SciencesPo
DEPARTMENT OF ECONOMICS

ScPoEconometrics: Advanced

Binary Response Models

Nikiforos Zampetakis based on Florian Oswald's slides
SciencesPo Paris
2024-03-27

Where Are We At?

Last Time

- Panel Data Estimation
- The *fixed effects estimator*
- `fixest` package

Today

1. Binary Response Models!
2. Another cool app! 😎

Binary Response Models

Binary Response Models

So far, our models looked like this:

$$
\begin{aligned}
y &= b_0 + b_1 x + e \\
e &\sim N\left(0, \sigma^2\right)
\end{aligned}
$$

- The distributional assumption on e:
- In priniciple implies that $y \in \mathbb{R}$.
- test scores, earnings, crime rates etc are all continuous outcomes. ✅

But some outcomes are clearly binary (i.e. either `TRUE` or `FALSE`):

- You either work or you don't,
- You either have children or you don't,
- You either bought a product or you didn't,
- You flipped a coin and it came up either heads or tails.

Binary Outcomes

- Outcomes restricted to `FALSE` vs `TRUE`, or `0` vs `1`.
- We'd have $y \in \{0,1\}$.
- In those situations we are primarily interested in estimating the **response probability** or the **probability of success**:

$$p(x) = \Pr(y = 1|x)$$

- how does $p(x)$ change as we change x?
- we ask

> If we increase x by one unit, how would the probability of $y = 1$ change?

Remembering Bernoulli Fun

Remember the Bernoulli Distribution?: We call a random variable $y \in \{0, 1\}$ such that

$$\begin{aligned} \Pr(y = 1) &= p \\ \Pr(y = 0) &= 1 - p \\ p &\in [0, 1] \end{aligned}$$

a *Bernoulli* random variable.

For us: *condition* those probabilities on a covariate x

$$\begin{aligned} \Pr(y = 1|X = x) &= p(x) \\ \Pr(y = 0|X = x) &= 1 - p(x) \\ p(x) &\in [0, 1] \end{aligned}$$

- Partcularly: *expected value* (i.e. the average) of Y given x

$$E[y|x] = p(x) \times 1 + (1 - p(x)) \times 0 = p(x)$$

- We often model **conditional expectations** 😉

The Linear Probability Model (LPM)

- The simplest option. Model the response probability as

$$\Pr(y=1|x) = p(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_K x_K$$

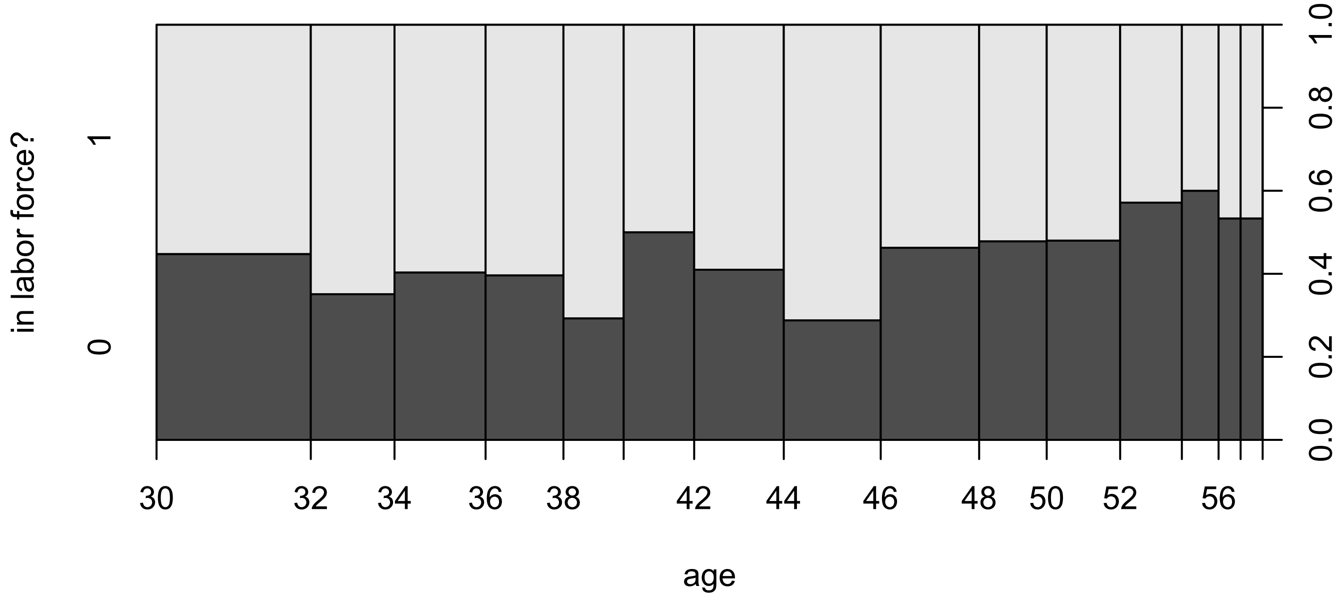
- Interpretation: *a 1 unit change in x_1, say, results in a change of $p(x)$ of β_1.*

