Tema 5: Regresión Logística Multinomial

Introducción

En este tema se considera un modelo de regresión logística donde la variable dependiente tiene más de dos categorías. La respuesta puede o bien ser nominal o bien ordinal. A su vez, las variables explicativas pueden ser categóricas o cuantitativas.

En los modelos de regresión multinomial se asume que los recuentos de las categorías de Y tienen una distribución multinomial. Esta distribución es, a su vez, una generalización de la distribución binomial.

Ver:

es.wikipedia.org/wiki/Distribución_multinomial

Modelos Logit para respuestas nominales

El número de categorías de Y se denota como J donde $\{\pi_1, \ldots, \pi_J\}$ son las probabilidades de las distintas categorías tal que $\sum_j \pi_j = 1$.

Se parte de n observaciones independientes que se localizan en las J categorías. La distribución de probabilidad del número de observaciones de las J categorías sigue una distribución multinomial. Esta modeliza la probabilidad de cada una de las posibles maneras en que n observaciones pueden repartirse entre las J categorías.

Al ser la escala de medida nominal, el orden entre las categorías es irrelevante.

Se toma una categoría como respuesta **base**, por ejemplo la última categoría (J), y se define un modelo logit con respecto a ella:

$$\log\left(\frac{\pi_j}{\pi_J}\right) = \alpha_j + \beta_j x$$

donde $j = 1, \ldots, J - 1$.

El modelo tiene $J-1$ ecuaciones con sus propios parámetros, y los efectos varían con respecto a la categoría que se ha tomado como base.

Cuando $J=2$, el modelo equivale a una única ecuación $\log(\pi_1/\pi_2) = \text{logit}(\pi_1)$ y se obtiene el modelo de regresión logística estándar.

La ecuación general logit con respecto a la categoría base J determina también los logits para cualquier pareja de categorías. Así, si consideramos dos categorías cualesquiera a y b,

$$\log\left(\frac{\pi_a}{\pi_b}\right) = \log\left(\frac{\pi_a/\pi_J}{\pi_b/\pi_J}\right) = \log\left(\frac{\pi_a}{\pi_J}\right) - \log\left(\frac{\pi_b}{\pi_J}\right) =$$

$$= (\alpha_a + \beta_a x) - (\alpha_b + \beta_b x) = (\alpha_a - \alpha_b) + (\beta_a - \beta_b)x$$

De este modo, la ecuación para las categorías a y b tiene también la forma $\alpha + \beta x$ donde $\alpha = (\alpha_a - \alpha_b)$ y $\beta = (\beta_a - \beta_b)$.

Ejemplo

Se trata de calcular las primas de seguros para los lugareños de una localidad de Florida (USA). Para ello se analizan los datos sobre los gustos alimentarios de 59 caimanes que viven por la zona.

La variable respuesta es la elección del tipo de comida principal que se encuentra en el estómago de los animales, con cinco posibles categorías: *peces, invertebrados, reptiles, pájaros* y *otros*. Como variables predictoras se consideran:

i) El lago en donde se captura al animal (L);

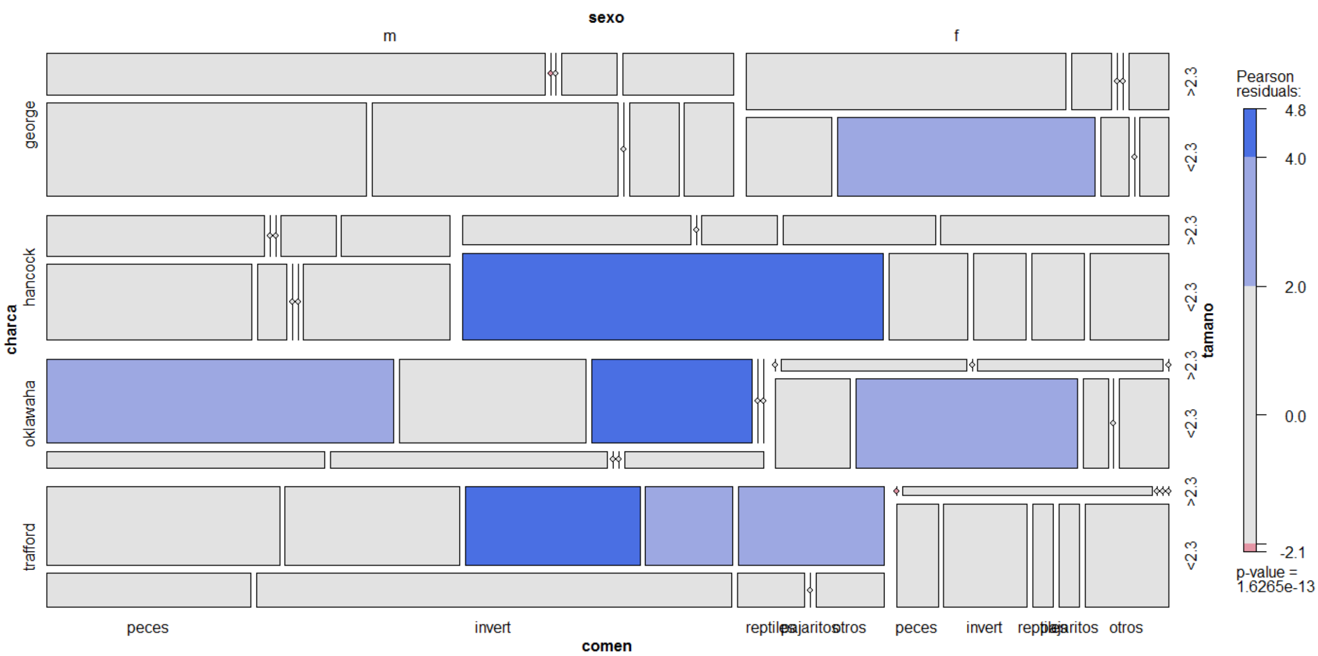
ii) Su género (G);

