Econ4274\_problem\_set\_4

Man Adrian Ho Hin

2023-05-02

# Question 1

rm(list=ls())  
DATA = read.csv("hospital\_choice.csv")  
  
LPM=lm(Ducla~income + distance + old, data=DATA)  
logit=glm(Ducla~income + distance + old, data=DATA,family = binomial(link='logit'))  
probit=glm(Ducla~income + distance + old, data=DATA,family = binomial(link='probit'))  
  
R2\_LPM = summary(LPM)$adj.r.squared  
  
LLF\_ur\_logit = logLik(logit)  
LLF\_r\_logit = logLik(glm(Ducla~1,data=DATA,family = binomial(link='logit')))  
R2\_logit = 1-(LLF\_ur\_logit/LLF\_r\_logit)  
  
LLF\_ur\_probit = logLik(probit)  
LLF\_r\_probit = logLik(glm(Ducla~1,data=DATA,family = binomial(link='probit')))  
R2\_probit = 1-(LLF\_ur\_probit/LLF\_r\_probit)  
  
R2\_LPM;R2\_logit;R2\_probit

## [1] 0.2219862

## 'log Lik.' 0.1949669 (df=4)

## 'log Lik.' 0.1978485 (df=4)

library(stargazer)

##   
## Please cite as:

## Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.

## R package version 5.2.3. https://CRAN.R-project.org/package=stargazer

stargazer(LPM, logit, probit, type="html", out="1.html")



# So in terms of the R^2, the performance of the models are at following ranking: linear > logit > probit

# Question 2a

rm(list=ls())  
set.seed(1)  
n = 200  
beta1 = 1  
beta2 = 0.5  
sigma = 2  
e = rnorm(n, 0, sigma)  
x = rgamma(n, 2)  
y = beta1 + beta2\*x + e  
  
sample = cbind(e,x,y)

# Question 2b

L<-function(theta){  
 v=-sum(log((1/sqrt(2\*pi\*theta[3]^2))\*exp(-((y-theta[1]-theta[2]\*x)^2)/(2\*theta[3]^2))))  
 return(v)  
}  
MLE=optim(c(1,1,1),L)  
  
MLE$par

## [1] 1.1888207 0.4425523 1.8513307

# Question 2c

OLS = lm(y~x)  
  
OLS\_coeff = c(OLS$coefficients, sd(OLS$residuals))  
OLS\_coeff; MLE$par

## (Intercept) x   
## 1.1895594 0.4423874 1.8560629

## [1] 1.1888207 0.4425523 1.8513307

-MLE$value;logLik(OLS)

## [1] -406.978

## 'log Lik.' -406.978 (df=3)

# we can observe that the coefficient of both MLE and OLS are close together, also close to the true parameters of interest. The OLS might be better as a slightly closer estimations for beta1 and sigma, lower in the beta2 with extremely small value.

# Question 3a

rm(list=ls())  
DATA = read.csv("heating.csv")  
# individual: idcase  
# alternative: alt  
# choice: depvar  
# This dataset is in long mode  
library(mlogit)

## Loading required package: dfidx

##   
## Attaching package: 'dfidx'

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

LDATA=mlogit.data(DATA,choice="depvar",shape="long", alt.var = "alt")

# Question 3b

DATA$region = as.factor(DATA$region)  
LDATA$region = as.factor(LDATA$region)  
  
m1=mlogit(depvar~ic+oc+0,data=LDATA)  
m2=mlogit(depvar~ic+oc+0 | income+agehed+rooms+0,data=LDATA)  
m3=mlogit(depvar~ic+oc+0 | income+agehed+rooms+region+0,data=LDATA)  
m0=mlogit(depvar~1,data=LDATA)  
  
m1R2=1-m1$logLik/m0$logLik  
m2R2=1-m2$logLik/m0$logLik  
m3R2=1-m3$logLik/m0$logLik  
m1R2;m2R2;m3R2

## 'log Lik.' -0.07142608 (df=2)

## 'log Lik.' 0.01469996 (df=14)

## 'log Lik.' 0.02291275 (df=26)

library(stargazer)  
stargazer(m1,m2,m3, type="html", out="3b.html")



# Question 3c

# (i)  
ic\_LDATA = LDATA  
ic\_LDATA$ic[which(ic\_LDATA$alt=="ec")] = ic\_LDATA$ic[which(ic\_LDATA$alt=="ec")]\*0.9  
m4=mlogit(depvar~ic+oc+0 | income+agehed+rooms+region+0,data=ic\_LDATA)  
  
# (ii)  
oc\_LDATA = LDATA  
oc\_LDATA$oc[which(oc\_LDATA$alt=="ec")] = oc\_LDATA$oc[which(oc\_LDATA$alt=="ec")]\*0.9  
m5=mlogit(depvar~ic+oc+0 | income+agehed++rooms+region+0,data=oc\_LDATA)  
  
choice.vec=DATA$alt[DATA$depvar==TRUE]  
N=length(choice.vec)  
Ob.CP=table(choice.vec)/N  
  
Pr.CP3=colMeans(fitted.values(m3,outcome=FALSE))  
Pr.CP4=colMeans(fitted.values(m4,outcome=FALSE))  
Pr.CP5=colMeans(fitted.values(m5,outcome=FALSE))  
  
Ob.CP;Pr.CP3; Pr.CP4;Pr.CP5

## choice.vec  
## ec er gc gr hp   
## 0.07111111 0.09333333 0.63666667 0.14333333 0.05555556

## ec er gc gr hp   
## 0.07217299 0.09007339 0.63419790 0.14602854 0.05752718

## ec er gc gr hp   
## 0.07234831 0.08998725 0.63409103 0.14604767 0.05752574

## ec er gc gr hp   
## 0.07264012 0.08987375 0.63398648 0.14601441 0.05748524



# It seems that the method (ii), reduce the operational cost of ec by 10% is predicted to be more effective in promoting ec as having the highest predicted market share of 7.26%.

