

Lab Guide

Written by Trent Mize for ICPSRCDA14 [Last updated: 16 July 2017]

1. The Lab Guide is divided into sections corresponding to class lectures. Each section should be reviewed **before** starting the assignments
2. The Lab Guide makes every effort to follow assignments as they are written, but as you are ultimately responsible for completing all sections of your assignment, specifics of assignment questions should be double-checked.
3. The lab guide uses the data set *icpsr_scireview4.dta*. **These data cannot be used to complete assignments.**
4. Throughout, we provide a few examples of interpretation in the review sections. **These are in shaded boxes.**
5. Although the command window can be used for exploring new commands, **assignments should always be completed using do-files.** If you are not sure how to use a do-file see *Getting Started Using Stata* for help.

Contents

1. Models for Binary Outcomes: Part 1	p. 3
2. Models for Binary Outcomes: Part 2	p. 10
3. Testing & Assessing Fit	p. 17
4. Models for Ordinal Outcomes	p. 24
5. Models for Nominal Outcomes	p. 31
6. Models for Count Outcomes	p. 41
7. A Few Advanced Commands & Techniques	p. 57

Section 1: Models for Binary Outcomes: Part 1

For details about binary models and related Stata commands, see Chapter 4 of L&F (2005). The file `icpsrcda01-brm-p1.do` contains these Stata commands.

___1.1a) Set-up your do-file.

```
capture log close //closes any open log file.
log using icpsrcda01-brm-p1, replace text //opens a log file that
//records both commands
//and results.

// Program: icpsrcda01-brm-p1.do
// Task: Review 1 - Binary Part 1
// Project: ICPSR CDA
// Author: Trent Mize \ 2014-06-24

**program setup
version 14.0 //this command is for version control,
//ensuring that results can be replicated.
clear all //this command removes data, value labels, matrices,
//scalars, saved results, etc. from memory.
set linesize 80 //specifies screen size width & prevents wrapping
```

___1.1b) Load the Data.

```
usecda cda_scireview4
```

___1.1c) Examine the Data and Select Variables. First, describe the dataset to see a list of variables and some information about the dataset:

```
codebook, compact
```

Produces this output:

```
. codebook, compact
```

Variable	Obs	Unique	Mean	Min	Max	Label
id	264	264	58556.74	57001	62420	ID Number.
cit1	264	48	11.33333	0	130	Citations: PhD yr -1 to 1.
cit3	264	54	14.68561	0	196	Citations: PhD yr 1 to 3.
cit6	264	59	17.58712	0	143	Citations: PhD yr 4 to 6.

```
:: output deleted ::
```

```
jobprst 264 4 2.348485 1 4 Rankings of University Job.
```

Use `keep` to select the dependent variable `faculty` and the independent variables, `fellow`, `phd`, `mcit3`, and `mnas` (remember: you only need three!), which we use in the regression models later:

```
keep faculty fellow phd mcit3 mnas
```

___1.1d) Drop cases with missing data and verify. Use `misschk` to review the missing data. Then, use the variable you generated with the `gen(m)` option to `keep` only those observations that are not missing on any of your selected variables.

```
misschk,    gen(m)
tab        mnumber
keep if    mnumber==0
```

Produces this output:

```
. misschk, gen(m)
```

Variables examined for missing values

#	Variable	# Missing	% Missing
1	fellow	0	0.0
2	mcit3	0	0.0
3	mnas	0	0.0
4	phd	0	0.0
5	faculty	0	0.0

Warning: this output does not differentiate among extended missing. To generate patterns for extended missing, use `extmiss` option.

```
Missing for |
which |
variables? |      Freq.    Percent    Cum.
-----+-----
_____ |           264    100.00    100.00
-----+-----
Total |           264    100.00
```

```
Missing for |
how many |
variables? |      Freq.    Percent    Cum.
-----+-----
0 |           264    100.00    100.00
-----+-----
Total |           264    100.00
```

```
. tab mnumber
```

```
Missing for |
how many |
variables? |      Freq.    Percent    Cum.
-----+-----
0 |           264    100.00    100.00
-----+-----
Total |           264    100.00
```

```
. keep if mnumber==0
(0 observations deleted)
```

___1.2) Describe your data. Copy and paste the results from `codebook`, `compact` and `sum`. These commands:

```
codebook faculty fellow phd mcit3 mnas, compact
sum faculty fellow phd mcit3 mnas
```

Produce the following output:

```
. codebook faculty fellow phd mcit3 mnas, compact
```

```
Variable   Obs Unique      Mean   Min    Max   Label
-----
faculty    264      2   .5340909   0     1   Faculty in Univ? (1=yes)
fellow     264      2   .4128788   0     1   Postdoctoral fellow? (1=yes)
phd        264     79   3.181894   1   4.66   Prestige of Ph.D. department.
mcit3      264     59  20.71591   0    129   Mentor's 3 yr citation.
mnas       264      2   .0833333   0     1   Was mentor in NAS? (1=yes)
```

```
. sum faculty fellow phd mcit3 mnas
```

```
Variable |      Obs      Mean   Std. Dev.      Min      Max
-----+-----
faculty |      264   .5340909   .4997839         0         1
fellow  |      264   .4128788   .4932865         0         1
  phd   |      264   3.181894   1.00518         1         4.66
mcit3   |      264  20.71591  25.44536         0        129
mnas    |      264   .0833333   .2769103         0         1
```

___1.3) Binary logit model. For all regressions, the dependent variable is listed first. A probit model is run by simply changing `logit` to `probit`. This command:

```
logit faculty fellow phd mcit3 mnas
```

Produces the output on the following page:

```
. logit faculty fellow phd mcit3 mnas
```

```
Iteration 0: log likelihood = -182.37674
Iteration 1: log likelihood = -164.019
Iteration 2: log likelihood = -163.55936
Iteration 3: log likelihood = -163.55534
Iteration 4: log likelihood = -163.55534
```

```
Logistic regression                               Number of obs   =       264
                                                    LR chi2(4)      =       37.64
                                                    Prob > chi2     =       0.0000
Log likelihood = -163.55534                       Pseudo R2      =       0.1032
```

faculty	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
fellow	1.250155	.2767966	4.52	0.000	.7076434	1.792666
phd	-.0637186	.1471307	-0.43	0.665	-.3520894	.2246522
mcit3	.0206156	.0071255	2.89	0.004	.0066498	.0345814
mnas	.3639082	.5571229	0.65	0.514	-.7280327	1.455849
_cons	-.5806031	.4498847	-1.29	0.197	-1.462361	.3011547

1.4) Computing factor change coefficients. The factor change in the odds as well as the standardized factor change can be obtained with the command `listcoef`. Note that `listcoef` can also be run after estimating a probit model but odds ratios cannot be computed for this model. Instead standardized beta coefficients are listed.

```
listcoef , help
```

Produces this output:

```
. listcoef , help
```

```
logit (N=264): Factor Change in Odds
```

```
Odds of: 1_Yes vs 0_No
```

faculty	b	z	P> z	e^b	e^bStdX	SDofX
fellow	1.25015	4.517	0.000	3.4909	1.8528	0.4933
phd	-0.06372	-0.433	0.665	0.9383	0.9380	1.0052
mcit3	0.02062	2.893	0.004	1.0208	1.6897	25.4454
mnas	0.36391	0.653	0.514	1.4389	1.1060	0.2769

b = raw coefficient

z = z-score for test of b=0

P>|z| = p-value for z-test

e^b = exp(b) = factor change in odds for unit increase in X

e^bStdX = exp(b*SD of X) = change in odds for SD increase in X

SDofX = standard deviation of X

1.5 & 1.6) Interpreting factor change coefficients. The shaded box below gives example interpretations of factor change coefficients for a logit model. Remember that x-standardized coefficients are not appropriate for binary variables.

Unstandardized: Obtaining a post-doctoral fellowship increases the odds of gaining a faculty position by a factor of 3.5, holding other variables constant.

X-standardized: A standard deviation increase in mentor's citations (about 25) increases the odds of gaining a faculty position by a factor of 1.7.

1.7a) Compute predicted probabilities. Use `mtable` to produce the predicted probability of being a faculty member when someone had a post-doctoral fellowship (fellow = 1) and when they did not have a post-doctoral fellowship (fellow = 0). Use the `atmeans` option to hold the other control variables at their mean value.

```
. mtable, at(fellow=(0 1)) atmeans
```

```
Expression: Pr(faculty), predict()
```

	fellow	Pr(y)
1	0	0.419
2	1	0.716

```
Specified values of covariates
```

	phd	mcit3	mnas
Current	3.18	20.7	.0833

An individual who held a post-doctoral fellowship has a 0.72 probability of currently holding a faculty position. An individual who did not hold a post-doctoral fellowship has a 0.42 probability of currently holding a faculty position.

1.7b) Use predicted probabilities to compute factor change coefficient I. Use `display` to compute the factor change coefficient for `fellow` using the predicted probabilities calculated above. Compare with the factor change coefficient given by `listcoef`, above.

```
display ((0.7159/(1-0.7159)) / (0.4192/(1-0.4192)))
```

Produces this output:

```
. display ((0.7159/(1-0.7159)) / (0.4192/(1-0.4192)))  
3.4912943
```

This number is similar—though not identical—to the factor change coefficient for `fellow` given above ($e^b=3.4909$). **HINT:** The more precision used in the calculation, the closer these two numbers will be.

1.8-1.9) Compare the coefficients from logit and probit. Run a probit model using the same variables and store the results. While you will need to use Excel or to create the full table for this question, the `estimates store` and `estimates table` commands can store regression results and allow you to list probit estimation results side-by-side with logit estimation results. Note that the logit coefficients are around 1.7 times as large as the probit estimates. Why is this?

```
quietly logit faculty fellow phd mcit3 mnas // 'quietly' = no output shown
estimates store estlogit
probit faculty fellow phd mcit3 mnas
estimates store estprobit
estimates table estlogit estprobit , b(%9.3f) se
```

Produces this output:

```
. quietly logit faculty fellow phd mcit3 mnas // 'quietly' = no output shown
. estimates store estlogit
. probit faculty fellow phd mcit3 mnas

Iteration 0:   log likelihood = -182.37674
Iteration 1:   log likelihood = -163.98754
Iteration 2:   log likelihood = -163.73877
Iteration 3:   log likelihood = -163.73838
```

```
Probit regression                               Number of obs   =           264
                                                LR chi2(4)      =           37.28
                                                Prob > chi2     =           0.0000
Log likelihood = -163.73838                    Pseudo R2      =           0.1022
```

faculty	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
fellow	.763915	.1675687	4.56	0.000	.4354863	1.092344
phd	-.0392676	.0897914	-0.44	0.662	-.2152556	.1367203
mcit3	.0118642	.003994	2.97	0.003	.0040362	.0196922
mnas	.2299521	.3252353	0.71	0.480	-.4074975	.8674016
_cons	-.3450294	.2743016	-1.26	0.208	-.8826506	.1925919


```
. estimates store estprobit
. estimates table estlogit estprobit , b(%9.3f) se
```

Variable	estlogit	estprobit
fellow	1.250	0.764
	0.277	0.168
phd	-0.064	-0.039
	0.147	0.090
mcit3	0.021	0.012
	0.007	0.004
mnas	0.364	0.230
	0.557	0.325
_cons	-0.581	-0.345
	0.450	0.274

legend: b/se

___1.END) Close Log File and Exit Do File.

```
log close
exit
```

Section 2: Models for Binary Outcomes: Part 2

For details about binary models and related Stata commands, see Chapter 4 of L&F (2005). The file `icpsrcda02-brm-p2.do` contains these Stata commands.

___2.1a) Set-up your do-file.

```
capture log close
log using icpsrcda02-brm-p2, replace text
```

```
// program:    icpsrcda02-brm-p2.do
// task:       Review 2 - Binary-part 2
// project:    ICPSR CDA
// author:     Trent Mize \ 2014-06-24
```

```
*program setup
version      14.0
clear        all
set          linesize 80
```

___2.1b) Load the Data.

```
usecda      cda_scireview4
```

___2.1c) Re-estimate your binary logit model. Results should match your results in BRM-Part 1.

```
logit faculty i.fellow phd mcit3 i.mnas
```

Produces this output:

```
. logit faculty i.fellow phd mcit3 i.mnas
```

```
Iteration 0:  log likelihood = -182.37674
Iteration 1:  log likelihood = -163.76405
Iteration 2:  log likelihood = -163.55602
Iteration 3:  log likelihood = -163.55534
Iteration 4:  log likelihood = -163.55534
```

```
Logistic regression                                Number of obs   =           264
                                                    LR chi2(4)      =           37.64
                                                    Prob > chi2     =           0.0000
Log likelihood = -163.55534                        Pseudo R2       =           0.1032
```

faculty	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
fellow					
1_Yes	1.250155	.2767966	4.52	0.000	.7076434 1.792666
phd	-.0637186	.1471307	-0.43	0.665	-.3520894 .2246522
mcit3	.0206156	.0071255	2.89	0.004	.0066498 .0345814
mnas					
1_Yes	.3639082	.5571229	0.65	0.514	-.7280327 1.455849
_cons	-.5806031	.4498847	-1.29	0.197	-1.462361 .3011547

2.2) Predicted Probabilities. We can compute and plot predicted probabilities for our observed data. Here we pick the name `prlogit` for the new variable that contains predicted values. Note that a predicted value is calculated for each respondent in the sample. **NOTE:** The “predict” command can predict many things. If run immediately following a logit model, it calculates predicted probabilities by default. These commands:

```
predict    prlogit
label var  prlogit "Logit: Predicted Probability"
sum        prlogit
dotplot    prlogit

graph export icpsrcda02-binary-fig1.png , width(1200) replace
```

Produces this output:

