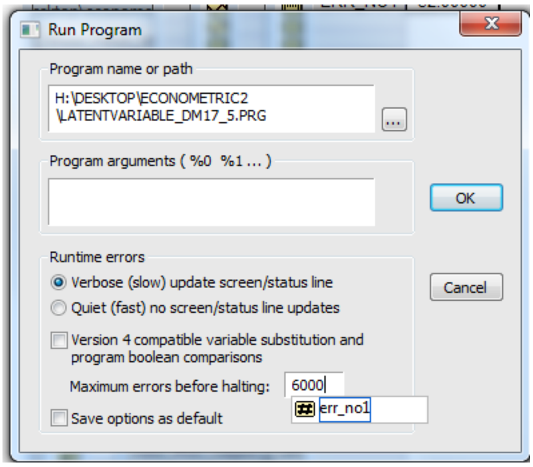


Figure 3: Manually changing of maximum errors before halting (EViews Run Program tab).

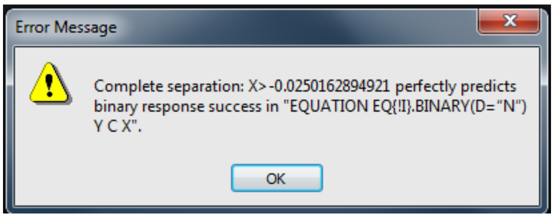


Figure 4: EViews Error Message about complete separation problem in the probit estimation.

If you are interested, you can check the results of each probit estimation: the coefficients estimates, their standard errors and the loglikelihood values are stored in matrices `eq_coeff`, `eq_stderrs` and `eq_logl`, respectively. But what we are truly after, is the error count variable, `err_no1`, which reports how many times an estimation error occurred. Notice, that we used the command `setmaxerr` to set the maximum number of error that the program may encounter before execution is halted. Alternatively, you can specify it in the box showing up after clicking on the `run` button, as in Figure 3. Without changing the value of maximum error allowed, the program would shortly break with the error message reporting the perfect separation problem, similar to the one from Figure 4.

The table below shows the proportion of the time that perfect classifiers were encountered for each of the five cases and each of the two sample sizes.

Parameters	$n = 20$	$n = 40$
$\beta_0 = 0,\ \beta_1 = 1$	0.012	0.000
$\beta_0 = 1,\ \beta_1 = 1$	0.074	0.001
$\beta_0 = -1,\ \beta_1 = 1$	0.056	0.002
$\beta_0 = 0,\ \beta_1 = 2$	0.143	0.008
$\beta_0 = 0,\ \beta_1 = 3$	0.286	0.052

The proportion of samples with perfect classifiers increases as both β_0 and β_1 increase in absolute value. When $\beta_0 = 0$, the unconditional expectation of y_i is 0.5.
As β_0 increases in absolute value, this expectation becomes larger, and the proportion of 1s in the sample increases.
As β_1 becomes larger in absolute value, the model fits better on average, which obviously increases the chance that it fits perfectly.
The results for parameters $(1, 1)$ are almost identical to those for parameters $(-1, 1)$ because, with x_i having mean 0, the fraction of 1s in the samples with parameters $(1, 1)$ is the same, on average, as the fraction of 0s in the samples with parameters $(-1, 1)$.

Comparing the results for $n = 20$ and $n = 40$, it is clear that the probability of encountering a perfect classifier falls very rapidly as the sample size increases.

3 Lecture Problems

Exercise 1.

This exercise is about the reason why we can use the standard normal (or standard logistic) distribution. Consider the binary probit model

$$\mathbb{P}(y_i = 1|x_i) = \Phi(\beta_0 + \beta_1 z_i),$$
$$\mathbb{P}(y_i = 0|x_i) = 1 - \Phi(\beta_0 + \beta_1 z_i),$$

where $\Phi(\cdot)$ is the cumulative distribution function (CDF) of the standard normal distribution. This stems from the assumption that

$$y_i^* = \beta_0 + \beta_1 z_i + e_i,$$

where e_i is an error term with standard normal distribution (independent of x_i), where

$$y_i = \mathbb{I}\{y_i^* > 0\} = \begin{cases} 1 & \text{if } y_i^* > 0, \\ 0 & \text{if } y_i^* \leq 0. \end{cases}$$

Suppose that we would assume that $e_i \sim \mathcal{N}(\mu, \sigma^2)$, where μ and σ^2 are parameters to be estimated (instead of setting $\mu = 0$ and $\sigma = 1$).

(1) Show that

$$\mathbb{P}(y_i = 1|x_i) = \Phi\left(\frac{\beta_0 + \beta_1 z_i + \mu}{\sigma}\right),$$
$$\mathbb{P}(y_i = 0|x_i) = 1 - \Phi\left(\frac{\beta_0 + \beta_1 z_i + \mu}{\sigma}\right).$$

Hint: use the 'standard' trick that $\frac{e_i - \mu}{\sigma} \sim \mathcal{N}(0,1)$ if $e_i \sim \mathcal{N}(\mu, \sigma^2)$.

We have

$$\begin{aligned}
\mathbb{P}(y_i = 1|x_i) &= \mathbb{P}(y_i^* > 0|x_i) \\
&= \mathbb{P}(x_i'\beta + e_i > 0|x_i) \\
&= \mathbb{P}(e_i > -x_i'\beta|x_i) \\
&= \mathbb{P}\left(\frac{e_i - \mu}{\sigma} > \frac{-x_i'\beta - \mu}{\sigma}\middle| x_i\right) \\
&\overset{(*)}{=} \mathbb{P}\left(\frac{e_i - \mu}{\sigma} < \frac{x_i'\beta + \mu}{\sigma}\middle| x_i\right) \\
&\overset{(**)}{=} \mathbb{P}\left(\frac{e_i - \mu}{\sigma} \leq \frac{x_i'\beta + \mu}{\sigma}\middle| x_i\right) \\
&\overset{(***)}{=} \mathbb{P}\left(\frac{e_i - \mu}{\sigma} \leq \frac{x_i'\beta + \mu}{\sigma}\right) \\
&= \Phi\left(\frac{x_i'\beta + \mu}{\sigma}\right),
\end{aligned}$$

where we used that the standard normal distribution of $\frac{e_i - \mu}{\sigma}$ is: $^{(*)}$ symmetric around 0, $^{(**)}$ continuous and $^{(***)}$ independent of x_i.