Example: Mroz (1987)

- Female labor market participation
- How does `inlf` (*in labor force*) status depend on non-wife household income, her education, age and number of small children?

Mroz 1987

```r
data(mroz, package = "wooldridge")
plot(factor(inlf) ~ age, data = mroz,
     ylevels = 2:1,
     ylab = "in labor force?")
```



Task 1 (5 Minutes)

1. What is the unit of observation in this data set?
2. How many rows and columns of data do we have?
3. What is the unconditional probability of being in the labor force?
4. What is the unconditional mean of being in the labor force?
5. What is the conditional probability of being in the labor force conditional on the number of kids less than 6 years old?

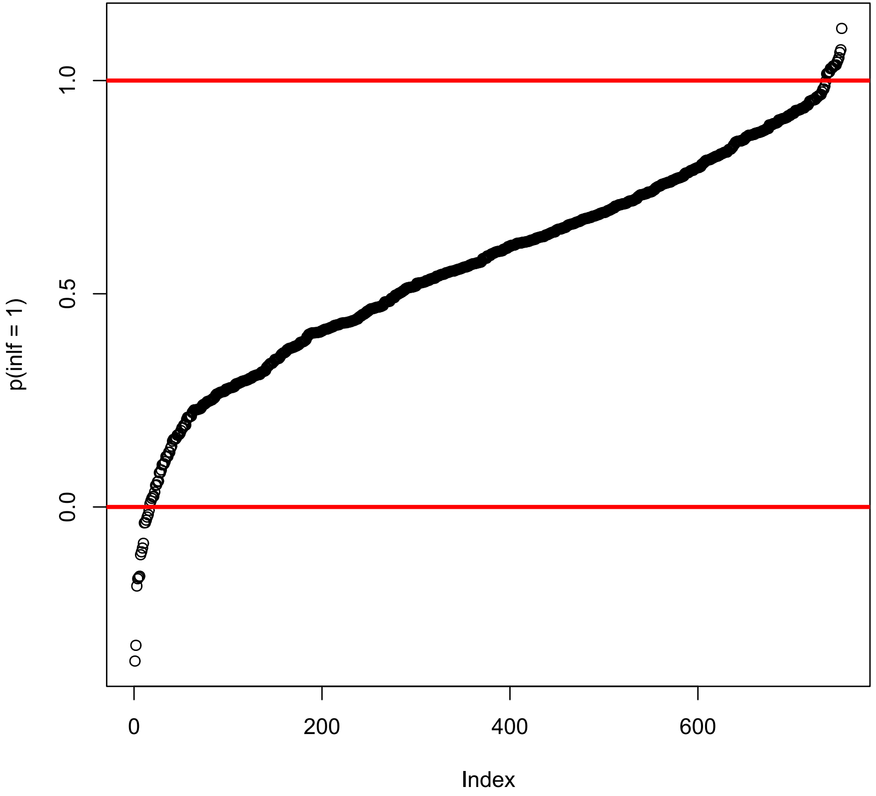
Running the LPM

- **identical** to our previous linear regression models
- Just `inlf` takes on only two values, 0 or 1.
- Results: non-wife income increases by 10 (i.e 10,000 USD), $p(x)$ falls by 0.034 (that's a small effect!),
- an additional small child would reduce the probability of work by 0.26 (that's large).
- So far, so simple. ✌️

```
LPM = lm(inlf ~ nwifeinc + educ + exper
         + I(exper^2) + age +I(age^2) + kidslt6, mroz)
broom::tidy(LPM)
```

```
## # A tibble: 8 × 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  0.322     0.486       0.662  5.08e- 1
## 2 nwifeinc    -0.00343   0.00145    -2.36   1.86e- 2
## 3 educ         0.0375    0.00735     5.10   4.33e- 7
## 4 exper        0.0383    0.00577     6.63   6.44e-11
## 5 I(exper^2)  -0.000565  0.000189   -2.98   2.96e- 3
## 6 age         -0.00112   0.0225     -0.0497 9.60e- 1
## 7 I(age^2)    -0.000182  0.000258   -0.706  4.80e- 1
## 8 kidslt6     -0.260     0.0341     -7.64   6.72e-14
```