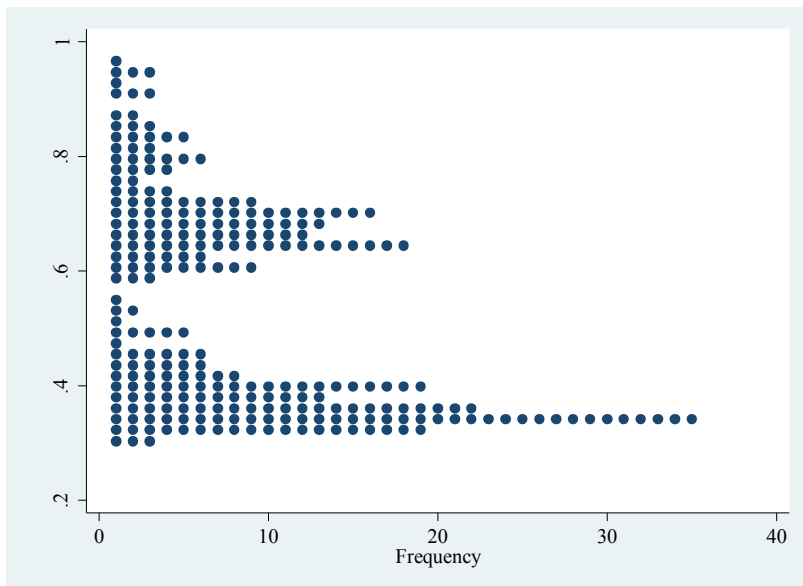
```
. predict prlogit
(option pr assumed; Pr(faculty))

. label var prlogit "Logit: Predicted Probability"

. sum prlogit
```

Variable	Obs	Mean	Std. Dev.	Min	Max
prlogit	264	.5340909	.1828654	.3035647	.9665072

```
. dotplot prlogit
```



```
. graph export icpsrcda02-binary-fig1.png , width(1200) replace
```

2.3) Discrete change. `mchange` computes the discrete change for all independent variables but does not calculate a confidence interval for the discrete change by default. Values for specific independent variables can be set using the `at()` option and all other control variables can be held at their means using the `atmeans` option. This command:

```
. mchange, atmeans centered
```

```
logit: Changes in Pr(y) | Number of obs = 264
```

```
Expression: Pr(faculty), predict(pr)
```

	Change	p-value
fellow		
1 Yes vs 0 No	0.297	0.000
phd		
+1 cntr	-0.016	0.665
+SD cntr	-0.016	0.665
Marginal	-0.016	0.665
mcit3		
+1 cntr	0.005	0.004
+SD cntr	0.129	0.003
Marginal	0.005	0.004
mnas		
1 Yes vs 0 No	0.088	0.500

```
Predictions at base value
```

	0_No	1_Yes
Pr(y base)	0.453	0.547

```
Base values of regressors
```

	1. fellow	phd	mcit3	1. mnas
at	.413	3.18	20.7	.0833

```
1: Estimates with margins option atmeans.
```

2.4) Discrete change with confidence interval. `mchange` reports a p-value by default. However, you can request other statistics including a 95% confidence interval by using the `stats()` option. You can also request the discrete changes calculated for only your variables of interest by specifying them in the `after` of the `mchange` command. **NOTE:** These are marginal effects at the mean because we are using the `atmeans` option.

```
. mchange fellow, atmeans stats(ci) centered
```

```
logit: Changes in Pr(y) | Number of obs = 264
```

```
Expression: Pr(faculty), predict(pr)
```

	Change	LL	UL
fellow			
1 Yes vs 0 No	0.297	0.178	0.416

```
Predictions at base value
```

	0_No	1_Yes
Pr(y base)	0.453	0.547

```
Base values of regressors
```

	1. fellow	phd	mcit3	1. mnas
at	.413	3.18	20.7	.0833

```
1: Estimates with margins option atmeans.
```

Interpretation: A scientist who receives a post-doctoral fellowship has a .30 higher probability of being faculty at a university than a scientist who does not receive a fellowship, holding other variables at their mean. This difference is significant (95% CI: 0.18, 0.42).

2.5) Discrete change for C + confidence interval. Note that `mchange` also produces a discrete change for a centered standard deviation increase (changing from ½ SD below the mean to ½ SD above the mean). **NOTE:** This would not be sensible to interpret for a binary variable (such as fellow), but is helpful for continuous variables such as mentor’s citations.

```
. mchange mcit3, atmeans stats(ci) centered
```

```
logit: Changes in Pr(y) | Number of obs = 264
```

```
Expression: Pr(faculty), predict(pr)
```

	Change	LL	UL
mcit3			
+1 cntr	0.005	0.002	0.009
+SD cntr	0.129	0.043	0.215
Marginal	0.005	0.002	0.009

```
Predictions at base value
```

	0_No	1_Yes
Pr(y base)	0.453	0.547

```
Base values of regressors
```

	fellow	phd	mcit3	mnas
at	.413	3.18	20.7	.0833

```
1: Estimates with margins option atmeans.
```

Interpretation: A standard deviation increase in the number of mentor’s citations (about 25 citations) centered around the mean increases the probability of obtaining a faculty position by .13, holding all other variables at their mean. This difference is significant (95% CI: 0.04, 0.21).

2.6) Plot predicted probabilities. It is often useful to compute predicted probabilities across the range of a continuous variable for two groups and then plot these. We do this using `mgen`. `mgen` generates a series of new variables containing predicted values and confidence intervals. The new variables begin with the stem you indicate in the option `stub()`. **HINT:** Keep your `stub()` short, 3-4 characters only. Longer stems may be truncated, which can make coding confusing. The `CI` option should be included if you wish to generate confidence intervals. These commands:

```
. mgen,          at(fellow=1 mcit3=(0(5)130)) atmeans stub(F)
**For Fellows, across the range of mentor citations

Predictions from: margins, at(fellow=1 mcit3=(0(5)130)) atmeans predict(pr)
```

Variable	Obs	Unique	Mean	Min	Max	Label
Fpr1	27	27	.8361422	.621785	.9599656	pr(y=1_Yes) from margins
Fll1	27	27	.748555	.5078947	.8969149	95% lower limit
Full1	27	27	.9237294	.7356753	1.023016	95% upper limit
Fmcit3	27	27	65	0	130	Mentor's 3 yr citation.

```
. label var      Fpr1 "Fellow"
```

```
. mgen,          at(fellow=0 mcit3=(0(5)130)) atmeans stub(NF)
**For Non-Fellows, across the range of mentor citations
```

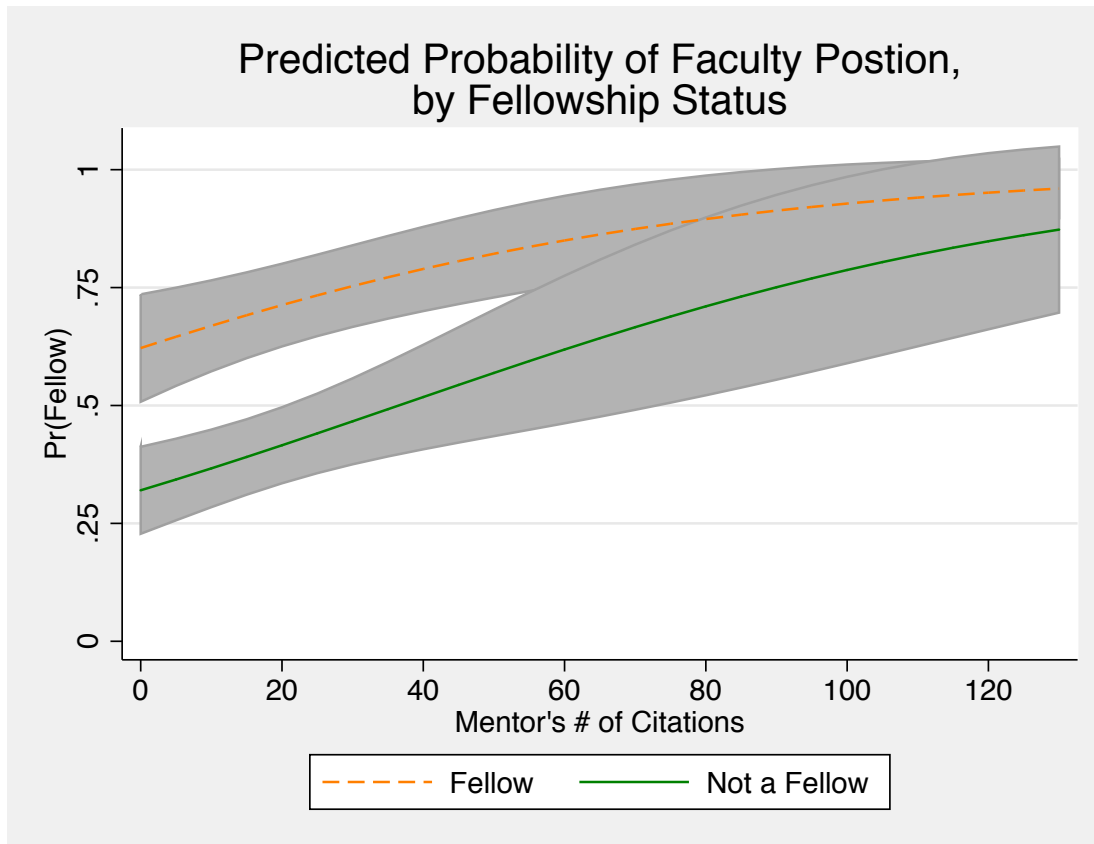
```
Predictions from: margins, at(fellow=0 mcit3=(0(5)130)) atmeans predict(pr)
```

Variable	Obs	Unique	Mean	Min	Max	Label
NFpr1	27	27	.6246303	.3201629	.8729175	pr(y=1_Yes) from margins
NFll1	27	27	.4768771	.2279714	.6968755	95% lower limit
NFull1	27	27	.7723835	.4123544	1.048959	95% upper limit
NFmcit3	27	27	65	0	130	Mentor's 3 yr citation.

```
. label var      NFpr1 "Not a Fellow"
```

```
. graph twoway ///
>   (rarea Full1 Fll1 Fmcit3, color(gs10)) ///
>   (rarea NFull1 NFll1 NFmcit3, color(gs10)) ///
>   (connected Fpr1 Fmcit3, lpattern(dash)) ///
>   lcolor(orange) msymbol(none)) ///
>   (connected NFpr1 NFmcit3, lpattern(solid)) ///
>   lcolor(green) msymbol(none)), ///
>   legend(on order(3 4)) ///
>   ylabel(0(.25)1) ytitle("Pr(Fellow)") ///
>   xlabel(0(20)130) xtitle("Mentor's # of Citations") ///
>   title("Predicted Probability of Faculty Postion," ///
>         "by Fellowship Status")
```

```
. graph export icpsrcda02-binary-fig2.png , width(1200) replace
(file icpsrcda02-binary-fig2.png written in PNG format)
```



Interpretation: For a scientist at an average prestige PhD-granting institution, receiving a fellowship increases the probability of being employed as a faculty member when mentor's citations are below 50 or so, by about .30. Above 50 mentor citations, there is no significant difference between scientists who received a fellowship and those who did not. As such, fellowships seem to be particularly useful when mentor's citations are low. For both fellow and non-fellows, the probability of having a faculty position increases as the number of mentor's citations increase. This effect is largest for non-fellows, whose predicted probability of being a faculty increases from .37 at 10 mentor citations to .79 at 100 mentor citations, an increase of .42. This change is significant (95% CI: 0.20, 0.64).

__2.END) Close Log File and Exit Do File.

```
log close
exit
```


Section 3: Testing and Assessing Fit

For a fuller discussion of testing and assessing fit, see Chapter 3 of L&F (2005).

The file `icpsrcda14-03-testing.do` contains these Stata commands.

___3.1a) Set-up your do-file.

```
capture log close
log using icpsrcda14-03-testing, replace text
```

```
// program:    icpsrcda14-03-testing.do
// task:      Review 3 - Testing & Fit
// project:   ICPSR CDA
// author:    Trent Mize \ 2014-06-24
```

```
*program setup
version      14.0
clear       all
set         linesize 80
```

___3.1b) Load the Data.

```
usecda      cda_scireview4
```

___3.1c) Examine data, keep variables, drop missing, and verify.

```
codebook,   compact
keep        faculty female fellow phd mcit3 mnas
misschk,    gen(m)
tab         mnumber
keep if     mnumber==0
codebook    faculty female fellow phd mcit3 mnas, compact
sum         faculty female fellow phd mcit3 mnas
```

___3.2) Computing a z-test. z-statistics are produced with the standard estimation commands. In the output below the z-statistics are in the 4th column. This command:

```
logit faculty i.female i.fellow phd mcit3 i.mnas, nolog
```

Produces the output on the following page:

```
. logit faculty i.female i.fellow phd mcit3 i.mnas, nolog
```

```
Logistic regression                               Number of obs   =       264
                                                  LR chi2(5)      =       41.72
                                                  Prob > chi2     =       0.0000
Log likelihood = -161.51514                    Pseudo R2      =       0.1144
```

faculty	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
female					
1_Yes	-.5869003	.2911944	-2.02	0.044	-1.157631 - .0161698
fellow					
1_Yes	1.118336	.2844612	3.93	0.000	.5608027 1.67587
phd	.002004	.1521298	0.01	0.989	-.2961648 .3001729
mcit3	.0190813	.0072584	2.63	0.009	.0048551 .0333075
mnas					
1_Yes	.3537104	.5652778	0.63	0.531	-.7542137 1.461635
_cons	-.5004836	.4539085	-1.10	0.270	-1.390128 .3891607

3.3a) Single Coefficient Wald Test. After estimation, the command `test` can compute a Wald test that a single coefficient is equal to zero. This command:

```
test female
```

Produces this output:

```
. test 1.female
```

```
( 1)  [faculty]1.female = 0

           chi2( 1) =      4.06
           Prob > chi2 =      0.0439
```

The effect of female is significant at the .05 level ($\chi^2=4.06, df=1, p=.04$).

3.4) Single Coefficient LR Test. To conduct a likelihood ratio test you begin by storing the estimation results using the `estimates store` command. We run the base model and then store the estimates with the name `base`. To test that the effect of female is zero, re-run the base model without `female` and then compare with the full model using `lrtest estname1 estname2`. These commands:

```
logit          faculty i.female i.fellow phd mcit3 i.mnas
estimates store base
logit          faculty i.fellow phd mcit3 i.mnas
estimates store nofemale
lrtest         base nofemale
```

Produces this output:

```
Likelihood-ratio test                               LR chi2(1) =      4.08
(Assumption: nofemale nested in base)              Prob > chi2 =      0.0434
```

The effect of female is significant at the .05 level ($LR\chi^2=4.08$, $df=1$, $p=.04$).

