Logit and Probit Models

Zahid Asghar

2023-12-20

Table of contents

Read Data on Insurance downloaded from Econometric Academy An...

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
library(tidyverse)

ins1<-read_csv("data/probit_insurance.csv") # insurance data
ins1<-ins1 %>% select(retire, age, hstatusg, hhincome, educyear, married, hisp ins)
```

Slide 3

```
table(ins1$ins)
```

```
   0    1
1965 1241
```

```
ins1 %>% group_by(ins) %>%
  summarise(count=n())
```

```
# A tibble: 2 x 2
    ins count
  <dbl> <int>
1     0  1965
2     1  1241
```

> **Warning**
>
> Never use OLS when dependent variable is binary.

Including Plots

You can also embed plots, for example:

```
Call:
glm(formula = ins ~ ., family = binomial(link = "logit"), data = ins1)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.715578   0.748622  -2.292 0.021926 *
retire       0.196930   0.084207   2.339 0.019354 *
age         -0.014596   0.011287  -1.293 0.195969
hstatusg     0.312265   0.091674   3.406 0.000659 ***
hhincome     0.002304   0.000762   3.023 0.002503 **
educyear     0.114263   0.014201   8.046 8.55e-16 ***
married      0.578636   0.093320   6.201 5.63e-10 ***
hisp        -0.810306   0.195751  -4.139 3.48e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 4279.5  on 3205  degrees of freedom
Residual deviance: 3989.8  on 3198  degrees of freedom
```

```
AIC: 4005.8

Number of Fisher Scoring iterations: 4

# A tibble: 8 x 5
  term        estimate std.error statistic  p.value
  <chr>          <dbl>     <dbl>     <dbl>    <dbl>
1 (Intercept) -1.72     0.749        -2.29 2.19e- 2
2 retire       0.197    0.0842        2.34 1.94e- 2
3 age         -0.0146   0.0113       -1.29 1.96e- 1
4 hstatusg     0.312    0.0917        3.41 6.59e- 4
5 hhincome     0.00230  0.000762      3.02 2.50e- 3
6 educyear     0.114    0.0142        8.05 8.55e-16
7 married      0.579    0.0933        6.20 5.63e-10
8 hisp        -0.810    0.196        -4.14 3.48e- 5
```

```r
## Round to 3 decimal places
tidy(logit) %>% mutate_if(is.numeric, round, 3)
```

```
# A tibble: 8 x 5
  term        estimate std.error statistic p.value
  <chr>          <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)   -1.72      0.749     -2.29   0.022
2 retire         0.197     0.084      2.34   0.019
3 age           -0.015     0.011     -1.29   0.196
4 hstatusg       0.312     0.092      3.41   0.001
5 hhincome       0.002     0.001      3.02   0.003
6 educyear       0.114     0.014      8.05   0
7 married        0.579     0.093      6.20   0
8 hisp          -0.81      0.196     -4.14   0
```

Slide 4

```r
# Logit model odds ratios round to 2 decimal places

exp(logit$coefficients) |> round(2)
```

```
(Intercept)      retire         age    hstatusg    hhincome    educyear
       0.18        1.22        0.99        1.37        1.00        1.12
    married        hisp
       1.78        0.44
```

Table 1

```
# Probit model coefficients
probit<- glm(ins ~ .,data=ins1, family=binomial (link="probit"))


# Round to 3 decimal places
tidy(probit) %>% mutate_if(is.numeric, round, 3)
```

```
# A tibble: 8 x 5
  term        estimate std.error statistic p.value
  <chr>          <dbl>     <dbl>     <dbl>   <dbl>
1 (Intercept)   -1.07      0.455     -2.35   0.019
2 retire         0.118     0.051      2.31   0.021
3 age           -0.009     0.007     -1.29   0.196
4 hstatusg       0.198     0.055      3.56   0
5 hhincome       0.001     0          2.82   0.005
6 educyear       0.071     0.008      8.33   0
7 married        0.362     0.056      6.46   0
8 hisp          -0.473     0.11      -4.29   0
```

Table 2

```
# Regression marginal effects
olsreg<-lm(ins~.,data = ins1)
coef(olsreg) |> round(3)
```

```
(Intercept)      retire         age    hstatusg    hhincome    educyear
      0.127       0.041      -0.003       0.066       0.000       0.023
    married        hisp
      0.123      -0.121
```

```
# Logit model average marginal effects
LogitScalar <- mean(dlogis(predict(logit, type = "link")))
LogitScalar * coef(logit) |> round(3)
```

```
 (Intercept)       retire          age     hstatusg     hhincome     educyear
-0.372614809  0.042776875 -0.003257122  0.067748147  0.000434283  0.024754131
```

```
     married         hisp
 0.125724927 -0.175884612
```

```
# Probit model average marginal effects
ProbitScalar <- mean(dnorm(predict(probit, type = "link")))
ProbitScalar * coef(probit)
```

```
  (Intercept)        retire           age      hstatusg      hhincome
-0.3792718940  0.0419770495 -0.0031457860  0.0701343649  0.0004371967
     educyear       married          hisp
 0.0250931581  0.1285130972 -0.1678031247
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```
# Regression predicted probabilities
polsreg<- predict(olsreg)
summary(polsreg)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
-0.1557  0.3055  0.4074  0.3871  0.4736  1.1972
```

```
# Logit model predicted probabilities
plogit<- predict(logit, type="response")
summary(plogit)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.03402 0.28493 0.39942 0.38709 0.47778 0.96496
```

```
# Probit model predicted probabilities
pprobit<- predict(probit, type="response")
summary(pprobit)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.02064 0.28663 0.40170 0.38611 0.47678 0.96473
```

```
# Percent correctly predicted values
table(true = ins1$ins, pred = round(fitted(probit)))
```

```
    pred
true    0    1
   0 1660  305
   1  906  335
```

```
table(true = ins1$ins, pred = round(fitted(logit)))
```

```
    pred
true    0    1
   0 1657  308
   1  896  345
```

```
# McFadden's Pseudo R-squared
probit0<-update(probit, formula= ins ~ 1)
McFadden<- 1-as.vector(logLik(probit)/logLik(probit0))
McFadden
```

```
[1] 0.06830054
```