Here: $x_i'\beta = \beta_0 + \beta_1 z_i$, so that

$$\mathbb{P}(y_i = 1|x_i) = \Phi\left(\frac{\beta_0 + \beta_1 z_i + \mu}{\sigma}\right).$$

Further, we have either $y_i = 0$ or $y_i = 1$, so that

$$\mathbb{P}(y_i = 0|x_i) + \mathbb{P}(y_i = 1|x_i) = 1,$$

so that

$$\mathbb{P}(y_i = 0|x_i) = 1 - \mathbb{P}(y_i = 1|x_i) = 1 - \Phi\left(\frac{\beta_0 + \beta_1 z_i + \mu}{\sigma}\right).$$

(2) *What happens to $\mathbb{P}(y_i = 1|x_i)$ if we change β_0 and μ into $\beta_0 + 1$ and $\mu - 1$?*

We have

$$\mathbb{P}(y_i = 1|x_i) = \Phi\left(\frac{(\beta_0 + 1) + \beta_1 z_i + (\mu - 1)}{\sigma}\right) = \Phi\left(\frac{\beta_0 + \beta_1 z_i + \mu}{\sigma}\right),$$

so nothing happens to $\mathbb{P}(y_i = 1|x_i)$ in this case.

Therefore β_0 and μ are **not identified**: the model with parameters $\beta_0 + a$ and $\mu - a$ $(-\infty < a < \infty)$ is the same Data Generating Process (DGP) as the model with parameters β_0 and μ: it yields the same probabilities $\mathbb{P}(y_i = 0|x_i)$ and $\mathbb{P}(y_i = 1|x_i)$ for each observation, and therefore exactly the same Bernoulli distributions and the same properties of the y_i (conditionally upon x_i). Even if we would have infinitely many observations, we could not distinguish between the model with parameters β_0 and μ and the model with parameters $\beta_0 + a$ and $\mu - a$. Therefore we can set $\mu = 0$ *without loss of generality.*

(3) *What happens to $\mathbb{P}(y_i = 1|x_i)$ if we change β_0, β_1, μ and σ into $2\beta_0$, $2\beta_1$, 2μ and 2σ?*

We have

$$\mathbb{P}(y_i = 1|x_i) = \Phi\left(\frac{2\beta_0 + 2\beta_1 z_i + 2\mu}{2\sigma}\right) = \Phi\left(\frac{\beta_0 + \beta_1 z_i + \mu}{\sigma}\right),$$

so nothing happens to $\mathbb{P}(y_i = 1|x_i)$ in this case.

Therefore β_0, β_1, μ and σ are **not identified**: the model with parameters $b \cdot \beta_0$, $b \cdot \beta_1$, $b \cdot \mu$ and $b \cdot \sigma$ is $(b > 0)$ is the same Data Generating Process (DGP) as the model with parameters β_0, β_1, μ and σ: it yields the same probabilities $\mathbb{P}(y_i = 0|x_i)$ and $\mathbb{P}(y_i = 1|x_i)$ for each observation, and therefore exactly the same Bernoulli distributions and the same properties of the y_i (conditional upon x_i). Even if we would have infinitely many observations, we could not distinguish between the model with parameters β_0, β_1, μ and σ and the model with parameters $b \cdot \beta_0$, $b \cdot \beta_1$, $b \cdot \mu$ and $b \cdot \sigma$. Therefore we can set $\sigma = 1$ *without loss of generality.*

(4) *What is the difference in $\mathbb{P}(y_i = 1|x_i)$ between the model with parameters β_0, β_1, μ and σ and the model with parameters $b \cdot (\beta_0 + a)$, $b \cdot \beta_1$, $b \cdot (\mu - a)$ and $b \cdot \sigma$ (with $-\infty < a < \infty$ and $b > 0$)?*

We have

$$\mathbb{P}(y_i = 1|x_i) = \Phi\left(\frac{b \cdot (\beta_0 + a) + b \cdot \beta_1 z_i + b \cdot (\mu - a)}{b \cdot \sigma}\right) = \Phi\left(\frac{\beta_0 + \beta_1 z_i + \mu}{\sigma}\right),$$

So there is no difference in $\mathbb{P}(y_i = 1|x_i)$ between the model with parameters β_0, β_1, μ and σ and the model with parameters $b \cdot (\beta_0 + a)$, $b \cdot \beta_1$, $b \cdot (\mu - a)$ and $b \cdot \sigma$ (with $-\infty < a < \infty$ and $b > 0$).

Therefore β_0, β_1, μ and σ are **not identified**. We we can set $\mu = 0$ and $\sigma = 1$ *without loss of generality.* Only after imposing these restrictions $\mu = 0$ and $\sigma = 1$, the parameters β_0 and β_1 are identified: a different value of (β_0, β_1) implies a different distribution of y_i (conditional upon x_i).

Execise 2.

The data are in the EViews file `bank_employees.wf1`.

(1) *Change the threshold from 0.5 to $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$. Compare the percentage correctly predicted between the binary probit and binary logit model.*

We have $n = 447$ observations, where $y_i = 0$ for 363 observations and $y_i = 1$ for 84 observations. So $\bar{y} = 84/447 = 0.1879$.

