MultiLogit_MaxEntropyClassif

2023-01-16

Multinomial Logit analysys or Maximun Entropy Classifier

1) First way: Multinomial logit analysis by proportional hazard regression model

```
library(survival)
choco_candy<-read.csv("C:\\Users\\Leobardo\\Documents\\GitHub\\MultiLogit_MaxEntropyClassif\\choco_candy
```

Coef: Multinomial Logit Parameter Estimates

se(coef): Estandard Error

Chi-Square: z^2

Pr > ChiSq: Pr(>|z|)

```
output1<-coxph(Surv(Subj,Choice)~Dark+Soft+Nuts, data=choco_candy,
               ties="breslow")
summary(output1)
```

```
## Call:
## coxph(formula = Surv(Subj, Choice) ~ Dark + Soft + Nuts, data = choco_candy,
##     ties = "breslow")
##
##   n= 80, number of events= 10
##
##         coef exp(coef) se(coef)      z Pr(>|z|)
## Dark  1.3863    4.0000   0.7906  1.754   0.0795 .
## Soft -2.1972    0.1111   1.0541 -2.084   0.0371 *
## Nuts  0.8473    2.3333   0.6901  1.228   0.2195
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      exp(coef) exp(-coef) lower .95 upper .95
## Dark    4.0000     0.2500   0.84943    18.836
## Soft    0.1111     9.0000   0.01408     0.877
## Nuts    2.3333     0.4286   0.60338     9.023
##
## Concordance= 0.769  (se = 0.078 )
## Likelihood ratio test= 12.86  on 3 df,   p=0.005
## Wald test            = 8.93  on 3 df,   p=0.03
## Score (logrank) test = 11.6  on 3 df,   p=0.009
```

Probability of choice

To get the probabilities of each alternative

```
predict(output1)
```

```
##  [1]  0.00000000  0.84729786 -2.19722458 -1.34992672  1.38629436  2.23359222
##  [7] -0.81093022  0.03636764  0.00000000  0.84729786 -2.19722458 -1.34992672
## [13]  1.38629436  2.23359222 -0.81093022  0.03636764  0.00000000  0.84729786
## [19] -2.19722458 -1.34992672  1.38629436  2.23359222 -0.81093022  0.03636764
## [25]  0.00000000  0.84729786 -2.19722458 -1.34992672  1.38629436  2.23359222
## [31] -0.81093022  0.03636764  0.00000000  0.84729786 -2.19722458 -1.34992672
## [37]  1.38629436  2.23359222 -0.81093022  0.03636764  0.00000000  0.84729786
## [43] -2.19722458 -1.34992672  1.38629436  2.23359222 -0.81093022  0.03636764
## [49]  0.00000000  0.84729786 -2.19722458 -1.34992672  1.38629436  2.23359222
## [55] -0.81093022  0.03636764  0.00000000  0.84729786 -2.19722458 -1.34992672
## [61]  1.38629436  2.23359222 -0.81093022  0.03636764  0.00000000  0.84729786
## [67] -2.19722458 -1.34992672  1.38629436  2.23359222 -0.81093022  0.03636764
## [73]  0.00000000  0.84729786 -2.19722458 -1.34992672  1.38629436  2.23359222
## [79] -0.81093022  0.03636764
```

Alternatives and predicted value (exponential)

```
exp_p<- cbind(choco_candy$Alt, data.frame(exp(predict(output1))))[1:8,]
exp_p
```

```
##   choco_candy$Alt exp.predict.output1..
## 1               A             1.0000000
## 2               B             2.3333333
## 3               C             0.1111111
## 4               D             0.2592593
## 5               E             4.0000000
## 6               F             9.3333333
## 7               G             0.4444444
## 8               H             1.0370370
```

Rename variables

```
names(exp_p)[1]<-"Alt"
names(exp_p)[2]<-"Exp"
```

Calculating probability of choice:

$$p(c_i|C) = \frac{exp(x_i\beta)}{\sum_{j=1}^{m} exp(x_j\beta)}$$

```
exp_p$prob<-exp_p$Exp/sum(exp_p$Exp)
exp_p$prob
```

```
## [1] 0.054 0.126 0.006 0.014 0.216 0.504 0.024 0.056
```

change the order of the probability of choice:

Choice of chocolate candies

```
exp_p[c(order(-exp_p$prob)),]
```

```
##   Alt       Exp  prob
## 6   F 9.3333333 0.504
## 5   E 4.0000000 0.216
## 2   B 2.3333333 0.126
## 8   H 1.0370370 0.056
```

```
## 1   A 1.0000000 0.054
## 7   G 0.4444444 0.024
## 4   D 0.2592593 0.014
## 3   C 0.1111111 0.006
```

2) Second way: Multinomial logit analysis by mlogit

```
library(mlogit)
```

```
## Loading required package: dfidx
```

```
##
## Attaching package: 'dfidx'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

Understanding data structure

```
head(choco_candy)
```

```
##   Subj Choice Alt Dark Soft Nuts
## 1    1      0   A    0    0    0
## 2    1      0   B    0    0    1
## 3    1      0   C    0    1    0
## 4    1      0   D    0    1    1
## 5    1      1   E    1    0    0
## 6    1      0   F    1    0    1
```

convert choco_candy data to a mlogit data format

```
TM<-mlogit.data(choco_candy,choice="Choice", shape="long",
                chid.var="Subj", alt.var="Alt", drop.index=TRUE)

