MA678 homework 05

Multinomial Regression

Weiling Li

Nov 18, 2019

Multinomial logit:

Using the individual-level survey data from the 2000 National Election Study (data in folder nes), predict party identification (which is on a 7-point scale) using ideology and demographics with an ordered multinomial logit model.

1. Summarize the parameter estimates numerically and also graphically.

```
#filter out cases where `partyid7` is NA
x=nes5200$partyid7
nes5200<-nes5200[!is.na(levels(x)[x]),]
#exclude apolitical to have an ordered outcome
nes5200<-subset(nes5200,partyid7!="apolitical")
nes5200$partyid7<-factor(nes5200$partyid7)
multi.log<-polr(partyid7~ideo+race+age_10,data=nes5200,Hess=TRUE)
```

```
## Warning in polr(partyid7 ~ ideo + race + age_10, data = nes5200, Hess =
## TRUE): design appears to be rank-deficient, so dropping some coefs
```

```
summary(multi.log)
```

```
## Call:
## polr(formula = partyid7 ~ ideo + race + age_10, data = nes5200,
##     Hess = TRUE)
##
## Coefficients:
##                                           Value Std. Error t value
## ideo1. liberal                          -1.6820   0.036170 -46.502
## ideo3. moderate ('middle of the road') -0.8847   0.038557 -22.945
## race2. black                            -1.7008   0.049578 -34.305
## race3. asian                             0.1335   0.119578   1.117
## race4. native american                  -0.3170   0.090396  -3.507
## race5. hispanic                         -0.8419   0.063198 -13.322
## race7. other                            -0.4121   0.404280  -1.019
## age_10                                  -0.1320   0.008827 -14.949
##
## Intercepts:
##                                                        Value    Std. Error
## 1. strong democrat|2. weak democrat                     -3.2756   0.0545
## 2. weak democrat|3. independent-democrat                -2.1541   0.0506
## 3. independent-democrat|4. independent-independent      -1.5343   0.0490
## 4. independent-independent|5. independent-republican  -1.1134   0.0483
## 5. independent-republican|6. weak republican            -0.4671   0.0477
## 6. weak republican|7. strong republican                  0.5794   0.0491
##                                                        t value
## 1. strong democrat|2. weak democrat                    -60.0839
## 2. weak democrat|3. independent-democrat               -42.6102
## 3. independent-democrat|4. independent-independent     -31.2845
```

```
## 4. independent-independent|5. independent-republican -23.0744
## 5. independent-republican|6. weak republican          -9.7970
## 6. weak republican|7. strong republican               11.7916
##
## Residual Deviance: 53114.60
## AIC: 53142.60
## (25245 observations deleted due to missingness)
```

```
#summary(nes_data_comp)
#modtest<-multinom(partyid7~ideo+race,family=logit,data=nes_data_comp)
#summary(modtest)
```

2. Explain the results from the fitted model.

```
#confint(multi.log)
```

3. Use a binned residual plot to assess the fit of the model.

```
residuals(multi.log)
```

```
## NULL
```

```
#binnedplot(predict(multi.log),resid(multi.log))
```

High School and Beyond

The hsb data was collected as a subset of the High School and Beyond study conducted by the National Education Longitudinal Studies program of the National Center for Education Statistics. The variables are gender; race; socioeconomic status; school type; chosen high school program type; scores on reading, writing, math, science, and social studies. We want to determine which factors are related to the choice of the type of program—academic, vocational, or general—that the students pursue in high school. The response is multinomial with three levels.

```
data(hsb)
?hsb
```

