INTERMEDIATE SOCIAL STATISTICS CLASSES

WEEK 5: ORDERED LOGIT AND PROBIT

Dr. Dave Armstrong (david.armstrong@politics.ox.ac.uk)
Dr. Michelle Jackson (michelle.jackson@politics.ox.ac.uk)

Objectives*:* Running and interpreting models for ordered data. Interpretation and presentation of results.

***STATA* Commands:** .

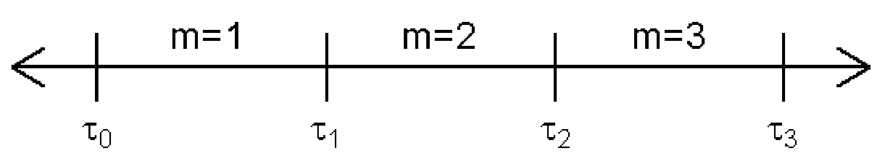
Dataset: British Election Study (BES), an extended version of last week's dataset: (**`besvoteplus.dta`**)

http://www.politics.ox.ac.uk/teaching/res_meths/reading_lists/intermstats/datasets.asp

---

1. ORDERED CHOICE MODELS

An ordinal variable is a variable for which the categories can be ranked from high to low; unlike interval variables, however, the distances between adjacent categories are unknown. Researchers often treat ordinal dependent variables as interval and use a linear regression model, but this is not always a sensible choice unless the thresholds are all about the same distances apart. When this is not the case, the linear regression model can give very misleading results. Moreover, when the dependent variable is ordinal, the errors are heteroscedastic and not normal, thus violating the assumptions of OLS. To deal with some of these problems, we use ordered logit or probit regression. The ordered models make what is called the 'Parallel Regression Assumption'. This suggests that most of the linear predictor is the same for each category. The only thing that changes is essentially the constant. We call these different constants 'cut-points' and for *m* categories in the dependent variable, we need to estimate *m-1* of these cut-points.



We only need to estimate the middle two cut-points here. We assume that $\tau_0 = -\infty$ and that $\tau_3 = \infty$. Both probit and logit have ordered models that again simply use different link functions to model the errors – standard logistic for the logit and standard normal for the probit. We will talk mostly about ordered logit here, but you should recognise that ordered probit does the same sort of thing. We use the following equation to obtain probabilities from ordered logit results:

$$\Pr(Y = m \mid X) = \frac{1}{1+\exp(-(\tau_m - x\beta))} - \frac{1}{1+\exp(-(\tau_{m-1} - x\beta))}$$

Note, that if we substitute -∞ for the first cut-point and ∞ for the last cut-point, that produces $1/1+\exp(-(\tau_m - x\beta))$ equal to 0 and 1, respectively.

With this little bit of theory out of the way, we can get down to interpreting some results.

The command to fit an ordered logit model in *STATA* is **`ologit.`** Let's look at an extended version of the dataset that we analysed in the previous class (uksmall.dta). We will run a simple model looking at satisfaction with democracy:

`.ologit demsat retnat trustgov`

```
Iteration 0:   log likelihood = -885.63502
Iteration 1:   log likelihood = -852.18385
Iteration 2:   log likelihood = -851.69555
Iteration 3:   log likelihood = -851.69488

Ordered logistic regression                       Number of obs   =        780
                                                  LR chi2(2)      =      67.88
                                                  Prob > chi2     =     0.0000
Log likelihood = -851.69488                       Pseudo R2       =     0.0383

------------------------------------------------------------------------------
      demsat |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      retnat |  -.3174962   .0918487    -3.46   0.001    -.4975164    -.137476
    trustgov |   .2300028   .0380836     6.04   0.000     .1553603    .3046452
-------------+----------------------------------------------------------------
       /cut1 |  -2.085778   .2982894                     -2.670415   -1.501142
       /cut2 |  -.5029909   .2853508                     -1.062268    .0562865
       /cut3 |   2.475154   .3038364                      1.879645    3.070662
------------------------------------------------------------------------------
```

The *demsat* variable has four categories, so we should see three cut-points estimated along with two coefficients. Note that we do not have an intercept because the intercept is not identified independent of the cut-points. We could choose to set one of the cut-points to zero and estimate a constant, but that is not how *STATA* does it.

The coefficients in the model are ordered log-odds (or logit) coefficients. Our model tells us that a one unit increase in the level of trust for government (*trustgov*) is associated with a 0.23 increase in the ordered log-odds of being in a higher *demsat* category, while holding economic perceptions (*retnat*) constant. Transforming to odds, we can say that the odds of being more satisfied with democracy are increased by 1.26 for each unit increase in the level of trust for government, holding economic perceptions constant.

