Modelos de Elección Discreta

Camilo Forero - Jhan Andrade - Germán Rodriguez

14/10/2020

Introducción

Los modelos de elección discreta se caracterizan por que su variable explicativa representan decisiones que toman los agentes. Ahora bien, hay múltiples modelos de elección discreta. Pero para efectos del cursos de Econometria II procederemos a abordar los modelos de elección binaria, es decir 2 opciones únicamente.

Lo anterior se refleja en el hecho de que nuestra variable explicada puede tomar los valores de 0 o 1, por lo tanto se pueden interpretar como modelos que predicen la *probabilidad* de tomar la decisión o no. Una vez dicho esto, hay que tener en cuenta que los modelos de probabilidad lineal (regresión lineal normal) no son los mejores para representar este tipo de decisiones, pues en muchas ocasiones estos modelos suelen pronosticar **probabilidades menores a cero y mayores a uno, lo cual va en contra de la teoría estadística**. Adicionalmente, estos modelos de probabilidad lineal por lo general arrojan un **coeficiente el cual es constante** a lo largo de la muestra.

Esto último representa un problema en el sentido que, por lo general la probabilidad de ocurrencia de un evento dependerá de características especificas del individuo y los cambios en la probabilidad de ocurrencia no son constantes. Por ejemplo: la probabilidad para una persona de 8, 17 y 22 años de que puedan entrar en un bar o les vendan licor en un almacén ancla no es la misma. Ahora bien, es claro que al niño de 8 años posiblemente no le vendan licor o entre a un bar, en caso de que cumpla 9 años, la probabilidad de ocurrencia de los eventos seguirá siendo nula. Para el caso de la persona con 17 años, puede que al cumplir 18 eventualmente si pueda entrar a bares y por tanto pueda comprar licor legalmente. Por otra parte, el joven de 22, al cumplir un año mas no percibirá un gran cambio en la probabilidad de ocurrencia como si puede ocurrir en el caso del individuo de 17 año, pues con 22 o 23 años ya puede comprar licor legalmente.

Hasta este momento podemos decir que las **limitaciones que tiene el modelo de probabilidad lineal** es que:

- Predice probabilidades mayores y menores a cero
- Los errores son muy grandes y no siguen una distribución normal
- Los coeficientes están sesgados

En realidad la probabilidad de ocurrencia de un evento sigue una forma de S, es decir dependiendo de las características generales del evento o del individuo habrá un determinado efecto sobre la probabilidad de ocurrencia del evento. Los modelos que se comportan de esta manera es decir **modelos NO lineales** siguen una distribución *bernoulli*. Para el caso del curso, analizaremos los modelos que hacen uso de funciones de distribución acumulada **logística (Logit)** y **probabilistica o normal (Probit)**.

Hay que tener cuidado si bien Logit y Probit no son modelos lineales, no significa que los parámetros no lo sean. Pues al final, usamos las funciones Logit y Probit para modelar las decisiones, pero estas terminan dependiendo de unos parámetros lineales, es decir :

- Modelo de Probabilidad Lineal:

$$Y = \beta_0 + \beta_1 X_1 + ... + \beta_k X_k + \epsilon$$

- Logit o probit:

$$Y = G(Y = \beta_0 + \beta_1 X_1 + ... + \beta_k X_k + \epsilon)$$

  donde G corresponde a la función de probabilidad logística o probabilistica que depende de una función lineal.

Ejercicio de monitoria

Para abordar los modelos de elección binaria debemos instalar los siguientes paquetes:

```
#library(foreign);library(car); library(lmtest); library(stargazer);
#library(wooldridge);library(dplyr);library(broom)
```

Cargamos la base de datos:

```
data<- read.dta("http://fmwww.bc.edu/ec-p/data/wooldridge/mroz.dta")
help(mroz)
attach(data)
```

Todo lo relacionado con la descripción de la base de datos la pueden encontrar en **help(mroz)**

Modelo de Probabilidad Lineal - MPL

Estimaremos un MPL para la probabilidad de que la mujer haya pertenecido a la fuera laboral.