Expectation-Prediction Evaluation for Binary Specification
Equation: BINARY_PROBIT
Success cutoff: C = 0.1879

	Estimated Equation			Constant Probability		
	Dep=0	Dep=1	Total	Dep=0	Dep=1	Total
P(Dep=1)<=C	333	5	338	0	0	0
P(Dep=1)>C	30	79	109	363	84	447
Total	363	84	447	363	84	447
Correct	333	79	412	0	84	84
% Correct	91.74	94.05	92.17	0.00	100.00	18.79
% Incorrect	8.26	5.95	7.83	100.00	0.00	81.21
Total Gain*	91.74	-5.95	73.38			
Percent Gain**	91.74	NA	90.36			

*Change in "% Correct" from default (constant probability) specification
**Percent of incorrect (default) prediction corrected by equation

Expectation-Prediction Evaluation for Binary Specification
Equation: BINARY_LOGIT
Success cutoff: C = 0.1879

	Estimated Equation			Constant Probability		
	Dep=0	Dep=1	Total	Dep=0	Dep=1	Total
P(Dep=1)<=C	333	5	338	0	0	0
P(Dep=1)>C	30	79	109	363	84	447
Total	363	84	447	363	84	447
Correct	333	79	412	0	84	84
% Correct	91.74	94.05	92.17	0.00	100.00	18.79
% Incorrect	8.26	5.95	7.83	100.00	0.00	81.21
Total Gain*	91.74	-5.95	73.38			
Percent Gain**	91.74	NA	90.36			

*Change in "% Correct" from default (constant probability) specification
**Percent of incorrect (default) prediction corrected by equation

Percentage correctly predicted = 92.17% in both models.

Note: this threshold 0.1878 (instead of 0.5) implies that we predict $\tilde{y}_i = 1$ more often (and $\tilde{y}_i = 0$ less often). Now we have 109 predictions $\tilde{y}_i = 1$ instead of 50. In this case this threshold 0.1878 leads to a better percentage correctly predicted of 92.17% instead of 89.71%; the latter does not need to be the case in general.

(2) Change the threshold from 0.5 *to* 0.4*. Compare the percentage correctly predicted between the binary probit and binary logit model.*

Expectation-Prediction Evaluation for Binary Specification
Equation: BINARY_PROBIT
Success cutoff: C = 0.4

	Estimated Equation			Constant Probability		
	Dep=0	Dep=1	Total	Dep=0	Dep=1	Total
P(Dep=1)<=C	357	40	397	363	84	447
P(Dep=1)>C	6	44	50	0	0	0
Total	363	84	447	363	84	447
Correct	357	44	401	363	0	363
% Correct	98.35	52.38	89.71	100.00	0.00	81.21
% Incorrect	1.65	47.62	10.29	0.00	100.00	18.79
Total Gain*	-1.65	52.38	8.50			
Percent Gain**	NA	52.38	45.24			

*Change in "% Correct" from default (constant probability) specification
**Percent of incorrect (default) prediction corrected by equation

Expectation-Prediction Evaluation for Binary Specification
Equation: BINARY_LOGIT
Success cutoff: C = 0.4

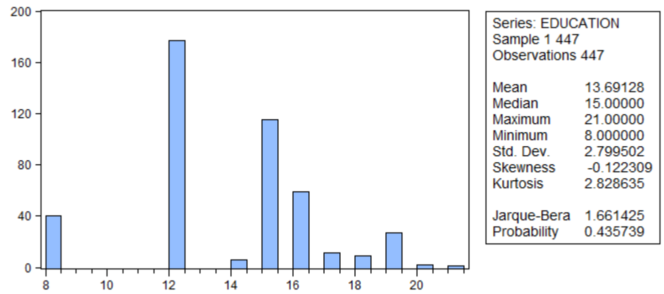
	Estimated Equation			Constant Probability		
	Dep=0	Dep=1	Total	Dep=0	Dep=1	Total
P(Dep=1)<=C	333	5	338	363	84	447
P(Dep=1)>C	30	79	109	0	0	0
Total	363	84	447	363	84	447
Correct	333	79	412	363	0	363
% Correct	91.74	94.05	92.17	100.00	0.00	81.21
% Incorrect	8.26	5.95	7.83	0.00	100.00	18.79
Total Gain*	-8.26	94.05	10.96			
Percent Gain**	NA	94.05	58.33			

*Change in "% Correct" from default (constant probability) specification
**Percent of incorrect (default) prediction corrected by equation

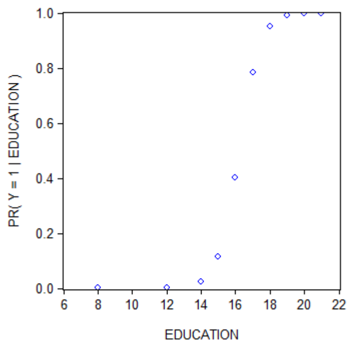
Percentage correctly predicted = 89.71% in binary probit model.
Percentage correctly predicted = 92.17% in binary logit model.
So, for this value of the threshold 0.4 the binary logit model has a better percentage correctly predicted than the binary probit model.

(3) Can you find a threshold so that $\sum_{i=1}^n \tilde{y}_i = \sum_{i=1}^n y_i$*? Motivate your answer.*

No. The reason is that the explanatory variable *education* (which is the only explanatory variable in this model) takes a finite number of values, so that the estimated probability $\hat{\mathbb{P}}(y_i = 1|x_i)$ is exactly the same for groups of individuals that have exactly the same *education*.



We have $\sum_{i=1}^n y_i = 84$ observations with $y_i = 1$ in the actual dataset.



There are 50 individuals with *education* ≥ 17; these have $\hat{\mathbb{P}}(y_i = 1|x_i) > 0.5$ in the binary logit model. There are 59 individuals with *education* $= 16$; these have $\hat{\mathbb{P}}(y_i = 1|x_i) = 0.4036$ in the binary logit model. So:

- We have 50 predictions with $\tilde{y}_i = 1$, if we take a threshold like 0.5, so that each individual with *education* ≥ 17 gets prediction $\tilde{y}_i = 1$ and so that each individual with *education* ≤ 16 gets prediction $\tilde{y}_i = 0$.
- We have $50 + 59 = 109$ predictions with $\tilde{y}_i = 1$, if we take a threshold like 0.4, so that each individual with *education* ≥ 16 gets prediction $\tilde{y}_i = 1$ and so that each individual with *education* ≤ 15 gets prediction $\tilde{y}_i = 0$.
- We can not get exactly 84 observations with $\tilde{y}_i = 1$.