As noted above, *STATA* sets the constant in the ordered logit model to 0 and estimates the cut-points for separating the different levels of *demsat*. **`/cut 1`** tells us the estimated cut-point on the latent variable used to differentiate *demsat*=1 from *demsat*=2/4 when values of the independent variables are evaluated at 0. **`/cut 2`** tells us the estimated cut-point on the latent variable used to differentiate *demsat*=1/2 from *demsat*=3/4 when values of the independent variables are evaluated at 0. **/cut 3** tells us the estimated cut-point on the latent variable used to differentiate *demsat*=1/3 from *demsat*=4 when values of the independent variables are evaluated at 0.

The *SPost* commands that we learned in the previous class work here, although now, we estimate the probability of being in each of the four categories, or the change in the probability of being in each category.

```
. prvalue, x(retnat 2 trustgov 4)

ologit: Predictions for demsat

Confidence intervals by delta method

                               95% Conf. Interval
     Pr(y=not_at_a|x): 0.0854  [ 0.0663,    0.1046]
     Pr(y=not_very|x): 0.2272  [ 0.1972,    0.2572]
     Pr(y=fairly_s|x): 0.5868  [ 0.5503,    0.6232]
     Pr(y=very_sat|x): 0.1006  [ 0.0797,    0.1215]

      retnat  trustgov
x=         2         4
```

This shows that when someone thinks that the economy stayed the same and had about the mean level of trust in government, the most likely category is ‘fairly satisfied’ at 0.587. The least likely category is ‘not at all satisfied’ at 0.085. Thus, if we had to guess, we would allocate this person to the third category. Note that it won’t always be the case that one of the categories has a probability of above 0.5. If this isn’t the case, our best guess would obviously be the category that had the highest probability, even though it is not higher than 0.5.

```
. prchange, rest(median)

ologit: Changes in Probabilities for demsat

retnat
            Avg|Chg|    not_at_a    not_very    fairly_s    very_sat
Min->Max   .06789253   .05005159   .08573347  -.07787329  -.05791175
   -+1/2   .03407033   .02486096   .04327971  -.03934854  -.02879209
  -+sd/2     .027607   .02012089   .03509311  -.03190899  -.02330501
MargEfct   .03411176    .0248057   .04341783  -.03948647  -.02873705

trustgov
            Avg|Chg|    not_at_a    not_very    fairly_s    very_sat
Min->Max   .18643029  -.13403197  -.23882861   .11813921   .25472137
   -+1/2   .02469573  -.01799094   -.0314005   .02855265   .02083883
  -+sd/2   .04957489  -.03631414  -.06283563   .05710822   .04204157
MargEfct   .02471148  -.01796991  -.03145305   .02860506   .02081789

          not_at_a   not_very   fairly_s   very_sat
Pr(y|x)  .08542685  .22716935  .58676392  .10063987

          retnat  trustgov
    x=         2         4
sd(x)=   .809956   2.01133
```

The results from **prchange** show that as economic perceptions become more favourable (note that higher values on *retnat* indicate less favourable perceptions) and as trust in government improves, people leave the bottom two categories in favour of the top two categories. This is not substantively surprising, but it's nice that it works this way. The row labelled 'Pr(y|x)' is the probability of each category given the x-values at 2 and 4 (medians) that were set by the 'rest(median)' argument to the *STATA* command.

Interpreting **prchange** output in detail:

Min->Max	The change in probability for *demsat* associated with changing x from its minimum to its maximum value
-+1/2	The change in probability for *demsat* associated with changing x from ½ unit below its mean value to ½ unit above its mean value
-+sd/2	The change in probability for *demsat* associated with changing x from ½ standard deviation below its mean value to ½ standard deviation above its mean value
MargEfct	Marginal effects are instantaneous rates of change, computed for an x variable while holding all other variables constant. They provide a good approximation for the amount of change in *demsat* that will be produced by a unit change in x. It is not appropriate to consider marginal effects for binary independent variables. We may contrast marginal effects with discrete effects (e.g. the change associated with a ½ unit change in x). Discrete effects and marginal effects need not be equal, but if the change in x occurs over a region of the probability curve that is approximately linear, the size of the two effects will be close. When changes in x are relatively small given the range of x, the marginal change and discrete change are likely to be similar, whereas when the range of change is relatively large, the marginal and discrete changes may be dissimilar.