Las variables explicativas a considerar son:

- Ingreso del esposo (un mayor ingreso puede causar que la sra. considere no trabajar)
- Educación (más educación abre la posibilidad de que considere querer trabajar)
- Experiencia
- Hijos menores de 6 años (Hijos menores de 6 años implican un mayor cuidado, por lo tanto pueden influir en la decisión de una mujer de pertenecer o no a la fuerza laboral)
- Hijos de 6 años o más

```
MPL=lm(inlf~nwifeinc+educ+exper+expersq+
         age+kidslt6+kidsge6,
       data = data)
summary(MPL)
```

```
##
## Call:
## lm(formula = inlf ~ nwifeinc + educ + exper + expersq + age +
##     kidslt6 + kidsge6, data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.93432 -0.37526  0.08833  0.34404  0.99417
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.5855192  0.1541780   3.798 0.000158 ***
## nwifeinc    -0.0034052  0.0014485  -2.351 0.018991 *
## educ         0.0379953  0.0073760   5.151 3.32e-07 ***
## exper        0.0394924  0.0056727   6.962 7.38e-12 ***
## expersq     -0.0005963  0.0001848  -3.227 0.001306 **
## age         -0.0160908  0.0024847  -6.476 1.71e-10 ***
## kidslt6     -0.2618105  0.0335058  -7.814 1.89e-14 ***
```

```
## kidsge6      0.0130122  0.0131960   0.986 0.324415
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4271 on 745 degrees of freedom
## Multiple R-squared:  0.2642, Adjusted R-squared:  0.2573
## F-statistic: 38.22 on 7 and 745 DF,  p-value: < 2.2e-16
```

Con el paquete broom:

```
# Summary statistic en forma de data frame
tidy_mpl = tidy(MPL); tidy_mpl
```

```
## # A tibble: 8 x 5
##   term         estimate std.error statistic  p.value
##   <chr>           <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)  0.586     0.154        3.80  1.58e- 4
## 2 nwifeinc    -0.00341   0.00145     -2.35  1.90e- 2
## 3 educ         0.0380    0.00738      5.15  3.32e- 7
## 4 exper        0.0395    0.00567      6.96  7.38e-12
## 5 expersq     -0.000596  0.000185    -3.23  1.31e- 3
## 6 age         -0.0161    0.00248     -6.48  1.71e-10
## 7 kidslt6     -0.262     0.0335      -7.81  1.89e-14
## 8 kidsge6      0.0130    0.0132       0.986 3.24e- 1
```

```
# Data frame con info. adicional sobre la regresión
glance_mpl = glance(MPL); glance_mpl
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic  p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.264         0.257 0.427      38.2 6.90e-46     7  -424.  866.  907.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```
# Data frame de datos expandidos con desviaciones estándar poblacional estimada,
#residuales, fitted values y más
augment_mpl = augment(MPL); augment_mpl
```

```
## # A tibble: 753 x 14
##     inlf nwifeinc  educ exper expersq   age kidslt6 kidsge6 .fitted .resid
##    <dbl>    <dbl> <dbl> <dbl>   <dbl> <dbl>   <dbl>   <dbl>   <dbl>  <dbl>
##  1     1    10.9     12    14     196    32       1       0   0.664 0.336
##  2     1    19.5     12     5      25    30       0       2   0.701 0.299
##  3     1    12.0     12    15     225    35       1       3   0.673 0.327
##  4     1     6.80    12     6      36    34       0       3   0.726 0.274
##  5     1    20.1     14     7      49    31       1       2   0.562 0.438
##  6     1     9.86    12    33    1089    54       0       0   0.793 0.207
##  7     1     9.15    16    11     121    37       0       2   0.955 0.0448
##  8     1    10.9     12    35    1225    54       0       0   0.787 0.213
##  9     1    17.3     12    24     576    48       0       2   0.841 0.159
## 10     1    12.9     12    21     441    39       0       2   0.962 0.0377
## # ... with 743 more rows, and 4 more variables: .std.resid <dbl>, .hat <dbl>,
## #   .sigma <dbl>, .cooksd <dbl>
```

```
help("augment")
```

```
## Help on topic 'augment' was found in the following packages:
##
```

```
##   Package               Library
##   generics              /home/germankux/R/x86_64-pc-linux-gnu-library/3.6
##   broom                 /home/germankux/R/x86_64-pc-linux-gnu-library/3.6
##
##
## Using the first match ...
```

```
#.hat   Diagonal of the hat matrix
#.sigma Estimate of residual standard deviation when corresponding observation is dropped
#from model
#.cooksd    Cooks distance, cooks.distance()
#.fitted    Fitted values of model
#.se.fit    Standard errors of fitted values
#.resid Residuals
#.std.resid Standardised residuals

#Estimar el MPL con erróres robustos a la heterocedasticidad p(1-p)
coeftest(MPL,vcov=hccm) # Heteroscedasticity-Corrected Covariance Matrix
```

```
##
## t test of coefficients:
##
##                Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)  0.58551923  0.15358032  3.8125  0.000149 ***
## nwifeinc    -0.00340517  0.00155826 -2.1852  0.029182 *
## educ         0.03799530  0.00733982  5.1766 2.909e-07 ***
## exper        0.03949239  0.00598359  6.6001 7.800e-11 ***
## expersq     -0.00059631  0.00019895 -2.9973  0.002814 **
## age         -0.01609081  0.00241459 -6.6640 5.183e-11 ***
## kidslt6     -0.26181047  0.03215160 -8.1430 1.621e-15 ***
## kidsge6      0.01301223  0.01366031  0.9526  0.341123
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Predicciones MPL

Vamos a realizar la predicción para dos mujeres fuera de la muestra y con características especificas:

- 1. Una mujer de 20 años con un esposo con un salario de $100, con 5 años de educación, 0 años de experiencia y 2 niños menores de 6 años.
- 2. La segunda mujer de 52 años, su esposo esta desempleado, tiene 17 años de educación, 30 años de experiencia.