iii) el tamaño S (si es menor o igual a 2.3 metros o mayor que este valor).

```
library(MASS)
library(nnet)

comen.f = factor(c("peces","invert","reptiles","pajaritos","otros"),
levels=c("peces","invert","reptiles","pajaritos","otros"))

tamano.f = factor(c("<2.3", ">2.3"), levels=c(">2.3", "<2.3"))
sexo.f = factor(c("m","f"), levels=c("m","f"))
```

```
charca.f = factor(c("hancock", "oklawaha","trafford", "george"),
levels=c("george","hancock","oklawaha","trafford"))

temp = c(7, 1, 0, 0, 5, 4, 0, 0, 1, 2, 16, 3, 2, 2, 3, 3, 0, 1, 2, 3,
2, 2, 0, 0, 1, 13, 7, 6, 0, 0, 3, 9, 1, 0, 2, 0, 1, 0, 1, 0, 3, 7, 1,
0, 1, 8, 6, 6, 3, 5, 2, 4, 1, 1, 4, 0, 1, 0, 0, 0, 13, 10, 0, 2, 2,
9, 0, 0, 1, 2, 3, 9, 1, 0, 1, 8, 1, 0, 0, 1)

tabla.array = expand.grid(comen=comen.f, tamano=tamano.f, sexo=sexo.f,
charca=charca.f)
tabla = tapply(temp, tabla.array[, 1:4], sum)

structable(~ charca + sexo + tamano + comen, tabla)
mosaic(~ charca + sexo + tamano + comen, tabla, shade=TRUE)
```

```
              sexo      m                                              f
              comen  peces invert reptiles pajaritos otros  peces invert reptiles pajaritos otros
charca   tamano
george   >2.3            9      0        0         1     2      8      1        0         0     1
         <2.3           13     10        0         2     2      3      9        1         0     1
hancock  >2.3            4      0        0         1     2      3      0        1         2     3
         <2.3            7      1        0         0     5     16      3        2         2     3
oklawaha >2.3           13      7        6         0     0      0      1        0         1     0
         <2.3            2      2        0         0     1      3      9        1         0     2
trafford >2.3            8      6        6         3     5      0      1        0         0     0
         <2.3            3      7        1         0     1      2      4        1         1     4
```




```
options(contrasts = c("contr.treatment", "contr.poly"))
fits1 = multinom(comen ~ tamano, data=tabla.array, weights=temp)
summary(fits1, cor=F)
```

```
Coefficients:
          (Intercept) tamano<2.3
invert      -1.034070  0.9489120
reptiles    -1.241705 -0.8583649
pajaritos   -1.727214 -0.5551882
otros       -1.241709  0.2943162

Std. Errors:
          (Intercept) tamano<2.3
invert      0.2910708  0.3568648
reptiles    0.3148729  0.5349960
pajaritos   0.3836949  0.6063277
otros       0.3148735  0.4149523

Residual Deviance: 589.2134
AIC: 605.2134
```

La categoría base que se toma es justamente la primera: **peces**.

Se pueden ajustar distintos modelos logit con la categoría base en *peces* y compararlos en términos de sus diferencias en *deviance*, es decir mediante las diferencias en la verosimilitud: $-2(\log(L(\hat{\mu}_1, y)) - \log(L(\hat{\mu}_2, y)))$.

```
fitS  = multinom(comen ~ charca*tamano*sexo, data=tabla.array,
weights=temp)
fitS2 = multinom(comen ~ charca*tamano, data=tabla.array,
weights=temp)

fit0 = multinom(comen ~  1, data=tabla.array, weights=temp)
fit1 = multinom(comen ~ charca, data=tabla.array, weights=temp)
fit2 = multinom(comen ~ tamano, data=tabla.array, weights=temp)
fit3 = multinom(comen ~ sexo, data=tabla.array, weights=temp)
fit4 = multinom(comen ~ charca + tamano, data=tabla.array,
weights=temp)
fit5 = multinom(comen ~ charca + tamano + sexo, data=tabla.array,
weights=temp)
```

```
# Se compara con el modelo que NO considera sexo
deviance(fitS2) - deviance(fitS)
```

```
[1] 35.39865
```

```
# Se usa un ANOVA con el modelo que NO considera sexo
anova(fitS2, fitS, test='Chisq')
```

```
Likelihood ratio tests of Multinomial Models

Response: comen
               Model Resid. df Resid. Dev   Test    Df LR stat.   Pr(Chi)
1      charca*tamano       288   523.0005
2 charca*tamano*sexo       256   487.6018 1 vs 2    32 35.39865 0.3108632
```

```r
deviance(fit0) -  deviance(fitS)
```

```
[1] 116.7611
```

```r
deviance(fit1) -  deviance(fitS)
```

```
[1] 73.56589
```

```r
deviance(fit2) -  deviance(fitS)
```

```
[1] 101.6116
```

```r
deviance(fit3) -  deviance(fitS)
```

```
[1] 114.6571
```

```r
deviance(fit4) -  deviance(fitS)
```

```
[1] 52.47848
```

```r
deviance(fit5) -  deviance(fitS)
```

```
[1] 50.26368
```

```r
# Se usa como modelo el que NO considera sexo
deviance(fit4) - deviance(fitS2)
anova(fit4, fitS2, test='Chisq')
```

```
[1] 17.07983

Likelihood ratio tests of Multinomial Models

Response: comen
          Model Resid. df Resid. Dev   Test    Df LR stat.   Pr(Chi)
1 charca+tamano       300   540.0803
2 charca*tamano       288   523.0005 1 vs 2    12 17.07983 0.1466191
```