MARGINAL EFFECTS: A DIGRESSION

The marginal effect is the partial first derivative of *y* with respect to *x*. To put it another way, it is the slope of the line tangent to the logit curve evaluated holding all variables at some value. The value at which we hold these variables basically provides us a starting point on the logit curve. Let's consider the binary model first. In the binary logit, the marginal effect is given by:

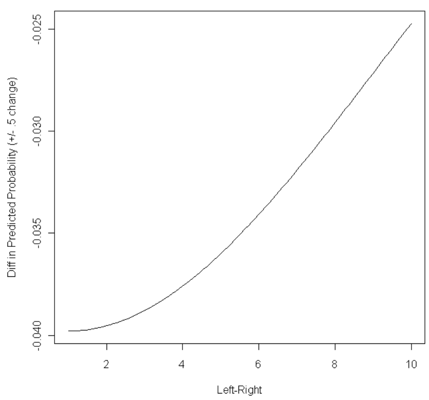
$$\frac{\partial \Pr(Y=1|X)}{\partial x_k} = \lambda(X\beta)\beta_k = \frac{\exp(X\beta)}{[1+\exp(X\beta)]^2}\beta_k$$

So, this is just the PDF of the logistic distribution evaluated at $X\beta$ multiplied by the coefficient β_k. This just says that the effect of x_k is going to depend on where you start on the logit curve.

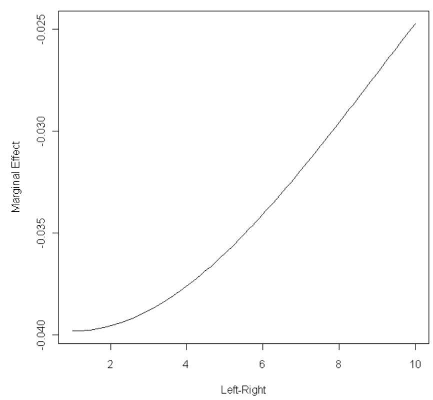
```
. xi: logit incumvote union swreg lrself urbrurl i.class i.retnat, nolog
i.class           _Iclass_1-4         (naturally coded; _Iclass_1 omitted)
i.retnat          _Iretnat_1-3        (naturally coded; _Iretnat_1 omitted)
```

```
Logistic regression                               Number of obs   =        607
                                                  LR chi2(9)      =      80.73
                                                  Prob > chi2     =     0.0000
Log likelihood = -294.50845                       Pseudo R2       =     0.1205

------------------------------------------------------------------------------
   incumvote |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       union |   .4525229   .2438244     1.86   0.063    -.0253641      .93041
       swreg |  -1.255679   .4923794    -2.55   0.011    -2.220725   -.2906328
      lrself |  -.1592278   .0455903    -3.49   0.000    -.2485831   -.0698725
     urbrurl |   .3844792   .2198014     1.75   0.080    -.0463236    .8152821
   _Iclass_2 |  -.3986732   .2741094    -1.45   0.146    -.9359177    .1385713
   _Iclass_3 |  -.6731731   .2331601    -2.89   0.004    -1.130158   -.2161876
   _Iclass_4 |  -2.688659   1.035281    -2.60   0.009    -4.717773   -.6595446
  _Iretnat_2 |  -.3581937   .2541471    -1.41   0.159    -.8563129    .1399255
  _Iretnat_3 |  -1.184982   .2595888    -4.56   0.000    -1.693767   -.6761976
       _cons |   .5561374   .3339491     1.67   0.096    -.0983907    1.210666
------------------------------------------------------------------------------
```

Let's look at the change in predicted probabilities for a +/- .5 unit change in left-right self placement versus the marginal effect of left-right self placement holding all other variables at their medians



Change in Predicted Probabilities



Marginal Effect

The marginal effect for ordered logit is a bit more difficult. It is:

$$\frac{\partial \Pr(Y = m|X)}{\partial x_k} = \beta_k[\lambda(\tau_{m-1} - X\beta) - \lambda(\tau_m - X\beta)]$$