___3.5) Multiple Coefficients Wald Test. We can also test if multiple coefficients are equal to zero. **HINT:** Remember to re-run your base model if it was not your last model run. Use the option `qui` or `quietly` to suppress output if the model is familiar. These commands:

```
qui logit    faculty i.female i.fellow phd mcit3 i.mnas
test        mcit3 1.mnas
```

Produce this output:

```
. test mcit 1.mnas

( 1)  [faculty]mcit3 = 0
( 2)  [faculty]1.mnas = 0

           chi2( 2) =      7.78
       Prob > chi2 =      0.0204
```

The hypothesis that the effects of mentor's citations and mentor's status as an NAS member are simultaneously equal to zero can be rejected at the .05 level ($\chi^2=7.78$, $df=2$, $p=.02$).

___3.6) Multiple Coefficients LR Test. To test if the effects of `mcit3` and `mnas` are jointly zero, run the comparison model without these variables, store the estimation results, and then compare models using `lrtest`. These commands:

```
logit          faculty 1.female 1.fellow phd
estimates store nomcit3mnas
lrtest         base nomcit3mnas
```

Produces this output:

```
Likelihood-ratio test                LR chi2(2) =      9.19
(Assumption: nomcit3mnas nested in base)  Prob > chi2 =      0.0101
```

The hypothesis that the effects of mentor's citations and mentor's status as an NAS member are simultaneously equal to zero can be rejected at the .01 level ($LR\chi^2=9.19$, $df=2$, $p=.01$).

___3.7) Wald Test All Coefficients are Zero. We can also test if all coefficients are equal to zero. **HINT:** Remember to re-run your base model if it was not your last model run. Use the option `qui` or `quietly` to suppress output if the model is familiar. These commands:

```
qui logit faculty i.female i.fellow phd mcit3 i.mnas
test 1.female 1.fellow phd mcit3 1.mnas
```

Produce the output on the following page:

```
. test                female fellow phd mcit3 mnas

( 1)  [faculty]female = 0
( 2)  [faculty]fellow = 0
( 3)  [faculty]phd = 0
( 4)  [faculty]mcit3 = 0
( 5)  [faculty]mnas = 0

           chi2( 5) =    33.78
       Prob > chi2 =    0.0000
```

3.8) LR Test All Coefficients are Zero. To test that all of the regressors have no effect, we estimate the model with only an intercept, store the estimation results again, and compare the models using `lrtest`. Note that this test statistic is identical to the one produced at the top of the estimation output for the full model (see page 16). These commands:

```
logit                faculty
estimates store      intercept
lrtest               base intercept
```

Produce this output:

```
. lrtest            base intercept
```

```
Likelihood-ratio test                LR chi2(5) =    41.72
(Assumption: intercept nested in base)  Prob > chi2 =    0.0000
```

We can reject the hypothesis that all coefficients except the intercept are zero at the .01 level ($LR\chi^2=41.72$, $df=5$, $p<.01$).

3.9) More Complicated Wald Tests. Two examples: we can test that the coefficients for `mcit3` and `mnas` are equal and that the effect of `female` is twice the effect of `mcit3`. These commands:

```
qui logit faculty i.female i.fellow phd mcit3 i.mnas
test mcit3 = 1.mnas
test 1.female = 2*mcit3
```

Produce this output:

```
. test                mcit3 = 1.mnas

( 1)  [faculty]mcit3 - [faculty]1.mnas = 0

           chi2( 1) =    0.35
       Prob > chi2 =    0.5545
```

```
. test          1.female = 2*mcit3

( 1)  [faculty]1.female - 2*[faculty]mcit3 = 0

      chi2( 1) =    4.64
      Prob > chi2 =    0.0313
```

The hypothesis that the effect of mentor's citations is equal to the effect of mentor's status as an NAS member cannot be rejected ($\chi^2=.35$, $df=1$, $p=.55$).

The hypothesis that the effect of gender is equal to twice the effect of mentor's citations is rejected at the 0.05 level ($\chi^2=4.64$, $df=1$, $p=0.03$).

3.10a) Compare BIC & AIC statistics across non-nested models. BIC & AIC allow comparison of non-nested models. Here we estimate a series of four non-nested models and then present all models in a table with their BIC and AIC statistics. These commands:

```
qui logit      faculty i.female i.fellow
estimates store m1

qui logit      faculty mcit3
estimates store m2

qui logit      faculty phd i.mnas
estimates store m3

qui logit      faculty phd i.mnas i.female i.fellow
estimates store m4

estimates table m1 m2 m3 m4, b(%9.3f) star stats(bic aic)
```

Produce this output:

```
. estimates table          m1 m2 m3 m4, b(%9.3f) star stats(bic aic)
```

Variable	m1	m2	m3	m4
female				
1_Yes	-0.624*			-0.695*
fellow				
1_Yes	1.182***			1.092***
mcit3		0.022***		
phd			0.167	0.165
mnas				
1_Yes			0.738	0.522
_cons	-0.115	-0.283	-0.451	-0.618
bic	351.254	361.088	376.048	359.127
aic	340.526	353.936	365.320	341.247

legend: * p<0.05; ** p<0.01; *** p<0.001

3.11) Compute and list residuals. Residuals can provide important information about which observations are least well-predicted by your model. We can use the command `predict, rs` to predict residuals, and `sort` and `list` to list the 10 most negative and 10 most positive residuals. Note that the question does ask for the *observations* with the most negative and most positive residuals, but rather the 10 residuals that are most negative and most positive. As observations may well have the same residual, you may need to list more than 10 residuals at any one time. The commands below start with 20 and can be increased or decreased as necessary. These commands:

```
logit      faculty phd i.mnas i.female i.fellow
predict    rstd, rs
sort       rstd
list       rstd faculty phd mnas female fellow in 1/20, clean // NB: 1=#
list       rstd faculty phd mnas female fellow in -20/1, clean // NB: 1=letter
```

Produce this output:

```
. list          rstd faculty phd mnas female fellow in 1/20, clean
```

	rstd	faculty	phd	mnas	female	fellow
1.	-2.218083	0_No	3.36	1_Yes	0_No	1_Yes
2.	-2.128379	0_No	4.66	1_Yes	0_No	0_No
3.	-2.128379	0_No	4.66	1_Yes	0_No	0_No
4.	-1.807865	1_Yes	4.29	0_No	0_No	1_Yes
5.	-1.807865	0_No	4.29	0_No	0_No	1_Yes
6.	-1.807865	0_No	4.29	0_No	0_No	1_Yes
7.	-1.716709	0_No	3.6	0_No	0_No	1_Yes
8.	-1.685745	0_No	2.96	0_No	1_Yes	0_No
9.	-1.685745	0_No	2.96	0_No	1_Yes	0_No

```

10.  -1.685745      0_No  2.96   0_No   1_Yes   0_No
11.  -1.685745      0_No  2.96   0_No   1_Yes   0_No
12.  -1.685745      0_No  2.96   0_No   1_Yes   0_No
13.  -1.685745      0_No  2.96   0_No   1_Yes   0_No
14.  -1.610768      0_No  2.83   0_No   0_No    1_Yes
15.  -1.546281      0_No  2.32   0_No   0_No    1_Yes
16.  -1.524043      0_No  1.76   0_No   1_Yes   1_Yes
17.  -1.524043      0_No  1.76   0_No   1_Yes   1_Yes
18.  -1.513915      0_No  2.05   0_No   0_No    1_Yes
19.  -1.504556      0_No  1.97   0_No   0_No    1_Yes
20.  -1.386946      0_No  3.36   0_No   0_No    0_No

```

```
. list          rstd faculty phd mnas female fellow in -20/1, clean
```

```

          rstd  faculty   phd    mnas   female  fellow
245.    1.540243   1_Yes   2.83   0_No   0_No   0_No
246.    1.540243   1_Yes   2.83   0_No   0_No   0_No
247.    1.578285   1_Yes   2.54   0_No   0_No   0_No
248.    1.578285   1_Yes   2.54   0_No   0_No   0_No
249.    1.594665   1_Yes   4.29   1_Yes  1_Yes  0_No
250.    1.594665   1_Yes   4.29   1_Yes  1_Yes  0_No
251.    1.707588   1_Yes   1.6    0_No   1_Yes  0_No
252.    1.734994   1_Yes   1.42   0_No   1_Yes  0_No
253.    1.753596   1_Yes   1.3    0_No   1_Yes  0_No
254.    2.880436   1_Yes   4.36   0_No   1_Yes  0_No
255.    2.880436   1_Yes   4.36   0_No   1_Yes  0_No
256.    2.880436   1_Yes   4.36   0_No   1_Yes  0_No
257.    2.880436   0_No    4.36   0_No   1_Yes  0_No
258.    2.880436   1_Yes   4.36   0_No   1_Yes  0_No
259.    2.880436   1_Yes   4.36   0_No   1_Yes  0_No
260.    2.880436   1_Yes   4.36   0_No   1_Yes  0_No
261.    2.880436   1_Yes   4.36   0_No   1_Yes  0_No
262.    2.880436   1_Yes   4.36   0_No   1_Yes  0_No
263.    2.880436   0_No    4.36   0_No   1_Yes  0_No
264.    2.880436   0_No    4.36   0_No   1_Yes  0_No

```

___3.END) Close the Log File and exit Do File.

```
log close
exit
```

Section 4: Models for Ordinal Outcomes

For a fuller discussion of models for ordinal outcomes, see Chapter 5 of L&F (2005).

The file `icpsrcda04-ordinal.do` contains these Stata commands.

___4.1a) Set-up your do-file.

```
capture log close
log using icpsrcda14-04-ordinal, replace text

// program:    icpsrcda14-04-ordinal.do
// task:      Review 4 - Ordinal Regression
// project:   ICPSR CDA
// author:    Trent Mize \ 2014-06-24
```

```
*program setup
version      14.0
clear       all
set         linesize 80
```

___4.1b) Load the Data.

```
usecda      cda_scireview4
```

___4.1c) Examine data, select variables, drop missing, and verify.

```
codebook,   compact
keep       jobprst publ phd female
misschk,   gen(m)
tab        mnumber
keep if    mnumber==0
codebook   jobprst publ phd female, compact
sum        jobprst publ phd female
```

___4.2) Produce table including distribution of output variable. Make sure to include the distribution of the outcome variable, in this case `jobprst`, in addition to the output produced above. The command:

```
tab1 jobprst, mis
```

Produces this output:

```
-> tabulation of jobprst
```

```
Rankings of |
University |
      Job. |      Freq.      Percent      Cum.
-----+-----
      1_Adeq |          29      10.98      10.98
      2_Good |         128      48.48      59.47
      3_Strong |          93      35.23      94.70
      4_Dist |          14       5.30     100.00
-----+-----
      Total |         264     100.00
```


4.3) Estimate the Ordered Logit. `ologit` and `oprobit` work in the same way. We only show `ologit`, but you might want to try what follows using `oprobit`. This command:

```
ologit jobprst pub1 phd i.female
```

Produces this output:

```
Iteration 0: log likelihood = -294.86055
Iteration 1: log likelihood = -255.97269
Iteration 2: log likelihood = -254.5157
Iteration 3: log likelihood = -254.51518
Iteration 4: log likelihood = -254.51518
```

```
Ordered logistic regression                               Number of obs   =          264
                                                         LR chi2(3)      =          80.69
                                                         Prob > chi2     =          0.0000
Log likelihood = -254.51518                             Pseudo R2       =          0.1368
```

jobprst	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
pub1	.1078786	.0481107	2.24	0.025	.0135833	.2021738
phd	1.130028	.1444046	7.83	0.000	.8470003	1.413056
female						
1_Yes	-.6973579	.2617103	-2.66	0.008	-1.210301	-.1844152
/cut1	.9274554	.4268201			.0909033	1.764007
/cut2	4.003182	.4996639			3.023859	4.982506
/cut3	7.034637	.6296717			5.800503	8.26877

4.4-6) Odds Ratios. The factor change in the odds can be computed for the ordinal logit model. Again we do this with the command `listcoef`. The `help` option presents a “key” to interpreting the headings of the output. The `percent` option presents change as a percentage. These commands:

```
listcoef, help
listcoef, percent help
```

Produce the output on the following page:

```
. listcoef, help
```

```
ologit (N=264): Factor change in odds
```

```
Odds of: >m vs <=m
```

	b	z	P> z	e^b	e^bStdX	SDofX
pub1	0.1079	2.242	0.025	1.114	1.321	2.581
phd	1.1300	7.825	0.000	3.096	3.114	1.005
female						
1_Yes	-0.6974	-2.665	0.008	0.498	0.717	0.476

```
b = raw coefficient
z = z-score for test of b=0
P>|z| = p-value for z-test
e^b = exp(b) = factor change in odds for unit increase in X
e^bStdX = exp(b*SD of X) = change in odds for SD increase in X
SDofX = standard deviation of X
```

```
. listcoef, percent help
```

```
ologit (N=264): Percentage change in odds
```

```
Odds of: >m vs <=m
```

	b	z	P> z	%	%StdX	SDofX
pub1	0.1079	2.242	0.025	11.4	32.1	2.581
phd	1.1300	7.825	0.000	209.6	211.4	1.005
female						
1_Yes	-0.6974	-2.665	0.008	-50.2	-28.3	0.476

```
b = raw coefficient
z = z-score for test of b=0
P>|z| = p-value for z-test
% = percent change in odds for unit increase in X
%StdX = percent change in odds for SD increase in X
SDofX = standard deviation of X
```

Interpretations: The odds of receiving a higher ranked job are .50 times smaller for women than men, holding other variables constant. This effect is significant at the $p < 0.01$ level ($z = -2.665$). For a standard deviation increase in publications, about 2.6, the odds of receiving a higher ranked job increase by 32 percent, holding other variables constant.

4.7) Compute Discrete Change for C and B using prchange. `mchange` computes discrete changes at specific values of the independent variables. By default, both multiple discrete changes and the marginal change are computed. Values for specific independent variables can be set using the `at()`. **NOTE:** By default, `mchange` will give you average marginal effects. You can produce marginal effects at the mean by simply adding the `atmeans` option. This command:

```
mchange          female pub1, atmeans centered
```

Produces this output:

```
. mchange female pub, atmeans centered

ologit: Changes in Pr(y) | Number of obs = 264

Expression: Pr(jobprst), predict(outcome())
```

	1 Adeq	2 Good	3 Strong	4 Dist
female				
1 Yes vs 0 No	0.047	0.116	-0.143	-0.019
p-value	0.020	0.006	0.007	0.019
pub1				
+1 cntr	-0.006	-0.019	0.023	0.003
p-value	0.035	0.029	0.028	0.043
+SD cntr	-0.017	-0.050	0.058	0.008
p-value	0.036	0.029	0.027	0.044
Marginal	-0.006	-0.019	0.023	0.003
p-value	0.035	0.029	0.028	0.043

Predictions at base value

	1_Adeq	2_Good	3_Strong	4_Dist
Pr(y base)	0.064	0.534	0.371	0.031

Base values of regressors

	pub1	phd	female
at	2.32	3.18	.345

1: Estimates with margins option atmeans.