Se trabaja con el modelo que utiliza como covariables la *charca* y el *tamaño*. Se obtienen los coeficientes de los odds en los que se compara cada tipo de comida con la categoría base **peces**.

```r
# Se usa el modelo con las variables charca y tamano
summary(fit4, cor=F)
```

```
Call:
multinom(formula = comen ~ charca + tamano, data = tabla)

Coefficients:
          (Intercept) charcahancock charcaoklawaha charcatrafford tamano<2.3
invert      -1.549021    -1.6581178    0.937237973       1.122002  1.4581457
reptiles    -3.314512     1.2428408    2.458913302       2.935262 -0.3512702
pajaritos   -2.093358     0.6954256   -0.652622721       1.088098 -0.6306329
otros       -1.904343     0.8263115    0.005792737       1.516461  0.3315514

Std. Errors:
          (Intercept) charcahancock charcaoklawaha charcatrafford tamano<2.3
invert      0.4249185     0.6128466      0.4719035      0.4905122  0.3959418
reptiles    1.0530577     1.1854031      1.1181000      1.1163844  0.5800207
pajaritos   0.6622972     0.7813123      1.2020025      0.8417085  0.6424863
otros       0.5258313     0.5575446      0.7765655      0.6214371  0.4482504

Residual Deviance: 540.0803
AIC: 580.0803
```

Se pueden calcular también los correspondientes p-valores usando el método de intervalos aproximados de *Wald*.

```r
# p-valores aproximados mediante tests de Wald
z = summary(fit4)$coefficients/summary(fit4)$standard.errors
pvalores = (1 - pnorm(abs(z),0,1))*2
round(pvalores, 4)
```

```
          (Intercept) charcahancock charcaoklawaha charcatrafford tamano<2.3
invert         0.0003        0.0068         0.0470         0.0222     0.0002
reptiles       0.0016        0.2944         0.0279         0.0086     0.5448
pajaritos      0.0016        0.3734         0.5872         0.1961     0.3263
otros          0.0003        0.1383         0.9940         0.0147     0.4595
```

Se pueden estimar, por ejemplo, también las probabilidades de cada tipo de comida cuando el tamaño de un animal es $> 2{,}3$ y está en el lago *Hancok*.

```r
predict(fit4, type="probs", newdata=data.frame(tamano=">2.3",
charca="hancock"))
```

```
     peces     invert   reptiles  pajaritos      otros
0.57018414 0.02307664 0.07182898 0.14089666 0.19401358
```

También se pueden estimar las probabilidades de cada una de las categorías para distintos valores de las variables explicativas.

```
predictions = predict(fit4, type="probs",
newdata=expand.grid(tamano=tamano.f, charca=charca.f))

cbind(expand.grid(tamano=tamano.f,charca=charca.f), predictions)
```

```
  tamano   charca     peces     invert   reptiles   pajaritos      otros
1   <2.3  hancock 0.5352844 0.09311221 0.04745855 0.070402770 0.25374209
2   >2.3  hancock 0.5701841 0.02307664 0.07182898 0.140896661 0.19401358
3   <2.3 oklawaha 0.2581899 0.60188004 0.07723295 0.008820525 0.05387663
4   >2.3 oklawaha 0.4584248 0.24864187 0.19484367 0.029424141 0.06866547
5   <2.3 trafford 0.1843017 0.51682297 0.08877041 0.035897986 0.17420698
6   >2.3 trafford 0.2957471 0.19296047 0.20240167 0.108228505 0.20066229
7   <2.3   george 0.4521217 0.41284675 0.01156715 0.029664777 0.09379957
8   >2.3   george 0.6574619 0.13968167 0.02389991 0.081046951 0.09790956
```

Modelos Acumulados para datos ordinales

Cuando las respuestas de la variable categórica son ***ordinales*** se pueden utilizar modelos logit *acumulados*.

La probabilidad acumulada de una variable Y es la probabilidad de que Y sea menor o igual que un determinado valor j.

Así, para una categoría dada j se define la *probabilidad acumulada* como

$$P(Y \leq j) = \pi_1 + \cdots + \pi_j$$

para $j = 1, \ldots, J$.

Las probabilidades acumuladas reflejan el orden entre las categorías:

$$P(Y \leq 1) \leq P(Y \leq 2) \leq \cdots \leq P(Y \leq J) = 1$$

Los logits de las probabilidades acumuladas son

$$\begin{aligned} \operatorname{logit}\left[P(Y \leq j)\right] &= \log\left[\frac{P(Y \leq j)}{1 - P(Y \leq j)}\right] = \\ \log\left[\frac{\pi_1 + \cdots + \pi_j}{\pi_{j+1} + \cdots + \pi_J}\right] &= \log\left[\frac{\sum_{i=1}^{j} \pi_i}{\sum_{i=j+1}^{J} \pi_i}\right] \end{aligned}$$

para $j = 1, \ldots, J - 1$.

