Categorical Data Analysis for Social Scientists: Labs

Brendan Halpin, Dept of Sociology, University of Limerick

June 20-21 2016

Contents

1 Lab 1: Monday morning

1.1 Intro to Stata

- Stata is in the Start menu / Programs / Specialist Software / Stata 12
- Text commands are entered in the command window
- Results show up in the main window
- Variables and the command history are on the left
- Enter this in the command window to load a data file (small BHPS extract, 2008):

```
use http://teaching.sociology.ul.ie/categorical/labs/example1
```

- Note the log window, and the command review
- Variables are shown in the variables window (click to put name in command window)

1.2 Basic operations

- Summarise variables

```
tab rjbstat
tab rjbstat rsex
tab rjbstat rsex, nol
su rfimn
su rfimn if rfimn>0
```

- Mark user-defined missing values (-9 thru -1) as missing

```
mvdecode r*, mv(-9/-1)
su rfimn
```

1.3 Create new variables

```
gen dob = rdoby + (rdobm - 1)/12
gen age = 2008 + (10-1)/12 - dob
recode rqfedhi 1/4=1 5/6=2 7/11=3 12/13=4, gen(educ)
label define educ4 1 "3rd level" 2 "Complete second plus" 3 "Incomplete second" 4 "Low/No"
label values educ educ4
```

1.4 Draw graphs

```
graph hbar, over(educ)
graph hbar (mean) rfimn, over(educ)
graph hbar (mean) rfimn if inrange(age,30,50), over(educ)
scatter rfimn age
lowess rfimn age if rfimn<=10000
```

1.5 Save data

- To save your data, assuming `f:\catdat\` exists as a folder

```
save f:/catdat/examplefile, replace
```

- Note that Stata prefers forward slashes so `f:\catdat\` becomes `f:/catdat/`
- The `replace` option is needed to over-write any existing file of that name

1.6 Do-file editor

- Entering commands one by one can be tedious
- You can enter and run blocks of commands in the "do-file editor"; click on this icon:

- You can run blocks of commands by highlighting them and clicking on this icon

- You can save and re-run files of commands using the do-file editor
- It also allows comments and multi-line commands
- You should use the do-file editor, for anything but very small tasks.

1.7 Stata add-ons

- Many user-written add-ons are available for Stata: from helpful utilities to state-of-the-art techniques
- Easy to install
- `tab_chi` is one

```
ssc install tab_chi
```

- This creates a new command, `tabchi`

```
help tabchi
tabchi educ rsex
tabchi educ rsex, pearson adjust
```

1.8 Tabular data

- The `tab_chi` package gives the `tabchi` command, which can generate tables with percentages, expected values, raw residuals, Pearson residuals, and adjusted residuals
- See if you can make sense of the help, and tabulate `rjbstat` by `educ` with expected, raw residual and adjusted residuals
- What sense can you make of those figures?

1.9 Creating tables with `input r c n`

- Categorical data is tabular; tables become data sets
- We can take a table off the page and put it in a Stata data set

```
                  |               qual
            class |      Univ  2nd level  Incomplet |     Total
------------------+---------------------------------+----------
         Prof/Man |      1025       1566        767 |      3358
Routine non-manual |       124        687        713 |      1524
    Skilled manual |        31        483        464 |       978
    Semi/unskilled |        18        361        716 |      1095
------------------+---------------------------------+----------
            Total |      1198       3097       2660 |      6955

Source: British Household Panel Survey 2001
```

- In this table there are 4*3=12 cells; we can treat it as a data set with 12 different combinations of variable values, and a weight:

```
clear // to discard existing data, if any
input class qual n
 1 1 1025
 1 2 1566
```

```
 1 3 767
 2 1 124
 2 2 687
 2 3 713
 3 1 31
 3 2 483
 3 3 464
 4 1 18
 4 2 361
 4 3 716
end

tab class qual
tab class qual [freq=n]
```