1. Fit a trinomial response model with the other relevant variables as predictors (untransformed).

```
mod2<-multinom(prog~gender+race+ses+schtyp+read+write+math+science+socst,hsb,trace=FALSE)
summary(mod2)
```

```
## Call:
## multinom(formula = prog ~ gender + race + ses + schtyp + read +
##     write + math + science + socst, data = hsb, trace = FALSE)
##
## Coefficients:
##          (Intercept)  gendermale raceasian racehispanic racewhite
## general     3.631901 -0.09264717  1.352739   -0.6322019 0.2965156
## vocation    7.481381 -0.32104341 -0.700070   -0.1993556 0.3358881
##               seslow sesmiddle schtyppublic        read       write
## general  1.09864111 0.7029621    0.5845405 -0.04418353 -0.03627381
## vocation 0.04747323 1.1815808    2.0553336 -0.03481202 -0.03166001
##                math    science       socst
## general  -0.1092888 0.10193746 -0.01976995
## vocation -0.1139877 0.05229938 -0.08040129
##
## Std. Errors:
```

```
##          (Intercept) gendermale raceasian racehispanic racewhite    seslow
## general     1.823452  0.4548778  1.058754    0.8935504 0.7354829 0.6066763
## vocation    2.104698  0.5021132  1.470176    0.8393676 0.7480573 0.7045772
##          sesmiddle schtyppublic       read      write       math
## general  0.5045938    0.5642925 0.03103707 0.03381324 0.03522441
## vocation 0.5700833    0.8348229 0.03422409 0.03585729 0.03885131
##             science      socst
## general  0.03274038 0.02712589
## vocation 0.03424763 0.02938212
##
## Residual Deviance: 305.8705
## AIC: 357.8705
```

2. For the student with id 99, compute the predicted probabilities of the three possible choices.

```
hsb[99,]
```

```
##    id gender     race ses schtyp     prog read write math science socst
## 99  1 female hispanic low public vocation   34    44   40      39    41
```

```
predict(mod2,newdata=hsb[99,],'prob')
```

```
##  academic   general  vocation
## 0.1939578 0.2830642 0.5229780
```

Happiness

Data were collected from 39 students in a University of Chicago MBA class and may be found in the dataset happy.

```
library(faraway)
data(happy)
?happy
```