El modelo logit de odds proporcionales, cuando se consideran d variables explicativas $\mathbf{x} = (x_1, \ldots, x_d)$ se puede expresar como

$$\operatorname{logit}\left[P(Y \leq j|\mathbf{x})\right] = \alpha_j + \sum_{i=1}^{d} \beta_i x_i$$

para $j = 1, \ldots, J - 1$.

Cada variable independiente tiene un solo coeficiente que no depende del valor j.

La dependencia con respecto al valor j aparece solo en el coeficiente o constante α_j.

Teniendo en cuenta que si $j' < j$ entonces

$$\operatorname{logit}\left[P(Y \leq j'|\mathbf{x})\right] \leq \operatorname{logit}\left[P(Y \leq j|\mathbf{x})\right]$$

se tiene que verificar que $\alpha_j \leq \alpha_{j'}$, es decir, los valores de α_j aumentan junto con los valores de j.

Si se toman dos vectores de variables independientes

$$\begin{aligned} \mathbf{x}_1 &= (x_{11}, \ldots, x_{1d})' \\ \mathbf{x}_2 &= (x_{21}, \ldots, x_{2d})' \end{aligned}$$

entonces

$$\operatorname{logit}\left[P\left(Y \leq j|\mathbf{x}_1\right)\right] - \operatorname{logit}\left[P\left(Y \leq j|\mathbf{x}_2\right)\right] = \sum_{i=1}^{d} \beta_i \left(x_{1i} - x_{2i}\right).$$

Pero, por otro lado,

$$\operatorname{logit}\left[P\left(Y \leq j|\mathbf{x}_1\right)\right] - \operatorname{logit}\left[P\left(Y \leq j|\mathbf{x}_2\right)\right] = \log \left[\frac{\dfrac{P\left(Y \leq j|\mathbf{x}_1\right)}{P\left(Y > j|\mathbf{x}_1\right)}}{\dfrac{P\left(Y \leq j|\mathbf{x}_2\right)}{P\left(Y > j|\mathbf{x}_2\right)}}\right]$$

por lo que se puede reescribir como

$$\log \left[\frac{\dfrac{P\left(Y \leq j|\mathbf{x}_1\right)}{P\left(Y > j|\mathbf{x}_1\right)}}{\dfrac{P\left(Y \leq j|\mathbf{x}_2\right)}{P\left(Y > j|\mathbf{x}_2\right)}}\right] = \sum_{i=1}^{d} \beta_i \left(x_{1i} - x_{2i}\right).$$

La parte izquierda de la ecuación es el llamado cociente de odds *acumulado*.

De este modo, los odds para un valor de respuesta menor o igual que j para $\mathbf{x} = \mathbf{x}_1$ es igual a $\exp\left[\sum_{i=1}^{d} \beta_i \left(x_{1i} - x_{2i}\right)\right]$ veces los odds para $\mathbf{x} = \mathbf{x}_2$ y esta proporcionalidad **no** depende del valor de j. Es por esto que al modelo se le denomina también modelo de *odds proporcionales*.

Supongamos que los vectores $\mathbf{x}_1$ y $\mathbf{x}_2$ son iguales salvo en la componente i-ésima. Es decir, ambos individuos coinciden en todos los valores salvo en dicha componente y supongamos, además, que difieren en una unidad en la i-ésima componente. Se tiene, bajo estas hipótesis, la siguiente ecuación:

$$\frac{\dfrac{P\left(Y \leq j|\mathbf{x}_1\right)}{P\left(Y > j|\mathbf{x}_1\right)}}{\dfrac{P\left(Y \leq j|\mathbf{x}_2\right)}{P\left(Y > j|\mathbf{x}_2\right)}} = e^{\beta_i (x_{1i} - x_{2i})} = e^{\beta_i}.$$

Así, manteniendo todas las demás variables constantes, el cambio de los odds de la función de distribución de Y condicionada a x_1 y a x_2 es igual a e^{β_i}.

Ejemplo: Enfermedad mental

En el libro *Categorical Data Analysis* (2002) de Agresti (pag. 279) se muestran los datos de un estudio sobre una enfermedad mental que se trata de relacionar con dos variables explicativas. La enfermedad mental se resume en una variable categórica con los siguientes niveles: *buen estado*, *síntomas leves*, *síntomas moderados* y *enfermedad*. Como variables predictoras tenemos:

$x_1 \equiv$ mide el número de sucesos impactantes en la vida de la persona en los últimos tres años (divorcios, fallecimientos, etc.).

$x_2 \equiv$ Estatus socio-económico con niveles 1 (*alto*) y 0 (*bajo*).