```
#Predicción para dos mujeres fuera de la muestra original
MPL.pred = list(nwifeinc=c(100,0),educ=c(5,17),exper=c(0,30),expersq=c(0,900),
                age=c(20,52),kidslt6=c(2,0),kidsge6=c(0,0))

predict(MPL,MPL.pred)
```

```
##          1          2
## -0.4104582  1.0428084
```

Como pueden ver, el MPL arroja predicciones fuera del rango de probabilidad [0,1].

Predicciones con el paquete broom

```
# Características de las dos mujeres fuera de la muestra
MPL.pred2 <- data.frame(nwifeinc=c(100,0),educ=c(5,17),exper=c(0,30),expersq=c(0,900),
                        age=c(20,52),kidslt6=c(2,0),kidsge6=c(0,0))

# predicción con broom para las dos mujeres fuera de la muestra
augment(MPL,newdata = MPL.pred2, type.predict="response")
```

```
## Warning: 'newdata' had 2 rows but variables found have 753 rows
```

```
## # A tibble: 2 x 8
##   nwifeinc  educ exper expersq   age kidslt6 kidsge6 .fitted
##      <dbl> <dbl> <dbl>   <dbl> <dbl>   <dbl>   <dbl>   <dbl>
## 1      100     5     0       0    20       2       0  -0.410
## 2        0    17    30     900    52       0       0   1.04
```

La columna *.fitted* es la que corresponde a las probabilidades esperadas, que como ya se menciono no tiene lógica bajo la teoría estadística.

MODELO LOGIT

Para estimar modelos Logit o Probit, debemos hacer uso del comando ***glm***. Tal como hemos visto en el transcurso del curso, la estructura de este tipo de comandos es muy sencilla:

$$glm(Y \sim X, family = binomial(link = logit), data)$$

```
LOGIT = glm(inlf~nwifeinc+educ+exper+expersq+age+kidslt6+kidsge6,
            family = binomial(link = logit),data = data); summary(LOGIT)
```

```
##
## Call:
## glm(formula = inlf ~ nwifeinc + educ + exper + expersq + age +
##     kidslt6 + kidsge6, family = binomial(link = logit), data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.1770  -0.9063   0.4473   0.8561   2.4032
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.425452   0.860365   0.495  0.62095
## nwifeinc    -0.021345   0.008421  -2.535  0.01126 *
## educ         0.221170   0.043439   5.091 3.55e-07 ***
## exper        0.205870   0.032057   6.422 1.34e-10 ***
## expersq     -0.003154   0.001016  -3.104  0.00191 **
## age         -0.088024   0.014573  -6.040 1.54e-09 ***
## kidslt6     -1.443354   0.203583  -7.090 1.34e-12 ***
## kidsge6      0.060112   0.074789   0.804  0.42154
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1029.75  on 752  degrees of freedom
```

```
## Residual deviance:  803.53  on 745  degrees of freedom
## AIC: 819.53
##
## Number of Fisher Scoring iterations: 4
```

Ahora usando el paquete broom:

```
tidy_logit = tidy(LOGIT)
glance_logit = glance(LOGIT)   # inlcuye logLik
augment_logit = augment(LOGIT)  # augmented data frame que incluye
#residuales y valores estimados como columnas

# predict sin más argumentos calcular los
#   z_ajustados = b_0 + b_1 * x_1 + b_2 * x_2 + ... + b_n * x_n (bs coeficientes estimados)
LOGIT.FIT = predict(LOGIT) #Valores ajustados dentro de la muestra.

# predict con type="response"
#   calcular P(Y = 1) = exp(z_ajustado)/(1 - exp(z_ajustado))
# que es la probabilidad de exito para los diferentes valores
#   z_ajustados en el modelo para un modelo logit
# Existen 3 formas diferentes de hacerlo:

# forma1
LOGIT.pred = predict(LOGIT, type="response")

# forma2
LOGIT.pred_2 = plogis(LOGIT.FIT) #plogis me da la función acumulada de probabilidad para una variable a

# forma3
LOGIT.pred_3 = exp(LOGIT.FIT)/(1 + exp(LOGIT.FIT))

# Las tres formas son equivalentes
```

Predicciones Logit

Usaremos las mismas características de las mujeres que mencionamos en el caso del MPL.