LPM: Predicting negative probabilities?!



- LPM predictions of $p(x)$ are not guaranteed to lie in unit interval $[0,1]$.
- Remember: $e \sim N\left(0, \sigma^2\right)$
- here, some probs smaller than zero!
- Particularly annoying if you want *predictions*: What is a probability of -0.3? 🤔

LPM in Saturated Model: No Problem!

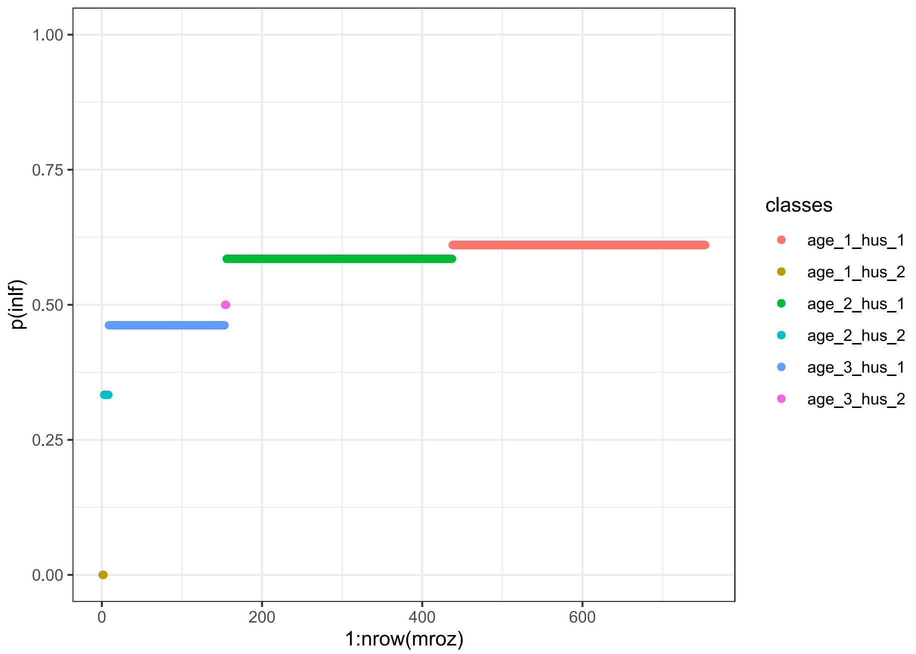
- *saturated model* : only have dummy explanatory variables
- Each class: $p(x)$ *within that cell.*

```
library(dplyr)
mroz %<>%
  # classify age into 3 and huswage into 2 classes
  mutate(age_fct = cut(age,breaks = 3,labels = FALSE),
         huswage_fct = cut(huswage, breaks = 2,labels = FALSE)) %>%
  mutate(classes = paste0("age_",age_fct,"_hus_",huswage_fct))

LPM_saturated = mroz %>%
  lm(inlf ~ classes, data = .)
broom::tidy(LPM_saturated)
```

```
## # A tibble: 6 × 5
##   term                estimate std.error statistic  p.value
##   <chr>                  <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)           0.611     0.0277    22.0   2.98e-83
## 2 classesage_1_hus_2  -0.611     0.350     -1.75  8.11e- 2
## 3 classesage_2_hus_1  -0.0257    0.0404    -0.635 5.25e- 1
## 4 classesage_2_hus_2  -0.277     0.203     -1.37  1.72e- 1
## 5 classesage_3_hus_1  -0.149     0.0494    -3.01  2.72e- 3
## 6 classesage_3_hus_2  -0.111     0.350     -0.317 7.51e- 1
```

LPM in Saturated Model: No Problem!