1. Build a model for the level of happiness as a function of the other variables.

```
happy$happyF<-factor(happy$happy)
happy$sexF<-factor(happy$sex)
happy$loveF<-factor(happy$love)
happy$workF<-factor(happy$work)

#A proportional odds model:
model1<-polr(happyF~money+sexF+loveF+workF,happy)
summary(model1)
```

```
##
## Re-fitting to get Hessian
```

```
## Call:
## polr(formula = happyF ~ money + sexF + loveF + workF, data = happy)
##
## Coefficients:
##           Value Std. Error  t value
## money   0.01783    0.01087  1.64024
## sexF1  -1.02504    0.93629 -1.09479
## loveF2  3.45757    1.56121  2.21467
## loveF3  7.85036    1.85200  4.23885
```

```
## workF2 -1.18912    1.68765 -0.70460
## workF3  0.01574    1.58056  0.00996
## workF4  1.84630    1.53696  1.20127
## workF5  0.64794    2.14983  0.30139
##
## Intercepts:
##      Value   Std. Error t value
## 2|3  -0.8390  1.8387    -0.4563
## 3|4   0.0100  1.7713     0.0056
## 4|5   2.4280  2.0149     1.2050
## 5|6   4.4745  2.1063     2.1243
## 6|7   5.0675  2.1243     2.3856
## 7|8   7.3973  2.2303     3.3168
## 8|9  11.3105  2.5925     4.3628
## 9|10 13.0849  2.7916     4.6872
##
## Residual Deviance: 90.47841
## AIC: 122.4784
```

```r
c(deviance(model1),model1$edf)
```

```
## [1] 90.47841 16.00000
```

```r
#AIC-based variable selection method:
model2<-step(model1)
```

```
## Start:  AIC=122.48
## happyF ~ money + sexF + loveF + workF
##
##         Df    AIC
## - sexF   1 121.68
## <none>     122.48
## - money  1 123.31
## - workF  4 123.81
## - loveF  2 149.91
##
## Step:  AIC=121.68
## happyF ~ money + loveF + workF
##
##         Df    AIC
## <none>     121.68
## - money  1 122.22
## - workF  4 124.43
## - loveF  2 148.55
```

```r
summary(model2)
```

```
##
## Re-fitting to get Hessian

## Call:
## polr(formula = happyF ~ money + loveF + workF, data = happy)
##
## Coefficients:
##          Value Std. Error t value
## money   0.01658    0.01064  1.5581
## loveF2  3.73131    1.55726  2.3961
```

```
## loveF3  7.61619    1.81550  4.1951
## workF2 -1.35110    1.67099 -0.8086
## workF3  0.17262    1.57968  0.1093
## workF4  1.92916    1.53483  1.2569
## workF5  1.65934    1.93487  0.8576
##
## Intercepts:
##      Value   Std. Error t value
## 2|3   0.0407  1.6528     0.0247
## 3|4   0.9203  1.5695     0.5864
## 4|5   3.3895  1.8365     1.8456
## 5|6   5.2892  1.9862     2.6630
## 6|7   5.8706  2.0123     2.9174
## 7|8   8.1744  2.1391     3.8214
## 8|9  11.9678  2.5214     4.7465
## 9|10 13.7191  2.7378     5.0110
##
## Residual Deviance: 91.68405
## AIC: 121.684
```

```r
c(deviance(model2),model2$edf)
```

```
## [1] 91.68405 15.00000
```

```r
#comparison
anova(model1,model2)
```

```
## Likelihood ratio tests of ordinal regression models
##
## Response: happyF
##                            Model Resid. df Resid. Dev   Test    Df LR stat.
## 1        money + loveF + workF        24   91.68405
## 2 money + sexF + loveF + workF        23   90.47841 1 vs 2     1 1.205641
##     Pr(Chi)
## 1
## 2 0.2721972
```

```r
#An ordered probit model:
model3<-polr(happyF~money+sexF+loveF+workF,method="probit",happy)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
summary(model3)
```

```
##
## Re-fitting to get Hessian

## Call:
## polr(formula = happyF ~ money + sexF + loveF + workF, data = happy,
##     method = "probit")
##
## Coefficients:
##           Value Std. Error  t value
## money   0.01043    0.00591  1.76544
## sexF1  -0.56913    0.50399 -1.12926
## loveF2  1.90594    0.88476  2.15419
## loveF3  4.45221    0.99665  4.46717
## workF2 -0.89118    0.97323 -0.91569
```

```
## workF3 -0.03890    0.97362 -0.03995
## workF4  0.99973    0.93493  1.06932
## workF5  0.34129    1.26428  0.26995
##
## Intercepts:
##      Value   Std. Error t value
## 2|3  -0.5175  1.0317    -0.5016
## 3|4  -0.0812  1.0283    -0.0790
## 4|5   1.2424  1.1662     1.0654
## 5|6   2.4330  1.2188     1.9963
## 6|7   2.7864  1.2288     2.2676
## 7|8   4.1646  1.2779     3.2590
## 8|9   6.4243  1.4221     4.5173
## 9|10  7.3709  1.4709     5.0113
##
## Residual Deviance: 89.64616
## AIC: 121.6462
```

```
c(deviance(model3),model3$edf)
```

```
## [1] 89.64616 16.00000
```

```
#AIC-based variable selection method:
model4<-step(model3)
```

```
## Start:  AIC=121.65
## happyF ~ money + sexF + loveF + workF
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
##         Df    AIC
## - sexF   1 120.93
## <none>     121.65
## - money  1 122.81
## - workF  4 125.16
## - loveF  2 150.58
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
##
## Step:  AIC=120.93
## happyF ~ money + loveF + workF
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
##         Df    AIC
## <none>     120.93
## - money  1 121.43
## - workF  4 125.84
## - loveF  2 148.73
```

```
summary(model3)
```

```
##
## Re-fitting to get Hessian
```

```
## Call:
## polr(formula = happyF ~ money + sexF + loveF + workF, data = happy,
```

```
##     method = "probit")
##
## Coefficients:
##           Value Std. Error  t value
## money   0.01043    0.00591  1.76544
## sexF1  -0.56913    0.50399 -1.12926
## loveF2  1.90594    0.88476  2.15419
## loveF3  4.45221    0.99665  4.46717
## workF2 -0.89118    0.97323 -0.91569
## workF3 -0.03890    0.97362 -0.03995
## workF4  0.99973    0.93493  1.06932
## workF5  0.34129    1.26428  0.26995
##
## Intercepts:
##      Value   Std. Error t value
## 2|3  -0.5175  1.0317    -0.5016
## 3|4  -0.0812  1.0283    -0.0790
## 4|5   1.2424  1.1662     1.0654
## 5|6   2.4330  1.2188     1.9963
## 6|7   2.7864  1.2288     2.2676
## 7|8   4.1646  1.2779     3.2590
## 8|9   6.4243  1.4221     4.5173
## 9|10  7.3709  1.4709     5.0113
##
## Residual Deviance: 89.64616
## AIC: 121.6462
```

```
c(deviance(model4),model4$edf)
```

```
## [1] 90.93076 15.00000
```

```
#comparison:
anova(model3,model4)
```

```
## Likelihood ratio tests of ordinal regression models
##
## Response: happyF
##                         Model Resid. df Resid. Dev   Test    Df LR stat.
## 1        money + loveF + workF        24   90.93076
## 2 money + sexF + loveF + workF        23   89.64616 1 vs 2     1 1.284597
##    Pr(Chi)
## 1
## 2 0.257046
```