Interpretation: For an average scientist, an additional publication increases the probability of receiving a *strong* job by .02.

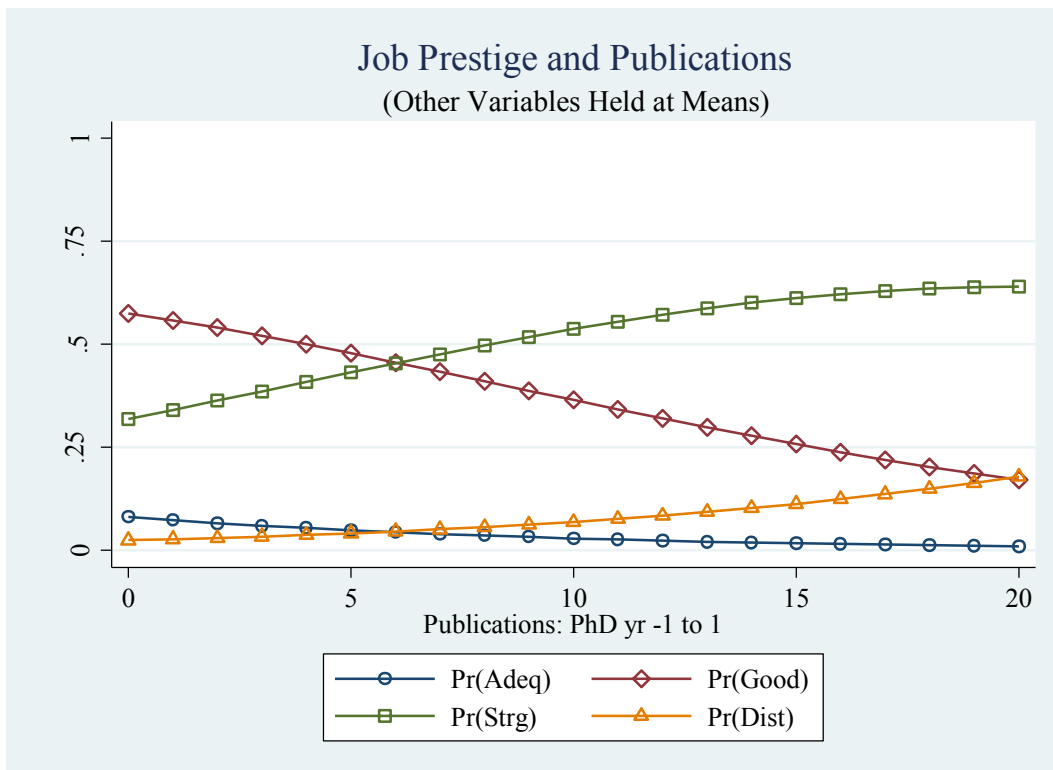
4.8) Graph Predicted Probabilities. Here we use the command `mgen` to generate variables for graphing. We look at individuals who are average on other characteristics and show how predicted probabilities are influenced by publications. `mgen` creates variables of both the predicted probabilities and the cumulative probabilities. Here we use `scatter` to plot the predicted probabilities and the cumulative probabilities. These commands:

```
mgen,          at(pub1=(0(1)20)) atmeans stub(pubpr)

label var pubprpr1 "Pr(Adeq)"
label var pubprpr2 "Pr(Good)"
label var pubprpr3 "Pr(Strg)"
label var pubprpr4 "Pr(Dist)"
label var pubprCpr1 "Pr(<=Adeq)"
label var pubprCpr2 "Pr(<=Good)"
label var pubprCpr3 "Pr(<=Strg)"
label var pubprCpr4 "Pr(<=Dist)"

**Graph predicted probabilities
graph twoway (connected pubprpr1 pubprpr2 pubprpr3 pubprpr4 pubprpub1, ///
             title("Job Prestige and Publications") ///
             subtitle("(Other Variables Held at Means)") ///
             ytitle("Predicted Pr(Job Prestige)") ///
             xtitle("Publications: PhD yr -1 to 1") ///
             xlabel(0(5)20) ylabel(0(.25)1, grid) ///
             msymbol(Oh Dh Sh Th))

graph export icpsrcda04-ordinal-fig1.png , width(1200) replace
```

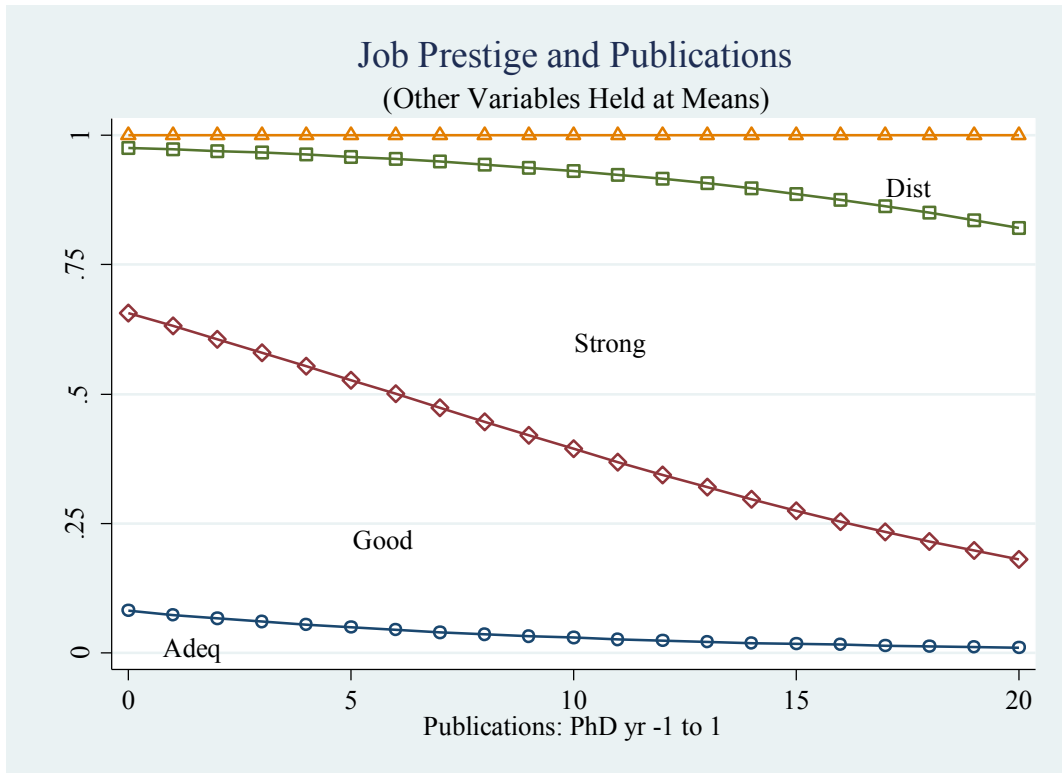


```

**Graph cumulative probabilities
graph twoway (connected pubprCpr1 pubprCpr2 pubprCpr3 pubprCpr4 pubprpub1, ///
             title("Job Prestige and Publications") ///
             subtitle("(Other Variables Held at Means)") ///
             ytitle("Cumulative Pr(Job Prestige)") ///
             xtitle("Publications: PhD yr -1 to 1") ///
             xlabel(0(5)20) ylabel(0(.25)1, grid) ///
             msymbol(Oh Dh Sh Th) name(tmp2, replace) ///
             text(.01 .75 "Adeq", place(e)) ///
             text(.22 5 "Good", place(e)) ///
             text(.60 10 "Strong", place(e)) ///
             text(.90 17 "Dist", place(e)), ///
             legend(off)

graph export icpsrcda04-ordinal-fig2.png , width(1200) replace

```



Interpretation of cumulative probability plot: First, the probability of obtaining a job in the lowest ranking category (adequate) is quite low for scientists regardless of the number of publications, peaking at only around .10. Similarly, the probability of attaining a distinguished position is quite low, even at the highest level of publication (around .20). However, individuals with more publications have a much higher probability of attaining a job that is either strong or distinguished compared to those who have published at lower levels. As such, the most dramatic change across the range of publications appears to be in the probability of obtaining a good job, which decreases as publications increase and is offset by an increase in the probability of obtaining a prestigious job.

___4.14) Testing the Parallel Regression Assumption. `brant` performs a Brant test of the parallel regressions assumptions for the ordered logit model estimated by `ologit`. This command:

```
brant, detail
```

Produces this output:

```
. brant, detail
```

Estimated coefficients from binary logits

Variable	y_gt_1	y_gt_2	y_gt_3
pub1	-0.002	0.109	0.143
	-0.02	1.66	1.73
phd	0.350	1.414	1.605
	1.73	7.44	3.27
female			
1_Yes	0.476	-1.122	-2.104
	1.03	-3.21	-1.96
_cons	0.891	-4.940	-8.897
	1.41	-7.52	-4.33

legend: b/t

Brant test of parallel regression assumption

	chi2	p>chi2	df
All	38.88	0.000	6
pub1	2.76	0.252	2
phd	22.68	0.000	2
1.female	11.26	0.004	2

A significant test statistic provides evidence that the parallel regression assumption has been violated.

___4.END) Close the Log File and Exit Do File.

```
log close  
exit
```

Section 5: Models for Nominal Outcomes

For details about models for multinomial outcomes and associated Stata commands, please read Chapter 6 of L&F (2005). File `icpsrcda05-nominal.do` contains these Stata commands.

___5.1a) Set-up your do-file.

```
capture log close
log using icpsrcda14-05-nominal, replace text

// program:    icpsrcda14-05-nominal.do
// task:      Review 5 - Nominal Regression
// project:   ICPSR CDA
// author:    Trent Mize \ 2014-06-24
```

```
*program setup
version      14.0
clear       all
set         linesize 80
```

___5.1b) Load the Data.

```
usecda      cda_scireview4
```

___5.1c) **Examine data, select variables, and drop missing.** In this case, we are using the same dependent variable we used in the ordinal assignment, as the Brant test showed violations of the parallel regression assumption. Also be sure to look at the distribution of the outcome variable, in this case `jobprst`, to ensure it has been labeled appropriately. These commands:

```
codebook,   compact
keep        jobprst female mcit3 publ phd
misschk,    gen(m)
tab         mnumber
keep if     mnumber==0
tab1        jobprst
```

Produce this output:

```
-> tabulation of jobprst
```

```
Rankings of |
University |
           Job. |           Freq.           Percent           Cum.
-----+-----
           1_Adeq |              29             10.98             10.98
           2_Good |             128             48.48             59.47
           3_Strong |              93             35.23             94.70
           4_Dist |              14              5.30            100.00
-----+-----
           Total |             264            100.00
```

___5.2) Verify data are clean.

```
. codebook          jobprst female mcit3 pub1 phd, compact
```

Variable	Obs	Unique	Mean	Min	Max	Label
jobprst	264	4	2.348485	1	4	Rankings of University Job.
female	264	2	.344697	0	1	Female? (1=yes)
mcit3	264	59	20.71591	0	129	Mentor's 3 yr citation.
pub1	264	14	2.32197	0	19	Publications: PhD yr -1 to 1.
phd	264	79	3.181894	1	4.66	Prestige of Ph.D. department.

```
. sum          jobprst female mcit3 pub1 phd
```

Variable	Obs	Mean	Std. Dev.	Min	Max
jobprst	264	2.348485	.7449179	1	4
female	264	.344697	.4761721	0	1
mcit3	264	20.71591	25.44536	0	129
pub1	264	2.32197	2.580736	0	19
phd	264	3.181894	1.00518	1	4.66

___5.4) **Multinomial Logit.** `mlogit` estimates the multinomial logit model. The option `baseoutcome()` allows you to set the comparison category. Stata will only give you the coefficients for the comparison to the base category. You can use the `listcoef` command to get all the coefficients for all contrasts. These commands:

```
//5.4a) Multinomial Logit.
```

```
mlogit          jobprst i.female mcit3 pub1 phd, baseoutcome(4) nolog
```

```
//5.4b) Multinomial Logit.
```

```
listcoef, help
```

Output on following page:


```
. mlogit jobprst i.female mcit3 pub1 phd, baseoutcome(4) nolog
```

```
Multinomial logistic regression      Number of obs   =      264
LR chi2(12)                          =      116.94
Prob > chi2                            =      0.0000
Log likelihood = -236.38913           Pseudo R2       =      0.1983
```

jobprst		Coef.	Std. Err.	z	P> z	[95% Conf. Interval]

1_Adeq						
	female					
	1_Yes	1.695191	1.189389	1.43	0.154	-.6359679 4.02635
	mcit3	-.0120625	.0116282	-1.04	0.300	-.0348533 .0107283
	pub1	-.147479	.1232017	-1.20	0.231	-.3889499 .0939919
	phd	-1.918716	.6037909	-3.18	0.001	-3.102124 -.7353071
	_cons	8.321049	2.349796	3.54	0.000	3.715533 12.92657

2_Good						
	female					
	1_Yes	2.601074	1.12741	2.31	0.021	.3913898 4.810757
	mcit3	-.0214454	.0103359	-2.07	0.038	-.0417034 -.0011874
	pub1	-.2096883	.1092039	-1.92	0.055	-.4237241 .0043474
	phd	-2.074765	.5772686	-3.59	0.000	-3.206191 -.9433394
	_cons	10.22683	2.295627	4.45	0.000	5.727482 14.72618

3_Strong						
	female					
	1_Yes	1.390078	1.108392	1.25	0.210	-.7823308 3.562487
	mcit3	-.0229483	.0085379	-2.69	0.007	-.0396824 -.0062143
	pub1	-.0937417	.0915408	-1.02	0.306	-.2731583 .085675
	phd	-.6595582	.5595541	-1.18	0.239	-1.756264 .4371477
	_cons	5.498777	2.262518	2.43	0.015	1.064323 9.93323