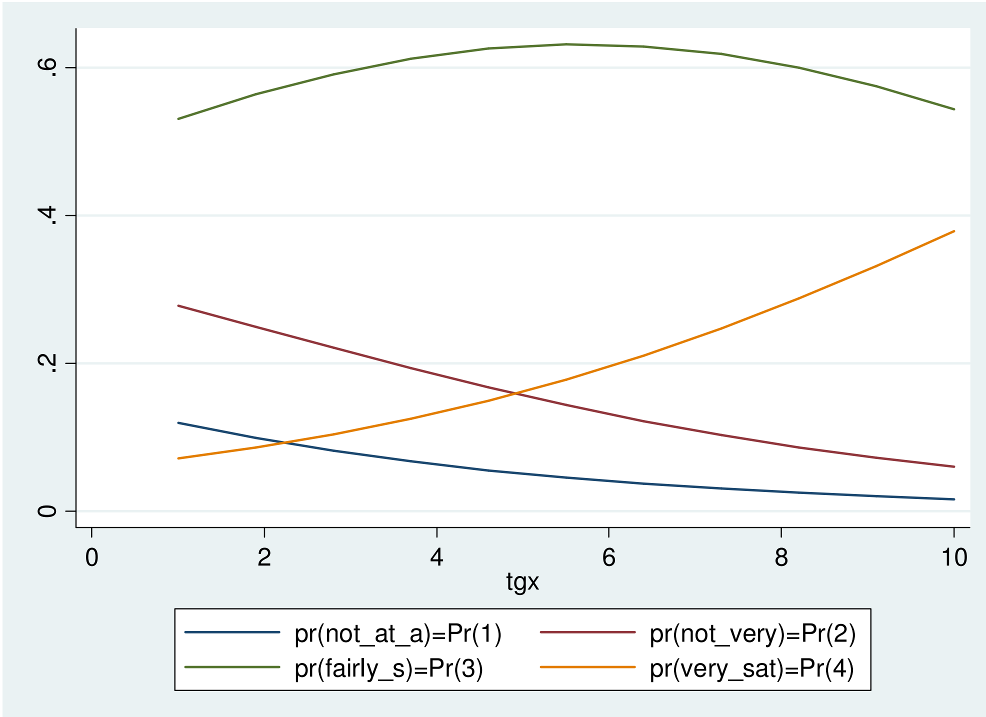
Given that λ is the PDF, the sum inside the brackets [ ] can be either positive or negative. So, the marginal effect on being in category *m* might be positive for some values of x_k and negative for others. It's a bit more difficult to think about this, but thinking about it in terms of predicted probabilities is a bit more intuitive and gives essentially the same information.

GRAPHING PREDICTED PROBABILITIES

Finally, you can also see how the probability of being in each category changes as a function of one of the continuous variables, holding the other variables constant, with the **prgen** command:

```
. prgen trustgov, gen(tg) ci rest(median)
```

```
. graph twoway line tgp1 tgp2 tgp3 tgp4 tgx
```



Notice that as a result of using the **ci** option in the **prgen** command, SPost generates upper and lower confidence bounds for the predicted probability lines. These could also be graphed, but it would make things considerably messier here. As an exercise for the interested reader, draw the confidence bounds around the 'fairly satisfied' line and see if any of the other lines are ever inside those bounds.

EXERCISE

Estimate a model with **demsat** as the dependent variable.

What is the predicted probability that a middle-class right-wing union member from an urban area in the southwest who owns a home has the highest level of trust in government and mid-level economic perceptions is fairly satisfied? What is the probability that a similar individual would be not at all satisfied?

2. Ordered Probit

As in the binary case, the choice of whether to use an ordered logit or probit will largely depend on which you feel more comfortable with. However, the ordered logit and ordered probit do differ in their assumptions about the distribution of errors. In an ordered logit model, errors have a logistic distribution with a mean of 0 and variance of $\pi^2/2$; in an ordered probit model, errors are distributed normally with a mean of 0 and variance of 1.

To run an ordered probit model:

.oprobit demsat retnat trustgov

```
Iteration 0:   log likelihood = -885.63502
Iteration 1:   log likelihood = -853.69549
Iteration 2:   log likelihood = -853.63779
Iteration 3:   log likelihood = -853.63778

Ordered probit regression                         Number of obs   =        780
                                                  LR chi2(2)      =      63.99
                                                  Prob > chi2     =     0.0000
Log likelihood = -853.63778                       Pseudo R2       =     0.0361

------------------------------------------------------------------------------
      demsat |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      retnat |  -.1964195   .0513751    -3.82   0.000    -.2971128   -.0957262
    trustgov |    .117262   .0209195     5.61   0.000     .0762606    .1582634
-------------+----------------------------------------------------------------
       /cut1 |  -1.292315   .1648977                     -1.615508   -.9691211
       /cut2 |  -.4022373   .1599246                     -.7156837   -.0887909
       /cut3 |   1.344969   .1653605                      1.020868     1.66907
------------------------------------------------------------------------------
```

As in the binary case, the coefficients estimated in an ordered logit model differ from those estimated in an ordered probit model by a factor of around 1.7. The probit coefficients are z-scores, and can be transformed into predicted probabilities using any table of the standard normal distribution.