La enfermedad mental, como variable *respuesta*, es un factor que presenta ordenación entre sus categorías.

```
dat = read.table(
"http://www.biostat.jhsph.edu/~bcaffo/aglm/files/
mentalestadoData.dat")

names(dat) = c("estado", "estatus", "sucesos")
dat$estado = as.ordered(dat$estado)
levels(dat$estado) = c("well", "mild", "mod", "imp")
dat$estatus = as.factor(dat$estatus)
levels(dat$estatus) = c("low", "high")

library(MASS)
fit = polr(estado ~ estatus + sucesos, data=dat)
summary(fit)
```

```
Call:
polr(formula = estado ~ estatus + sucesos, data = dat)

Coefficients:
              Value Std. Error t value
estatushigh -1.1112     0.6109  -1.819
sucesos      0.3189     0.1210   2.635

Intercepts:
          Value   Std. Error t value
well|mild -0.2819  0.6423    -0.4389
mild|mod   1.2128  0.6607     1.8357
mod|imp    2.2094  0.7210     3.0644

Residual Deviance: 99.0979
AIC: 109.0979
```

Se calculan ahora intervalos de confianza para los coeficientes originales y su expresión en términos de la exponencial.

```
# Intervalos de confianza
(ci = confint(fit))
```

```
                  2.5 %     97.5 %
estatushigh -2.34715412 0.06410162
sucesos      0.09203235 0.57185762
```

```
# Exponencial de los parametros e IC
exp(cbind(OR=coef(fit), ci))
```

```
                   OR      2.5 %   97.5 %
estatushigh 0.3291535 0.09564096 1.066201
sucesos     1.3755605 1.09640030 1.771555
```

```
# Seleccionas los de estatus 1 y mas de 5 sucesos
que = subset(dat, estatus=="high" & sucesos>5)

# Prediccion para el grupo anterior
cbind(que, predict(fit, data.frame(que), type="probs"))
```

```
   estado estatus sucesos      well      mild       mod       imp
2    well    high       9 0.1150236 0.2518325 0.2439856 0.3891584
10   well    high       7 0.1973878 0.3255946 0.2251307 0.2518869
17   mild    high       8 0.1516700 0.2918553 0.2399334 0.3165413
21   mild    high       9 0.1150236 0.2518325 0.2439856 0.3891584
29    mod    high       6 0.2527800 0.3485130 0.2020681 0.1966389
32    imp    high       8 0.1516700 0.2918553 0.2399334 0.3165413
34    imp    high       7 0.1973878 0.3255946 0.2251307 0.2518869
38    imp    high       8 0.1516700 0.2918553 0.2399334 0.3165413
```

Otra opción es usar la función `vglm`. Esta función está en la librería `VGAM`.

```
library(VGAM)
fit = vglm(estado ~ estatus + sucesos,
family=cumulative(parallel=TRUE),data=dat)

summary(fit)
```

```
Pearson residuals:
                    Min       1Q  Median     3Q   Max
logit(P[Y<=1]) -1.568 -0.7048 -0.2102 0.8070 2.713
logit(P[Y<=2]) -2.328 -0.4666  0.2657 0.6904 1.615
logit(P[Y<=3]) -3.688  0.1198  0.2039 0.4194 1.892

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept):1  -0.2819     0.6231  -0.452  0.65096
(Intercept):2   1.2128     0.6511   1.863  0.06251 .
(Intercept):3   2.2094     0.7171   3.081  0.00206 **
estatushigh     1.1112     0.6143   1.809  0.07045 .
sucesos        -0.3189     0.1194  -2.670  0.00759 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Number of linear predictors:  3

Names of linear predictors: logit(P[Y<=1]), logit(P[Y<=2]), logit(P[Y<=3])

Residual deviance: 99.0979 on 115 degrees of freedom

Log-likelihood: -49.5489 on 115 degrees of freedom

Number of iterations: 5

Exponentiated coefficients:
estatushigh     sucesos
  3.0380707   0.7269742
```

Se puede considerar el mismo modelo con SAS University:

```
OPTIONS nodate ls=75;
/* Para SAS University */
/* ODS listing file='/folders/myfolders/sale.lst'; */
DATA mental;
INFILE "/folders/myfolders/DatosMental.txt";
INPUT estado estatus sucesos;

PROC genmod;
model estado = sucesos estatus / dist=multinomial
link=clogit lrci type3;
RUN;
/* ODS listing close; */
```

O en SAS estándar:

```
OPTIONS nodate ls=65 formchar='|----|+|---+=|-/\<>*';
x 'cd "c:\dondeSea"';

DATA mental;
INFILE 'DatosMental.txt';
INPUT estado estatus sucesos;

PROC genmod;
model estado = sucesos estatus / dist=multinomial
link=clogit lrci type3;
RUN;
```

Criterio para evaluar bondad de ajuste			
Criterio	DF	Valor	Valor/DF
Verosimilitud log		-49.5489	
Verosimilitud log completa		-49.5489	
AIC (mejor más pequeño)		109.0979	
AICC (mejor más pequeño)		110.8626	
BIC (mejor más pequeño)		117.5423	

Algoritmo convergido.

Análisis de estimadores de parámetro de verosimilitud máxima							
Parámetro	DF	Estimación	Error estándar	Límites de confianza de ratio de verosimilitud al 95%		Chi-cuadrado de Wald	Pr > ChiSq
Intercept1	1	-0.2819	0.6423	-1.5615	0.9839	0.19	0.6607
Intercept2	1	1.2128	0.6607	-0.0507	2.5656	3.37	0.0664
Intercept3	1	2.2094	0.7210	0.8590	3.7123	9.39	0.0022
sucesos	1	-0.3189	0.1210	-0.5718	-0.0920	6.95	0.0084
estatus	1	1.1112	0.6109	-0.0641	2.3471	3.31	0.0689
Escala	0	1.0000	0.0000	1.0000	1.0000		

Note: The scale parameter was held fixed.

Estadísticos LR para análisis de tipo 3			
Origen	DF	Chi-cuadrado	Pr > ChiSq
sucesos	1	7.78	0.0053
estatus	1	3.43	0.0641

Ejemplo

Se tiene una muestra de 735 personas a los que se pregunta por sus preferencias en cuanto a tres marcas (*brands*) de algunos productos. Se considera además el género y la edad de las personas de la encuesta.