4_Dist						
		(base outcome)				

```
. listcoef, help
```

```
mlogit (N=264): Factor change in the odds of jobprst
```

```
Variable: 1.female (sd=0.476)
```

			b	z	P> z	e^b	e^bStdX

1_Adeq	vs 2_Good		-0.9059	-1.854	0.064	0.404	0.650
1_Adeq	vs 3_Strong		0.3051	0.568	0.570	1.357	1.156
1_Adeq	vs 4_Dist		1.6952	1.425	0.154	5.448	2.242
2_Good	vs 1_Adeq		0.9059	1.854	0.064	2.474	1.539
2_Good	vs 3_Strong		1.2110	3.274	0.001	3.357	1.780
2_Good	vs 4_Dist		2.6011	2.307	0.021	13.478	3.451
3_Strong	vs 1_Adeq		-0.3051	-0.568	0.570	0.737	0.865
3_Strong	vs 2_Good		-1.2110	-3.274	0.001	0.298	0.562
3_Strong	vs 4_Dist		1.3901	1.254	0.210	4.015	1.939
4_Dist	vs 1_Adeq		-1.6952	-1.425	0.154	0.184	0.446
4_Dist	vs 2_Good		-2.6011	-2.307	0.021	0.074	0.290
4_Dist	vs 3_Strong		-1.3901	-1.254	0.210	0.249	0.516

Variable: mcit3 (sd=25.445)

		b	z	P> z	e^b	e^bStdX
1_Adeq	vs 2_Good	0.0094	0.876	0.381	1.009	1.270
1_Adeq	vs 3_Strong	0.0109	1.087	0.277	1.011	1.319
1_Adeq	vs 4_Dist	-0.0121	-1.037	0.300	0.988	0.736
2_Good	vs 1_Adeq	-0.0094	-0.876	0.381	0.991	0.788
2_Good	vs 3_Strong	0.0015	0.187	0.852	1.002	1.039
2_Good	vs 4_Dist	-0.0214	-2.075	0.038	0.979	0.579
3_Strong	vs 1_Adeq	-0.0109	-1.087	0.277	0.989	0.758
3_Strong	vs 2_Good	-0.0015	-0.187	0.852	0.998	0.962
3_Strong	vs 4_Dist	-0.0229	-2.688	0.007	0.977	0.558
4_Dist	vs 1_Adeq	0.0121	1.037	0.300	1.012	1.359
4_Dist	vs 2_Good	0.0214	2.075	0.038	1.022	1.726
4_Dist	vs 3_Strong	0.0229	2.688	0.007	1.023	1.793

Variable: publ (sd=2.581)

		b	z	P> z	e^b	e^bStdX
1_Adeq	vs 2_Good	0.0622	0.683	0.495	1.064	1.174
1_Adeq	vs 3_Strong	-0.0537	-0.570	0.569	0.948	0.871
1_Adeq	vs 4_Dist	-0.1475	-1.197	0.231	0.863	0.683
2_Good	vs 1_Adeq	-0.0622	-0.683	0.495	0.940	0.852
2_Good	vs 3_Strong	-0.1159	-1.604	0.109	0.891	0.741
2_Good	vs 4_Dist	-0.2097	-1.920	0.055	0.811	0.582
3_Strong	vs 1_Adeq	0.0537	0.570	0.569	1.055	1.149
3_Strong	vs 2_Good	0.1159	1.604	0.109	1.123	1.349
3_Strong	vs 4_Dist	-0.0937	-1.024	0.306	0.911	0.785
4_Dist	vs 1_Adeq	0.1475	1.197	0.231	1.159	1.463
4_Dist	vs 2_Good	0.2097	1.920	0.055	1.233	1.718
4_Dist	vs 3_Strong	0.0937	1.024	0.306	1.098	1.274

Variable: phd (sd=1.005)

		b	z	P> z	e^b	e^bStdX
1_Adeq	vs 2_Good	0.1560	0.610	0.542	1.169	1.170
1_Adeq	vs 3_Strong	-1.2592	-4.400	0.000	0.284	0.282
1_Adeq	vs 4_Dist	-1.9187	-3.178	0.001	0.147	0.145
2_Good	vs 1_Adeq	-0.1560	-0.610	0.542	0.856	0.855
2_Good	vs 3_Strong	-1.4152	-6.606	0.000	0.243	0.241
2_Good	vs 4_Dist	-2.0748	-3.594	0.000	0.126	0.124
3_Strong	vs 1_Adeq	1.2592	4.400	0.000	3.522	3.546
3_Strong	vs 2_Good	1.4152	6.606	0.000	4.117	4.148
3_Strong	vs 4_Dist	-0.6596	-1.179	0.239	0.517	0.515
4_Dist	vs 1_Adeq	1.9187	3.178	0.001	6.812	6.880
4_Dist	vs 2_Good	2.0748	3.594	0.000	7.963	8.049
4_Dist	vs 3_Strong	0.6596	1.179	0.239	1.934	1.941

b = raw coefficient

z = z-score for test of b=0

P>|z| = p-value for z-test

e^b = exp(b) = factor change in odds for unit increase in X

e^bStdX = exp(b*SD of X) = change in odds for SD increase in X

5.5) Single Coefficient Wald Test. The `mlogtest` command can be used to compute both Wald and LR tests of single coefficients. This command:

```
mlogtest 1.female mcit3, wald lr
```

Produces this output:

```
. mlogtest 1.female mcit3, wald lr
```

```
LR tests for independent variables (N=264)
```

```
Ho: All coefficients associated with given variable(s) are 0
```

	chi2	df	P>chi2
1.female	16.623	3	0.001
mcit3	8.140	3	0.043

```
Wald tests for independent variables (N=264)
```

```
Ho: All coefficients associated with given variable(s) are 0
```

	chi2	df	P>chi2
1.female	14.501	3	0.002
mcit3	7.816	3	0.050

Interpretation: The effect of gender on job prestige is significant at the .01 level ($LRX^2=16.62$, $df=3$, $p=.001$).

5.6) Plot Odds Ratios. One of the difficulties in interpreting nominal models is that many coefficients need to be considered. To help you sort out all the information, odds ratios can be plotted using the `mlogitplot` command.

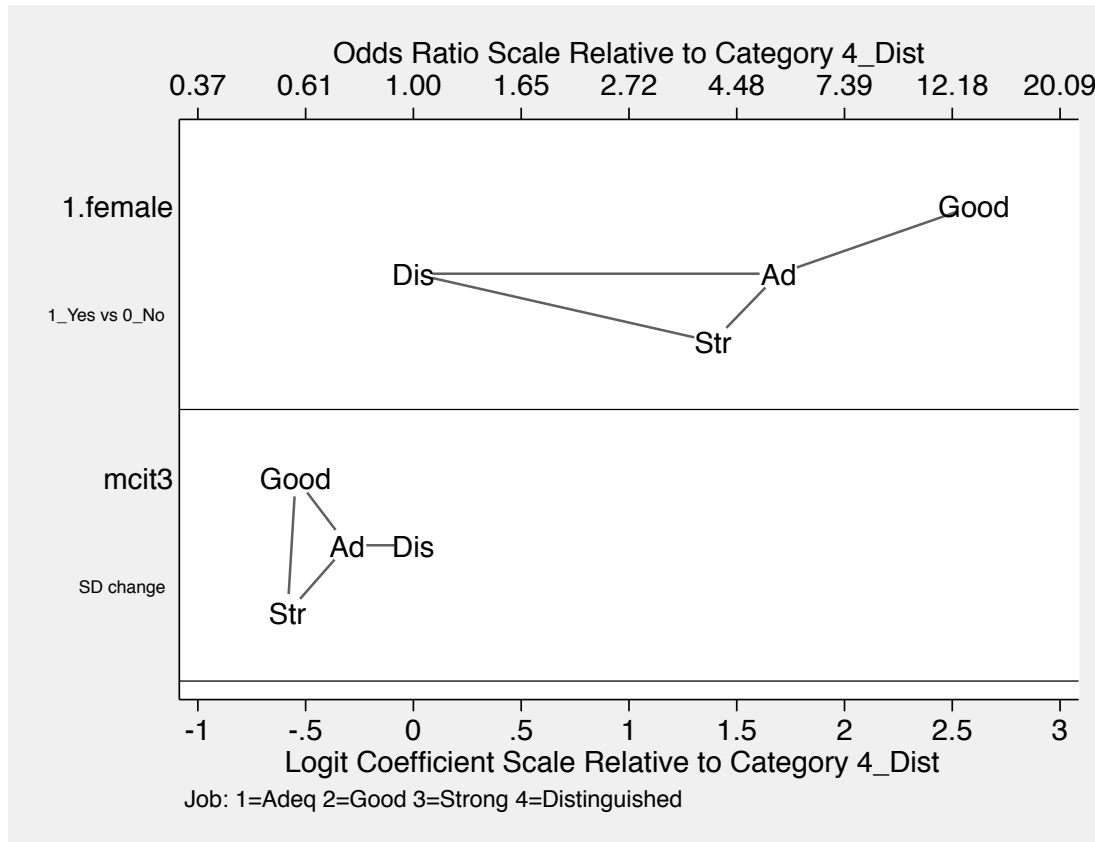
The assignment requires you to indicate significance at the 0.05 level by drawing a line between coefficients that are not differentiated (the default is .05 level, but you can change this using the `linep()` option). You will also need to specify the amount of change you want to plot for non-factor variables—a one-unit change, a standard deviation change, a 0-1 change or a change from min to max. This is done with the `amount()` option and indicating `one`, `bin`, `range`, or `sd`. Do not set a range for factor variables.

For example, if you want to specify that your first two non-factor variables should increase by one unit, and the third non-factor variable should increase across the range you can specify `amount(one one range)`

You should also add a `note` to the plot that includes the values and value labels. You can further specify the symbols for the markers. These should be abbreviations so they are easily readable on the graph. These are listed in order of the outcome categories using the `symbols()` option.

Example:

```
mlogitplot female mcit3, amount(sd) ///  
symbols(Ad Good Str Dis) min(-1) max(3) gap(0.5) ///  
note(Job: 1=Adeq 2=Good 3=Strong 4=Distinguished)
```



Interpretation: The effect of mentor’s citations is small, with a standard deviation increase in publications increasing the odds of obtaining a *distinguished* (4) job compared to either a *good* (2) or *strong* (3) job, but does not distinguish between a *distinguished* (4) job and an *adequate* (1) job. Females have larger odds of obtaining a *good* (2) job relative to either *strong* (3) or *distinguished* (4) jobs, but gender does not distinguish between obtaining a *good* (2) job and an *adequate* (1) job, or between a *strong* (3) and a *distinguished* (4) job.

5.8) Calculating Discrete Change. We can use `mchange` to calculate discrete change. **NOTE:** Remember this will calculate average marginal effects by default unless you specify the `atmeans` option.

(Output on following page)

. mchange, atmeans centered

mlogit: Changes in Pr(y) | Number of obs = 264

Expression: Pr(jobprst), predict(outcome())

	1 Adeq	2 Good	3 Strong	4 Dist
female				
1 Yes vs 0 No	-0.046	0.279	-0.209	-0.024
p-value	0.325	0.000	0.002	0.115
mcit3				
+1 cntr	0.001	-0.001	-0.001	0.000
p-value	0.320	0.786	0.595	0.167
+SD cntr	0.028	-0.013	-0.022	0.007
p-value	0.320	0.786	0.594	0.169
Marginal	0.001	-0.001	-0.001	0.000
p-value	0.320	0.786	0.595	0.167
publ				
+1 cntr	0.001	-0.026	0.022	0.002
p-value	0.878	0.110	0.140	0.216
+SD cntr	0.004	-0.067	0.058	0.006
p-value	0.878	0.109	0.139	0.218
Marginal	0.001	-0.026	0.022	0.002
p-value	0.878	0.111	0.140	0.215
phd				
+1 cntr	-0.047	-0.267	0.292	0.023
p-value	0.062	0.000	0.000	0.028
+SD cntr	-0.048	-0.268	0.293	0.023
p-value	0.062	0.000	0.000	0.028
Marginal	-0.049	-0.275	0.302	0.022
p-value	0.064	0.000	0.000	0.033

Predictions at base value

	1_Adeq	2_Good	3_Strong	4_Dist
Pr(y base)	0.129	0.514	0.343	0.014

Base values of regressors

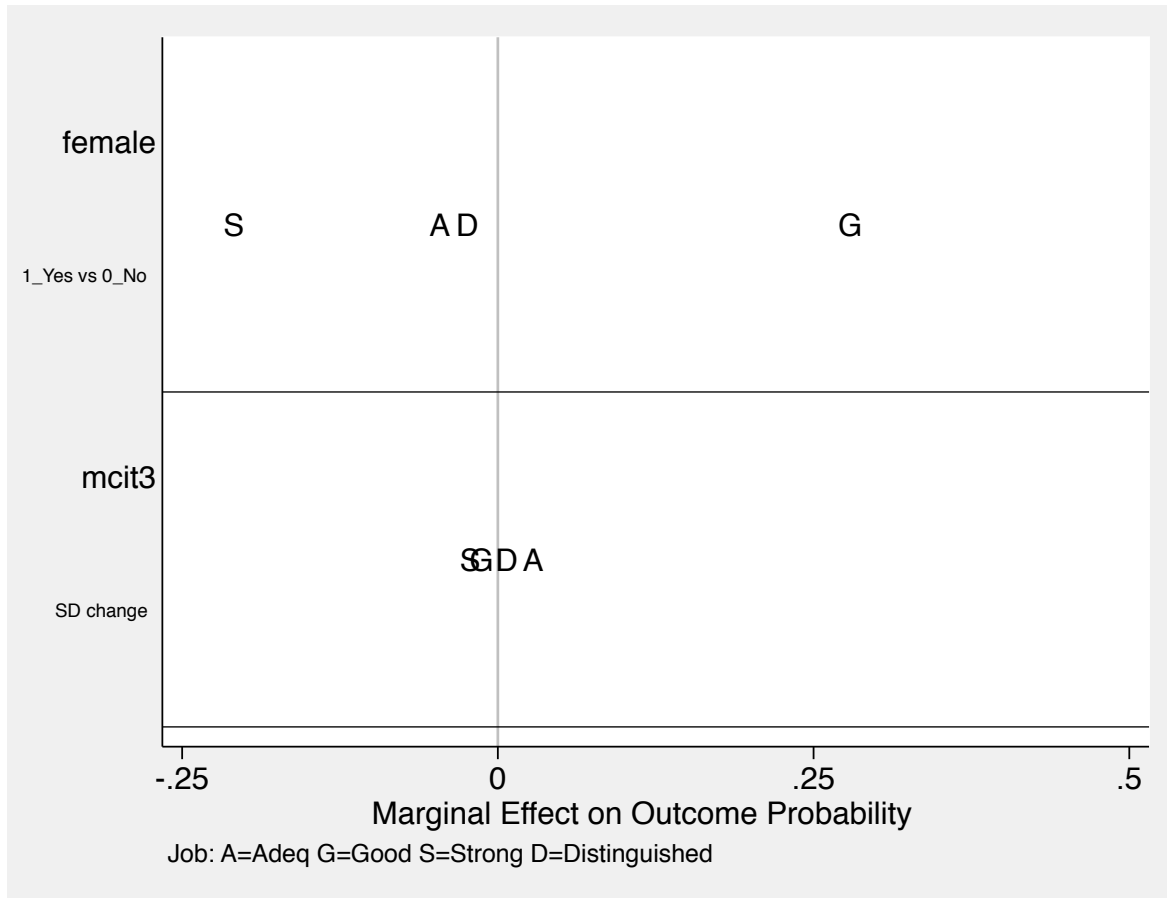
	1. female	mcit3	publ	phd
at	.345	20.7	2.32	3.18

1: Estimates with margins option atmeans.