(4) Change the value of education of the last observation from 12 to 120 (to create an extreme outlier with enormous education and $y_i = 0$).
Note: first you may need to click the `Edit +-` *button above the spreadsheet with values. Re-estimate the binary probit and logit models. Compare the percentage correctly predicted between the binary probit and binary logit model. Can you explain the difference in quality between the probit and logit models?*

Note: Here we have a binary logit/probit model, where we have added an outlier by changing the *education* of the last observation to 120 (instead of 12), where $y_i = 0$ for this last observation. So, we created an extreme observation with an extremely high value of education and still $y_i = 0$. The person of the last observation has an administrative job, not a management job.

Note: if the last observation would have $y_i = 1$, then this would not be an outlier! Then the extremely high value of education would simply 'match' with $y_i = 1$, so that changing 12 into 120 would hardly affect the parameter estimates.

For threshold = 0.5 we obtain:

Expectation-Prediction Evaluation for Binary Specification
Equation: BINARY_PROBIT
Success cutoff: C = 0.5

	Estimated Equation			Constant Probability		
	Dep=0	Dep=1	Total	Dep=0	Dep=1	Total
P(Dep=1)<=C	362	84	446	363	84	447
P(Dep=1)>C	1	0	1	0	0	0
Total	363	84	447	363	84	447
Correct	362	0	362	363	0	363
% Correct	99.72	0.00	80.98	100.00	0.00	81.21
% Incorrect	0.28	100.00	19.02	0.00	100.00	18.79
Total Gain*	-0.28	0.00	-0.22			
Percent Gain**	NA	0.00	-1.19			

*Change in "% Correct" from default (constant probability) specification
**Percent of incorrect (default) prediction corrected by equation

Expectation-Prediction Evaluation for Binary Specification
Equation: BINARY_LOGIT
Success cutoff: C = 0.5

	Estimated Equation			Constant Probability		
	Dep=0	Dep=1	Total	Dep=0	Dep=1	Total
P(Dep=1)<=C	359	48	407	363	84	447
P(Dep=1)>C	4	36	40	0	0	0
Total	363	84	447	363	84	447
Correct	359	36	395	363	0	363
% Correct	98.90	42.86	88.37	100.00	0.00	81.21
% Incorrect	1.10	57.14	11.63	0.00	100.00	18.79
Total Gain*	-1.10	42.86	7.16			
Percent Gain**	NA	42.86	38.10			

*Change in "% Correct" from default (constant probability) specification
**Percent of incorrect (default) prediction corrected by equation

Now we see a substantial difference in percentage correctly predicted between the binary probit model and the binary logit model: 80.98% versus 88.37%. (The binary probit model does not even beat the approach of simply predicting $\tilde{y}_i = 0$ for each observation, which has percentage correctly predicted of 81.21%.)

Explanation: the tails of the logistic distribution are fatter than the tails of the normal distribution: outliers can occur in the logistic distribution, so that the parameter estimates are relatively less affected by outliers. In the normal distribution, the presence of one outlier can have a huge effect on the parameter estimates.

Roughly stated, in the binary probit model the parameter estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ are enormously changed in order to "keep the last observation with ($education = 120, y = 0$) out of the extreme tail".

4 Exercises[3]

W17/1

(i) For a binary response y, let $\bar{y}$ be the proportion of ones in the sample (which is equal to the sample average of the y_i). Let $\hat{q}_0$ be the percent correctly predicted for the outcome $y = 0$ and let $\hat{q}_1$ be the percent correctly predicted for the outcome $y = 1$. If $\hat{p}$ is the overall percent correctly predicted, show that $\hat{p}$ is a weighted average of $\hat{q}_0$ and $\hat{q}_1$:

$$\hat{p} = (1 - \bar{y})\hat{q}_0 + \bar{y}\hat{q}_1.$$

Let n_0 denote the number of observations when $y_i = 0$, n_1 be the number of observations when $y_i = 1$ and $n = n_0 + n_1$ the sample size. Moreover, let m_0 denote the number (not the percent) correctly predicted

[3] In "WX/Y" W refers to the book by Wooldridge (Edition 5, but NOT the international edition, which my have slightly different numbers/stories in the exercises than the international edition), X to the chapter number there and Y to the exercise number.

when $y_i = 0$ (so the prediction is also zero) and let m_1 be the number correctly predicted when $y_i = 1$. Then, the proportion correctly predicted is $\frac{m_0+m_1}{n}$. By simple algebra, we can write this as follows:

$$\frac{m_0+m_1}{n} = \frac{n_0}{n} \cdot \frac{m_0}{n_0} + \frac{n_1}{n} \cdot \frac{m_1}{n_1} = (1-\bar{y})\frac{m_0}{n_0} + \bar{y}\frac{m_1}{n_1},$$

where we have used the fact that $\bar{y} = \frac{n_1}{n}$ (the proportion of the sample with $y_i = 1$) and $1 - \bar{y} = \frac{n_0}{n}$ (the proportion of the sample with $y_i = 0$).

Next, notice that $\frac{m_0}{n_0}$ is the proportion correctly predicted when $y_i = 0$, and $\frac{m_1}{n_1}$ is the proportion correctly predicted when $y_i = 1$. Therefore, we have

$$\frac{m_0+m_1}{n} = (1-\bar{y})\frac{m_0}{n_0} + \bar{y}\frac{m_1}{n_1}.$$

Multiplying both sides by 100 yields

$$\hat{p} = (1-\bar{y})\hat{q}_0 + \bar{y}\hat{q}_1, \tag{6}$$

where we use the fact that, by definition,

$$\hat{p} = 100 \cdot \frac{m_0+m_1}{n}, \qquad \hat{q}_0 = 100 \cdot \frac{m_0}{n_0}, \qquad \hat{q}_1 = 100 \cdot \frac{m_1}{n_1}.$$

(ii) In a sample of 300, suppose that $\bar{y} = 0.70$, so that there are 210 outcomes with $y_1 = 1$ and 90 with $y_i = 0$. Suppose that the percent correctly predicted when $y = 0$ is 80, and the percent correctly predicted when $y = 1$ is 40. Find the overall percent correctly predicted.