2. Interpret the parameters of your chosen model.

The interpretation are done using the proportional odds model with the covariates money, love and work. The chosen model is created so that the default level is money=0, love=1,work=1, corresponding to a person that has no annual family income, is lonely and has no job. The log-odds for this default person to be happyniess category 2 or smaller against 3 or higher is 0.0389, hence the odds is exp(0.0389)=1.04. The coefficients in the output corresponds to the beta, and can be interpreted in the following way. If the income is increased by one unit ($1000) the odds of moving from a given happiness category to one category higher increase by a factor of exp(0.01657)=1.0167. This is equivalent as to say thay standing in happiness category 2, the log-odds for being in that category or lower will be smaller if the money-variable is increased with 3 units.

3. Predict the happiness distribution for subject whose parents earn $30,000 a year, who is lonely, not sexually active and has no job.

```
#predict with the proportional odds model:
round(predict(model2,data.frame(money=30,sexF="0",loveF="1",workF="1"),type="probs"),3)
```

```
##     2     3     4     5     6     7     8     9    10
## 0.388 0.216 0.343 0.044 0.004 0.004 0.000 0.000 0.000
```

```
#check the predictive performance for the proportional odds model:
skattningar1<-predict(model2)
table(skattningar1,happy$happy)
```

```
##
## skattningar1  2  3  4  5  6  7  8  9 10
##           2   0  0  1  0  0  0  0  0  0
##           3   0  0  0  0  0  0  0  0  0
##           4   1  1  2  1  0  0  0  0  0
##           5   0  0  1  2  1  2  0  0  0
##           6   0  0  0  0  0  0  0  0  0
##           7   0  0  0  2  0  3  1  0  0
##           8   0  0  0  0  1  3 12  2  1
##           9   0  0  0  0  0  0  1  1  0
##           10  0  0  0  0  0  0  0  0  0
```

```
#predict with the ordered probit model:
round(predict(model4,data.frame(money=30,sexF="0",loveF="1",workF="1"),type="probs"),3)
```

```
##     2     3     4     5     6     7     8     9    10
## 0.358 0.189 0.386 0.062 0.003 0.001 0.000 0.000 0.000
```

```
#check the predictive performance for the ordered probit model:
skattningar2<-predict(model4)
table(skattningar2,happy$happy)
```

```
##
## skattningar2  2  3  4  5  6  7  8  9 10
##           2   0  0  1  0  0  0  0  0  0
##           3   0  0  0  0  0  0  0  0  0
##           4   1  1  2  1  0  0  0  0  0
##           5   0  0  1  2  1  2  0  0  0
##           6   0  0  0  0  0  0  0  0  0
##           7   0  0  0  2  1  3  1  0  0
##           8   0  0  0  0  0  3 13  3  1
##           9   0  0  0  0  0  0  0  0  0
##           10  0  0  0  0  0  0  0  0  0
```