Los datos (en formato SAS) se pueden descargar en

`http://www.ats.ucla.edu/stat/sas/dae/mlogit.sas7bdat`

La variable dependiente es `brand`. La variable `female` se codifica como 0 para hombres y 1 para mujeres.

```
library(sas7bdat)

lacosa = read.sas7bdat(
"http://www.ats.ucla.edu/stat/sas/dae/mlogit.sas7bdat")
colnames(lacosa) = c("brand","female","age")
attach(lacosa)
```

```
# Se considera un analisis descriptivo
table(brand)
```

```
  1   2   3
207 307 221
```

```
table(female)
```

```
female
  0   1
269 466
```

```
summary(age)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   24.0    32.0    32.0    32.9    34.0    38.0
```

```
xtabs(~ female + brand)
```

```
         brand
female    1   2   3
     0   92  99  78
     1  115 208 143
```

Se aplica un modelo de regresión multinomial. Se usa el comando `mlogit.data` para expandir el formato de los datos. Así, para cada observación de la base de datos original, se obtienen 3 observaciones: una para cada una de los valores de la variable `brand`.

Usando el nuevo formato, el valor de `brand` aparece en la variable `alt`, y si una rama ha sido o no seleccionada por la persona se indica como `TRUE` o `FALSE`.

```
library(mlogit)
lacosa$brand = as.factor(lacosa$brand)

# Con mlogit.data se expanden los datos
mldata = mlogit.data(lacosa, varying=NULL, choice="brand",
shape="wide")

head(mldata)
```

```
    brand female age chid alt
1.1  TRUE      0  24    1   1
1.2 FALSE      0  24    1   2
1.3 FALSE      0  24    1   3
2.1  TRUE      0  26    2   1
2.2 FALSE      0  26    2   2
2.3 FALSE      0  26    2   3
...
```

```
# brand 1 es el nivel de referencia
mlogit.model = mlogit(brand~0|female+age, data=mldata, reflevel="1")
summary(mlogit.model)
```

```
Frequencies of alternatives:
      1       2       3
0.28163 0.41769 0.30068

nr method
5 iterations, 0h:0m:0s
g'(-H)^-1g = 0.00158
successive function values within tolerance limits

Coefficients :
                Estimate Std. Error  t-value  Pr(>|t|)
2:(intercept) -11.774478   1.774612  -6.6350 3.246e-11 ***
3:(intercept) -22.721201   2.058028 -11.0403 < 2.2e-16 ***
2:female        0.523813   0.194247   2.6966  0.007004 **
3:female        0.465939   0.226090   2.0609  0.039316 *
2:age           0.368201   0.055003   6.6942 2.169e-11 ***
3:age           0.685902   0.062627  10.9523 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Log-Likelihood: -702.97
McFadden R^2:  0.11676
Likelihood ratio test : chisq = 185.85 (p.value = < 2.22e-16)
```

En los resultados anteriores se obtienen los coeficientes y sus p-valores. Cada variable independiente y el `intercept` aparecen dos veces, una para cada una de las categorías denominadas `alt2` y `alt3`. Corresponden a dos ecuaciones:

$$\log\left(\frac{P(\text{brand}=2)}{P(\text{brand}=1)}\right) = b_{10} + b_{11}\cdot\text{female} + b_{12}\cdot\text{age}$$

$$\log\left(\frac{P(\text{brand}=3)}{P(\text{brand}=1)}\right) = b_{20} + b_{21}\cdot\text{female} + b_{22}\cdot\text{age}$$

donde los b_{ij} son los coeficientes de regresión de la salida.

Por ejemplo, por cada aumento en una unidad de la variable `age`, el logaritmo del ratio de las dos probabilidades, $P(\text{brand}=2)/P(\text{brand}=1)$, se incrementa en 0.368, y el logaritmo del ratio de las dos probabilidades, $P(\text{brand}=3)/P(\text{brand}=1)$, se incrementa en 0.686. Por tanto, en general, cuanto mayor sea una persona tendrá más preferencia por `brand` igual a 2 ó a 3, que por `brand` igual a 1.

A continuación, se muestran los resultados del modelo en términos de las probabilidades. Por ejemplo, se muestra un rango de distintas edades y se calculan las probabilidades de escoger cada categoría de `brand` para mujeres y hombres. Se generan los valores predichos en la escala logit usando los coeficientes del modelo. En `brand` $=1$, el valor se fija en 0.