A note on including categorical variables in a regression command in *STATA*. Remember that you can use the **xi** command as an alternative to creating the dummy variables yourself (the default reference category is the first category of your categorical variable). So, we could include *class* in this regression model as follows:

.xi:oprobit demsat retnat trustgov i.class

EPCP AND EPRE

So far, we have suggested a way to adjudicate between two nested models through a likelihood ratio test. However, we have not offered a method for adjudicating the fit of a model to the data. For this, we need to think of the concepts – percent correctly predicted and proportional reduction in error. Here, we are really comparing our model to the null model – a model without variables (only intercepts). This model will not "explain" anything. It will simply reproduce the marginal observed probabilities in the dataset. So, if in the democratic satisfaction variable, there are 30% of people who are very satisfied, in the null model, Pr(Y=Very Satisfied|X) = 0.3 for everyone.

The Percent correctly predicted simply tells us what percentage of observations we got right. The PCP measure is:

$$PCP = \frac{1}{N} \sum_{pr(Y=1)>0.5} y_i + \sum_{pr(Y=1)<0.5} (1 - y_i)$$

The null model will predict only the number of cases in the modal category right. In fact, the Null model will predict everyone in the modal category as that is the way that it gets the most possible cases right for sure.

The proportional reduction in error characterizes how much better we do than the Null model in percentage terms. Specifically, it calculates:

$$PRE = \frac{Errors_{NULL} - Errors_{FULL}}{Errors_{NULL}}$$

In a piece in *Political Analysis* (the journal of the Methodology section of the American Political Science Association), Michael Herron suggested another way of considering percent correctly prediction and proportional reduction in error. He proposed "expected" pcp and pre. For those using their own copies of Stata, you can download the routine from here: http://www.cnlawrence.com/data/epcp.zip. Here, rather than the category predictions, we sum the probabilities of being in the observed category.

$$ePCP = \sum_{y=m} \Pr\,(\widehat{y_i = m})$$

$$ePRE = \frac{\sum_{y_i=m} 1 - \Pr_{NULL}(y_i = m) - \sum_{y_i=m} 1 - \Pr_{FULL}(y_i = m)}{\sum_{y_i=m} 1 - \Pr_{NULL}(y_i = m) -}$$

Luckily, there is a Stata routine that calculates all of these quantities for you. It's called epcp.

```
Ordered logistic regression                       Number of obs   =        780
                                                  LR chi2(2)      =      67.88
                                                  Prob > chi2     =     0.0000
Log likelihood = -851.69488                       Pseudo R2       =     0.0383

------------------------------------------------------------------------------
      demsat |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      retnat |  -.3174962   .0918487    -3.46   0.001    -.4975164    -.137476
    trustgov |   .2300028   .0380836     6.04   0.000     .1553603    .3046452
-------------+----------------------------------------------------------------
       /cut1 |  -2.085778   .2982894                     -2.670415   -1.501142
       /cut2 |  -.5029909   .2853508                     -1.062268    .0562865
       /cut3 |   2.475154   .3038364                      1.879645    3.070662
------------------------------------------------------------------------------
```