newspaper survey on Vietnam War

A student newspaper conducted a survey of student opinions about the Vietnam War in May 1967. Responses were classified by sex, year in the program and one of four opinions. The survey was voluntary. The data may be found in the dataset `uncviet`. Treat the opinion as the response and the sex and year as predictors. Build a proportional odds model, giving an interpretation to the estimates.

```
data(uncviet)
?uncviet
modelfit<-polr(policy~sex+year,uncviet)
summary(modelfit)
```

```
##
## Re-fitting to get Hessian
## Call:
## polr(formula = policy ~ sex + year, data = uncviet)
##
## Coefficients:
##                  Value Std. Error    t value
## sexMale     -7.183e-16     0.5657 -1.270e-15
## yearGrad     5.802e-16     0.8944  6.487e-16
## yearJunior   4.442e-16     0.8944  4.966e-16
## yearSenior   8.773e-16     0.8944  9.808e-16
## yearSoph    -6.661e-16     0.8944 -7.448e-16
##
## Intercepts:
##     Value   Std. Error t value
## A|B -1.0986  0.7303    -1.5043
## B|C  0.0000  0.7071     0.0000
## C|D  1.0986  0.7303     1.5043
##
## Residual Deviance: 110.9035
## AIC: 126.9035
```

Taking the level of political opinion as outcome, sex and year as predictors. If the tested person is a male, the odds ratio of political opinion would decrease logit inverse (-7.183e-16). If the tested student is a graduate student, the odds ratio of political opinion would increase logit inverse (5.902e-16). Same for the other three year related coefficients.

pneumonoconiosis of coal miners

The pneumo data gives the number of coal miners classified by radiological examination into one of three categories of pneumonoconiosis and by the number of years spent working at the coal face divided into eight categories.

```r
library(faraway)
data(pneumo,package="faraway")
?pneumo
```

```
## Help on topic 'pneumo' was found in the following packages:
##
##   Package               Library
##   VGAM                  /Library/Frameworks/R.framework/Versions/3.6/Resources/library
##   faraway               /Library/Frameworks/R.framework/Versions/3.6/Resources/library
##
##
## Using the first match ...
```

1. Treating the pneumonoconiosis status as response variable as nominal, build a model for predicting the frequency of the three outcomes in terms of length of service and use it to predict the outcome for a miner with 25 years of service.

2. Repeat the analysis with the pneumonoconiosis status being treated as ordinal.

3.Now treat the response variable as hierarchical with top level indicating whether the miner has the disease and the second level indicating, given they have the disease, whether they have a moderate or severe case.

4. Compare the three analyses.

(optional) Multinomial choice models:

Pardoe and Simonton (2006) fit a discrete choice model to predict winners of the Academy Awards. Their data are in the folder academy.awards.

name	description
No	unique nominee identifier
Year	movie release year (not ceremony year)
Comp	identifier for year/category
Name	short nominee name
PP	best picture indicator
DD	best director indicator
MM	lead actor indicator
FF	lead actress indicator
Ch	1 if win, 2 if lose
Movie	short movie name
Nom	total oscar nominations
Pic	picture nom
Dir	director nom
Aml	actor male lead nom
Afl	actor female lead nom
Ams	actor male supporting nom
Afs	actor female supporting nom
Scr	screenplay nom
Cin	cinematography nom
Art	art direction nom
Cos	costume nom
Sco	score nom
Son	song nom
Edi	editing nom
Sou	sound mixing nom
For	foreign nom
Anf	animated feature nom
Eff	sound editing/visual effects nom
Mak	makeup nom
Dan	dance nom
AD	assistant director nom
PrNl	previous lead actor nominations
PrWl	previous lead actor wins
PrNs	previous supporting actor nominations
PrWs	previous supporting actor wins
PrN	total previous actor/director nominations
PrW	total previous actor/director wins
Gdr	golden globe drama win
Gmc	golden globe musical/comedy win
Gd	golden globe director win
Gm1	golden globe male lead actor drama win
Gm2	golden globe male lead actor musical/comedy win
Gf1	golden globe female lead actor drama win
Gf2	golden globe female lead actor musical/comedy win
PGA	producer's guild of america win
DGA	director's guild of america win
SAM	screen actor's guild male win
SAF	screen actor's guild female win

name	description
PN	PP*Nom
PD	PP*Dir
DN	DD*Nom
DP	DD*Pic
DPrN	DD*PrN
DPrW	DD*PrW
MN	MM*Nom
MP	MM*Pic
MPrN	MM*PrNl
MPrW	MM*PrWl
FN	FF*Nom
FP	FF*Pic
FPrN	FF*PrNl
FPrW	FF*PrWl

1. Fit your own model to these data.
2. Display the fitted model on a plot that also shows the data.
3. Make a plot displaying the uncertainty in inferences from the fitted model.