- Each line segment: $p(x)$ *within that cell.*
- E.g. women from the youngest age category and lowest husband income (class `age_1_hus_1`) have the highest probability of working (0.611).

Task 2 (10 Minutes): Saturated LPM

Define a *saturated* LPM as before

$$\Pr(y = 1|x) = p(x) = \beta_0 + \beta_1 x_1 + \cdots + \beta_K x_K$$

but restrict all $x_j \in \{0, 1\}$.

1. Create a binary indicator `age_lt_50 = 1` for age smaller than 50 and `0` else and same for `husage_lt_50`.
2. Run a full interactions model (use the `*` syntax in your formula) of `age_lt_50 = 1` interacted with `husage_lt_50`. I.e. run the following LPM:

$$\Pr(y = 1|x) = \beta_0 + \beta_1 \text{age\_lt\_50} + \beta_2 \text{husage\_lt\_50} + \beta_3 \times \text{age\_lt\_50} \times \text{husage\_lt\_50}$$

3. `predict` $\Pr(y = 1|x)$ for each observation using your LPM.
4. What's the probability for a woman younger than 50 with a husband younger than 50?
5. make a plot similar to the one on the previous slide.

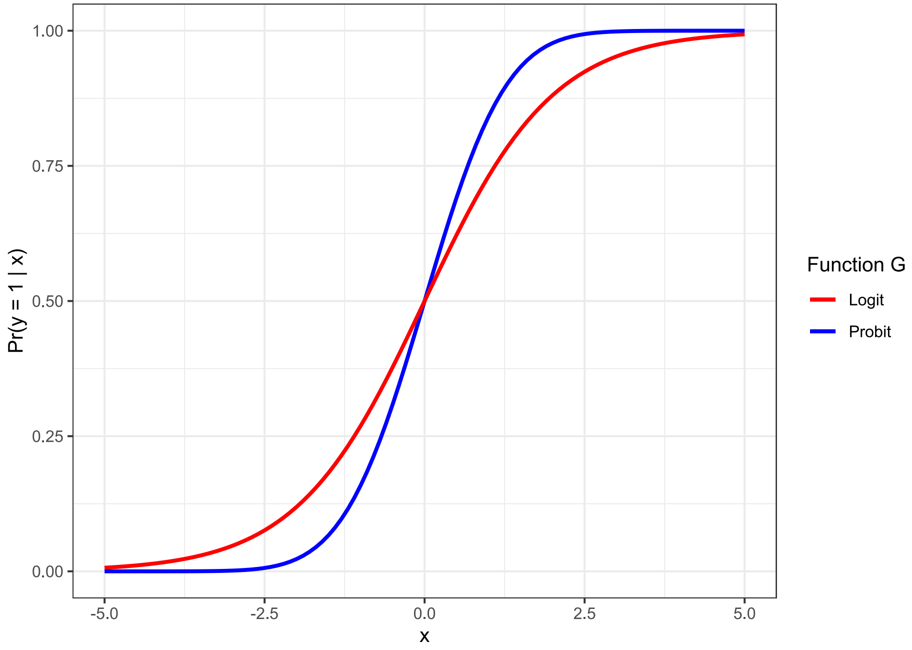
Nonlinear Binary Response Models

In this class of models we change the way we model the response probability $p(x)$. Instead of the simple linear structure from above, we write

$$\Pr(y = 1|x) = p(x) = G\left(\beta_0 + \beta_1 x_1 + \cdots + \beta_K x_K\right)$$

- *almost* identical to LPM!
- except the *linear index* $\beta_0 + \beta_1 x_1 + \cdots + \beta_K x_K$ is now inside some function $G(\cdot)$.
- Main property of G: transforms any $z \in \mathbb{R}$ into a number in the interval $(0, 1)$.
- This immediately solves our problem of getting weird predictions for probabilities.