- Most Stata commands accept a frequency weight subcommand
- Alternatively we can `expand` the data set: turn a case with weight `n` into `n` identical cases

```
tab class qual
count // show how many cases there are
expand n
drop if n==0 // not necessary in this case, but if n==0 we don't want the case
tab class qual
count
```

1.10 Labelling data

```
label define quallab 1 "Univ" 2 "2nd level" 3 "Incomplete"
label define classlab 1 "Prof/Man" ///
                      2 "Routine non-manual" ///
                      3 "Skilled manual" ///
                      4 "Semi/unskilled"
label values qual quallab
label values class classlab
```

#** Calculating ORs

1.11 Logistic regression: compare LPM with BRM

- In a clear Stata session, enter to following command to load a data set relating to O-ring failure and temperature in Space Shuttle launches:

```
use http://teaching.sociology.ul.ie/categorical/labs/shuttledata
```

- We are interested in the relationship between temperature and the chance of failure
- What does `ttest temp, by(fail)` tell you?

- Fit and interpret a linear regression with failure "explained" by temperature

```
reg fail temp
predict lpm
scatter lpm fail temp
```

- Examine the residuals too:

```
predict lpmres, res
scatter lpmres temp
histogram lpmres, normal
```

- The sample is very small but it certainly looks like the residuals are odd, and the predictions include impossible values.
- Try the logistic regression instead:

```
logit fail temp
```

- Calculate a few predicted values by hand, using observed values for temperature:

$p = \frac{e^{a+bx}}{1+e^{a+bx}}$

- In many ways that formula is the hardest part of logistic regression! Do it the easy way and check your results:

```
predict lgp
```

- Plot the LPM and Logit results to compare:

```
scatter lpm lgp fail temp
```

1.12 Drawing functions and accessing parameter estimates

- We can access parameter estimates, and draw functions with Stata

```
logit // re-display the most recent results
display _b[temp] // The _b[] vector gives us access to the parameter estimates
twoway function ///
   exp(_b[_cons] + x*_b[temp])/(1+exp(_b[_cons] + x*_b[temp])) ///
   , range(temp) ///
 || scatter lgp lpm fail temp
```

- This should make the sigmoid curve all the clearer
- Note that because the parameter for `temp` is negative, the curve slopes down

2 Lab 2: Monday afternoon

2.1 Logistic with more explanatory variables

2.1.1 Housing tenure data

- Load the following file which contains information on housing tenure and

related details:

```
use http://teaching.sociology.ul.ie/categorical/labs/hten
```

- Examine the contents, and create a binary variable indicating home ownership, e.g.

```
recode tenure 1/2=1 3/99=0, gen(owner)
```

2.2 Hypothesis testing

- We can examine the implications in terms of signficance of parameter estimats

2.2.1 Wald test

- The Wald test ($\frac{\hat{\beta}}{SE} \sim \mathcal{N}(0,1)$ or $(\frac{\hat{\beta}}{SE})^2 \sim \chi^2$) tests each parameter estimate individually against the null of no effect

```
. logit owner age

Iteration 0:   log likelihood = -8462.2668
Iteration 1:   log likelihood = -8396.1105
Iteration 2:   log likelihood = -8395.8593
Iteration 3:   log likelihood = -8395.8593

Logistic regression                               Number of obs   =      15499
                                                  LR chi2(1)      =     132.81
                                                  Prob > chi2     =     0.0000
Log likelihood = -8395.8593                       Pseudo R2       =     0.0078

------------------------------------------------------------------------------
       owner |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |   .0119102   .0010456    11.39   0.000     .0098608    .0139596
       _cons |   .6421907   .0495884    12.95   0.000     .5449992    .7393822
------------------------------------------------------------------------------
```

- Here the PE for age is clearly significant

2.2.2 LR for 1 variable

- The likelihood ratio test comparing the model with versus without that parameter tests the same hypothesis and is likely more stable with small data sets:

```
logit owner // The null model with no explanatory variables
estimates store null
logit owner age
lrtest age
```

- Note this code will not work, for a good reason: age has a missing value so the data set changes between the two models
- Try this approach instead

```
logit owner age
estimates store model1
logit owner if e(sample)
lrtest model1
```

- The `e(sample)` is temporary information created by the previous logistic regression, indicating which cases were use to fit it. This strategy means the two models are fitted on the same data.