5.8) Plotting Discrete Change. We can use `mchangeplot` to plot the discrete changes.

Significance is not necessary here, so you simply need to select the amount of change for each non-factor variable and include your note.

```
mchangeplot female mcit3, amount(sd) ///  
          symbols(A G S D) min(-.25) max(.5) gap(0.25) ///  
          note(Job: A=Adeq G=Good S=Strong D=Distinguished)
```



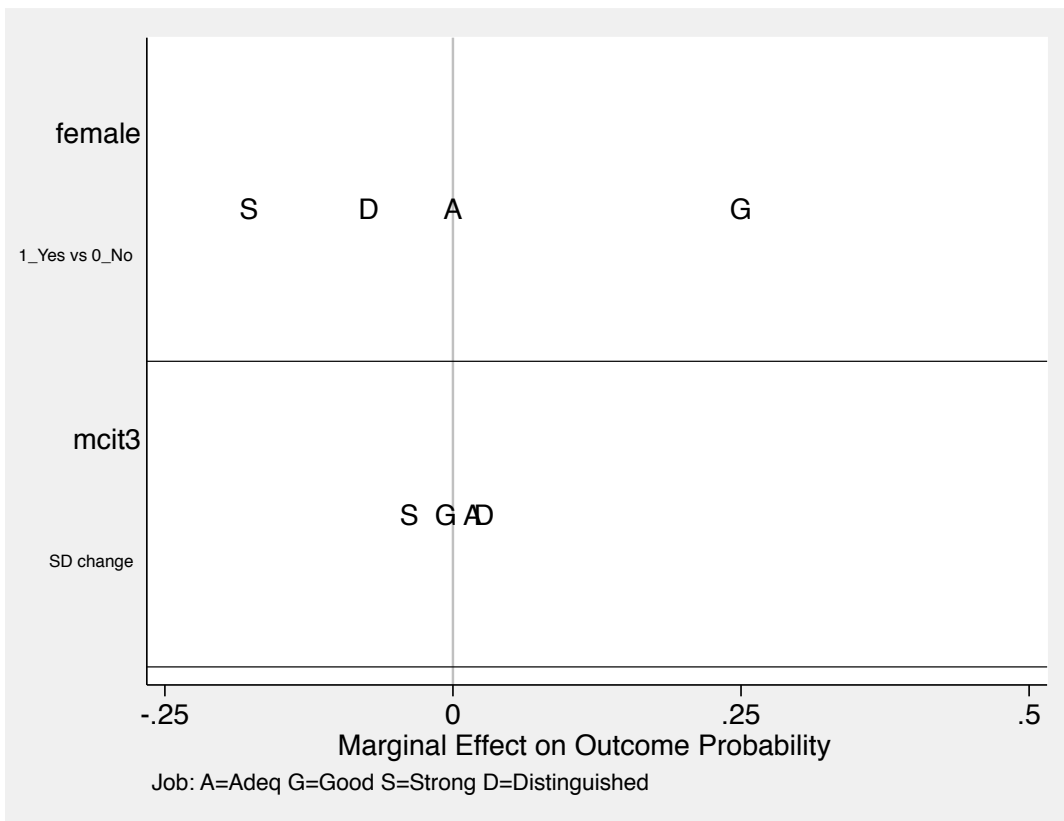
Interpretation: A standard deviation increase in mentor’s citations generally has a small effect on predicted probabilities, increasing the probability of obtaining an *adequate* (1) job by .03 and decreasing the probability of obtaining a *strong* (3) job by about .02. Females have a higher predicted probability than men of being in a *good* (2) job of about .28, but have a .20 lower predicted probability than men of having a *strong* (3) job, for average scientists. Neither of the variables have much of an impact on the probability of obtaining an *adequate* (1) or *distinguished* (4) jobs.

5.8) Calculating and Plotting Discrete Change II. Repeat the steps in 5.8, this time selecting a new location for the other variables. In the example below, we look at individuals from high prestige universities with a larger number of publications to see the effect of female and mentor's citations. We can use `mchange` to calculate discrete change. NOTE: To ensure direct comparability across plots, you may want to specify a min and max in your plot, based on your first plot. These commands:

```
mchange, at(phd=4 pub1=4) atmeans centered
mchangeplot female mcit3, amount(sd) ///
      symbols(A G S D) min(-.25) max(.5) gap(0.25) ///
      note(Job: A=Adeq G=Good S=Strong D=Distinguished)
graph export icpsrcda05-nominal-fig3.png , width(1200) replace
```

Produce this output and plot:

```
. mchange, at(phd=4 pub1=4) atmeans centered
<snip>
. mchangeplot female mcit3, amount(sd) ///
      symbols(A G S D) min(-.25) max(.5) gap(0.25) ///
      note(Job: A=Adeq G=Good S=Strong D=Distinguished)
graph export icpsrcda05-nominal-fig3.png , width(1200) replace
```



Interpretation: For highly productive scientists from high-prestige PhD programs, a standard deviation increase in mentor's citations generally has only a small effect on predicted probabilities, increasing the probability of obtaining a *distinguished* (4) job by .02 and decreasing the probability of obtaining a *strong* (3) job by about .04. Highly-productive females from high-prestige universities have a predicted probability of attaining a *good* (2) job that is about .20 higher than their male counterparts, and similarly a .18 lower predicted probability of attaining a *strong* (3) job, as well as a .07 lower predicted probability of obtaining a *distinguished* (4) job. Neither of the variables have much of an impact on the probability of obtaining an *adequate* (1) job.

___5.END) Close Log File and Exit Do File.

```
log close  
exit
```


Section 6: Models for Count Outcomes Review

For more details about models for count outcomes, please read Chapter 7 of L&F (2005). The file `icpsrcda06-count.do` contains these Stata commands.

___6.1a) Set-up your do-file.

```
capture log close
log using icpsrcda14-06-count, replace text
```

```
// program:    icpsrcda14-06-count.do
// task:       Review 6 - Count Models
// project:    ICPSR CDA
// author:     Trent Mize \ 2014-06-24
```

```
*program setup
version      14.0
clear       all
set         linesize 80
```

___6.1b) Load the Data.

```
usecda      cda_scireview4
```

___6.1c) Examine data, select variables, drop missing, and verify. Include output from `codebook`, `compact` and `sum`. Also make sure to look at the distribution of the outcome variable, in this case, `pub6`.

```
codebook,    compact
keep        pub6 female phd enrol
misschk,    gen(m)
tab         mnumber
keep if     mnumber==0
tab1        pub6
sum         pub6 female phd enrol
codebook    pub6 female phd enrol, compact
```

___6.3) Estimate the Poisson and NBReg Regression Models. Using the `estimates store` and `estimates table` command may help with question 4.

```
. poisson pub6 i.female phd enrol, nolog
```

```
Poisson regression
Number of obs = 264
LR chi2(3) = 78.72
Prob > chi2 = 0.0000
Pseudo R2 = 0.0448
Log likelihood = -839.78052
```

pub6	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
female						
1_Yes	-.2408113	.069001	-3.49	0.000	-.3760508	-.1055719
phd	.1882524	.0321844	5.85	0.000	.1251721	.2513328
enrol	-.1325456	.0240756	-5.51	0.000	-.179733	-.0853582
_cons	1.532594	.1699269	9.02	0.000	1.199543	1.865644

```
. estimates store prm
```

```
. nbreg pub6 i.female phd enrol, nolog
```

```
Negative binomial regression
Number of obs = 264
LR chi2(3) = 20.59
Dispersion = mean
Prob > chi2 = 0.0001
Pseudo R2 = 0.0158
Log likelihood = -642.723
```

pub6	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
female						
1_Yes	-.2822292	.1382637	-2.04	0.041	-.553221	-.0112373
phd	.1995909	.0651859	3.06	0.002	.0718288	.327353
enrol	-.150895	.0480431	-3.14	0.002	-.2450578	-.0567322
_cons	1.607418	.3379749	4.76	0.000	.9449989	2.269836
/lnalpha	-.203673	.1255831			-.4498113	.0424654
alpha	.8157291	.1024418			.6377485	1.04338

```
Likelihood-ratio test of alpha=0: chibar2(01) = 394.12 Prob>=chibar2 = 0.000
```

```
. estimates store nbreg
```

```
. esttab          prm nbreg, z mti wide
```

	(1) prm		(2) nbreg	
pub6				
0.female	0	(.)	0	(.)
1.female	-0.241***	(-3.49)	-0.282*	(-2.04)
phd	0.188***	(5.85)	0.200**	(3.06)
enrol	-0.133***	(-5.51)	-0.151**	(-3.14)
_cons	1.533***	(9.02)	1.607***	(4.76)
lnalpha				
_cons			-0.204	(-1.62)
N	264		264	

z statistics in parentheses
 * p<0.05, ** p<0.01, *** p<0.001

___6.7) Testing the NBRM vs. the PRM. The NBRM regression output includes the overdispersion parameter (alpha) and a likelihood ratio test for overdispersion. This command:

```
. nbreg          pub6 female phd enrol, nolog
```

```
Negative binomial regression          Number of obs   =          264
LR chi2(3)                            =          20.59
Dispersion = mean                      Prob > chi2     =          0.0001
Log likelihood = -642.723              Pseudo R2      =          0.0158
```

pub6	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
female						
1_Yes	-.2822292	.1382637	-2.04	0.041	-.553221	-.0112373
phd	.1995909	.0651859	3.06	0.002	.0718288	.327353
enrol	-.150895	.0480431	-3.14	0.002	-.2450578	-.0567322
_cons	1.607418	.3379749	4.76	0.000	.9449989	2.269836
/lnalpha	-.203673	.1255831			-.4498113	.0424654
alpha	.8157291	.1024418			.6377485	1.04338

```
Likelihood-ratio test of alpha=0:  chibar2(01) = 394.12 Prob>=chibar2 = 0.000
```

Because there is significant evidence of overdispersion ($G^2=394.12, p<.001$), the negative binomial regression model is preferred to the Poisson regression model.

6.8) Factor Changes. `listcoef` computes the factor change coefficients. Note that factor change coefficients can also be computed after estimating the PRM, but since the NBRM is our preferred model, we show only the output for NBRM. This command:

```
listcoef, help
```

Produces the following output:

```
. listcoef, help
```

```
nbreg (N=264): Factor change in expected count
```

```
Observed SD: 4.3103
```

	b	z	P> z	e^b	e^bStdX	SDofX
female	-0.2822	-2.041	0.041	0.754	0.874	0.476
phd	0.1996	3.062	0.002	1.221	1.222	1.005
enrol	-0.1509	-3.141	0.002	0.860	0.804	1.443
constant	1.6074	4.756	0.000	.	.	.
alpha						
lnalpha	-0.2037
alpha	0.8157

```
LR test of alpha=0: 394.12 Prob>=LRX2 = 0.000
```

```
b = raw coefficient
```

```
z = z-score for test of b=0
```

```
P>|z| = p-value for z-test
```

```
e^b = exp(b) = factor change in expected count for unit increase in X
```

```
e^bStdX = exp(b*SD of X) = change in expected count for SD increase in X
```

```
SDofX = standard deviation of X
```

Interpretation: Expected publications by female scientists are decreased by a factor of .75 compared to expected publications by male scientists, holding all other variables constant. A standard deviation increase in the number of years from enrollment to completion of PhD, about 1.4 years, decreases the expected number of publications by 14%, holding other variables constant.

6.10) Calculating Discrete Changes. It is also possible to use the `mchange` command to compute the discrete change in the expected count/rate. This command:

```
. mchange, atmeans centered
```

```
nbreg: Changes in mu | Number of obs = 264
```

```
Expression: Predicted number of pub6, predict()
```

	Change	p-value
female		
+1 cntr	-1.050	0.043
+SD cntr	-0.499	0.042
Marginal	-1.047	0.042
phd		
+1 cntr	0.741	0.002
+SD cntr	0.745	0.002
Marginal	0.740	0.002
enrol		
+1 cntr	-0.560	0.002
+SD cntr	-0.809	0.002
Marginal	-0.560	0.002

```
Prediction at base value
```

```
3.709
```

```
Base values of regressors
```

	female	phd	enrol
at	.345	3.18	5.53

```
1: Estimates with margins option atmeans.
```

On average, female scientists have expected productivity that is approximately 1 publication lower than their male counterparts. A standard deviation increase (centered around the mean) in the number of years from enrollment to completion of PhD, about 1.4 years, on average, decreases the expected rate of productivity by .75 publications.

6.11) Predicted Rate and Probabilities. We can use the `mchange` command to generate confidence intervals for the discrete changes by simply adding the `stats(ci)` option. We can also calculate discrete changes at each value of our outcome using the `pr()` option. Here we are comparing men and women's expected productivity at each level of publications.

```
. mchange female, atmeans stats(ci)
```

. mchange female, stats(ci) atmeans

nbreg: Changes in mu | Number of obs = 264

Expression: Predicted number of pub6, predict()

	Change	LL	UL
female			
+1 cntr	-1.050	-2.069	-0.031
+SD cntr	-0.499	-0.980	-0.017
Marginal	-1.047	-2.056	-0.038

Prediction at base value

3.709

Base values of regressors

	female	phd	enrol
at	.345	3.18	5.53

1: Estimates with margins option atmeans.