G: probit and logit



For both **probit** and **logit** we see that:

1. any value x results in a value $p(x)$ between 0 and 1
2. the higher x, the higher the resulting $p(x)$.
3. Logit has *fatter tails* than Probit.

Running probit and logit in R: the `glm` function

- We use the `glm` function to run a **generalized linear model**
- This *generalizes* our standard linear model. We have to specify a `family` and a `link`:

```
probit <- glm(inlf ~ age,
                    data = mroz,
                    family = binomial(link = "probit"))

logit <- glm(inlf ~ age,
                    data = mroz,
                    family = binomial(link = "logit"))
```

Interpretation

```
modelsummary::modelsummary(list("probit" = probit,'
```

	probit	logit
(Intercept)	0.707	1.136
	(0.248)	(0.398)
age	-0.013	-0.020
	(0.006)	(0.009)
Num.Obs.	753	753
AIC	1028.9	1028.9
BIC	1038.1	1038.1
Log.Lik.	-512.442	-512.431
F	4.828	4.858
RMSE	0.49	0.49

- probit coefficient for `age` is -0.013
- logit: -0.02 for logit,
- impact of age on the prob of working is **negative**
- However, **how** negative? We can't tell!

Interpretation

The model is

$$\Pr(y = 1 | \text{age}) = G\left(x\beta\right) = G\left(\beta_0 + \beta_1 \text{age}\right)$$

and the *marginal effect* of `age` on the response probability is

$$\frac{\partial \Pr(y = 1 | \text{age})}{\partial \text{age}} = g\left(\beta_0 + \beta_1 \text{age}\right) \beta_1$$

- function g is defined as $g(z) = \frac{dG}{dz}(z)$ - the first derivative function of G (i.e. the *slope* of G).
- given G that is nonlinear, this means that g will be non-constant. You are able to try this out yourself using this app here:

```
ScPoApps::launchApp("marginal_effects_of_logit_probit")
```

or online

Interpretation

So you can see that there is not one single *marginal effect* in those models, as that depends on *where we evaluate* the previous expression. In practice, there are two common approaches:

1. report effect at the average values of x:

$$g(\bar{x}\beta)\beta_j$$

2. report the sample average of all marginal effects:

$$\frac{1}{n}\sum_{i=1}^{N} g(x_i\beta)\beta_j$$

Thankfully there are packages available that help us to compute those marginal effects fairly easily. One of them is called `mfx`, and we would use it as follows:

Interpretation

```r
f <- "inlf ~ age + kidslt6 + nwifeinc" # setup a formula
glms <- list()
glms$probit <- glm(formula = f,
                    data = mroz,
                    family = binomial(link = "probit"))
glms$logit <- glm(formula = f,
                    data = mroz,
                    family = binomial(link = "logit"))
# now the marginal effects versions
glms$probitMean <- mfx::probitmfx(formula = f,
                    data = mroz, atmean = TRUE)
glms$probitAvg <- mfx::probitmfx(formula = f,
                    data = mroz, atmean = FALSE)
glms$logitMean <- mfx::logitmfx(formula = f,
                    data = mroz, atmean = TRUE)
glms$logitAvg <- mfx::logitmfx(formula = f,
                    data = mroz, atmean = FALSE)
```

Interpretation

```
## Call:
## mfx::probitmfx(formula = f, data = mroz, atmean = TRUE)
##
## Marginal Effects:
##               dF/dx  Std. Err.       z     P>|z|
## age      -0.0136710  0.0026087 -5.2406 1.601e-07 ***
## kidslt6  -0.3139105  0.0435115 -7.2144 5.416e-13 ***
## nwifeinc -0.0044957  0.0016463 -2.7308  0.006317 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

```
## Call:
## mfx::probitmfx(formula = f, data = mroz, atmean = FALSE)
##
## Marginal Effects:
##               dF/dx  Std. Err.       z     P>|z|
## age      -0.0126459  0.0022822 -5.5412 3.005e-08 ***
## kidslt6  -0.2903722  0.0358252 -8.1053 5.264e-16 ***
## nwifeinc -0.0041586  0.0015000 -2.7724  0.005564 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Goodness of Fit in Binary Models

GOF in Binary Models

- There is no universally accepted R^2 for binary models.