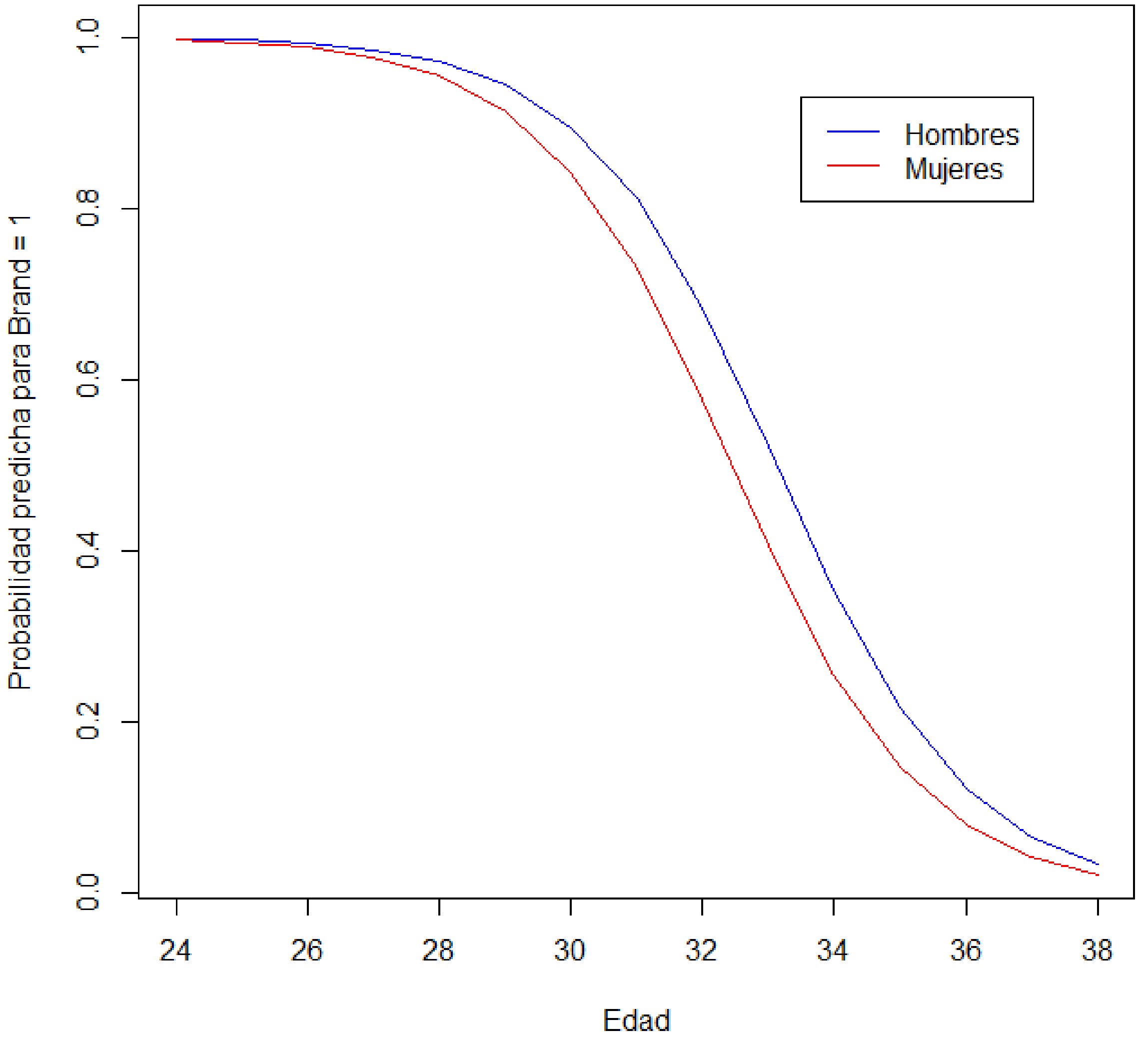
Las columnas etiquetadas como `pred.1`, `pred.2`, y `pred.3`, contienen las probabilidades predichas de que `brand` sea igual a 1, 2 y 3 respectivamente.

```
newdata = data.frame(cbind(age=rep(24:38, 2),
female=c(rep(0, 15), rep(1, 15))))
logit1 = rep(0, 30)

logit2 = mlogit.model$coefficients[[1]] +
mlogit.model$coefficients[[3]]*newdata$female +
mlogit.model$coefficients[[5]]*newdata$age

logit3 = mlogit.model$coefficients[[2]] +
mlogit.model$coefficients[[4]]*newdata$female +
mlogit.model$coefficients[[6]]*newdata$age

logits = cbind(logit1, logit2, logit3)
p.unscaled = exp(logits)
p = cbind(newdata, (p.unscaled / rowSums(p.unscaled)))
colnames(p) = c("age", "female", "pred.1", "pred.2", "pred.3")
print(p)
```

```
   age female    pred.1     pred.2      pred.3
1   24      0 0.94795544 0.05023196 0.001812592
2   25      0 0.92560550 0.07088034 0.003514161
3   26      0 0.89429225 0.09896621 0.006741533
4   27      0 0.85114170 0.13611844 0.012739858
5   28      0 0.79312711 0.18330129 0.023571598
6   29      0 0.71787603 0.23976170 0.042362268
7   30      0 0.62506819 0.30169310 0.073238707
8   31      0 0.51809475 0.36137225 0.120533003
9   32      0 0.40487183 0.40810394 0.187024226
10  33      0 0.29639567 0.43175036 0.271853968
11  34      0 0.20299473 0.42732003 0.369685239
12  35      0 0.13058003 0.39723991 0.472180052
13  36      0 0.07951592 0.34957297 0.570911107
14  37      0 0.04627661 0.29400375 0.659719642
15  38      0 0.02598254 0.23855066 0.735466793
16  24      1 0.91531696 0.08189411 0.002788936
17  25      1 0.88078797 0.11388332 0.005328709
18  26      1 0.83412301 0.15585707 0.010019920
19  27      1 0.77287121 0.20869457 0.018434221
20  28      1 0.69561310 0.27144346 0.032943439
21  29      1 0.60315223 0.34013099 0.056716789
22  30      1 0.49958719 0.40713475 0.093278061
23  31      1 0.39239931 0.46212850 0.145472190
24  32      1 0.29086432 0.49503122 0.214104459
25  33      1 0.20320727 0.49979166 0.297001073
26  34      1 0.13411370 0.47668396 0.389202343
27  35      1 0.08404322 0.43168573 0.484271045
28  36      1 0.05034227 0.37368477 0.575972963
29  37      1 0.02903251 0.31143314 0.659534350
30  38      1 0.01623160 0.25162236 0.732146040
```