. mchange female, stats(ci) atmeans pr(0/9)

nbreg: Changes in Pr(y) | Number of obs = 264

Expression: Pr(pub), predict(pr(0))

	0	1	2	3	4	5
female						
+1 cntr	0.047	0.032	0.017	0.006	-0.002	-0.006
LL	0.002	0.001	0.000	-0.001	-0.006	-0.013
UL	0.093	0.062	0.033	0.012	0.002	0.001
+SD cntr	0.022	0.015	0.008	0.003	-0.001	-0.003
LL	0.001	0.000	0.000	-0.000	-0.003	-0.006
UL	0.044	0.030	0.016	0.006	0.001	0.000
Marginal	0.047	0.032	0.017	0.006	-0.002	-0.006
LL	0.002	0.001	0.000	-0.001	-0.005	-0.013
UL	0.093	0.063	0.033	0.012	0.002	0.001
	6	7	8	9		
female						
+1 cntr	-0.009	-0.010	-0.010	-0.009		
LL	-0.018	-0.019	-0.019	-0.018		
UL	0.000	-0.000	-0.000	-0.000		
+SD cntr	-0.004	-0.005	-0.005	-0.004		
LL	-0.009	-0.009	-0.009	-0.009		

UL		0.000	0.000	-0.000	-0.000
Marginal		-0.009	-0.010	-0.010	-0.009
LL		-0.018	-0.020	-0.020	-0.019
UL		0.000	0.000	-0.000	-0.000

Predictions at base value

		0	1	2	3	4	5
-----+							
Pr(y base)		0.181	0.167	0.140	0.113	0.090	0.070
		6	7	8	9		
-----+							
Pr(y base)		0.055	0.043	0.033	0.025		

Base values of regressors

		female	phd	enrol
-----+				
at		.345	3.18	5.53

1: Estimates with margins option atmeans.

On average, female scientists are expected to have approximately one fewer publications than male scientists, with estimated bounds for the 95% confidence interval at -2.20 and -.01. Women are more likely than men to have between 0 and 3 publications and men are more likely than women to have more than 3 publications.

6.12) ZIP and ZINB Models. The `zip` and `zinb` command with the `inf(indvars)` option estimates Zero-Inflated Poisson or Zero-Inflated Negative Binomial Models. You can “inflate” the same set of variables that are used in the main portion of the model, a subset of these variables or an entirely different set of variables. Here we “inflate” the variable `phd`. These commands:

```
. zip pub6 female phd enrol, inf(phd) nolog
```

```
Zero-inflated Poisson regression          Number of obs   =       264
                                           Nonzero obs     =       212
                                           Zero obs        =        52
```

```
Inflation model = logit                  LR chi2(3)      =       48.74
Log likelihood = -758.0032                Prob > chi2     =       0.0000
```

	pub6	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	

pub6							
	female	-.1210631	.0710846	-1.70	0.089	-.2603864	.0182602
	phd	.1400257	.0334849	4.18	0.000	.0743964	.205655
	enrol	-.1306837	.0250179	-5.22	0.000	-.1797178	-.0816496
	_cons	1.838966	.1749225	10.51	0.000	1.496124	2.181808

inflate							
	phd	-.2383082	.1657934	-1.44	0.151	-.5632572	.0866408
	_cons	-.7539084	.5332584	-1.41	0.157	-1.799076	.291259

```
. zinb pub6 female phd enrol, inf(phd) nolog
```

```
Zero-inflated negative binomial regression  Number of obs   =       264
                                           Nonzero obs     =       212
                                           Zero obs        =        52
```

```
Inflation model = logit                  LR chi2(3)      =       18.91
Log likelihood = -642.2026                Prob > chi2     =       0.0003
```

	pub6	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	

pub6							
	female	-.2708994	.1371918	-1.97	0.048	-.5397905	-.0020084
	phd	.1745669	.0695427	2.51	0.012	.0382657	.3108682
	enrol	-.1527173	.047032	-3.25	0.001	-.2448984	-.0605362
	_cons	1.739814	.3498874	4.97	0.000	1.054047	2.42558

inflate							
	phd	-.5440498	.8665119	-0.63	0.530	-2.242382	1.154282
	_cons	-1.456929	2.082817	-0.70	0.484	-5.539175	2.625316

```

      /lnalpha |  -.3514184   .2107589   -1.67   0.095   -.7644982   .0616614
-----+-----
      alpha |   .7036893   .1483088
-----+-----

```

6.13) Factor Change. As with Poisson and Negative Binomial, factor change coefficients can be computed after estimating the ZIP or ZINB models using `listcoef`. Here we show the output for ZINB. The top half of the output, labeled Count Equation, contains coefficients for the factor change in the expected count for those in the Not Always Zero group. The bottom half, labeled Binary Equation, contains coefficients for the factor change in the odds of being in the Always Zero group compared with the Not Always Zero group. These commands:

```
zinb (N=264): Factor change in expected count
```

```
Observed SD: 4.3103
```

```
Count equation: Factor change in expected count for those not always 0
```

```

-----+-----
      |          b          z    P>|z|      e^b    e^bStdX    SDofX
-----+-----
female |   -0.2709   -1.975    0.048    0.763    0.879    0.476
  phd |    0.1746    2.510    0.012    1.191    1.192    1.005
  enrol |  -0.1527   -3.247    0.001    0.858    0.802    1.443
constant |   1.7398    4.972    0.000      .        .        .
-----+-----
alpha  |
  lnalpha |  -0.3514      .        .        .        .        .
  alpha |   0.7037      .        .        .        .        .
-----+-----

```

```

      b = raw coefficient
      z = z-score for test of b=0
      P>|z| = p-value for z-test
      e^b = exp(b) = factor change in expected count for unit increase in X
      e^bStdX = exp(b*SD of X) = change in expected count for SD increase in X
      SDofX = standard deviation of X

```

```
Binary equation: factor change in odds of always 0
```

```

-----+-----
      |          b          z    P>|z|      e^b    e^bStdX    SDofX
-----+-----
  phd |   -0.5440   -0.628    0.530    0.580    0.579    1.005
constant |  -1.4569   -0.699    0.484      .        .        .
-----+-----

```

```

      b = raw coefficient
      z = z-score for test of b=0
      P>|z| = p-value for z-test
      e^b = exp(b) = factor change in odds for unit increase in X
      e^bStdX = exp(b*SD of X) = change in odds for SD increase in X
      SDofX = standard deviation of X

```

Interpretation: Among those who have the opportunity to publish, a standard deviation increase in PhD prestige increases the expected rate of publication by a factor of 1.2, holding other variables constant. A standard deviation increase in PhD prestige decreases the odds of not having the opportunity to publish by a factor of .58, although this is not significant ($z=-0.63$, $p=0.53$).

6.14) Plot $Pr(y=0)$ across the range of C. `mgen` can be used here to calculate the probability of any of a range of counts. In this case, we will be plotting the probability of a zero count from both the Poisson model and from the Negative Binomial model.

```
. qui      poisson          pub6 female phd enrol, nolog
. mgen,    at(phd=(1(1)5)) atmeans stub(prm) pr(0/9)
```

Predictions from: margins, at(phd=(1(1)5)) atmeans predict(pr(9))

Variable	Obs	Unique	Mean	Min	Max	Label
prmpr0	5	5	.0361123	.0052585	.0844589	pr(y=0) from margins
prml10	5	5	.0233283	.0019256	.0506958	95% lower limit
prmul0	5	5	.0488964	.0085913	.1182219	95% upper limit
prmphd	5	5	3	1	5	Prestige of Ph.D. depart...

<snip>

```
. qui      nbreg          pub6 female phd enrol, nolog
. mgen,    at(phd=(1(1)5)) atmeans stub(nbrm) pr(0/9)
```

Predictions from: margins, at(phd=(1(1)5)) atmeans predict(pr(9))

Variable	Obs	Unique	Mean	Min	Max	Label
nbrmpr0	5	5	.1919387	.128004	.2646877	pr(y=0) from margins
nbrml10	5	5	.1403318	.0810974	.1895822	95% lower limit
nbrmul0	5	5	.2435456	.1749106	.3397932	95% upper limit
nbrmphd	5	5	3	1	5	Prestige of Ph.D. depart...

<snip>

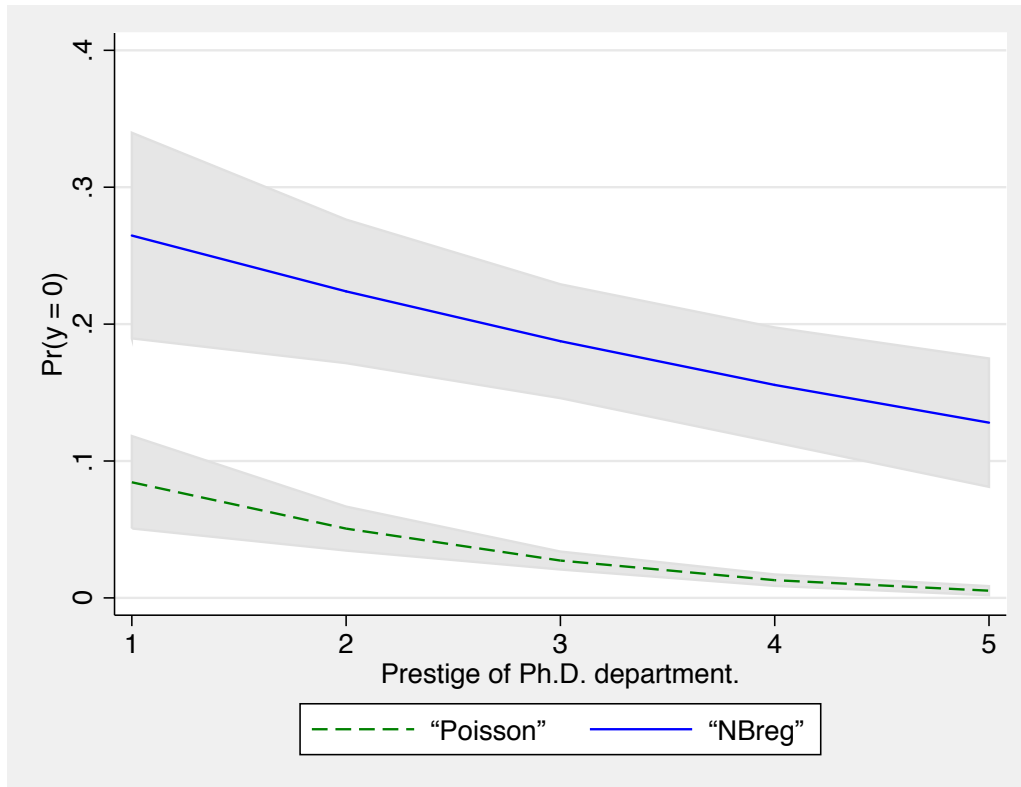
```

.label var prml10 Poisson

.label var nbrml10 NBreg

. graph twoway ///
>     (rarea prml10 prmul0 prmphd, col(gs14)) ///
>     (rarea nbrml10 nbrmul0 prmphd, col(gs14)) ///
>     (connected prmpr0 prmphd, lpat(dash) lcol(green) msym(i) ) ///
>     (connected nbrmpr0 prmphd, lpat(solid) lcol(blue) msym(i) ), ///
>     ylab(0(.1).4, grid gmax gmin) ytitle("Pr(y = 0)") ///
>     legend(on order(3 4))

```



```

. graph export icpsrcda06-count-fig1.png, width(1200) replace
(file icpsrcda06-count-fig1.png written in PNG format)

```

6.15) Compare model using countfit. `countfit` compares the fit of PRM, NBRM, ZIP, and ZINB, optionally generating a table of estimates, a table of differences between observed and average estimated probabilities, a graph of these differences, and various tests and measures of fit. This command:

```
. countfit pub6 female phd enrol, inf( phd) ///
> graph(icpsrcda06-count-fig2.png , width(1200) replace)
```

Variable		PRM	NBRM	ZIP	ZINB

pub6					
	Female? (1=yes)	0.786	0.754	0.895	0.836
		-3.49	-2.04	-1.57	-1.19
	Prestige of Ph.D. department.	1.207	1.221	1.151	1.231
		5.85	3.06	4.19	3.19
	Years from BA to P..	0.876	0.860	0.879	0.871
		-5.51	-3.14	-5.14	-2.82
	Constant	4.630	4.990	6.213	4.532
		9.02	4.76	10.44	4.45

lnalpha					
	Constant		0.816		0.735
			-1.62		-2.14

inflate					
	Female? (1=yes)			2.006	2.60e+06
				2.04	0.02
	Prestige of Ph.D. department.			0.759	1.430
				-1.66	0.49
	Years from BA to P..			1.028	1.370
				0.23	0.68
	Constant			0.351	0.000
				-1.24	-0.02

Statistics					
	alpha		0.816		
	N	264	264	264	264
	ll	-839.781	-642.723	-755.914	-641.263
	bic	1701.865	1313.326	1556.436	1332.709
	aic	1687.561	1295.446	1527.828	1300.526

legend: b/t

Comparison of Mean Observed and Predicted Count

Model	Maximum Difference	At Value	Mean Diff
PRM	0.163	0	0.051
NBRM	0.038	6	0.015
ZIP	0.100	1	0.033
ZINB	0.037	6	0.012

PRM: Predicted and actual probabilities

Count	Actual	Predicted	Diff	Pearson
0	0.197	0.034	0.163	205.490
1	0.144	0.100	0.044	4.992
2	0.129	0.161	0.032	1.688
3	0.121	0.185	0.064	5.777
4	0.095	0.170	0.075	8.815
5	0.053	0.133	0.080	12.712
6	0.091	0.092	0.001	0.003
7	0.023	0.057	0.035	5.546
8	0.042	0.033	0.009	0.589
9	0.023	0.018	0.005	0.371
Sum	0.917	0.983	0.507	245.982