We just use formula (6) from part *(i)*:

$$\hat{p} = 0.30 \cdot 80 + 0.70 \cdot 40 = 52.$$

Therefore, overall we correctly predict only 52% of the outcomes. This is because, while 80% of the time we correctly predict $y = 0$, the observations where $y_i = 0$ account for only 30 percent of the outcomes. More weight (i.e. 0.70) is given to the predictions when $y_i = 1$, and we do much less well predicting that outcome (getting it right only 40% of the time).

W17/2

Let grad be a dummy variable for whether a student-athlete at a large university graduates in five years. Let hsGPA and SAT be high school grade point average and SAT score, respectively. Let study be the number of hours spent per week in an organized study hall. Suppose that, using data on 420 student-athletes, the following logit model is obtained:

$$\hat{\mathbb{P}}(grad = 1 | hsGPA, SAT, study) = \Lambda(-1.17 + 0.24\, hsGPA + 0.00058\, SAT + 0.073\, study),$$

where $\Lambda(z) = \exp(z)/[1+\exp(z)]$ is the logit function. Holding hsGPA fixed at 3.0 and SAT fixed at 1,200, compute the estimated difference in the graduation probability for someone who spent 10 hours per week in study hall and someone who spent 5 hours per week.

We first need to compute the estimated probability at $hsGPA = 3.0$, $SAT = 1,200$, and $study = 10$, second at $hsGPA = 3.0$, $SAT = 1,200$, and $study = 5$, and then the former from the latter.

To obtain the first probability, we start by computing the linear function inside $\Lambda(\cdot)$:

$$-1.17 + 0.24 \cdot hsGPA + 0.00058 \cdot SAT + 0.073 \cdot study =$$
$$-1.17 + 0.24 \cdot 3.0 + 0.00058 \cdot 1,200 + 0.073 \cdot 10 = 0.9760.$$

Next, we plug this into the logit function:

$$\frac{\exp(0.9760)}{1+\exp(0.9760)} \approx 0.7263.$$

This is the estimated probability that a student-athlete with the given characteristics graduates in five years.

For the student-athlete who attended study hall five hours a week, we compute:

$$-1.17 + 0.24 \cdot 3.0 + 0.00058 \cdot 1,200 + 0.073 \cdot 5 = 0.6110.$$

Evaluating the logit function at this value gives

$$\frac{\exp(0.6110)}{1 + \exp(0.6110)} \approx 0.6482.$$

Therefore, the difference in estimated probabilities is

$$0.7263 - 0.6482 = 0.0781,$$

which is under 0.10.

Note how far off the calculation would be if we simply use the coefficient on study (in the linear function inside Λ) to conclude that the difference in probabilities is $0.073 \cdot (10–5) = 0.365$.

5 Computer Exercises

W17/C1

Use the data in `pntsprd.wf1`[4] *for this exercise.*

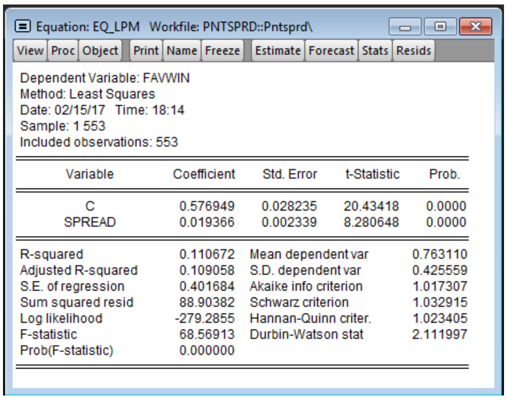
(i) *The variable favwin is a binary variable if the team favoured by the Las Vegas point spread wins. A linear probability model to estimate the probability that the favoured team wins is*

$$\mathbb{P}(favin = 1|spread) = \beta_0 + \beta_1 spread.$$

Explain why, if the spread incorporates all relevant information, we expect $\beta_0 = 0.5$.

If *spread* is zero, there is no favourite, and the probability that the team we (arbitrarily) label the favourite should have a 50% chance of winning.

(ii) *Estimate the model from part (i) by OLS. Test* $H_0 : \beta_0 = 0.5$ *against a two-sided alternative.*


Equation: EQ_LPM Workfile: PNTSPRD::Pntsprd\

View | Proc | Object | Print | Name | Freeze | Estimate | Forecast | Stats | Resids

Dependent Variable: FAVWIN
Method: Least Squares
Date: 02/15/17 Time: 18:14
Sample: 1 553
Included observations: 553

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.576949	0.028235	20.43418	0.0000
SPREAD	0.019366	0.002339	8.280648	0.0000

R-squared	0.110672	Mean dependent var	0.763110
Adjusted R-squared	0.109058	S.D. dependent var	0.425559
S.E. of regression	0.401684	Akaike info criterion	1.017307
Sum squared resid	88.90382	Schwarz criterion	1.032915
Log likelihood	-279.2855	Hannan-Quinn criter.	1.023405
F-statistic	68.56913	Durbin-Watson stat	2.111997
Prob(F-statistic)	0.000000		

The linear probability model estimated by OLS gives

$$\begin{aligned}\widehat{favwin} = &0.577 + 0.0194\, spread \\ &(0.028)\ (0.0023)\end{aligned}$$

with $n = 553$ and $R^2 = 0.111$, where the usual standard errors are in parentheses. Using the usual standard error, the t statistic for $H_0 : \beta_0 = 0.5$ is

$$\frac{0.577 - 0.500}{0.028} = 2.75,$$

which leads to rejecting H_0 against a two-sided alternative at the 1% level (critical value ≈ 2.58).

[4] $N = 553$, cross-sectional gambling point spread data for the 1994–1995 men's college basketball seasons. The spread is for the day before the game was played.