```
# predicción para el logit con el comando broom

# Características de las dos mujeres fuera de la muestra
LOGIT.pred2 <- data.frame(nwifeinc=c(100,0),educ=c(5,17),exper=c(0,30),expersq=c(0,900),
                          age=c(20,52),kidslt6=c(2,0),kidsge6=c(0,0))

z = augment(LOGIT,newdata = LOGIT.pred2)  # para encontrar z
p_exito = augment(LOGIT,newdata = LOGIT.pred2, type.predict="response")
# para encontrar P(Y = 1) = exp(z)/ 1 - exp(z)
p_exito
```

```
## # A tibble: 2 x 8
##   nwifeinc  educ exper expersq   age kidslt6 kidsge6 .fitted
##      <dbl> <dbl> <dbl>   <dbl> <dbl>   <dbl>   <dbl>   <dbl>
## 1      100     5     0       0    20       2       0 0.00522
## 2        0    17    30     900    52       0       0 0.950
```

Si se dan cuenta, la columna *.fitted* ya tiene consistencia con los valores de una probabilidad. De tal manera que nuestra primera mujer, tiene una probabilidad muy cercana a cero de pertenecer al mercado laboral,

mientras que la segunda mujer tiene una alta probabilidad (0,9) de pertenecer al mercado laboral.

Pseudo R^2 de McFadden

El pseudo R^2 de McFadden es una medida de bondad de ajuste, esta medida es importante en el sentido de que estamos hablando de modelos no lineales. Es por ello que carece de sentido fijarnos en el R^2 de los modelos de regresión lineal. Luego, entre mayor sea el pseudoR^2, mayor será la bondad de ajuste del modelo.

```
#Pseudo R^2 de McFadden para comparar dos modelos de elección
#discreta con las mismas variables
# 1- (logaritmo verosimilitud modelo completo)/
#  (logaritmo verosimilitud del modelo restringido con solo un intercepto)
# is defined as 1- L1/L0, where L0 represents the log likelihood for the "constant-only" model and L1 i

# La función loglik permite calcular la función
# log likelihood para el modelo LOGIT
logLik(LOGIT)
```

```
## 'log Lik.' -401.7652 (df=8)
```

```
#Pseudo McFadden R^2 usando Residual deviance y NUll deviance
1 - LOGIT$deviance/LOGIT$null.deviance
```

```
## [1] 0.2196814
```

```
#Pseudo McFadden R^2 usando el log de la funx. de max. verosimilitud para el modelo completo y para el
# solo con el intercepto
## loglik(LOGIT): funx. de max. verosimilitud modelo completo
## loglik(LOGIT_NULL): funx. de max. verosimilitud modelo solo intercepto
LOGIT_null <- glm(inlf~1, family = binomial, data = data)
1- logLik(LOGIT)/logLik(LOGIT_null)
```

```
## 'log Lik.' 0.2196814 (df=8)
```

Interpretación de los resultados:

Al estar trabajando con modelos no lineales, no podemos hacer una interpretación directa de los estimadores, lo único que hasta el momento podemos interpretar son los signos de los coeficientes, y por tanto no tenemos un orden de magnitud que nos permita entender cómo una variable afecta o no la probabilidad de ocurrencia de un evento.

Los odds ratio se interpretan como el número de veces que es más probable que ocurra el fenómeno $P(Y = 1|X)$ frente al hecho de que no ocurra$P(Y = 0|X)$:

- Odd=1 entonces $P(Y = 1|X) = P(Y = 0|X)$
- Odd<1 entonces $P(Y = 1|X) < P(Y = 0|X)$
- Odd>1 entonces $P(Y = 1|X) > P(Y = 0|X)$

La forma de calcular los Odds Ratios es la siguiente:

```
odds=exp(z$.fitted);odds
```

```
## [1]  0.005245373 19.019665970
```

Para más detalle en la forma en como se calculan estos odds, pueden remitirse al script.

Los cocientes entre odds ratios, se entienden como que tan probable es que ocurra la alternativa i frente a la alternativa j

```
#c.odds=exp(coefficients(LOGIT));c.odds    # Para cada parámetro estimado
#log.odds= coefficients(LOGIT);log.odds
```