2.2.3 LR for multiple variables

- The advantage for the LR test for single variables is greater in small data sets, but it is very important where we add groups of variables, either blocks that go together for theoretical reasons, or as groups of dummies indicating categorical variables

```
logit owner age i.sex i.mastat
est store model2
logit owner age i.sex if e(sample)
...
lrtest model2
```

```
Likelihood-ratio test                                 LR chi2(6)  =    928.68
(Assumption: . nested in model2)                      Prob > chi2 =    0.0000
```

- The LR χ^2 attributable to adding marital status to the model is 928.68. Since `mastat` has 7 values, this requires 6 parameter estimates, hence 6 DF – this is a very strong effect

2.2.4 Model search

- Armed with these tools, search for a persuasive model to predict house-ownership

2.3 Grouped data example

- Where there are a limited number of "settings" of values of variables logistic regression can operate on the data as grouped or tabular
- For large datasets, the grouped format can be concise and quick to work with
- Take this dataset, `http://teaching.sociology.ul.ie/categorical/labs/groupable` (ESS data on spouse-pair education, 4 levels for the woman, binary university/other for the man, and cohort) and examine it, fitting `logit manu i.wom4 i.cohort`
- You can group it by using the `collapse` command:

```
gen n = 1
collapse (sum) n, by(cohort wom4 manu)
```

- Examine how that compares with the original format
- I have prepared a slightly different grouped version at `http://teaching.sociology.ul.ie/categorical/labs/grouped.dta` which has one line per "setting", plus the number of yeses, noes and total
- This structure is suited for "block logit":

```
blogit nyes total i.wom4 i.cohort
```

- This is explicitly modelling the binomial distribution of `nyes` outcomes out of `total` trials
- The second big advantage of grouped or block logit is that you can compare predicted and observed values for each block

```
predict n1
```

- This predicts the number of cases: compare it with the observed number of cases
- Test this model against the saturated model, which has DF=0 as it has as many parameters as settings. Do a LR test

```
blogit nyes total i.wom4##i.cohort
```

- Compare the predicted and observed values: what do you notice?
- If you fit these two models on the raw data you will get the same result, including for the LR test, but not the predicted values (and it will be slower, but not enough to matter much)

2.4 Fit

2.4.1 estat gof

- For individual data, we can't compare predicted and observed values in quite the same way. `estat gof` generates the Pearson goodness of fit statistic, and `estat gof, group(10)` the Hosmer-Lemeshow test: both are analagous to the LR test against the saturated model for grouped data, but are far from perfect.
- Use the `example1` data set as you have changed and saved it, or the tenure data to experiment with this:

```
gen univ = educ==1
logit univ age
estat gof, group(10)
```

2.4.2 fitstat

- Long and Freese's book comes with a suite of add-ons called `SPost`

```
ssc install spost
```

- This includes the `fitstat` command which outputs a large range of fit-related statistics: examine these for a few of the models you have fitted.

2.4.3 estat class and lroc

- Also explore the classification table: `estat class` for a few models
- Examine the ROC curves for a few models using `lroc`

3 Lab 3: Tuesday morning

3.1 Marginal effects

3.1.1 Graphing marginal effects

- Consider a linear regression with a non-linear RHS:

```
use http://teaching.sociology.ul.ie/categorical/labs/example1
keep if rdoby>0
gen age = 2008 -rdoby
reg rfimn c.age##c.age
predict pi
scatter pi age, msize(0.2)
```

- The fitted effect of age is nonlinear, rising to a peak in the late 40s, then falling
- The marginal effect of age is (by differentiation) $\beta_1 + 2\beta_2 \times x$