(*iii*) *Is spread statistically significant? What is the estimated probability that the favoured team wins when* $spread = 10$*?*

The t–statistic for $H_0 : \beta_1 = 0.0$ is

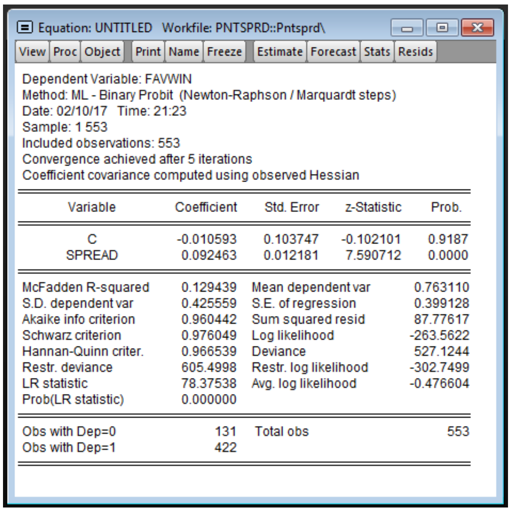
$$\frac{0.0194 - 0}{0.0023} = 8.4348,$$

so as we expect, *spread* is very statistically significant.

If $spread = 10$ the estimated probability that the favoured team wins is

$$0.577 + 0.0194 \cdot 10 = 0.771.$$

(*iv*) *Now, estimate a probit model for* $P(favwin = 1|spread)$*. Interpret and test the null hypothesis that the intercept is zero. [Hint: Remember that* $\Phi(0) = 0.5$*.]*



Equation: UNTITLED Workfile: PNTSPRD::Pntsprd\

View | Proc | Object | Print | Name | Freeze | Estimate | Forecast | Stats | Resids

Dependent Variable: FAVWIN
Method: ML - Binary Probit (Newton-Raphson / Marquardt steps)
Date: 02/10/17 Time: 21:23
Sample: 1 553
Included observations: 553
Convergence achieved after 5 iterations
Coefficient covariance computed using observed Hessian

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	-0.010593	0.103747	-0.102101	0.9187
SPREAD	0.092463	0.012181	7.590712	0.0000

McFadden R-squared	0.129439	Mean dependent var	0.763110
S.D. dependent var	0.425559	S.E. of regression	0.399128
Akaike info criterion	0.960442	Sum squared resid	87.77617
Schwarz criterion	0.976049	Log likelihood	-263.5622
Hannan-Quinn criter.	0.966539	Deviance	527.1244
Restr. deviance	605.4998	Restr. log likelihood	-302.7499
LR statistic	78.37538	Avg. log likelihood	-0.476604
Prob(LR statistic)	0.000000		

Obs with Dep=0	131	Total obs	553
Obs with Dep=1	422		

In the Probit model

$$\mathbb{P}(favwin = 1|spread) = \Phi(\beta_0 + \beta_1 spread),$$

where $\Phi(\cdot)$ denotes the standard normal cdf. If $\beta_0 = 0$, then

$$\mathbb{P}(favwin = 1|spread) = \Phi(\beta_1 spread)$$

and, in particular,

$$\mathbb{P}(favwin = 1|spread = 0) = \Phi(0) = 0.5.$$

This is the analog of testing whether the intercept is 0.5 in the LPM. From the EViews output, the t statistic (or, actually , the z statistic, only valid asymptotically) for testing $H_0 : \beta_0 = 0$ is only about -0.102 so there are no grounds to reject H_0.

(*v*) *Use the probit model to estimate the probability that the favoured team wins when* $spread = 10$*. Compare this with the LPM estimate from part (iii).*

When $spread = 10$ the predicted response probability from the estimated probit model is

$$\Phi(-0.0106 + 0.0925 \cdot 10) = \Phi(0.9144) \approx 0.820.$$

This is somewhat above the estimate for the LPM.

(*vi*) *Add the variables* $favhome$*,* $fav25$*, and* $und25$ *to the probit model and test joint significance of these variables using the likelihood ratio test. (How many df are in the chi-square distribution?) Interpret this result, focusing on the question of whether the spread incorporates all observable information prior to a game.*

Equation: EQ_PROBIT2 Workfile: PNTSPRD::Pntsprd\

View | Proc | Object | Print | Name | Freeze | Estimate | Forecast | Stats | Resids

Dependent Variable: FAVWIN
Method: ML - Binary Probit (Newton-Raphson / Marquardt steps)
Date: 02/15/17 Time: 18:47
Sample: 1 553
Included observations: 553
Convergence achieved after 4 iterations
Coefficient covariance computed using observed Hessian

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	-0.055180	0.128763	-0.428540	0.6683
SPREAD	0.087884	0.012949	6.786915	0.0000
FAVHOME	0.148575	0.137057	1.084039	0.2783
FAV25	0.003068	0.158690	0.019333	0.9846
UND25	-0.219808	0.250584	-0.877183	0.3804

McFadden R-squared	0.132479	Mean dependent var	0.763110
S.D. dependent var	0.425559	S.E. of regression	0.399241
Akaike info criterion	0.967963	Sum squared resid	87.34770
Schwarz criterion	1.006981	Log likelihood	-262.6418
Hannan-Quinn criter.	0.983207	Deviance	525.2835
Restr. deviance	605.4998	Restr. log likelihood	-302.7499
LR statistic	80.21622	Avg. log likelihood	-0.474940
Prob(LR statistic)	0.000000		

Obs with Dep=0	131	Total obs	553
Obs with Dep=1	422		

When $favhome$, $fav25$, and $und25$ are added to the probit model, the value of the loglikelihood becomes –262.64, while it used to be −263.56. Therefore, the likelihood ratio statistic is

$$2 \cdot [-262.64–(-263.56)] = 2 \cdot (263.56–262.64) = 1.84.$$

The p-value from the χ^2_3 (df = 3 because we add 3 variables) distribution is about 0.61, so $favhome$, $fav25$, and $und25$ are jointly very insignificant. Once spread is controlled for, these other factors have no additional power for predicting the outcome.