NBRM: Predicted and actual probabilities

Count	Actual	Predicted	Diff	Pearson
0	0.197	0.187	0.010	0.142
1	0.144	0.167	0.023	0.834
2	0.129	0.136	0.008	0.115
3	0.121	0.109	0.013	0.382
4	0.095	0.086	0.009	0.255
5	0.053	0.067	0.014	0.781
6	0.091	0.053	0.038	7.394
7	0.023	0.041	0.018	2.176
8	0.042	0.032	0.009	0.728
9	0.023	0.025	0.003	0.069
Sum	0.917	0.903	0.145	12.875

ZIP: Predicted and actual probabilities

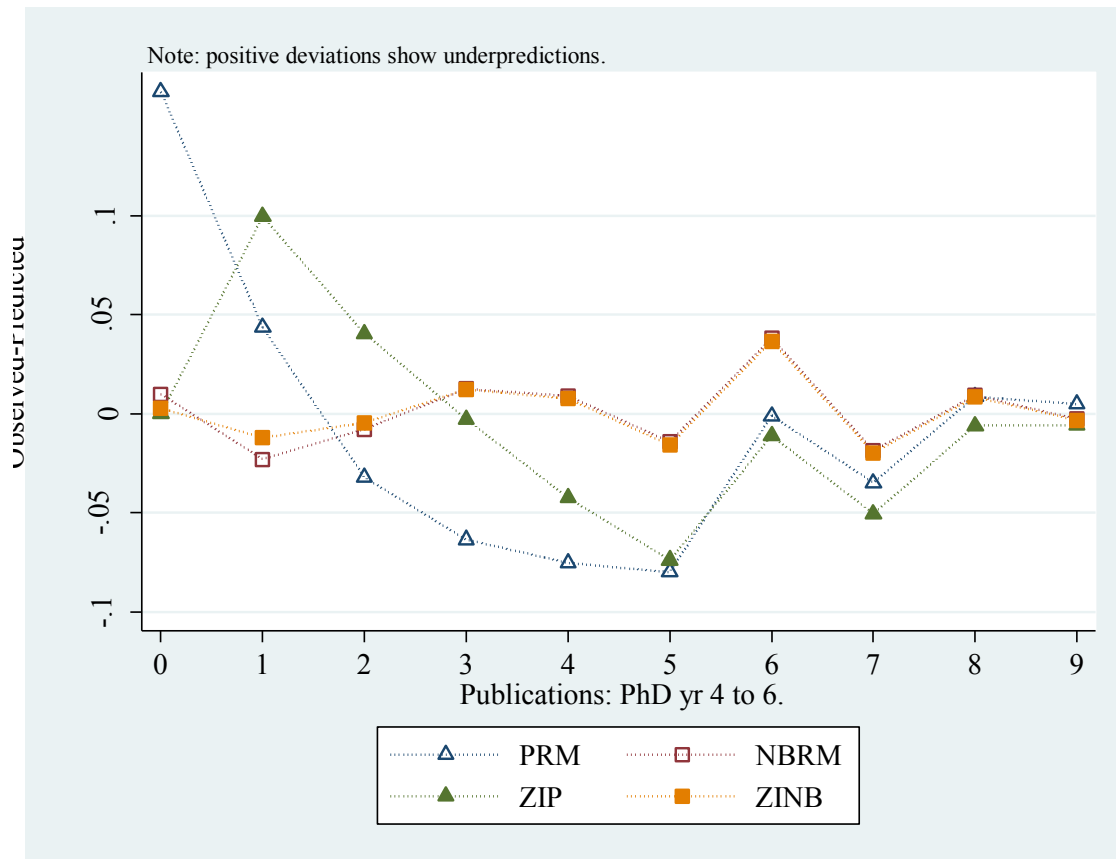
Count	Actual	Predicted	Diff	Pearson
0	0.197	0.197	0.000	0.000
1	0.144	0.044	0.100	59.264
2	0.129	0.088	0.041	4.940
3	0.121	0.124	0.003	0.016
4	0.095	0.137	0.042	3.465
5	0.053	0.127	0.074	11.353
6	0.091	0.102	0.011	0.320
7	0.023	0.073	0.050	9.169
8	0.042	0.048	0.006	0.193
9	0.023	0.028	0.006	0.306
Sum	0.917	0.969	0.332	89.027

ZINB: Predicted and actual probabilities

Count	Actual	Predicted	Diff	Pearson
0	0.197	0.194	0.003	0.010
1	0.144	0.156	0.012	0.247
2	0.129	0.133	0.004	0.040
3	0.121	0.109	0.012	0.373
4	0.095	0.087	0.008	0.179
5	0.053	0.069	0.016	0.958
6	0.091	0.054	0.037	6.598
7	0.023	0.042	0.020	2.416
8	0.042	0.033	0.008	0.567
9	0.023	0.026	0.003	0.109
Sum	0.917	0.904	0.123	11.496

Tests and Fit Statistics

PRM	BIC=	1701.865	AIC=	1687.561	Prefer	Over	Evidence
vs NBRM	BIC=	1313.326	dif=	388.539	NBRM	PRM	Very strong
	AIC=	1295.446	dif=	392.115	NBRM	PRM	
	LRX2=	394.115	prob=	0.000	NBRM	PRM	p=0.000
vs ZIP	BIC=	1556.436	dif=	145.429	ZIP	PRM	Very strong
	AIC=	1527.828	dif=	159.733	ZIP	PRM	
	Vuong=	4.358	prob=	0.000	ZIP	PRM	p=0.000
vs ZINB	BIC=	1332.709	dif=	369.155	ZINB	PRM	Very strong
	AIC=	1300.526	dif=	387.035	ZINB	PRM	
NBRM	BIC=	1313.326	AIC=	1295.446	Prefer	Over	Evidence
vs ZIP	BIC=	1556.436	dif=	-243.110	NBRM	ZIP	Very strong
	AIC=	1527.828	dif=	-232.382	NBRM	ZIP	
vs ZINB	BIC=	1332.709	dif=	-19.384	NBRM	ZINB	Very strong
	AIC=	1300.526	dif=	-5.080	NBRM	ZINB	
	Vuong=	0.834	prob=	0.202	ZINB	NBRM	p=0.202
ZIP	BIC=	1556.436	AIC=	1527.828	Prefer	Over	Evidence
vs ZINB	BIC=	1332.709	dif=	223.726	ZINB	ZIP	Very strong
	AIC=	1300.526	dif=	227.302	ZINB	ZIP	
	LRX2=	229.302	prob=	0.000	ZINB	ZIP	p=0.000



__6.END) Close the Log File.

```
log close
exit
```


Section 7: A Few Advanced Commands & Techniques

This section provides some more advanced commands and techniques for exploring your data and models or completing the exercises listed in the assignments. The use of these commands and methods is not required for this course, so exploring these methods is at your own behest (and risk!). The file `icpsrcda14-07-advanced.do` contains these Stata commands.

___7.1) Set-up your do-file and load your data.

```
capture log close
log using icpsrcda14-07-advanced, replace text

// program:    icpsrcda14-07-advanced.do
// task:       Advanced techniques
// project:    ICPSR CDA
// author:     Trent Mize \ 2014-06-24

*program setup
version      14.0
clear       all
set         linesize 80

//7.1) Load Data
usecda      cda_scireview4
```

___7.2) Using locals. Locals are a type of temporary macro used by Stata, or a way to store variables or commands using a shorthand so that they can be referred to later by other commands. Locals can be helpful for performing complicated analyses as they allow you to change one command (the local) and subsequently change all of the commands that refer to that local. **NOTE:** One thing to remember with locals is that the command that *defines* the local must be run at the same time as all commands that *refer* to that local.

Simple local example: Using locals to refer to a list of variable. Here, we create a local called “ivs” for our independent variables. We can then refer to the local to call up all of these variables, such as in the sum command below:

```
. local ivs "fellow phd mcit3 mnas"

. sum `ivs'
```

Variable	Obs	Mean	Std. Dev.	Min	Max
fellow	264	.4128788	.4932865	0	1
phd	264	3.181894	1.00518	1	4.66
mcit3	264	20.71591	25.44536	0	129
mnas	264	.0833333	.2769103	0	1

Example—setting up a logit model. In the binary assignment, we ran this model:

```
logit faculty fellow phd mcit3 mnas
```

[We'll store it so we can compare to the results from the model using locals:]

```
est store reg
```

However, we could have also set up the model using locals, first by defining them & then by referring back to them:

```
local dv "faculty" // sets up local called dv which contains our dependent var
local ivs "fellow phd mcit3 mnas" // local called ivs with our independent vars
logit `dv' `ivs' // runs the logit; note different quotation marks—lhs uses the
                // quotation mark on the top left of most keyboards, rhs uses
                // regular singular quote mark
```

Note again that for these commands to work, they all need to be run **together**. Again we will store this model and then use `esttab` to show that the local approach produces results identical to the regular approach.

```
est store reg
esttab reg local
```

This produces this output:

```
. esttab reg local
```

```
-----
                (1)                (2)
                faculty            faculty
-----
faculty
fellow                1.250***            1.250***
                   (4.52)            (4.52)

phd                   -0.0637            -0.0637
                   (-0.43)            (-0.43)

mcit3                 0.0206**           0.0206**
                   (2.89)            (2.89)

mnas                  0.364              0.364
                   (0.65)            (0.65)

_cons                 -0.581            -0.581
                   (-1.29)            (-1.29)
-----
N                      264              264
-----
```

t statistics in parentheses

* p<0.05, ** p<0.01, *** p<0.001

7.3) Using loops to perform multiple analyses. Loops are ways of telling Stata to do something multiple times for multiple values or multiple variables. Loops can be defined and used in many ways—far beyond the scope of this simple guide—but here we provide a couple of examples for how you might want to use them. As a rule, anytime I need to do something more than once, I use a loop!

Example—recoding a variable. We often want to recode a continuous or ordinal variable into a series of binary variables. Loops make this easy. The following loop, for example, generates a series of new binary variables indicating whether PhD prestige was above or below a certain cutpoint. The first line defines the local `v` and the values of the cutpoints (2 through 5; `/` indicates we want to increase in increments of 1). The open bracket starts the code, which refers back to the local ``v'`, and the close bracket indicates the end of the loop. So these commands, run at the same time:

```
forvalues v = 2/5 {
    gen phdlt_`v' = phd < `v' if phd < .
    label var phdlt_`v' "PhD<`v'"
}
codebook phdlt*, compact
```

Produce this output:

```
. forvalues v = 2/5 {
  2.      gen phdlt_`v' = phd < `v' if phd < .
  3.      label var phdlt_`v' "PhD<`v'"
  4.      }

. codebook phdlt*, compact
```

Variable	Obs	Unique	Mean	Min	Max	Label
phdlt_2	264	2	.1439394	0	1	PhD<2
phdlt_3	264	2	.4734848	0	1	PhD<3
phdlt_4	264	2	.7007576	0	1	PhD<4
phdlt_5	264	1	1	1	1	PhD<5

Example II—generating interaction terms. Another frequent use of loops is generating multiple interaction terms. Here we generate interactions between female and a number of other variables. Note that this time our loop uses the `foreach` term at the beginning, rather than `forvalues`. `Foreach` allows non-numbers to be stored and looped through; `forvalues` is strictly for numerical terms. So these commands, run at the same time:

```
foreach v in mcit3 phd fellow {
    gen femX`v' = female * `v'
    label var femX`v' "Female x `v'"
}
codebook femX*, compact
```

Produce this output:

```
. foreach v in mcit3 phd fellow {
  2.      gen femX`v'= female * `v'
  3.      label var femX`v' "Female x `v'"
  4.      }

. codebook femX*, compact
```

Variable	Obs	Unique	Mean	Min	Max	Label
femXmcit3	264	30	5.965909	0	110	Female x mcit3
femXphd	264	25	1.156098	0	4.36	Female x phd
femXfellow	264	2	.0871212	0	1	Female x fellow

7.4) Generating a matrix of values. One of the most pressing problems in this age of data analysis can be making sense of large amount of data and output. Although many of Stata’s commands are easy to read and keep track of, generating matrices can sometimes make taking in data even easier. Matrix commands can be a bit confusing—but once you have the hang of it, are easy to copy and replicate.

Example—Matrix of summary statistics by gender. This is a simple matrix that uses locals and loops to collect the means of a set of binary variables by gender. The code below contains several steps: first, defining a local of the important variables; second defining a [blank] matrix, with two columns and rows that correspond to the variables in our local ``var'`; third, looping through these variables using the `sum` command, saving the appropriate means in new locals and using those locals to populate the matrix; and finally printing the matrix. These commands, run together:

```
*set up locals
local var fellow faculty phdlt_2 phdlt_3 phdlt_4
local q quietly

*define matrix
local nvars: word count `var'
matrix outmat= J(`nvars',2,.)
matrix colnames outmat = femaleMN maleMN
matrix rownames outmat=`var'
local irow=0

*loop through variables
foreach v of varlist `var' {
  local ++irow
  `q' sum `v' if female==1
  local fem = r(mean)
  `q' sum `v' if female==0
  local male = r(mean)
}

* populate matrix
mat outmat[`irow',1] = `fem'
mat outmat[`irow',2] = `male'
}
```

```
*print matrix
mat list outmat, f(%6.3f) ///
    title("Means for select variables, by gender")
```

Produce this output:

```
. *define locals
. local var fellow faculty phdlt_2 phdlt_3 phdlt_4
. local q quietly

. *define matrix
. local nvars: word count `var'

. matrix outmat= J(`nvars',2,.)
. matrix colnames outmat = femaleMN maleMN
. matrix rownames outmat=`var'
. local irow=0

. *loop through variables
. foreach v of varlist `var' {
2.     local ++irow
3.     `q' sum `v' if female==1
4.     local fem = r(mean)
5.     `q' sum `v' if female==0
6.     local male = r(mean)
7. * populate matrix
.     mat outmat[`irow',1] = `fem'
8.     mat outmat[`irow',2] = `male'
9.     }

. *print matrix
. mat list outmat, f(%6.3f) ///
>     title("Means for select variables, by gender")
```

```
outmat[5,2]:  Means for select variables, by gender
      femaleMN    maleMN
fellow    0.253    0.497
faculty    0.396    0.607
phdlt_2    0.121    0.156
phdlt_3    0.429    0.497
phdlt_4    0.527    0.792
```

Because all of the variables included here are binary, this provides an easy way of knowing what proportion of males and females belong to each group—for example, 25% of female scientists have fellowships compared to nearly 50% of male scientists. **QUESTION:** How could you modify this code to include a column for the difference between the two means?