- We can think of a *pseudo* R^2 which compares our model to one without any regressors:

```
glms$probit0 <- update(glms$probit, formula = . ~ 1)  # intercept model only
1 - as.vector(logLik(glms$probit)/logLik(glms$probit0))
```

```
## [1] 0.07084972
```

- But that's not super informative (unlike the standard R^2). Changes in likelihood value are highly non-linear, so that's not great.

- Let's check **accuracy** - what's the proportion correctly predicted! `round(fitted(x))` assigns `1` if the predicted prob > 0.5.

```
prop.table(table(true = mroz$inlf, pred = round(fitted(glms$probit))))
```

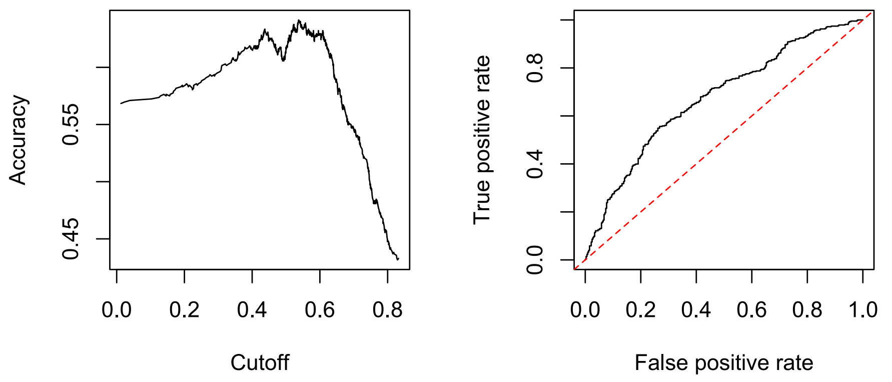
```
##     pred
## true         0         1
##    0 0.1699867 0.2616202
##    1 0.1221780 0.4462151
```

GOF in Binary Models: ROC Curves

- The 0.5 cutuff is arbitrary. What if all predicted probs are > 0.5 but in the data there are about 50% of zeros?
- Let's choose an *arbitrary cutoff* $c \in (0, 1)$ and check accuracy for each value. This gives a better overview.
- Also, we can confront the **true positives rate** (TPR) with the **false positives rate** (FPR).
    1. TPR: number of women correctly predicted to work divided by num of working women.
    2. FPR: number of women incorrectly predicted to work divided by num of non-working women.
- Plotting FPR vs TPR for each c defines the **ROC** (Receiver Operating Characteristics) Curve.
- A good model has a ROC curve in the upper left corner: FPR = 0, TPR = 1.

GOF in Binary Models: ROC Curves

```r
library(ROCR)
pred <- prediction(fitted(glms$probit), mroz$inlf)
par(mfrow = c(1,2), mar = lowtop)
plot(performance(pred,"acc"))
plot(performance(pred,"tpr","fpr"))
abline(0,1,lty = 2, col = "red")
```



- Best accuracy at around $c = 0.6$
- ROC always above 45 deg line. Better than random assignment (flipping a coin)! Yeah!

Task 3 (10 Minutes): `SwissLabor`

1. Load the `SwissLabor` Dataset from the `AER` package with `data(SwissLabor, package = "AER")`
2. `skim` the data to get a quick overview. How many foreigners are in the data?
3. Run a probit model of `participation` on all other variables plus age squared. Which age has the largest impact on participation?
4. What is the marginal effect at the mean of all x of being a foreigner on participation?
5. Produce a ROC curve of this probit model and discuss it!

SciencesPo
DEPARTMENT OF ECONOMICS

END

	nikiforos.zampetakis@sciencespo.fr
	Slides
	Book
	@ScPoEcon
	@ScPoEcon