W17/C2

Use the data in `loanapp.wf1`[5] *for this exercise; see also Computer Exercise C8 in Chapter 7.*

(i) *Estimate a probit model of approve on white. Find the estimated probability of loan approval for both whites and nonwhites. How do these compare with the linear probability estimates?*

Equation: EQ_PROBIT Workfile: LOANAPP::Loanapp\

View | Proc | Object | Print | Name | Freeze | Estimate | Forecast | Stats | Resids

Dependent Variable: APPROVE
Method: ML - Binary Probit (Newton-Raphson / Marquardt steps)
Sample (adjusted): 1 1988
Included observations: 1988 after adjustments
Convergence achieved after 3 iterations
Coefficient covariance computed using observed Hessian

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	0.546946	0.075435	7.250563	0.0000
WHITE	0.783615	0.086714	9.036738	0.0000

McFadden R-squared	0.053274	Mean dependent var	0.877264
S.D. dependent var	0.328217	S.E. of regression	0.320172
Akaike info criterion	0.707023	Sum squared resid	203.5846
Schwarz criterion	0.712652	Log likelihood	-700.7813
Hannan-Quinn criter.	0.709091	Deviance	1401.563
Restr. deviance	1480.431	Restr. log likelihood	-740.2157
LR statistic	78.86870	Avg. log likelihood	-0.352506
Prob(LR statistic)	0.000000		

Obs with Dep=0	244	Total obs	1988
Obs with Dep=1	1744		

As there is only one explanatory variable that takes on just two values, there are only two different predicted values: the estimated probabilities of loan approval for white and nonwhite applicants. Rounded to three decimal places these are:

$$\mathbb{P}(approve = 1|white = 0) = \Phi(\beta_0 + \beta_1 \cdot 0) = \Phi(0.547) = 0.708,$$
$$\mathbb{P}(approve = 1|white = 1) = \Phi(\beta_0 + \beta_1 \cdot 1) = \Phi(0.547 + 0.784) = 0.908,$$

[5] N = 1989, cross-sectional individual data. These data were originally used in a famous study by researchers at the Boston Federal Reserve Bank. See A. Munnell, G.M.B. Tootell, L.E. Browne, and J. McEneaney (1996), "Mortgage Lending in Boston: Interpreting HMDA Data", *American Economic Review* 86, 25–53.

for nonwhites and whites, respectively. Without rounding errors, these are identical to the fitted values from the linear probability model. This must always be the case when the independent variables in a binary response model are mutually exclusive and exhaustive binary variables. Then, the predicted probabilities, whether we use the LPM, probit, or logit models, are simply the cell frequencies.

(In other words, 0.708 is the proportion of loans approved for nonwhites and 0.908 is the proportion approved for whites.)

(ii) Now, add the variables hrat, obrat, loanprc, unem, male, married, dep, sch, cosign, chist, pubrec, mortlat1, mortlat2, and vr to the probit model. Is there statistically significant evidence of discrimination against nonwhites?

Equation: EQ_PROBIT2 Workfile: LOANAPP::Loanapp\

View | Proc | Object | Print | Name | Freeze | Estimate | Forecast | Stats | Resids

Dependent Variable: APPROVE
Method: ML - Binary Probit (Newton-Raphson / Marquardt steps)
Sample (adjusted): 1 1988
Included observations: 1971 after adjustments
Convergence achieved after 3 iterations
Coefficient covariance computed using observed Hessian

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	2.062327	0.313176	6.585194	0.0000
WHITE	0.520253	0.096959	5.365707	0.0000
HRAT	0.007876	0.006962	1.131394	0.2579
OBRAT	-0.027692	0.006049	-4.577783	0.0000
LOANPRC	-1.011969	0.237240	-4.265600	0.0000
UNEM	-0.036685	0.017481	-2.098594	0.0359
MALE	-0.037001	0.109927	-0.336599	0.7364
MARRIED	0.265747	0.094252	2.819528	0.0048
DEP	-0.049576	0.039057	-1.269304	0.2043
SCH	0.014650	0.095842	0.152851	0.8785
COSIGN	0.086071	0.245751	0.350238	0.7262
CHIST	0.585281	0.095971	6.098491	0.0000
PUBREC	-0.778741	0.126320	-6.164823	0.0000
MORTLAT1	-0.187624	0.253113	-0.741265	0.4585
MORTLAT2	-0.494356	0.326556	-1.513847	0.1301
VR	-0.201062	0.081493	-2.467220	0.0136

McFadden R-squared	0.186602	Mean dependent var	0.876205
S.D. dependent var	0.329431	S.E. of regression	0.299475
Akaike info criterion	0.625338	Sum squared resid	175.3347
Schwarz criterion	0.670686	Log likelihood	-600.2710
Hannan-Quinn criter.	0.642002	Deviance	1200.542
Restr. deviance	1475.959	Restr. log likelihood	-737.9793
LR statistic	275.4167	Avg. log likelihood	-0.304551
Prob(LR statistic)	0.000000		

Obs with Dep=0	244	Total obs	1971
Obs with Dep=1	1727		

With the set of controls added, the probit estimate on white becomes about 0.520 with the standard error of around 0.097. Therefore, there is still very strong evidence of discrimination against nonwhites.

[We can divide this by 2.5 to make it roughly comparable to the LPM estimate in part *(iii)* of Computer Exercise C7.8: $0.520/2.5 \approx 0.208$, compared with 0.129 in the LPM.]

(iii) Estimate the model from part (ii) by logit. Compare the coefficient on white to the probit estimate.

Equation: EQ_LOGIT2 Workfile: LOANAPP::Loanapp\

View | Proc | Object | Print | Name | Freeze | Estimate | Forecast | Stats | Resids

Dependent Variable: APPROVE
Method: ML - Binary Logit (Newton-Raphson / Marquardt steps)
Sample (adjusted): 1 1988
Included observations: 1971 after adjustments
Convergence achieved after 4 iterations
Coefficient covariance computed using observed Hessian

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	3.801710	0.594707	6.392572	0.0000
WHITE	0.937764	0.172904	5.423603	0.0000
HRAT	0.013263	0.012880	1.029730	0.3031
OBRAT	-0.053034	0.011280	-4.701462	0.0000
LOANPRC	-1.904951	0.460443	-4.137212	0.0000
UNEM	-0.066579	0.032809	-2.029310	0.0424
MALE	-0.066385	0.206429	-0.321588	0.7478
MARRIED	0.503282	0.177998	2.827452	0.0047
DEP	-0.090734	0.073334	-1.237261	0.2160
SCH	0.041229	0.178404	0.231098	0.8172
COSIGN	0.132059	0.446094	0.296034	0.7672
CHIST	1.066577	0.171212	6.229570	0.0000
PUBREC	-1.340665	0.217366	-6.167781	0.0000
MORTLAT1	-0.309882	0.463520	-0.668541	0.5038
MORTLAT2	-0.894675	0.568581	-1.573522	0.1156
VR	-0.349828	0.153725	-2.275671	0.0229