Se pueden presentar gráficamente las predicciones. Por ejemplo, se dibujan las probabilidades predichas de que una persona seleccione `brand` $= 1$ en función de la variable `age`, cuando es hombre (`female` $= 0$, en *azul*), y cuando es mujer (`female` $= 1$, en *rojo*).

```
plot(p$pred.1[p$female==0]~p$age[p$female==0],type="l",col="blue",
lwd=1,ylab="Probabilidad predicha para Brand = 1", xlab="Edad")

lines(p$pred.1[p$female==1]~p$age[p$female==1],col="red",lwd=1)
legend(33.5,.93,c("Hombres","Mujeres"),col=c("blue","red"),lwd=c(1,1))
```


Hombres
Mujeres
1.0
0.8
0.6
0.4
0.2
0.0
Probabilidad predicha para Brand = 1
24
26
28
30
32
34
36
38
Edad

Se usa ahora SAS Unversity para tratar el mismo conjunto de datos.

```
OPTIONS nodate ls=75;
/* ODS listing file='/folders/myfolders/sale.lst'; */
DATA market;
INFILE '/folders/myfolders/market.txt';
INPUT brand female age;

PROC freq data=market;
tables brand;

PROC sort data=market;
by brand;

PROC means data=market;
by brand;
var age female;

PROC logistic data=market;
class brand (ref = "1");
model brand = female age / link = glogit;
RUN;
/* ODS listing close; */
```

O en SAS estándar

```
OPTIONS nodate ls=65 formchar='|----|+|---+=|-/\<>*';
x 'cd "c:\deSAS"';

DATA market;
INFILE 'market.txt';
INPUT brand female age;

... Igual que el anterior programa ...
```

Procedimiento FREQ

brand	Frecuencia	Porcentaje	Frecuencia acumulada	Porcentaje acumulado
1	207	28.16	207	28.16
2	307	41.77	514	69.93
3	221	30.07	735	100.00

Procedimiento MEANS

brand=1

Variable	N	Media	Desv. est.	Mínimo	Máximo
age	207	31.4879227	2.1083742	24.0000000	38.0000000
female	207	0.5555556	0.4981086	0	1.0000000

brand=2

Variable	N	Media	Desv. est.	Mínimo	Máximo
age	307	32.8436482	1.8243945	28.0000000	38.0000000
female	307	0.6775244	0.4681870	0	1.0000000

brand=3

Variable	N	Media	Desv. est.	Mínimo	Máximo
age	221	34.3031674	2.3478111	27.0000000	38.0000000
female	221	0.6470588	0.4789695	0	1.0000000

Perfil de respuesta		
Valor ordenado	brand	Frecuencia total
1	1	207
2	2	307
3	3	221

Los logits modelados usan brand=1 como categoría de referencia.

Estado de convergencia del modelo
Criterio de convergencia (GCONV=1E-8) satisfecho.

Estadísticos de ajuste del modelo		
Criterio	Sólo término independiente	Término independiente y covariables
AIC	1595.792	1417.941
SC	1604.991	1445.541
-2 LOG L	1591.792	1405.941

Probando hipótesis nula global: BETA=0			
Test	Chi-cuadrado	DF	Pr > ChiSq
Ratio de verosim	185.8502	4	<.0001
Puntuación	163.9538	4	<.0001
Wald	129.7966	4	<.0001

Análisis de efectos Tipo 3			
Efecto	DF	Chi-cuadrado de Wald	Pr > ChiSq
female	2	7.6704	0.0216
age	2	123.3880	<.0001

Análisis de estimación de verosimilitud máxima						
Parámetro	brand	DF	Estimación	Error estándar	Chi-cuadrado de Wald	Pr > ChiSq
Intercept	2	1	-11.7746	1.7746	44.0239	<.0001
Intercept	3	1	-22.7214	2.0580	121.8897	<.0001
female	2	1	0.5238	0.1942	7.2719	0.0070
female	3	1	0.4659	0.2261	4.2472	0.0393
age	2	1	0.3682	0.0550	44.8133	<.0001
age	3	1	0.6859	0.0626	119.9541	<.0001

Estimadores de ratio de probabilidades				
Efecto	brand	Estimador de punto	Límites de confianza de Wald al 95%	
female	2	1.688	1.154	2.471
female	3	1.594	1.023	2.482
age	2	1.445	1.297	1.610
age	3	1.986	1.756	2.245

Así, se obtiene que las variables `female` y `age` son significativas en los dos modelos. Las mujeres parecen preferir `brand` igual a 2 ó igual a 3 en comparación con `brand` igual a 1. Por otro lado, cuanta más edad tiene una persona es más probable que prefiera `brand` igual a 2 ó a 3 que `brand` igual a 1.

Se observa que con el cambio en una unidad en la variable `age` (un año mayor), se espera que la razón de odds entre elegir `brand` $= 2$ respecto de `brand` $= 1$ se incrementa en $e^{0{,}3682} = 1{,}45$.

En el caso del sexo de las personas, `female`, la razón de odds de elegir `brand` $= 2$ respecto de 1 se incrementa en $e^{0{,}5238} = 1{,}69$.