```
stargazer(c.odds, log.odds, type="text")
```

```
##
## ===============================================================
## (Intercept) nwifeinc educ  exper expersq  age  kidslt6 kidsge6
## ---------------------------------------------------------------
## 1.530        0.979   1.248 1.229  0.997  0.916  0.236   1.062
## ---------------------------------------------------------------
##
## ================================================================
## (Intercept) nwifeinc educ  exper expersq  age   kidslt6 kidsge6
## ----------------------------------------------------------------
## 0.425        -0.021  0.221 0.206 -0.003  -0.088 -1.443   0.060
## ----------------------------------------------------------------
```

Para este caso, un año más de educación hace 1.24 veces más probable entrar al mercado laboral.

MODELO PROBIT

De la misma manera que con LOGIT, procedemos a usar el comando ***glm***, de tal manera que la estructura para este tipo de modelos es la siguiente:

$$glm(Y \sim X, family = binomial(link = probit), data)$$

```
PROBIT = glm(inlf~nwifeinc+educ+exper+expersq+age+kidslt6+kidsge6,
             family=binomial(link=probit),data=data)
summary(PROBIT)
```

```
##
## Call:
## glm(formula = inlf ~ nwifeinc + educ + exper + expersq + age +
##     kidslt6 + kidsge6, family = binomial(link = probit), data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2156  -0.9151   0.4315   0.8653   2.4553
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.2700736  0.5080782   0.532  0.59503
## nwifeinc    -0.0120236  0.0049392  -2.434  0.01492 *
## educ         0.1309040  0.0253987   5.154 2.55e-07 ***
## exper        0.1233472  0.0187587   6.575 4.85e-11 ***
## expersq     -0.0018871  0.0005999  -3.145  0.00166 **
## age         -0.0528524  0.0084624  -6.246 4.22e-10 ***
## kidslt6     -0.8683247  0.1183773  -7.335 2.21e-13 ***
## kidsge6      0.0360056  0.0440303   0.818  0.41350
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1029.7  on 752  degrees of freedom
## Residual deviance:  802.6  on 745  degrees of freedom
```

```
## AIC: 818.6
##
## Number of Fisher Scoring iterations: 4
```

Haciendo uso del paquete broom:

```
tidy_probit =tidy(PROBIT)
glance_probit =glance(PROBIT)   # inlcuye logLik
augment_probit = augment(PROBIT)
# augmented data frame que incluye residuales y valores estimados como columnas

# predict sin más argumentos calcular los
# z_ajustados = b_0 + b_1 * x_1 + b_2 * x_2 + ... + b_n * x_n (bs coeficientes estimados)
PROBIT.FIT = predict(PROBIT) #Valores ajustados dentro de la muestra.

# predict con type="response" calcular
# que es la probabilidad de éxito para los diferentes valores
#z_ajustados en el modelo para un modelo logit


# Existen 3 formas diferentes de hacerlo:

# forma1
PROBIT.pred = predict(PROBIT, type="response")

# forma2
PROBIT.pred_2 = pnorm(PROBIT.FIT) #pnorm me da la función acumulada de probabilidad para una variable a

# No se agrega la forma integral porque seria la integral asociada a la funciona de distribución acumul
```

Predicción

De la misma manera que con el caso de logit, usaremos las mismas características de las mujeres consideradas para el caso de MPL.

```
# Características de las dos mujeres fuera de la muestra
PROBIT.pred2 <- data.frame(nwifeinc=c(100,0),educ=c(5,17),exper=c(0,30),expersq=c(0,900),
                           age=c(20,52),kidslt6=c(2,0),kidsge6=c(0,0))

z_norm = augment(PROBIT,newdata = PROBIT.pred2)  # para encontrar z
p_exito_norm = augment(PROBIT,newdata = PROBIT.pred2, type.predict="response") # para encontrar P(Y = 1
p_exito_norm
```

```
## # A tibble: 2 x 8
##   nwifeinc  educ exper expersq   age kidslt6 kidsge6 .fitted
##      <dbl> <dbl> <dbl>   <dbl> <dbl>   <dbl>   <dbl>   <dbl>
## 1      100     5     0       0    20       2       0 0.00107
## 2        0    17    30     900    52       0       0 0.960
```

De acuerdo con la columna *.fitted* la probabilidad de la primera mujeres es muy cercana a cero, mientras que la probabilidad de la segunda mujer es casi cercana a 1. Lo anterior, va en linea con que estos valores son coherentes con la teoria estadística dado que se encuentran entre en el rango de probabilidad de [0,1].

Pseudo R^2 de McFadden

```
logLik(PROBIT)
```

```
## 'log Lik.' -401.3022 (df=8)
```

```
1 - PROBIT$deviance/PROBIT$null.deviance  #los modelos son estadisticamente equivalentes.
```