# Question 4a

rm(list=ls())  
  
M = 50  
J = 3  
N = 10000  
  
DATA=read.csv("product.csv")  
DATA["MS"]=DATA$sales/N  
  
market\_share = NULL  
for (i in 1:M){  
 market\_share = rbind(market\_share,c(1-sum(DATA$sales[which(DATA$market==i)]/N),DATA$sales[which(DATA$market==i)]/N))  
}  
  
market\_share = as.data.frame(market\_share)  
colnames(market\_share) = c("MS0","MS1","MS2","MS3")  
market\_share

## MS0 MS1 MS2 MS3  
## 1 0.0021 0.0921 0.2451 0.6607  
## 2 0.0025 0.0855 0.2447 0.6673  
## 3 0.0019 0.0934 0.2413 0.6634  
## 4 0.0012 0.0897 0.2471 0.6620  
## 5 0.0012 0.0862 0.2329 0.6797  
## 6 0.0019 0.0881 0.2461 0.6639  
## 7 0.0016 0.0895 0.2410 0.6679  
## 8 0.0023 0.0925 0.2447 0.6605  
## 9 0.0018 0.0921 0.2452 0.6609  
## 10 0.0019 0.0881 0.2453 0.6647  
## 11 0.0015 0.0938 0.2476 0.6571  
## 12 0.0010 0.0907 0.2446 0.6637  
## 13 0.0023 0.0862 0.2413 0.6702  
## 14 0.0018 0.0864 0.2452 0.6666  
## 15 0.0011 0.0878 0.2488 0.6623  
## 16 0.0019 0.0863 0.2474 0.6644  
## 17 0.0013 0.0886 0.2522 0.6579  
## 18 0.0018 0.0882 0.2417 0.6683  
## 19 0.0012 0.0895 0.2423 0.6670  
## 20 0.0020 0.0898 0.2480 0.6602  
## 21 0.0020 0.0983 0.2397 0.6600  
## 22 0.0012 0.0867 0.2477 0.6644  
## 23 0.0015 0.0898 0.2445 0.6642  
## 24 0.0021 0.0898 0.2451 0.6630  
## 25 0.0020 0.0906 0.2534 0.6540  
## 26 0.0012 0.0923 0.2388 0.6677  
## 27 0.0021 0.0892 0.2406 0.6681  
## 28 0.0017 0.0952 0.2469 0.6562  
## 29 0.0017 0.0907 0.2462 0.6614  
## 30 0.0015 0.0879 0.2473 0.6633  
## 31 0.0017 0.0915 0.2460 0.6608  
## 32 0.0018 0.0869 0.2500 0.6613  
## 33 0.0014 0.0914 0.2449 0.6623  
## 34 0.0017 0.0880 0.2443 0.6660  
## 35 0.0022 0.0849 0.2512 0.6617  
## 36 0.0011 0.0931 0.2415 0.6643  
## 37 0.0017 0.0902 0.2483 0.6598  
## 38 0.0014 0.0886 0.2411 0.6689  
## 39 0.0021 0.0871 0.2495 0.6613  
## 40 0.0014 0.0888 0.2404 0.6694  
## 41 0.0018 0.0902 0.2432 0.6648  
## 42 0.0013 0.0887 0.2482 0.6618  
## 43 0.0014 0.0870 0.2449 0.6667  
## 44 0.0017 0.0907 0.2455 0.6621  
## 45 0.0022 0.0897 0.2425 0.6656  
## 46 0.0018 0.0880 0.2477 0.6625  
## 47 0.0018 0.0889 0.2474 0.6619  
## 48 0.0018 0.0879 0.2494 0.6609  
## 49 0.0016 0.0865 0.2488 0.6631  
## 50 0.0019 0.0911 0.2422 0.6648

# Question 4b

b=rep(0.001,4)  
  
Q<-function(beta){  
 RSS\_matrix = NULL  
 X=as.matrix(DATA[which(DATA$market==1),][1:4])  
 for (i in 1:50){  
 Ob.MS=DATA$sales[which(DATA$market==i)]/N  
 uvec=X%\*%beta  
 Pr.MS=exp(uvec)/(1+sum(exp(uvec)))  
 d=Ob.MS-Pr.MS  
 RSS=(1/J)\*sum(d^2)  
 RSS\_matrix = c(RSS\_matrix, RSS)  
 }  
 return(sum(RSS\_matrix)/M)  
}  
  
initial=b  
nls=optim(initial,Q,hessian=T)  
betahat=nls$par  
betahat

## [1] 2.7374785 -1.1368645 3.7445506 -0.8673358