- Find the marginal effect of age at its mean: this is the slope of the curve at the mean value of age. Is it a useful summary?
- Now find the average marginal effect of age at each observed value of age and summarise it

```
gen marg = _b[age] + 2*age*_b[c.age#c.age]
su marg
```

- Compare your results with the official `margins` command:

```
margins, dydx(age)
```

3.1.2 Margins with logistic regression

- Logistic regression is non-linear in the probability
- The marginal effect is the slope of the tangent: $\beta p(1-p)$

```
use http://teaching.sociology.ul.ie/categorical/labs/shuttledata, clear
logit fail temp
predict p
gen slope = _b[temp]*p*(1-p)
su slope
margins, dydx(temp)
```

3.1.3 Example with multiple variables

- Things are a little more complex with multiple variables

```
use http://teaching.sociology.ul.ie/categorical/labs/hten
recode tenure 1/2=1 3/99=0, gen(owner)
logit owner i.sex age
predict p //
```

3.2 MNL

3.2.1 Fit and interpret

Load the data at `http://teaching.sociology.ul.ie/so5032/bhpsqual.dta` and examine the variables. The variable `vote` has four categories. Examine bivariate relationships between vote and some of the other variables, and then search for a multinomial logistic regession that makes sense:

```
mlogit vote ...
```

- It is better to have a base category that is easily interpretable, and the default here will use the biggest category ("Other/nationalist"). To avoid this, use the `baseoutcome(1)` option, which will force category one as the base.

3.2.2 Inference

- Use the likelihood-ratio test for each variable, since there are three times as many parameters as usual

3.2.3 Calculate predictions

- Use `predict` to generate predicted values (note you have to supply one variable name to hold the prediction for each category). How often is the most probable predicted category the same as the observed one?

```
predict v1-v4
```

3.3 Ordinal

3.3.1 Start with `mlogit`

- Fit a multinomial logit with `qual` as the dependent variable

```
mlogit qual c.age##c.age i.sex pahgs
```

- Note any patterns in the predictors: do you think the relationship between them and `qual` suggests ordinality

3.3.2 slogit

- Fit a stereotype logit:

```
slogit qual c.age##c.age i.sex pahgs
```

- This constrains the `mlogit` as if the dependent variable could be placed on a single continuum (ideally but not necessarily in its "natural" order)
- How well does it work?

3.3.3 ologit

- We can also try ordered logistic regression:

```
ologit qual c.age##c.age i.sex pahgs
```

- How do the results compare?
- Are you happy with the assumption of proportional odds?
- We can test this with the Brant test (part of `spostado`). However, the `brant` command is a little old and doesn't work with the latest syntax. We need an older format of the model specification:

```
gen age2 = age*age
xi: ologit qual age age2 i.sex pahgs
brant
```

- If the Brant test is significant, try generalised ordinal logit

```
ssc install gologit2
xi: gologit2 qual age age2 i.sex pahgs
```

- Compare the output with `mlogit` : it is equally imparsimonious but is structured differently

3.3.4 seqlogit

- `seqlogit` is also available from SSC:

```
ssc install seqlogit
seqlogit qual c.age##c.age i.sex pahgs, tree(1 : 2 3 4 , 2 : 3 4 , 3 : 4 )
```

- You can constrain effects to be equal across contrasts:

```
constraint 1 [_3_4v2=_4v3]: 2.sex
seqlogit qual c.age##c.age i.sex pahgs, ///
 tree(1 : 2 3 4 , 2 : 3 4 , 3 : 4 ) ///
 constraint (1)
```

- You can test whether a constraint is allowable using the `lrtest` command

3.3.5 Comparing multinomial and ordinal models

- Fit each of the models again, and run `fitstat` after each one, noting the BIC statistics
- Which gives the highest value of BIC?

4 Lab 4: Tuesday afternoon

4.1 Count models

4.1.1 Poisson

4.1.2 nbreg

4.1.3 zero inflated

4.1.4 zero truncated

4.2 Alternative specific logit

4.3 EHA