McFadden R-squared	0.186297	Mean dependent var	0.876205
S.D. dependent var	0.329431	S.E. of regression	0.299487
Akaike info criterion	0.625567	Sum squared resid	175.3487
Schwarz criterion	0.670915	Log likelihood	-600.4962
Hannan-Quinn criter.	0.642230	Deviance	1200.992
Restr. deviance	1475.959	Restr. log likelihood	-737.9793
LR statistic	274.9664	Avg. log likelihood	-0.304666
Prob(LR statistic)	0.000000		

Obs with Dep=0	244	Total obs	1971
Obs with Dep=1	1727		

When we use logit instead of probit, the coefficient on *white* becomes 0.938 with the standard error of 0.173.

[Recall that to make probit and logit estimates roughly comparable, we can multiply the logit estimates by 0.625. The scaled logit coefficient becomes: $0.625 \cdot 0.938 \approx 0.586$, which is reasonably close to the probit estimate of 0.520. A better comparison would be to compare the predicted probabilities by setting the other controls at interesting values, such as their average values in the sample.]

(iv) Use equation

$$n^{-1}\sum_{i=1}^{n}\Big\{G\big[\hat{\beta}_0+\hat{\beta}_1 x_{i1}+\cdots+\hat{\beta}_{k-1}x_{ik-1}+\hat{\beta}_k(c_k+1)\big]-G\big[\hat{\beta}_0+\hat{\beta}_1 x_{i1}+\cdots+\hat{\beta}_{k-1}x_{ik-1}+\hat{\beta}_k c_k\big]\Big\} \quad (17.17)$$

to estimate the sizes of the discrimination effects for probit and logit.

Note that (17.17) is the average partial effect for a discrete explanatory variable. Unfortunately, it seems there is no build-in function for this measure in EViews, so we need to calculate it ourselves using the estimation results from the "augmented" probit and logit models. Figure 5 presents a code to carry out such computations.

We consider all the variables but *white*. Instead, for each individual we consider two counterfactual scenarios: as if he or she was white and otherwise (new generated variables `white1` and `white0`), which we use to create two `groups` (`variables_white1` and `variables_white0`). Then, we use the coefficients from two estimations (`coef_probit` and `coef_logit`) to sum all the variables multiplied by their respective coefficient.

This gives us the arguments inside $G(\cdot)$ in (17.17). To evaluate $G(\cdot)$ we need to apply the appropriate function for each model. For probit, it is $\Phi(z)$, the cdf of the standard normal distribution; for logit, it is $\frac{1}{1+\exp(-z)}$. Finally, we subtract the vector with $G(\cdot)$ applied to the sum under the "nonwhites scenario" from that under the "whites scenario" and average out. The obtained values are $APE_{probit} = 0.1042$ and $APE_{logit} = 0.1009$, hence quite similar.

Program: LOANAPP - (h:\desktop\loanapp.prg)

Run | Print | Save | SaveAs | Cut | Copy | Paste | InsertTxt | Find | Replace | Wrap+/- | LineNum+/- | Encrypt

```
equation eq_probit.binary(d=n) approve c white hrat obrat loanprc unem male married dep sch cosign chist pubrec mortlat1 mortlat2 vr
equation eq_logit.binary(d=l) approve c white hrat obrat loanprc unem male married dep sch cosign chist pubrec mortlat1 mortlat2 vr

vector(16) coef_probit
coef_probit = eq_probit.@coefs

vector(16) coef_logit
coef_logit = eq_logit.@coefs

' counterfactual scenarios
genr white1 = 1
genr white0 = 0

' all variables under counterfactual scenarios
group variables_white1 white1 hrat obrat loanprc unem male married dep sch cosign chist pubrec mortlat1 mortlat2 vr
group variables_white0 white0 hrat obrat loanprc unem male married dep sch cosign chist pubrec mortlat1 mortlat2 vr

' sum inside the G functions
series sum_white0_probit
series  sum_white1_probit
series  sum_white0_logit
series  sum_white1_logit

' start summing with the intercept (beta0)
sum_white0_probit = coef_probit(1)
sum_white1_probit = coef_probit(1)
sum_white0_logit = coef_logit(1)
sum_white1_logit = coef_logit(1)

' add subsequent variables multiplied by their coefficients
' (there are more coefs because the one for the constant term is also there - hence !i-1 for the grouped variables)
for !i = 2 to 16
    series temp =  coef_probit(!i)* variables_white0(!i-1)
    sum_white0_probit = sum_white0_probit + temp

    series temp =  coef_probit(!i)* variables_white1(!i-1)
    sum_white1_probit = sum_white1_probit + temp

    series temp =  coef_logit(!i)* variables_white0(!i-1)
    sum_white0_logit = sum_white0_logit + temp

    series temp =  coef_logit(!i)* variables_white1(!i-1)
    sum_white1_logit = sum_white1_logit + temp
next

' for probit: compute G as the cdf of the standard normal distribution
series G_white0_probit  = @cnorm(sum_white0_probit )
series G_white1_probit  = @cnorm(sum_white1_probit )
series diff_probit = G_white1_probit  - G_white0_probit
scalar apf_probit = @mean(diff_probit )

' for logit: compute G as the logistic function
series G_white0_logit  = 1/(1+@exp(-sum_white0_logit))
series G_white1_logit  = 1/(1+@exp(-sum_white1_logit))
series diff_logit = G_white1_logit  - G_white0_logit
scalar apf_logit = @mean(diff_logit )
```

Figure 5: EViews code for computing APE for probit and logit models, where we are interested in the effect of being white or not on loan approval.