```
## [1] 0.2205805
```

```
#Pseudo McFadden R^2 usando el log de la funx. de max. verosimilitud para el modelo completo y para el
PROBIT_null <- glm(inlf~1, family = binomial, data = data)
1- logLik(PROBIT)/logLik(PROBIT_null)
```

```
## 'log Lik.' 0.2205805 (df=8)
```

```
stargazer(MPL,LOGIT,PROBIT, column.labels = c("MPL","LOGIT","PROBIT"), type="text")
```

```
##
## ===============================================================
##                                 Dependent variable:
##                     ------------------------------------------
##                                         inlf
##                               OLS           logistic   probit
##                               MPL             LOGIT    PROBIT
##                               (1)               (2)       (3)
## ---------------------------------------------------------------
## nwifeinc                   -0.003**          -0.021**  -0.012**
##                             (0.001)           (0.008)   (0.005)
##
## educ                       0.038***          0.221***  0.131***
##                             (0.007)           (0.043)   (0.025)
##
## exper                      0.039***          0.206***  0.123***
##                             (0.006)           (0.032)   (0.019)
##
## expersq                    -0.001***         -0.003*** -0.002***
##                            (0.0002)           (0.001)   (0.001)
##
## age                        -0.016***         -0.088*** -0.053***
##                             (0.002)           (0.015)   (0.008)
##
## kidslt6                    -0.262***         -1.443*** -0.868***
##                             (0.034)           (0.204)   (0.118)
##
## kidsge6                      0.013             0.060     0.036
##                             (0.013)           (0.075)   (0.044)
##
## Constant                   0.586***            0.425     0.270
##                             (0.154)           (0.860)   (0.508)
##
## ---------------------------------------------------------------
## Observations                  753               753       753
## R2                           0.264
## Adjusted R2                  0.257
## Log Likelihood                                -401.765  -401.302
## Akaike Inf. Crit.                              819.530   818.604
## Residual Std. Error     0.427 (df = 745)
## F Statistic          38.218*** (df = 7; 745)
## ===============================================================
```

```
## Note:                                *p<0.1; **p<0.05; ***p<0.01
```

Ahora bien, ya hemos visto la forma en como se estiman los modelos LOGIT y PROBIT, en muchas ocasiones se preguntan:

- *Oiga monitor, pero ya que hemos visto esto ¿Cuándo se que modelo usar?*

La respuesta a esa pregunta es: Ambos modelos son correctos, la única diferencia está en las colas de las funciones de probabilidad de la función logística y estándar. Es decir, el modelo LOGIT en muchas ocasiones permite la ocurrencia de eventos más raros que el modelo RPOBIT. En otras palabras, dependiendo de la decisión que se deseé modelar y el comportamiento de los datos, se elige si PROBIT o LOGIT.

Efectos parciales:

Tal como ya lo mencionamos arriba, es importante que tengan en cuenta que **nunca se pueden interpretar de manera directa los coeficientes de los modelos LOGIT y PROBIT**. Es por ello que recurrimos a los ODDS RATIO o en el mejor de los casos a los efectos parciales, la forma en como se calculan estos efectos parciales varian y dependen del modelo.

El efecto parcial para la variable k se calcula dependiendo del modelo de la siguiente manera:

- MPL:
$$\beta_k$$
- Logit:
$$\Lambda(X\beta) * [1 - \Lambda(X\beta)] * \beta_k$$
- Probit:
$$\phi(X\beta) * \beta_k$$

Donde $\Lambda()$ hace referencia a la función de distribución logística acumulada estándar y $\phi()$ a la función de distribución de probabilidad normal (OJO no es la acumulada).

Recuerde que cuando se hace referencia a un efecto marginal o parcial, se hace referencia a una derivada.

Para hacer el cálculo de los efectos marginales se usará el paquete ***margins***. No obstante es importante que tengan en cuenta que cuando tienen interacción de variables o variables elevadas al cuadrado, para e cálculo de los efectos marginales se deben incluir en la forma funcional de la forma ***I(Var1xVar2)*** o ***I(Var^2)***. De esta manera se garantiza que el efecto marginal se calcule correctamente.

Para usar el paquete *margins* en el ejemplo de la monitoria debemos incluir ***I(exper^2)*** en la estimación de los modelos Logit y Probit.