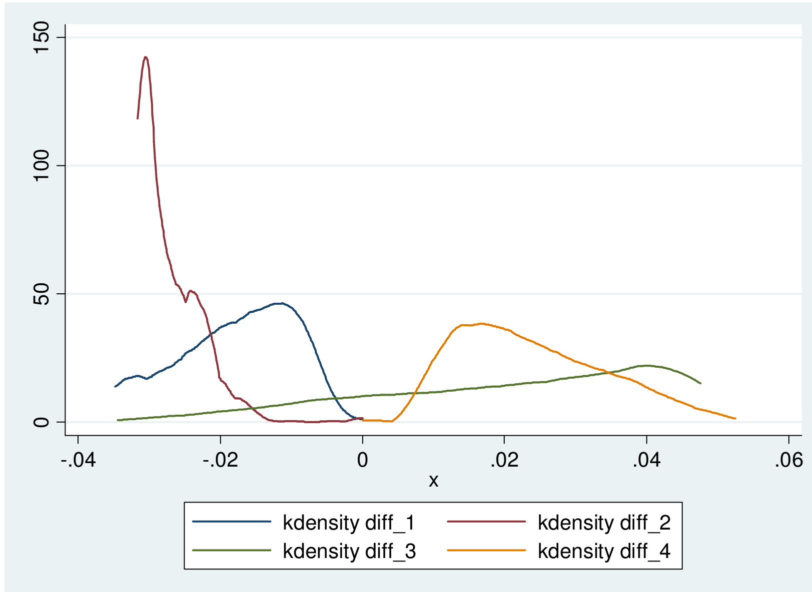
```
. epcp

Displaying classification results for ologit
Dependent variable: demsat

Classification table

 Predicted values of |           Actual values of demsat
              demsat | Not at al  Not very   Fairly sa  Very sati |     Total
---------------------+-------------------------------------------+----------
    Fairly satisfied |        76       179       437         88 |       780
---------------------+-------------------------------------------+----------
               Total |        76       179       437         88 |       780

    Percent Correctly Predected =  56.03%
      Percent in Modal Category =  56.03%
Proportional Reduction in Error =   0.00%

expected PCP =  40.62%
expected PMC =  38.88%
expected PRE =   2.86%
```

This shows that despite significant coefficients, the estimated model is not interestingly different from the null model. That is, they both explain about the same number of cases right.

SUBSTANTIVE EFFECTS: THE RAY WAY

We used SPost to get a sense of substantive effects, but Ray suggested another way in class. This way is a bit more difficult, but it actually may make a bit more sense. Specifically, it gets us out of the potential problem of making inferences about individuals that don't (and probably wouldn't) exist. This way requires generating two different predictions. You can use the following steps. Again, for this example I'll use the model with demsat as a function of retnat and trustgov. We'll consider the effects of a one-unit change in trustgov.

1. Generate predicted probabilities for each individual.
   `predict olprob1-olprob4, pr`
2. Rename the variable of interest, it's original name plus "_old"
   `rename trustgov trustgov_old`
3. Generate a new variable called trustgov that equals the values of the original trustgov variable (now trustgov_old) plus one
   `gen trustgov = trustgov_old + 1`
4. Fix values that got moved outside the original variable's range. We don't want to move extreme observations to points that are theoretically impossible.
   `replace trustgov = 10 if trustgov > 10 & trustgov < .`
5. Generate new predictions based on this new variable.
   `predict olprob2_1-olprob2_4, pr`
6. Generate new variables that are the difference between the first and second predicted probabilities for each category.
   `for num 1/4: gen diff_X = olprob2_X-olprobX`
7. Summarize the new diff variables and the means will give you a sense of what a likely change might be in predicted probabilities.
8. You could also make a density plot of the new diff variables to show you what the values are:
   `twoway (kdensity diff_1) || (kdensity diff_2) || (kdensity diff_3) || (kdensity diff_4)`



9. Drop the newly generated trustgov variable and rename trustgov_old to trustgov:
```
drop trustgov
rename trustgov_old trustgov
```

We also might wonder whether this one-unit change would change anyone's predicted category membership. I adapted the code from the epcp routine so it would generate predicted category memberships called. The new program is called "predcat2". You simply type predcat and then the name you want to give to the new predicted category variable. The whole process, then, would look something like this:

```
ologit demsat trustgov retnat
predict olprob1_1-olprob1_4, pr
predcat2 pcat_orig
epcp
rename trustgov trustgov_old
gen trustgov = trustgov_old + 1
replace trustgov = 10 if trustgov > 10 & trustgov < .
predict olprob2_1-olprob2_4, pr
predcat2 pcat_new
for num 1/4: gen diff_X = olprob2_X - olprob1_X
tab pcat_orig pcat_new
drop trustgov
rename trustgov_old trustgov
```

EXERCISE

Using the dataset from last class (uksmall.dta), estimate the binary logit model of vote for the incumb as a function of income, union, swreg, lrself urbrur1 class (as dummy variables) and retnat (as dummy variables).

Using the same logic as above, see what effect a one-unit (positive) change in lrself for everyone has in the average change in probabilities of voting for the incumbent. Would this change the predicted vote for anyone?

USEFUL REFERENCES

Long, J. Scott (1997) *Regression Models for Categorical and Limited Dependent Variables*. Thousand Oaks, CA: Sage Publications.

Long, J. S., and J. Freese (2006) *Regression Models for Categorical Dependent Variables Using Stata*. Stata Press.