TM
```

```
## ~~~~~~~
##  first 10 observations out of 80
## ~~~~~~~
##    Choice Dark Soft Nuts idx
## 1   FALSE    0    0    0 1:A
## 2   FALSE    0    0    1 1:B
## 3   FALSE    0    1    0 1:C
## 4   FALSE    0    1    1 1:D
## 5    TRUE    1    0    0 1:E
## 6   FALSE    1    0    1 1:F
## 7   FALSE    1    1    0 1:G
## 8   FALSE    1    1    1 1:H
## 9   FALSE    0    0    0 2:A
## 10  FALSE    0    0    1 2:B
##
## ~~~ indexes ~~~~
##    chid alt
## 1     1   A
## 2     1   B
## 3     1   C
```

```
## 4     1   D
## 5     1   E
## 6     1   F
## 7     1   G
## 8     1   H
## 9     2   A
## 10    2   B
## indexes:  1, 2
```

We need to add 0 because we do not have an intercept, if not, there is an error.

```
output2<-mlogit(Choice ~ 0 + Dark+Soft+Nuts, data=TM)
summary(output2)
```

```
##
## Call:
## mlogit(formula = Choice ~ 0 + Dark + Soft + Nuts, data = TM,
##     method = "nr")
##
## Frequencies of alternatives:choice
##   A   B   C   D   E   F   G   H
## 0.0 0.2 0.0 0.0 0.2 0.5 0.1 0.0
##
## nr method
## 5 iterations, 0h:0m:0s
## g'(-H)^-1g = 1.98E-05
## successive function values within tolerance limits
##
## Coefficients :
##      Estimate Std. Error z-value Pr(>|z|)
## Dark  1.38629    0.79057  1.7535  0.07951 .
## Soft -2.19722    1.05409 -2.0845  0.03712 *
## Nuts  0.84730    0.69007  1.2279  0.21950
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -14.363
```

Probability of choice

To get the probabilities of each alternative

```
predict(output2)
```

```
##           A           B           C           D           E           F
## 0.053999953 0.125999889 0.006000047 0.014000111 0.215999810 0.503999557
##           G           H
## 0.024000190 0.056000443
```

Alternatives and predicted value (exponential)

```
exp_p<- cbind(choco_candy$Alt, data.frame(exp(predict(output2))))[1:8,]
```

```
## Warning in data.frame(..., check.names = FALSE): row names were found from a
## short variable and have been discarded
```

```
exp_p
```

```
##   choco_candy$Alt exp.predict.output2..
## 1               A              1.055485
## 2               B              1.134282
## 3               C              1.006018
## 4               D              1.014099
## 5               E              1.241102
## 6               F              1.655329
## 7               G              1.024291
## 8               H              1.057598
```

Rename variables

```
names(exp_p)[1]<-"Alt"
names(exp_p)[2]<-"Exp"
```

Calculating probability of choice:

$$p(c_i|C) = \frac{exp(x_i\beta)}{\sum_{j=1}^{m} exp(x_j\beta)}$$

```
exp_p$prob<-exp_p$Exp/sum(exp_p$Exp)
exp_p$prob
```

```
## [1] 0.1148739 0.1234498 0.1094902 0.1103696 0.1350756 0.1801580 0.1114789
## [8] 0.1151039
```

change the order of the probability of choice:

Choice of chocolate candies

```
exp_p[c(order(-exp_p$prob)),]
```

```
##   Alt      Exp      prob
## 6   F 1.655329 0.1801580
## 5   E 1.241102 0.1350756
## 2   B 1.134282 0.1234498
## 8   H 1.057598 0.1151039
## 1   A 1.055485 0.1148739
## 7   G 1.024291 0.1114789
## 4   D 1.014099 0.1103696
## 3   C 1.006018 0.1094902
```

3) Third way. Rchoice library

```
library(Rchoice)
```

```
## Loading required package: Formula
```

```
## Loading required package: maxLik
```

```
## Loading required package: miscTools
```

```
##
## Please cite the 'maxLik' package as:
## Henningsen, Arne and Toomet, Ott (2011). maxLik: A package for maximum likelihood estimation in R. C
##
## If you have questions, suggestions, or comments regarding the 'maxLik' package, please use a forum o
## https://r-forge.r-project.org/projects/maxlik/
```

```r
output3<-Rchoice(Choice ~ 0 + Dark + Soft + Nuts, data = TM,
                 family =binomial("logit") )
summary(output3)
```

```
##
## Model: binomial
## Model estimated on: dom. nov. 12 11:28:24 p. m. 2023
##
## Call:
## Rchoice(formula = Choice ~ 0 + Dark + Soft + Nuts, data = TM,
##     family = binomial("logit"), method = "nr")
##
##
## Frequencies of categories:
## y
## FALSE  TRUE
## 0.875 0.125
## The estimation took: 0h:0m:0s
##
## Coefficients:
##      Estimate Std. Error z-value Pr(>|z|)
## Dark  -0.2136     0.5085  -0.420  0.67440
## Soft  -3.3065     1.0378  -3.186  0.00144 **
## Nuts  -0.5940     0.5191  -1.144  0.25248
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Optimization of log-likelihood by Newton-Raphson maximisation
## Log Likelihood: -31.12
## Number of observations: 80
## Number of iterations: 6
## Exit of MLE: gradient close to zero (gradtol)
```

Coefficients

```r
coef<-output3$coefficients
coef
```

```
##       Dark       Soft       Nuts
## -0.2136363 -3.3065001 -0.5940417
```

```r
coef_percent<-(exp(coef)-1)*100
coef_percent
```

```
##      Dark      Soft      Nuts
## -19.23580 -96.33558 -44.79086
```