```
# Modelo Logit
LOGIT = glm(inlf~nwifeinc+educ+exper+I(exper^2)+age+kidslt6+kidsge6,
            family = binomial(link = logit),data = data)
# Modelo Probit
PROBIT = glm(inlf~nwifeinc+educ+exper+I(exper^2)+age+kidslt6+kidsge6,
             family=binomial(link=probit),data=data)
```

Para el caso de la monitoria calcularemos el PEA (Partial effect on the average) y el APE (Average Partial Effect). El PEA es el cálculo del efecto de determinada variable X sobre Y, en el caso específico de la media de la muestra, mientras que el APE es el efecto parcial promedio de una variable X sobre Y en todos los casos de la muestra.

Calcular el APE

La estructura del coligo corresponde a $margins(MODELO, type = "response")$. Es importante indicar el *response* para que se cáclule el APE.

```
margins(LOGIT, type = "response")
```

```
## Average marginal effects
```

```
## glm(formula = inlf ~ nwifeinc + educ + exper + I(exper^2) + age +     kidslt6 + kidsge6, family = bi
```

```
##   nwifeinc    educ   exper      age kidslt6 kidsge6
##  -0.003812 0.0395 0.02543 -0.01572 -0.2578 0.01073
```

```
margins(PROBIT, type = "response")
```

```
## Average marginal effects
```

```
## glm(formula = inlf ~ nwifeinc + educ + exper + I(exper^2) + age +     kidslt6 + kidsge6, family = bi
```

```
##   nwifeinc    educ   exper     age kidslt6 kidsge6
##  -0.003616 0.03937 0.02558 -0.0159 -0.2612 0.01083
```

calcular el PEA

Para este caso es necesario proveerle a Rstudio un data.frame con las variables evaluadas en sus medias.

```
margins(LOGIT, type = "response",
        data.frame(nwifeinc = mean(nwifeinc),
                   educ = mean(educ),
                   age = mean(age),
                   exper = mean(exper),
                   kidslt6 = mean(kidslt6),
                   kidsge6 = mean(kidsge6)))
```

```
## Average marginal effects
```

```
## glm(formula = inlf ~ nwifeinc + educ + exper + I(exper^2) + age +     kidslt6 + kidsge6, family = bi
```

```
##   nwifeinc    educ  exper      age kidslt6 kidsge6
##  -0.004966 0.05146 0.0323 -0.02048 -0.3358 0.01399
```

```
margins(PROBIT, type = "response",
        data.frame(nwifeinc= mean(nwifeinc),
                   educ = mean(educ),
                   age = mean(age),
                   exper = mean(exper),
                   kidslt6 = mean(kidslt6),
                   kidsge6 = mean(kidsge6)))
```

```
## Average marginal effects
```

```
## glm(formula = inlf ~ nwifeinc + educ + exper + I(exper^2) + age +     kidslt6 + kidsge6, family = bi
```

```
##   nwifeinc    educ   exper      age kidslt6 kidsge6
##  -0.004545 0.04948 0.03146 -0.01998 -0.3282 0.01361
```

Calcular el PEA para variables dicotómincas o discretas

Variables dicotómicas El cálculo del PEA para la dummy de *kidslt6* se puede hacer de dos maneras, de manera manual o con el paquete *margins*, no obstante ambos resultados son equivalentes. Recuerde que para hacer este procedimiento debe calcular dos probabilidades, una cuando la dummy toma el valor de 1 y

otra cuando la dummy toma el valor de 0, en ambos casos el resto de variables evaluadas en la media de la muestra. En este caso el efecto parcial será la resta entre las Probabilidades cuando la dummy es 1 y cuando la dummy toma el valor de 0.

```
# De manera manual para logit
p_1_logis = plogis(0.425 - 0.0213 * mean(nwifeinc) + 0.221 * mean(educ)
            + 0.206 * mean(exper) - 0.0032 * mean(expersq) - 0.0880 * mean(age)
            + 0.0601 - 1.44 )

p_2_logis = plogis(0.425 + -0.0213 * mean(nwifeinc) + 0.221 * mean(educ)
            + 0.206 * mean(exper) - 0.0032 * mean(expersq) -0.0880 * mean(age)
            + 0.0601)

marg_effect_1_logis = p_1_logis - p_2_logis;marg_effect_1_logis
```

```
## [1] -0.3448272
```

```
# De manera manual para probit

p_1_norm = pnorm(0.270 - 0.012 * mean(nwifeinc) + 0.131 * mean(educ)
             + 0.123 * mean(exper) - 0.0019 * mean(expersq) - 0.053 * mean(age)
             + 0.036 - 0.868 )

p_2_norm = pnorm(0.270 - 0.012 * mean(nwifeinc) + 0.131 * mean(educ)
                 + 0.123 * mean(exper) - 0.0019 * mean(expersq)
                 - 0.053 * mean(age) + 0.036)

marg_effect_1_norm = p_1_norm - p_2_norm;marg_effect_1_norm
```

```
## [1] -0.3353904
```

Con el paquete *margins* en la parte del data.frame hay que indicarle en este caso que la dummy toma el valor de 0.

```
margins(LOGIT, type = "response", data.frame(nwifeinc=mean(nwifeinc), educ = mean(educ),
                                             age = mean(age), exper = mean(exper),
                                             kidslt6 = 0, kidsge6 = 1))
```

```
## Average marginal effects
```

```
## glm(formula = inlf ~ nwifeinc + educ + exper + I(exper^2) + age +     kidslt6 + kidsge6, family = bi
```

```
##   nwifeinc    educ   exper      age kidslt6 kidsge6
##  -0.004458 0.04619 0.02899 -0.01838 -0.3014 0.01255
```

```
margins(PROBIT, type = "response", data.frame(nwifeinc=mean(nwifeinc), educ = mean(educ),
                                              age = mean(age), exper = mean(exper),
                                              kidslt6 = 0, kidsge6 = 1))
```

```
## Average marginal effects
```

```
## glm(formula = inlf ~ nwifeinc + educ + exper + I(exper^2) + age +     kidslt6 + kidsge6, family = bi
```

```
##   nwifeinc    educ   exper     age kidslt6 kidsge6
##  -0.004185 0.04557 0.02897 -0.0184 -0.3023 0.01253
```

Luego con estos cálculos podemos determinar que en promedio tener hijos menores de 6 años, reduce la probabilidad de pertenecer en el mercado laboral en aproximadamente un 30%.

Calcular el PEA para variables discretas

En este caso evaluaremos el efecto marginal de pasar de 10 años de experiencia en el mercado laboral a 11 años de experiencia. Nuevamente, recuerde que para el resto de variables debe usar la media.

```
# De manera manual para logit
p_1_logis = plogis(0.425 - 0.0213 * mean(nwifeinc) + 0.221 * mean(educ)
             + 0.206 * 11 - 0.0032 * 11^2 - 0.0880 * mean(age) + 0.0601)

p_2_logis = plogis(0.425 + -0.0213 * mean(nwifeinc) + 0.221 * mean(educ)
             + 0.206 * 10 - 0.0032 * 10^2 -0.0880 * mean(age) + 0.0601)

marg_effect1 = p_1_logis - p_2_logis; marg_effect1
```

```
## [1] 0.02925198
```

```
# De manera manual para probit
p_1_norm = pnorm(0.270 - 0.012 * mean(nwifeinc) + 0.131 * mean(educ)
                 + 0.123 * 11 - 0.0019 * 11^2 - 0.053 * mean(age) + 0.036)

p_2_norm = pnorm(0.270 - 0.012 * mean(nwifeinc) + 0.131 * mean(educ)
                 + 0.123 * 10 - 0.0019 * 10^2 - 0.053 * mean(age) + 0.036)

marg_effect2 = p_1_norm - p_2_norm;marg_effect2
```

```
## [1] 0.0292346
```

Con el paquete *margins* en la parte del data.frame hay que indicarle en este caso que la variable *expert* toma el valor de 10.

```
margins(LOGIT, type = "response", data.frame(nwifeinc=mean(nwifeinc), educ = mean(educ),
                                             age = mean(age), exper = 10,
                                             kidslt6 = 0, kidsge6 = 1))
```

```
## Average marginal effects
```

```
## glm(formula = inlf ~ nwifeinc + educ + exper + I(exper^2) + age +     kidslt6 + kidsge6, family = bi
```

```
##   nwifeinc    educ   exper      age kidslt6 kidsge6
##  -0.004613 0.0478 0.03086 -0.01903  -0.312 0.01299
```

```
margins(PROBIT, type = "response", data.frame(nwifeinc=mean(nwifeinc), educ = mean(educ),
                                              age = mean(age), exper = 10,
                                              kidslt6 = 0, kidsge6 = 1))
```

```
## Average marginal effects
```

```
## glm(formula = inlf ~ nwifeinc + educ + exper + I(exper^2) + age +     kidslt6 + kidsge6, family = bi
```

```
##   nwifeinc    educ  exper      age kidslt6 kidsge6
##  -0.004297 0.04678 0.0306 -0.01889 -0.3103 0.01287
```

Luego, podemos concluir que en promedio pasar de 10 años de experiencia a 11 años, aumenta la probabilidad de pertenecer al mercado laboral en aproximadamente 3%.

Una vez obtengamos estos resultados, ya es posible interpretar esos efectos parciales de como las variables afectan la probabilidad de ocurrencia:

Ya para terminar es importante que recuerden que para el caso de cálculos manuales de probabilidad se usan **las funciones de probabilidad acumulada (las que tienen forma de S)** mientras que para el cálculo de los efectos marginales se usan **las funciones de distribución (las campanas)**.